A Minimalistic Approach to Segregation in Robot Swarms

Peter Mitrano[1], Jordan Burklund[1], Michael Giancola[1], Carlo Pinciroli[1]

***Abstract*— We present a decentralized algorithm to achieve segregation into an arbitrary number of groups with swarms of autonomous robots. The distinguishing feature of our approach is in the minimalistic assumptions on which it is based. Specifically, we assume that (i) Each robot is equipped with a ternary sensor capable of detecting the presence of a single nearby robot, and, if that robot is present, whether or not it belongs to the same group as the sensing robot; (ii) The robots move according to a differential drive model; and (iii) The structure of the control system is purely reactive, and it maps directly the sensor readings to the wheel speeds with a simple 'if' statement. We present a thorough analysis of the parameter space that enables this behavior to emerge, along with idealized conditions for convergence and a study of non-ideal aspects in the robot design.**

I. Introduction

Group formation is one of the most fundamental mechanisms a robot swarm must exhibit [1]. Group formation can occur in several forms to satisfy different requirements. Segregation is a particular type of group formation in which the focus is on creating local aggregates of robots that share a common property. Segregation can be seen as a precursor to object sorting, task allocation, or self-assembly. For example, swarms may need to split into arbitrary groups to diffuse and search different areas, or segregate by skill or capability in order to form useful heterogeneous teams.

Segregation is an example of the broader class of spatially organizing behaviors, whose purpose is to impose a structure in the environment (e.g., object clustering [2], collective construction [3]) or in the distribution of the robots (e.g., aggregation [4], pattern formation [5], self-assembly [6]).

A recent line of research in spatially organizing behaviors focuses on the *minimal* assumptions a swarm of robots must fulfill in order to perform the task. Johnson and Brown [7] and Brown *et al.* [8] characterized the set of possible behaviors that can be obtained using primordial control strategies based on a simple 'if/then/else' structure, binary sensors, and differential-drive robots. Gauci *et al.* provided the specific conditions for the emergence of aggregation [9] and object clustering [2], while St.-Onge *et al.* [10] studied the emergence of circular formations. While more efficient control strategies have been proposed to achieve these behaviors, studying the minimal assumptions required for their emergence is an important step towards principled 'swarm engineering' practices. In addition, these minimal behaviors might offer last-resort solutions in case of sensor failures in remote environments such as in planetary exploration missions.

This paper furthers this line of inquiry by studying the minimal assumptions for N-class segregation to emerge from local, decentralized interactions among robots. The term 'N-class' refers to the creation of N spatially distinct groups. We show that, for segregation to emerge, it is sufficient to equip an 'if/then/else', differential drive robot with a *ternary* sensor. This sensor detects the presence of a robot in range. When a robot is detected, the sensor can distinguish whether it is a *kin*, i.e., it belongs to the same group as the sensing robot, or a *non-kin*, i.e., it belongs to a different group. When multiple robots are in range, the sensor returns information on the closest one of them.

The main contributions of this paper are *(i)* A study of the parameter space that enables the emergence of N-class segregation; *(ii)* A study of why convergence is guaranteed for the best parameter choice found; and *(iii)* An analysis of the robustness of the algorithm to non-idealities in the robot design.

II. Related Work

Segregation is a common behavior in nature, and it can be observed across scales. For example, cell segregation is a basic building block of embryogeneis in tissue generation processes [11], [12]; while social insects, such as ants, organize their brood into ring-like structures [13].

In robotics, segregation is a problem that has not received considerable attention. The main methods that have been proposed so far are based on some variation of the artificial potential approach [14], which assumes that the robots can detect each other and estimate relative distance vectors.

Groß *et al.* [15] proposed an algorithm inspired by the Brazil Nut effect, in which the robots form regular layers simulating gravity by sharing a common direction. This study was later extended to work on e-pucks robots [16]. To simulate gravity, this approach requires the robots to share a common target vector, which can be obtained through centralized controllers or a distributed consensus algorithm.

Kumar *et al.* [17] introduced the concept of "differential potential", whereby two robots experience a different artificial potential depending on their being part of the same class or not. The convergence of this approach is guaranteed for two classes, but when more classes are employed local minima prevent segregation from emerging.

Santos *et al.* [18] took inspiration from [17] to devise an approach based on the Differential Adhesion Hypothesis,

[1] Robotics Engineering, Worcester Polytechnic Institute, MA, USA. Email: cpinciroli@wpi.edu

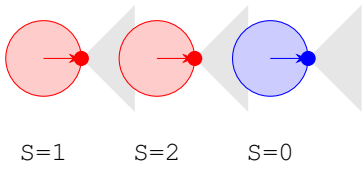

Fig. 1. A diagram of the ternary sensor in which classes are depicted as colors. The left red robot detects a kin robot, so its sensor returns 1. The middle red robot detects a blue robot, so its sensor returns 2. The right robot detects no robot, so its sensor returns 0.

Algorithm 1 The segregation control algorithm.

if $S = 0$ **then**
  set wheel speeds to $v_{\text{left}}^{S=0}$, $v_{\text{right}}^{S=0}$
else if $S = 1$ **then**
  set wheel speeds to $v_{\text{left}}^{S=1}$, $v_{\text{right}}^{S=1}$
else
  set wheel speeds to $v_{\text{left}}^{S=2}$, $v_{\text{right}}^{S=2}$
end if

which states that kin cells tend to adhere stronger than non-kin cells. However, one limitation is the assumption that the robots have global knowledge about the positions of other robots.

