

Segregation without Computation

Peter Mitrano*, Jordan Burkland*, Michael Giancola*, Carlo Pinciroli*

*Worcester Polytechnic Institute, Worcester, Massachusetts

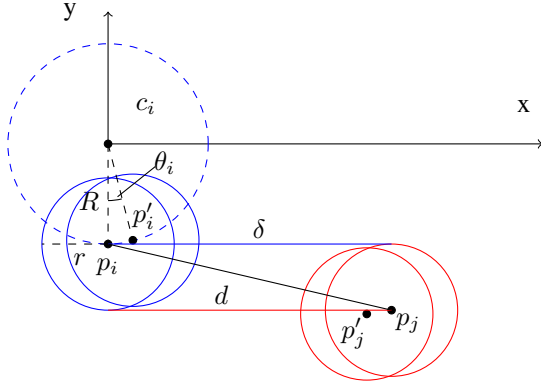


Figure 1. The configuration of a robot aggregating to another kin where both robots see each other. R is the radius of the robots path around the ICC.

I. CONTROLLER ANALYSIS

We found from grid search that the parameters $[1, -0.6667, 0.3333, 1.0, 1.0, 0.0]$ best achieve segregation as defined by our cluster metric cost function. The actual wheel speeds, which are scaled by 0.2 to convert to ms^{-1} are therefore $[0.2, -0.1333, 0.06667, 0.2, 0.2, 0.0]$. Now that we have found parameters that achieve what we consider segregation, we define the conditions under segregation is guaranteed. We first define the conditions under which aggregation between two isolated kin robots is guaranteed. Then we show that kin will aggregate with a static ring of kin. Finally, we show that non-kin will aggregate faster to kin than non-kin. All together, this explains much of the emergent behavior. More generally it demonstrates that we can prove useful things about complex emergent swarm behavior of these simple robots and controllers. In all of these proofs we will consider the controller found via grid search since it is the best controller. We also assume the controllers are in sync, that each motion is execute for the same Δt at the same point in time, and that the robot has a fixed positive inter wheel distance W and radius r .

A. Aggregation of Two Kin

The set of all possible initial configurations of two robots i and j can be split into three possible states. Either neither robot sees the other $(0, 0)$, one robot sees the other $(1, 0)$, or both robots see each other $(1, 1)$. In order to show that aggregation is guaranteed in all scenarios, we consider how these states evolve as a finite state machine (Figure I-A). For each sensor state $I = 0, 1, 2$ there is a corresponding radius of rotation around the ICC which we call R_0, R_1, R_2 respectively. For

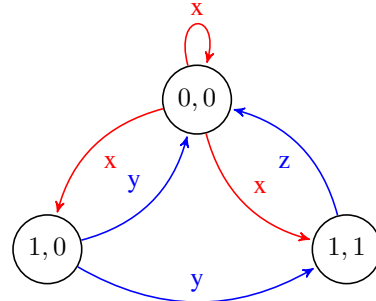


Figure 2. State machine representation for two kin aggregation. $(1, 0)$ means robot i has sensor state $I = 0$, and robot j has $I = 1$. Blue arrows indicate transitions where the robots aggregate, and red indicates they may not aggregate. Letters indicate identical transitions. One can see visually in Figure 1 that impossible to transition from $(1, 1)$ to $(1, 0)$ because both robots rotate away from each other.

each sensor state, There are three unique transitions between states in this state machine, which are called x , y and z . For example, from state $(0, 0)$ both robots will drive with speeds $v_{0l} = 0.2, v_{0r} = -0.1333$, and as a result either the robots will continue to see nothing and transition back to $(0, 0)$ or it could transition to the other two states.

We show that the distance by which the robots aggregate during the blue transitions y or z is always greater than the distance they could possibly separate in any sequence of x transitions. In other words, if we started at $(0, 0)$, we can continue to loop there but when we eventually transition to another state, and the again transition out of that state (now along a blue arrow), we prove that the robots must be closer at then end then when they started.