To the best of our knowledge, this paper is the first to propose a segregation algorithm that is not based on global information nor on communication and sensing of multiple neighbors.

III. Methodology

A. Problem Formulation

Motion model. We consider a set of robots executing the same controller in a two-dimensional, obstacle-free environment. The robots are equipped with two wheels for which $[v_{\text{left}}, v_{\text{right}}]$ denote their *normalized* linear speeds. By 'normalized' we mean that the speed values are in the range $[-1, 1]$. Using normalized speeds allows us to reason in a general way over the specific speeds attainable by any robot. To transform from normalized speeds $[v_{\text{left}}, v_{\text{right}}]$ into actual speeds $[V_{\text{left}}, V_{\text{right}}]$, we introduce a parameter V_{max} that denotes the maximum linear speed possible with a specific robot and define

$$V_{\text{left}} = V_{\text{max}} v_{\text{left}} \quad (1)$$

$$V_{\text{right}} = V_{\text{max}} v_{\text{right}}. \quad (2)$$

The robots' motion assuming constant wheel velocities is modeled by the well-known differential-drive equations, shown in Equation (5).

Sensor model. The robots are also equipped with a ternary sensor that is able to detect the presence of nearby robots and their "kinness". Two robots are *kin* if they belong to the same class (denoted by color in our experiments); they are *non-kin* otherwise. The sensor is assumed to have infinite range (we consider non-infinite range in Sec. VI-E). As depicted in Fig. 1, the sensor returns a reading $S = 0$ when no robot is detected, $S = 1$ when a kin robot is detected, and $S = 2$ when a non-kin robot is detected. We allow for any number of classes, but the sensor need not distinguish between different non-kin classes—it only detects whether a nearby robot belongs to the same class or not.

Control logic. The control logic followed by the robots is formalized in Alg. 1. It is a simple 'if/then/else' structure, which maps the sensor readings S directly into normalized wheel speeds $[\, v_{\text{left}} \;\; v_{\text{right}} \,]$. The latter are the parameters whose value we intend to study, and they can be encoded as a six-dimensional vector

$$[\, v_{\text{left}}^{S=0} \;\; v_{\text{right}}^{S=0} \;\; v_{\text{left}}^{S=1} \;\; v_{\text{right}}^{S=1} \;\; v_{\text{left}}^{S=2} \;\; v_{\text{right}}^{S=2} \;\;].$$

Objective. The objective of our study is to find the values of the speed parameters for which the robots group into clusters, such that all the robots of the same class are packed into one cluster with no non-kin robots.

B. Simulation Environment and Robots

Simulation platform. We utilized the ARGoS multi-robot simulator [19] to search for controller parameters and evaluate them. ARGoS offers accurate models for several differential-drive robots, such as the foot-bot [20], the Khepera IV[1], and the Kilobot [21].

Simulated Robot platform. For the simulated experiments we opted to use the foot-bot, because of the possibility to utilize its range-and-bearing communication system as a base for the ternary sensor. The advantage of using the range-and-bearing system is that its model is simple and, as a consequence, a large number of simulations could be completed in a short time. The range-and-bearing system allows two robots to exchange messages when they are in direct line-of-sight; upon receiving a message, a robot can also estimate the relative position of the sender. This sensor, in principle, receives messages from all the nearby robots. To simulate the ternary sensor, our robot controller kept the message of the closest robot. The message payload was an integer that encoded the id of the group to which the sender belonged. The range-and-bearing sensor is simulated through ray casting. This allowed us to assume that the sensor reading is infinitely thin, a choice that simplifies the mathematical analysis presented in Sec. V. However, in practical applications, the sensor can be expected to cast a cone-shaped sensory range with non-zero aperture angle. We explore the performance effect of various angles in Sec. VI-D.

Real Robot platform. We used the Turtlebot 3 robots equipped with Raspberry Pi cameras. To implement the ternary sensor, we placed bightly color skirts on each robot and used OpenCV blob-detection to identify the color of the robot. Each robot knows it's own color, this way kin versus non-kin robots can be identified. We use a threshold on blob size and color values to distinguish whether there are any robots in our line of sight.

[1] https://www.k-team.com/khepera-iv

C. Grid Search

Trial setup. In order to exhaustively search the space of possible controllers, we conducted a grid search of the 6-dimensional parameter space. Due to limited computational resources, we were only able to search with a resolution of 7 values per parameter: $[-1, -2/3, -1/3, 0, 1/3, 2/3, 1]$. In total we evaluated $7^6 = 117,649$ parameter sets. For each parameter set, we tested 36 different initial configurations (8 with 1-class, 8 with 2-class, 20 with 4-class), with 100 simulated seconds for each trial. These initial configurations consisted of uniformly random placement, clusters, and lines of robots distributed throughout the environment. We chose to include some structured configurations (clusters and lines) because we discovered that they affected significantly the performance with respect to uniform random configurations. Hence, by explicitly evaluating diverse initial configurations, we could better estimate the best parameter values in the general case. Examples of these starting configurations can be seen in our supplementary videos [2].

Clusters. To define our cost function, we first establish the notion of a cluster. A cluster can be intuitively defined as an island of connected kin robots. Formally, a cluster is a set of kin robots which are *connected*. Denoting with r the radius of the body of a robot, and defining $\mathbf{p}_i(t) = [x_i(t)\ y_i(t)]$ as the x and y coordinates of the robot at time t, we consider any two kin robots to be connected if

$$\|\mathbf{p}_i(t) - \mathbf{p}_j(t)\| \leq 2r + \epsilon \qquad (i \neq j, \epsilon \in \mathbb{R}^+) \qquad (3)$$

In our experiments, we set $\epsilon = 5\,\text{cm}$. We find clusters by first constructing an adjacency matrix and then performing a depth-first search. Since we are interested in segregating the robots in N classes, the final result of a trial is expected to be a set of N distinct clusters composed of kin robots.

Cost function. To measure the difference between the ideal, perfectly segregated result and any configuration achieved by the robots over time, we first calculate, for every class i, the number of robots in the largest cluster $c_i(t)$ formed by robots of class i. Since, in principle, different classes might involve different numbers of robots, in our cost function we employ the ratio

$$\gamma_i(t) = -\frac{c_i(t)}{C_i}$$

This ratio should be maximized, so the negative sign assigns larger clusters a lower, more negative, cost. Here C_i is the number of robots that belongs to class i. At each time step, the cost is

$$\gamma(t) = \frac{1}{N}\sum_{i=1}^{N}\gamma_i(t)$$

Our complete cost function is then

$$c_{\text{total}} = \sum_{t=0}^{T-1} t\gamma(t) \qquad (4)$$

[2]https://goo.gl/z8UAuB

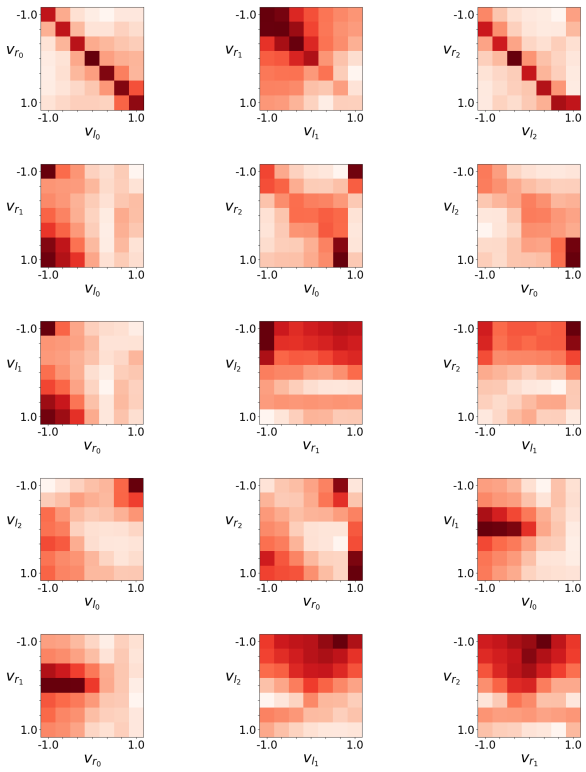

Fig. 2. Heatmaps that relate relevant pairs of wheel speeds. Darker is worse.

in which we denote the total trial time (100 seconds) with T. The effect of multiplying $\gamma_i(t)$ by t is to highlight the emergence of clusters as the trial time proceeds: we cannot expect large clusters to be present at the beginning of a trial, but good parameter settings should grow (and maintain) clusters over time. In our experiments, we found that c_{total} correctly assigns cost in most scenarios, thus fitting well our analysis purposes. However, this cost function considers a straight line of robots to be a cluster, and as such it might not be ideal for scenarios in which the clusters are required to be tight.

IV. The Emergent Behavior

Visualizing grid search. The results of grid search are reported in Fig. 2. Because the search space is six-dimensional, we chose to visualize it by plotting every pair of parameters against each other. For example, we consider how the cost changes as $v_{\text{left}}^{S=0}$ and $v_{\text{right}}^{S=0}$ change. As an example of reading these plots, we can tell from the plot of $v_{\text{left}}^{S=1}$ and $v_{\text{right}}^{S=1}$ that there were no good controllers where the left and right wheel speeds were equal and negative (dark squares in the upper left), and that the best controllers had unequal values close to 1 (lightest squares in the bottom left). These plots also convey the presence of sharp discontinuities where performance changes dramatically.

The emergent behavior. After running the grid search, the parameters with the lowest mean cost across all 36 starting

configurations

$$[-1, 1/3, 1/3, 1, -1, 1]. \quad \text{(C)}$$

The resulting behavior is for a robot to turn sharply until it detects a kin, then turn less sharply while it sees the kin. This change in the radius of rotation moves the robot closer to its kin. When a non-kin is seen, the robot turns in place. In isolation, the emergent behavior between kin and non-kin is similar to the aggregation behavior found by Gauci *et al.* [9] in that the robots are always turning in the same direction (counter-clockwise for our parameters). To our surprise, we found that the best segregation behavior requires that non-kin also aggregate. This was not only true for the best parameters, but for all of the top-scoring parameters we observed. To better appreciate the dynamics of this behavior, we invite the interested reader to watch the videos at https://goo.gl/z8UAuB.