Intuitively, if the robots execute $(0, 0)$ repeatedly they simply spin in a small circle, and so they're dispersion is bounded. The upper bound on how much the robots disperse in transition x we call s , and it is equal to twice the radius of the curvature of the robots path R_0 . Equation (1) formally defines s .

$$s = 2R_0 = 2 \frac{W}{2} \left(\frac{v_{l1} + v_{r1}}{v_{l1} - v_{r1}} \right) = \frac{W}{4} \left(\frac{0.2 + -0.133}{0.2 - (-0.133)} \right) = 0.8W \quad (1)$$

We need to show that, in transitions y and z , the kin robots always aggregate more than s . We first demonstrate that the magnitude of aggregation, which can be described as $d' - d$ in Figure 1, always decreases as the sensor ray length δ decreases. In other words, the closer the robots are, the less by which they aggregate. The conditions under which this is true are

derived in Theorem 2. From this we know that if the robots aggregate by more than s at the smallest δ , then they must also aggregate for all larger δ . This second condition is derived in Theorem 3 and 4, and the final conditions are shown below. We know the lower bound of δ is $\sqrt{2}r$ because this is the point at which the two robots of radius r are touching.

$$\frac{1}{-W - r + W \cos(\theta_i)} \leq \sqrt{2} \quad (2)$$

$$\begin{aligned} & \sqrt{(\sqrt{2}r_j - W \sin(\theta_i))^2 + (-W - r_j + W \cos(\theta_i))^2} \\ & - \sqrt{3}r_j < \frac{\Delta t}{15W} + 1.6W \sin\left(\frac{\Delta t}{6W}\right) \end{aligned} \quad (3)$$

$$\begin{aligned} & \sqrt{(\sqrt{2}r_j - 2W \sin(\theta_i))^2 + (-r_j - 2W + 2W \cos(\theta_i))^2} \\ & - \sqrt{3}r_j < \frac{\Delta t}{15W} \end{aligned} \quad (4)$$

Ultimately, if Equations (2), (3), and (4) are satisfied, we can say that from any initial configuration, two kin robots will eventually aggregate in a sort of two-steps-forward one-step-back fashion. This behavior is clearly demonstrated in our simulation and real robot experiments.

B. Aggregation with Ring of Kin

In this scenario, imagine an isolated kin robot which has sensed a ring of kin. Imagine the ring of kin as robot j but with a different (larger) radius than the isolated robot i . If we consider the center of the ring static, we can define the conditions under which robot i is guaranteed to be closer to the center of the ring. Robot i starts at p_i with sensor reading $I = 1$. It then executes $v_{l1} = 0.06667$ and $v_{r1} = 0.2$ and arcs with radius R_1 to the left and ends up at position p'_i . We show that for a small Δt that the robot moves closer to the center of the ring at p_j .

The condition we need to satisfy is shown in Equation (5). The variable r_j now denotes the radius of this ring, whereas r still refers to the radius of a single robot.

$$\|p'_i - p_j\| < \|p_i - p_j\| \quad (5)$$

We can expand Equation (5) and derive a simple condition which describes for which values of r , r_j , W , Δt , this holds true for all δ . The full derivation can be found in Theorem 6, and the result is shown in Equation (6).

$$\begin{aligned} & \sqrt{(\sqrt{2}r_j - 2W \sin(\theta_i))^2 + (-r_j - 2W + 2W \cos(\theta_i))^2} \\ & - \sqrt{3}r_j < 0 \end{aligned} \quad (6)$$

Ultimately, we can use these geometric proof to make a useful assertions about the behavior of an individual in our swarm and of the swarm as a whole. While we have not shown that segregation is guaranteed in all cases, this may serve as a building block for more general claims.

Table I
AGGREGATION IS GUARANTEED FOR MANY DIFFERENTIAL DRIVE SWARM ROBOTS.