V. Behavior Analysis

Using the parameter settings (C), we now break down and analyze the emerging behavior. While we are not able to formally explain the segregation behavior among an arbitrary number of robots, we can instead consider simplified scenarios with two robots. In this section, we provide simple conditions under which two kin robots in isolation are guaranteed to aggregate with each other. Interestingly, we also show that non-kin robots aggregate, and provide the condition under which this occurs. From these two conditions, we can then hypothesize that large numbers of kin and non-kin robots also aggregate, which partially explains the emergent behavior observed in our experiments.

Critical to this analysis are the assumptions that the sensor is an infinitely thin beam, and that both robot controller are synchronized such that both robots take sensor readings at the same time and act for the same Δt time step. Lastly, we assume the robots have the same radius.

In the proofs, we employ the well-known equations that govern the instantaneous radius of curvature R and rotation speed ω of the path followed a differential-drive robot with l denoting the interwheel distance [22]:

$$\begin{aligned} R &= \frac{l}{2}\frac{V_{\text{right}} + V_{\text{left}}}{V_{\text{right}} - V_{\text{left}}} = \frac{l}{2}\frac{v_{\text{right}} + v_{\text{left}}}{v_{\text{right}} - v_{\text{left}}} \\ \omega &= \frac{V_{\text{right}} - V_{\text{left}}}{l} = \frac{V_{\max}(v_{\text{right}} - v_{\text{left}})}{l}. \end{aligned} \quad (5)$$

Table V shows values of R and ω for each sensor value, which we then use in our proofs.

We derive conditions under which the robots eventually step closer to each other regardless of robot positions or sensor value. We consider all of the possible sensor values ($S = 0$, $S = 1$, $S = 2$) for a robot i and a second robot j in separate cases. However, the cases ($S_i = 1$, $S_j = 0$) and ($S_i = 0$, $S_j = 1$) and the cases ($S_i = 2$, $S_j = 0$) and ($S_i = 0$, $S_j = 2$) are equivalent so we need only prove it for one of each pair. Furthermore, for two isolated robots, the cases where one robot sees a kin and the other sees a non-kin ($S_i = 1$, $S_j = 2$) and ($S_i = 1$, $S_j = 2$) are impossible. This leaves just 5 cases, and we derive a condition for each.

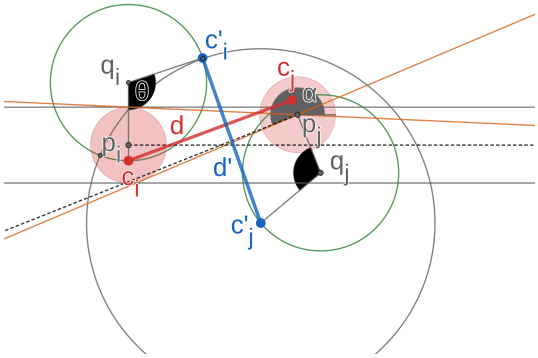

Fig. 3. Two kin in an arbitrary valid configuration, where robot i and robot j see each other. The robots are shown as red circles. The dashed lines indicate the sensor beam of the robots. p_i and p_j are the positions of robots i and j respectively. The robots rotate around the points c_i,c_j when $S = 0$ and around the points q_i,q_j when $S = 1$.

We use x, y, and α, collectively called a configuration, to indicate the location of robot j relative to robot i. In order to prove that robots aggregate for any x, y, and α, we define a worst-case configuration and derive the conditions under this configuration.

We define worst-case based on the angle θ, where θ is defined as the angle at which the distance between c_i' and c_j' equals the distance between c_i and c_j. In detail, the distance from c_i and c_j initially decreases, reaches a minimum, and then increases. At some point, this distance will be equal to the original distance, and the angle at which this occurs is defined as θ. For there to be aggregation, we require $\omega\Delta t \leq \theta$. Therefore, the stricted condition on the amounts the robot can rotate $\omega\Delta t$ is worst when the condition on the angle of rotation θ is smallest.

Rather than showing that the center point of the robots p_i and p_j aggregate, we show that the points c_i and c_j aggregate. These are the points around which the robot rotates in the $S = 0$ case. These points are at a distance $\frac{l}{4}$ from the center of the robot, and therefore when $c_i = c_j$ the points p_i and p_i are at most $\frac{l}{2}$ apart. On a real robot, the half of the inner wheel distance l must be less than the robot radius r (or the wheels would protrude from the robot). Hence aggregation of c_i and c_j is equivalent to the aggregation of p_i and p_j.

Theorem 1 (Nothing Detected): Two robots seeing nothing ($S_i = 0$, $S_j = 0$) will not separate, and will eventually transition to another case if

Proof: ■

Theorem 2 (Kin Robots #1): In the case where robot i sees kin robot j and robot j sees robot i ($S_i = 1$, $S_j = 1$), the robots aggregate when

$$V\Delta t \leq 3l \tan^{-1}\left(\frac{\sqrt{3}r}{2l + r}\right) \quad (6)$$

Proof:

To find a condition under which c_i moves towards c_j for any valid configuration (x, y, α), we define valid by excluding all the configurations which are impossible for

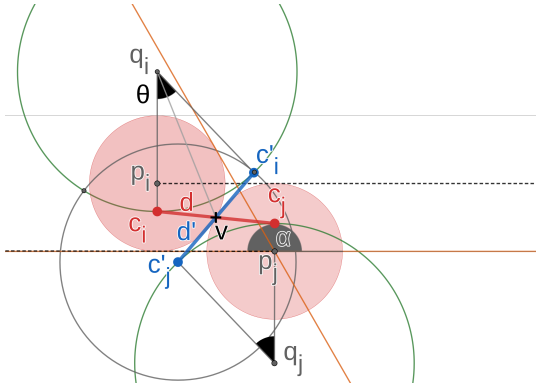

Fig. 4. Theorem 2: Two kin in the worst-case configuration for which θ is minimized.

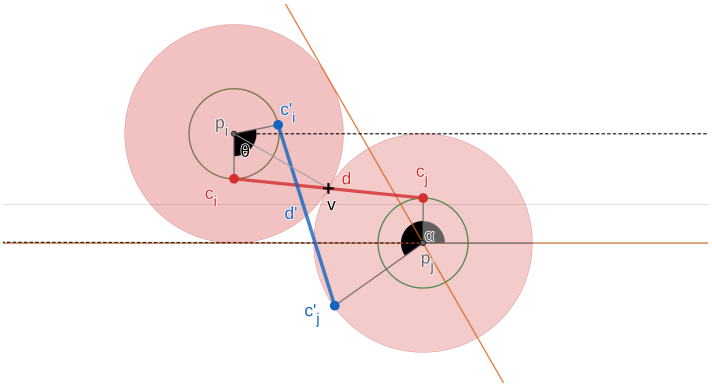

Fig. 5. Theorem 3: Two non-kin in the worst-case configuration for which θ is minimized.

this setting where $S_i = 1$ and $S_j = 1$. This is visualized in Figure 3 as the region bounded by the horizontal grey lines on top and bottom, and the radius of robot i on the left. The region is unbounded on the right. Any x, y position of robot j outside this region is invalid because then robot i would not hit robot j.

Next, we first find the valid configuration which gives the tightest bound on the angle θ. We defined θ to be the maximum angle through which c_i and c_j may rotate such that the distance d' between c'_i and c'_j (the blue line) is no greater than the original distance d (the red line). The worst-case configuration is found by letting α be such that the beam of robot j is tangent to the bottom of robot i, setting $y = -r$, and $x = \sqrt{3}r$. To understand why this is the worst-case, observe that as α is constrained to lie between the orange lines because we are considering the case where robot j sees robot i. As α increase within this range, the point c_j moves counter clockwise towards c_i and therefore d strictly decreases.

Furthermore, as d decreases θ must also decrease to maintain the defining property of θ – which is that $d' \leq d$. Therefore the worst-case for α is when it obtains its maximum value, which occurs when the beam of robot j is tangent to the bottom of robot i. Next, we observe that θ also strictly decreases as y decreases and x decreases. Since $y \in [-r, r]$ take y to be the minimum value of $y = -r$. In this case, the minimum value of x is then $x = \sqrt{3}r$. This worst-case configuration is shown in Figure 4. From this worst-case, we can now define the exact positions $q_i = [0, 0]$, $p_i = [0, -l]$, $v = [\sqrt{3}r, -l - r/2]$. Reasoning on a right triangle with vertices q_i and v, we find that

$$\frac{\theta}{2} = \tan^{-1}\left(\frac{|v_x|}{|v_y - q_{iy}|}\right) = \tan^{-1}\left(\frac{\sqrt{3}r}{2l + r}\right) \tag{7}$$

From this we require that the angle the robot rotates in one time step be less than this angle, and thus the condition under which the kin aggregate in this scenario is

$$V\Delta t \leq 3l \tan^{-1}\left(\frac{\sqrt{3}r}{2l + r}\right) \tag{8}$$

■

Theorem 3 (Non-Kin Robots #1): In the case where robot i sees non-kin robot j and robot j sees robot i ($S_i = 2$, $S_j = 2$), the robots aggregate when

$$2V\Delta t \leq l\pi \tag{9}$$

Proof:
This proof follows the same logic and geometry as the previous case (Theorem 2) so for brevity we omit the full proof. Ultimately, we arrive at the worst-case configuration shown in Figure 5. The coordinates are now $p_i = [0, 0]$ and $v = [\sqrt{3}r/2, r/2]$, which gives

$$\frac{\theta}{2} = \tan^{-1}\left(\frac{|v_x|}{|v_y - p_{iy}|}\right) = \tan^{-1}\left(\frac{\frac{\sqrt{3}r}{2}}{\frac{r}{2}}\right) = \frac{\pi}{3} \tag{10}$$

$$\begin{aligned} \frac{4V}{3l}\Delta t &\leq \frac{2\pi}{3} \\ 2V\Delta t &\leq l\pi \end{aligned} \tag{11}$$

■

Theorem 4 (Kin Robots #2): In the case where robot i sees kin robot j and robot j sees nothing ($S_i = 1$, $S_j = 0$), the robots aggregate when

$$\begin{aligned} 2V\Delta t &\leq 3l(\pi - 2\tan^{-1}(\eta_1) - 2\sin^{-1}(\eta_2)) \\ \eta_1 &= \frac{5l + 4r}{4\sqrt{3}r} \\ \eta_2 &= \frac{l}{\sqrt{25l^2 + 40lr + 64r^2}} \end{aligned} \tag{12}$$

Proof:
In this case, the point c_j does not move because $S_j = 0$. Again we arrive at the worst-case configuration shown in

	$S = 0$	$S = 1$	$S = 2$
v_l	-1	$1/3$	-1
v_r	$1/3$	1	1
R	$-l/4$	l	0
ω	$\frac{4V}{3l}$	$\frac{4V}{3l}$	$4V/3l$

TABLE I
TURNING RADII AND ROTATIONAL VELOCITIES FOR THE BEST PARAMETERS.