Robot	r (m)	W (m)	maximum r_j (m)
foot-bot ¹	0.085	0.14	74.619
Khepera IV ²	0.07	0.1054	34.725
E-Puck 2 ³	0.035	0.053	4.288
Kilobot ⁴	0.0165	0.030 ⁵	0.594

C. Segregation

So far all of the behaviors we have discussed are actually just aggregation behaviors. Here we partially explain why segregation occurs by showing that between two robots i and j , the kin robots aggregate faster than the non-kin robots. First, we need to explain why non-kin robots do in fact aggregate at all. It is counter intuitive that the optimal controller for segregation aggregates non-kin. The key is that they aggregate more slowly, so only in absense of kin do non-kin aggregate. In our supplementary videos we include an example of an environment with only non-kin to demonstrate this.

The conditions for non-kin aggregating follows the same steps as the proof for kin aggregating, except with different coordinates for p_i , p_j , and p'_i . The derivation is shown in Theorem 7. If the resulting conditions are met, then non-kin are guaranteed to aggregate to kin.

Next, to show that non-kin aggregate more slowly than kin, we show that the displacement during their kin step (the arc left with speeds 0.01666, 0.2) is greater than the displacement during their non-kin step (with speeds 0.2, 0.0).

Theorem 1. *The distance moved in the kin step is always greater than the distance moved in the non-kin step.*

Proof. We compare the arc lengths derived from the wheels speeds.

$$\begin{aligned} \theta_1 R_1 &> \theta_2 R_2 \\ \frac{2\Delta t}{15W} W &> \frac{2\Delta t}{10W} \frac{W}{2} \\ \frac{2}{15} &> \frac{1}{10} \end{aligned}$$

□

Note that we don't show that the kin step actually gets closer to the other robot, so in theory the longer kin step could aggregate less than the shorter non-kin step, but this at least partially justifies why kin aggregate faster.

D. Application to Real Swarm Robots

The above conditions do not hold true for all scenarios (such as those with large Δt), but we show that they do hold true for many popular robots in Table I.

¹ARGoS file footbot_entity.cpp

²Khepera IV User Manual, page 66.

³<http://projects.gctronic.com/epuck2/e-puck2-flyer.pdf>

II. EXPERIMENTAL RESULTS

A. Evaluating the centroid-of-centroids cost function

When we implemented the centroid-of-centroids style cost function, we quickly found examples of configurations which were ranked undesirably. One example is shown below in Figure 3. The behavior that looks like aggregation was given lower cost of $-8 \cdot 10^9$, versus the behavior that looks like segregation was given a cost of $-5 \cdot 10^9$.

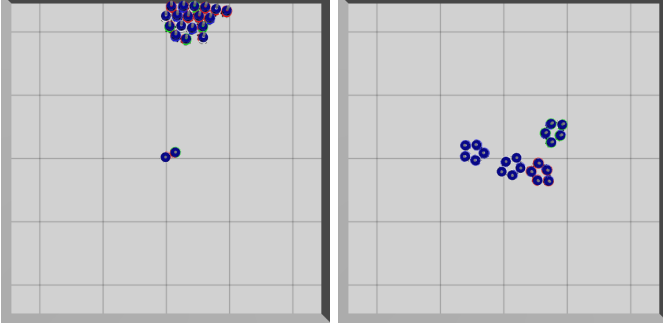


Figure 3. The left picture was given lower cost than the right, which is not desirable.

B. Scalability Study

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

a) *Fixed number of robots per class:* Because our cost function is independent of the number of classes, we first considered having 10 robots for each class. However, this also means that in the trial with 25 classes there are 25 times the total number of robots than in the 1-class trial. This means that occlusion of robots is more likely and so the cost increases with more classes. The results of this are plotted in Figure 5.

b) *Fixed total number of robots:* Another scenario is to consider a fixed number of robots and split them into more and more classes. We choose 100 robots because we are still able to get 4 robots per class at 25 classes. As you can see in Figure 4, the cost no longer increases as the number of classes increases. However, the cost for just a few classes is much higher. This is expected, because when there are many robots of a single class, our controller forms very large sparse rings where robots are too far apart to be considered clustered.

Ultimately, we can say that our controller scales well to many classes with few robots, but not well to many robots of the same class.