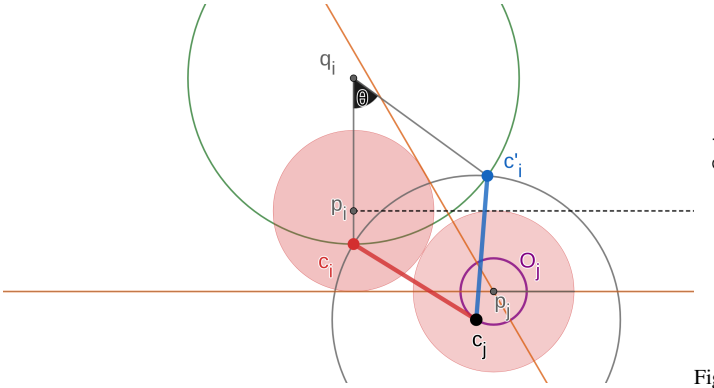

Fig. 6. Theorem 4: Two kin in the worst-case configuration for which θ is minimized.

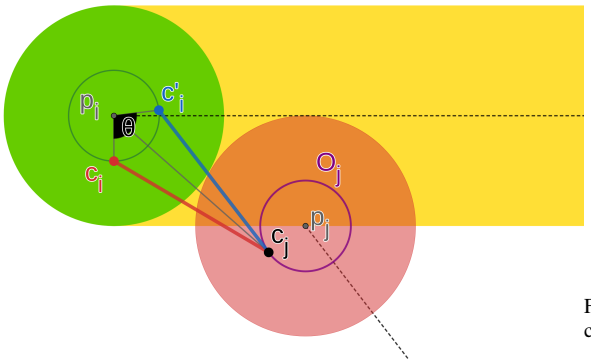

Fig. 7. Theorem 5: Two non-kin in the worst-case configuration for which θ is minimized.

Figure 5. This case is derived by noting for any position of p_j, the point c_j lies somewhere on a circle O_j of radius $l/4$ centered at p_j. The location of c_j which minimizes θ is when the line from q_i to c_j is tangent on the bottom of O_j. Like in the Theorem 2 the worst configuration is achieved by minimizing x and y. The coordinates are now $q_i = [0, 0]$ and $v = [\sqrt{3}r/2, r/2]$, which gives ■

Theorem 5 (Non-Kin Robots #2): In the case where robot i sees robot j and robot j sees nothing, the robots aggregate when

$$\begin{aligned} 2V\Delta t &\leq 3l(\pi - 2\tan^{-1}(\eta_1) - 2\sin^{-1}(\eta_2)) \\ \eta_1 &= \frac{l + 4r}{4\sqrt{3}r} \\ \eta_2 &= \frac{l}{\sqrt{l^2 + 8lr + 64r^2}} \end{aligned} \tag{13}$$

Proof:

This proofs is the same as the proof of Theorem 4, and the worst-case configuration is shown in Figure 7

■

VI. Experimental Results

A. Real Robot Study

B. Scalability Study

In this experiment we investigate how the segregation behavior scales with the number of classes and the number of robots in the environment. We varied the number of classes from 1 to 25 and ran 100 trials with robots uniformly randomly distributed. All trials lasted 180 seconds.

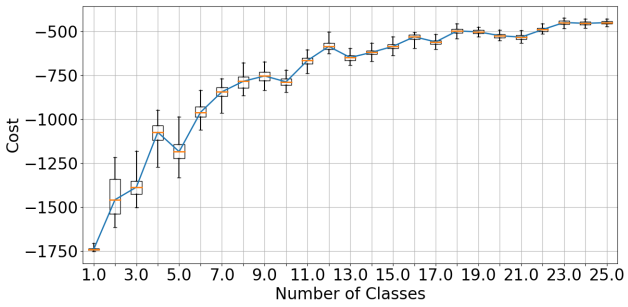

Fig. 8. The average cost over 100 trials with N classes, 10 robots per class.

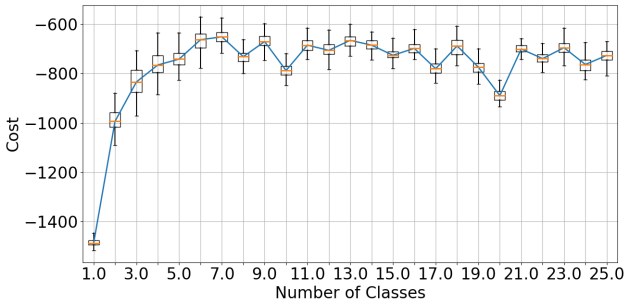

Fig. 9. The average cost over 100 trials with 100 robots divided into N classes.

Fixed number of robots per class. We studied the case in which every class has 10 robots. The results of this are plotted in Fig. 8. We observe that the cost increases with the number of classes. As the number of classes increases, the density of the robots increases too. Hence, line-of-sight occlusions between robots are more likely, navigation is more difficult, and the clusters do not have a chance to coalesce. The important trend is that as the number of robots increases, the cost increase sublinearly. This suggests that our method scales well with the total number of robots.

Fixed total number of robots. We considered the scenario in which a fixed number of robots is split into an increasing number of classes. We set the number of robots to 100, so with 25 classes 4 robots were still assigned to each class. As reported in Fig. 9, the cost is very low for small numbers of classes and above 5 classes has no trend in cost. The initial low cost is due to the fact that with 100 robots to divided among few classes, line-of-sight to a kin is highly likely. Here the trend indicates that our methods scaled well with the number of classes.

C. The Effect of Implementation Details of the Sensor

We observed that the implementation details of the sensor have a significant effect on the behavior of the controller.

Initially, our method for determining sensor value from the simulated range-and-bearing system was to consider all

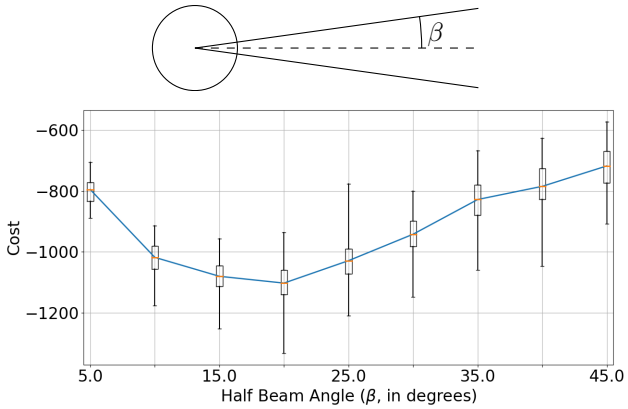

Fig. 10. A 15° degree half beam angle is best for segregation. Lower cost is better.

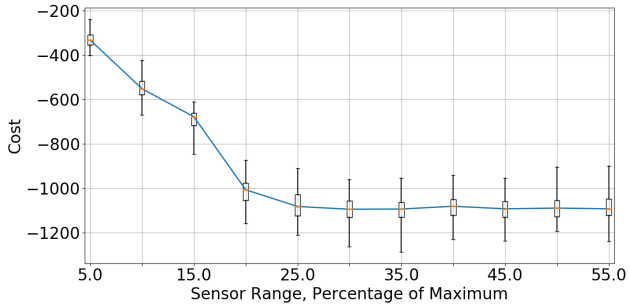

Fig. 11. Segregation is robust to small sensor beam ranges. The performance at 25% of maximum range is indistinguishable from infinite range.

the robots within some small angle in front of the robot and pick the closest one. This is very similar to what would be provided by a real-world camera that uses colored skirts on each robot and picks the largest blob as the robot to be detected. This sensor implementation works well and was used in all our experiments. However, we found later that, if the robots instead always prefer to react to kin over non-kin, clusters form more quickly and robustly. For example, if there are two robots within the field of view of a robot's sensor and the non-kin robot is closer, the robot would ignore it and execute the $S = 1$ logic, which drives the robot towards the farther kin robot. Exploring exactly which of the various implementation details have what effect on cost is left for future work.

D. The Effect of the Beam Angle

On a real robot, there must be some finite beam angle to the theoretically line-of-sight sensor. We ran 100 trials 180 seconds each in simulation with uniformly random initial distributions of 40 robots with various beam angles. Fig. 10 shows the results, along with a diagram showing how we define beam angle. The best beam angle we tested was 20°, and angles smaller or larger became progressively worse. We found that at lower beam angles, it was possible for a robot to become stuck in groups of two or three, and the robots spent all their time looking at each other instead of peeking around them to find kin. At larger angles, we suspect the behavior fails because larger beam angles cause the rings to enlarge faster. When the rings become too large and the space between robots exceeds $2r + \epsilon$ and are no longer a cluster as defined by (3).

E. The Effect of Beam Length

We consider what happens if the theoretically infinite-range sensor has finite range. We use 15° half beam angle and the same experimental setup as with the beam angle experiments. We consider the maximum range of the sensor as the diagonal length of the square in which the robot are initially distributed. In all our experiments, this square was $5\,\mathrm{m}$ on each side, so we consider a range of $7.07\,\mathrm{m}$ to be effectively unlimited. We report the costs for beam ranges as a fraction of this maximum range. As shown in Fig. 11, a beam range of 25% of the theoretical maximum performs just as well as an infinite sensor. Below this, the performance degrades.

VII. Conclusion

In this paper, we show how robots with only a ternary sensor and a controller which maps sensor readings to wheel speeds is capable of N-class segregation. This controller is invariant to the number of classes, and using the best found parameters we are able to construct theoretical guarantees on the emergent behavior in a simplified setting. We performed a grid search to learn about the full parameter space, and we investigated the effect of sensor implementation details and the number of robots and classes on performance. Our findings indicate that robust segregation with non-ideal sensors in reality is possible, although not guaranteed.

References

[1] 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.

[2] M. Gauci, J. Chen, W. Li, T. J. Dodd, and R. Groß "Clustering Objects with Robots That Do Not Compute," in *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-agent Systems*, ser. AAMAS '14. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2014, pp. 421–428. [Online]. Available: http://dl.acm.org/citation.cfm?id=2615731.2615800

[3] A. Bolger, M. Faulkner, D. Stein, L. White, and D. Rus, "Experiments in decentralized robot construction with tool delivery and assembly robots," in *2010 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS 2010)*. IEEE Press, 2010, pp. 5085–5092.