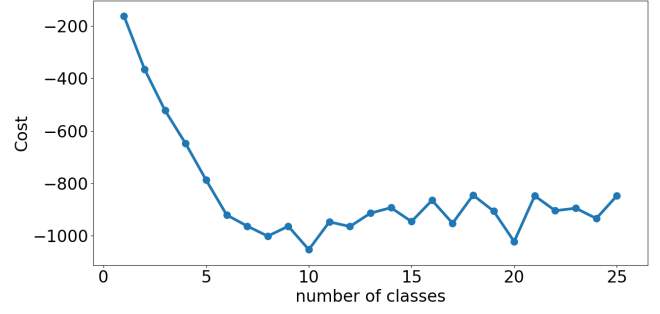


Figure 4. The average cost with 100 robots divided into N classes. More classes are lower cost partially because kin robots stay close enough to be considered clusters.

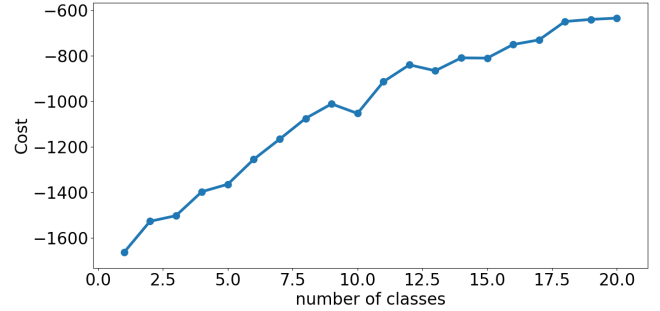


Figure 5. The average cost with N classes, 10 robots per class. More classes are higher cost because other robots obstruct your view making it difficult to find kin.

C. The Effect of Implementation Details of the sensor

We found in our experiments that the implementation details of the line-of-sight sensor have a significant effect on the behavior of the controller.

Initially, our method for determining sensor state from our simulated range-and-bearing sensors was to consider all 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 genetic algorithm and grid search experiments. However, we found later that if the robots instead always prefer to react to kin over non-kin, you can form larger rings more quickly and robustly. For example, if there are two robots within the field of view of your sensor and the non-kin robot is closer, you will ignore it and execute the $I = 1$ state which will drive you towards the farther away kin robot. Exploring exactly which of these 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 in simulation with uniformly random initial distributions of 40

⁴<https://www.k-team.com/mobile-robotics-products/kilobot/specifications>

⁵only approximate

robots with various half beam angles. Figure 6 shows the results, as well as a diagram showing how we define half beam angle. The best half beam angle we tested was 15° , 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 where the robots spent all their time looking at each other and not peeking around them to find kin. At larger angles, we believe the behavior fails because larger beam angles cause the rings to enlarge faster, which in turn causes the rings to be so large that they are not considered a cluster anymore and so the cost rises.

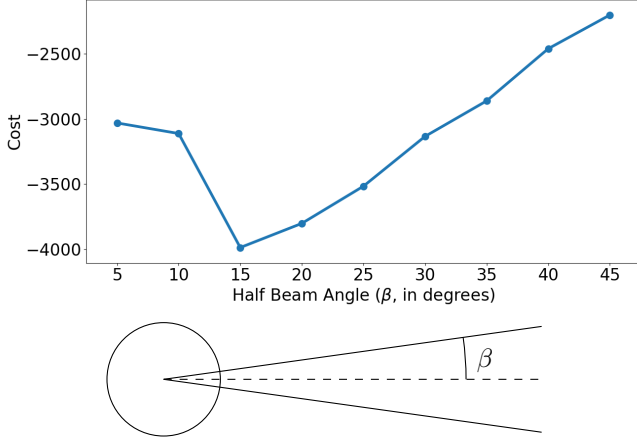


Figure 6. A 15° degree half beam angle is best for segregation. Lower cost is better.