[4] N. E. Shlyakhov, I. V. Vatamaniuk, and A. L. Ronzhin, "Survey of Methods and Algorithms of Robot Swarm Aggregation," *Journal of Physics: Conference Series*, vol. 803, p. 012146, Jan. 2017. [Online]. Available: http://stacks.iop.org/1742-6596/803/i=1/a=012146?key=crossref.a3ceb0f2b8113f73c39aa6e56ddeeb8a

[5] C. Pinciroli, A. Gasparri, E. Garone, and G. Beltrame, "Decentralized progressive shape formation with robot swarms," in *13th International Symposium on Distributed Autonomous Robotic Systems (DARS2016)*, ser. Springer Proceedings in Advanced Robotics. Springer, November 2016, pp. 433–445.

[6] R. Groß and M. Dorigo, "Self-assembly at the macroscopic scale," *Proceedings of the IEEE*, vol. 96, no. 9, pp. 1490–1508, 2008.

[7] M. Johnson and D. Brown, "Evolving and Controlling Perimeter, Rendezvous, and Foraging Behaviors in a Computation-Free Robot Swarm," in *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (Formerly BIONETICS)*, ser. BICT'15. ICST, Brussels, Belgium, Belgium: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2016, pp. 311–314. [Online]. Available: http://dx.doi.org/10.4108/eai.3-12-2015.2262390

[8] D. S. Brown, R. Turner, O. Hennigh, and S. Loscalzo, "Discovery and Exploration of Novel Swarm Behaviors Given Limited Robot Capabilities," in *Distributed Autonomous Robotic Systems*, ser. Springer Proceedings in Advanced Robotics. Springer, Cham, 2018, pp. 447–460. [Online]. Available: https://link.springer.com.ezproxy.wpi.edu/chapter/10.1007/978-3-319-73008-0_31

[9] 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*, ser. Springer Tracts in Advanced Robotics. Springer, Berlin, Heidelberg, 2014, pp. 355–367. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-642-55146-8_25

[10] D. St-Onge, C. Pinciroli, and G. Beltrame, "Circle formation with computation-free robots shows emergent behavioral structure," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2018)*. Madrid, Spain: IEEE Press, October 2018, in press.

[11] E. Batlle and D. G. Wilkinson, "Molecular Mechanisms of Cell Segregation and Boundary Formation in Development and Tumorigenesis," *Cold Spring Harbor Perspectives in Biology*, vol. 4, no. 1, p. a008227, Jan. 2012. [Online]. Available: http://cshperspectives.cshlp.org/content/4/1/a008227

[12] M. S. Steinberg, "Reconstruction of tissues by dissociated cells," *Science*, vol. 141, pp. 401–408, 1963.

[13] N. N. Franks and A. Sendova-Franks, "Brood sorting by ants: distributing the workload over the work-surface," *Behavioral Ecology and Sociobiology*, vol. 30, no. 2, pp. 109–123, 1992.

[14] W. M. Spears, D. F. Spears, J. C. Hamann, and R. Heil, "Distributed, Physics-Based Control of Swarms of Vehicles," *Autonomous Robots*, vol. 17, no. 2/3, pp. 137–162, Sep. 2004.

[15] R. Groß S. Magnenat, and F. Mondada, "Segregation in swarms of mobile robots based on the Brazil nut effect," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct. 2009, pp. 4349–4356.

[16] J. Chen, M. Gauci, M. J. Price, and G. R, "Segregation in swarms of e-puck robots based on the brazil nut effect," in *Proc. of the 11th Int. Conf. on Auton. Agents and Multi-agent Syst.*, 2012, pp. 163–170.

[17] M. Kumar, D. P. Garg, and V. Kumar, "Segregation of Heterogeneous Units in a Swarm of Robotic Agents," *IEEE Transactions on Automatic Control*, vol. 55, no. 3, pp. 743–748, Mar. 2010.

[18] V. G. Santos, L. C. A. Pimenta, and L. Chaimowicz, "Segregation of multiple heterogeneous units in a robotic swarm," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 1112–1117.

[19] C. Pinciroli, V. Trianni, R. O'Grady, G. Pini, A. Brutschy, M. Brambilla, N. Mathews, E. Ferrante, G. D. Caro, F. Ducatelle, M. Birattari, L. M. Gambardella, and M. Dorigo, "ARGoS: a Modular, Parallel, Multi-Engine Simulator for Multi-Robot Systems," *Swarm Intelligence*, vol. 6, no. 4, pp. 271–295, 2012.

[20] M. Bonani, V. Longchamp, S. Magnenat, P. Rétornaz, D. Burnier, G. Roulet, F. Vaussard, H. Bleuler, and F. Mondada, "The marXbot, a miniature mobile robot opening new perspectives for the collective-robotic research," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Piscataway, NJ: IEEE Press, 2010, pp. 4187–4193.

[21] M. Rubenstein, C. Ahler, and R. Nagpal, "Kilobot: A low cost scalable robot system for collective behaviors," *2012 IEEE International Conference on Robotics and Automation*, pp. 3293–3298, May 2012.

[22] G. Dudek and M. Jenkin, *Computational Principles of Mobile Robotics*, 2nd ed. Cambridge University Press, 2010.