E. The Effect of Beam Length

We also consider what happens if our theoretically infinite sensor now has finite range. We use 15° half beam angle and the same experimental setup as with the beam angle experiments. Analogously to [?], 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 box was 5m, so we consider a range of 7.07m to be effectively unlimited. We report the costs for beam lengths as a fraction of this maximum range. As shown in Figure 7, a beam length of 35% of the theoretical maximum performs just as well. Below this, the performance degrades. However, even a beam length of 7% of unlimited is more effective than zero length at segregation. In our experiment watching many simulations, beam length should also be scaled with the total number of robots as well as the initial distance between robots. When the beam length is short, there are many cases where two rings of kin robots form at different points in the environment, and in order for these two rings to join the robots must have sufficiently long beam length in order to detect each other.

III. CONCLUSION

In this paper, we demonstrate that memoryless, computeless robots are capable of n -class segregation. We use a simple controller design consisting of a 6-tuple. This controller is invariant to the number of classes, so any given controller can

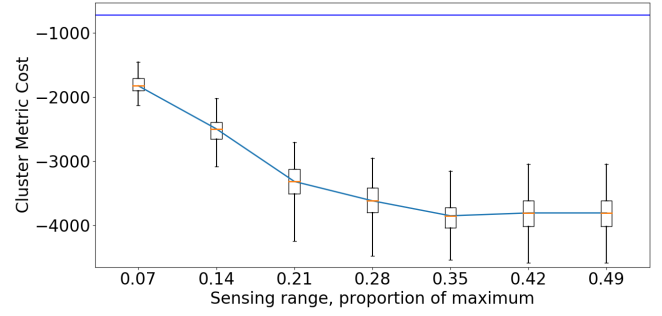


Figure 7. Segregation is robust to very finite sensor beam lengths. The blue bar indicates the worst (highest) attainable cost in which all robots are isolated from any kin.

work for any number of classes. To quickly find a quality controller, we evolved one using a basic genetic algorithm, but we also performed a grid search to make claims about the full parameter space. We investigated the effect of sensor implementation details and the number of robots and classes on performance. We find that robust segregation is possible, although not guaranteed. Instead, we prove that aggregation of kin robots is guaranteed when reasonable conditions on controller frequency and inner wheel distance are met.

APPENDIX

$$\begin{aligned}\theta_i &= \Delta t \omega = \Delta t \frac{v_{r_1} - v_{l_1}}{W} = \Delta t \frac{0.2 - 0.06667}{W} = \frac{2\Delta t}{15W} \\ R_1 &= \frac{W}{2} \left(\frac{v_{r_1} + v_{l_1}}{v_{r_1} - v_{l_1}} \right) = \frac{W}{2} \left(\frac{0.2 + 0.06667}{0.2 - 0.06667} \right) = W\end{aligned}\tag{7}$$

From Figure 1 we can define the coordinates of p_i , p_j , p'_i , and p'_j .

$$\begin{aligned}p_i &= \begin{bmatrix} 0 \\ -R_1 \end{bmatrix} \\ p_j &= \begin{bmatrix} \delta \\ -(R_1 + r_j) \end{bmatrix} \\ p'_i &= \begin{bmatrix} R_1 \sin(\theta_i) \\ -R_1 \cos(\theta_i) \end{bmatrix} \\ p'_j &= \begin{bmatrix} \delta - R_1 \sin(\theta_i) \\ -(R_1 + r) - (R_1 - R_1 \cos(\theta_i)) \end{bmatrix}\end{aligned}\tag{8}$$

Theorem 2. *Robots aggregate by less the closer they are*

Proof. We show that $d' - d$ always decreases as δ decreases (until its minimum of $\sqrt{2}r$). This is true if the derivative $\frac{\partial}{\partial \delta}(d' - d) \leq 0$, $\forall \delta > \sqrt{2}r$.

$$\begin{aligned}
& \frac{\partial}{\partial \delta} (d' - d) < 0 \\
& \frac{\partial}{\partial \delta} (\|p_j - p'_i\| - \|p_j - p_i\|) \leq 0 \\
& \frac{\partial}{\partial \delta} \left(\sqrt{(p_{j_x} - p'_{i_x})^2 + (p_{j_y} - p'_{i_y})^2} - \sqrt{(p_{j_x} - p_{i_x})^2 + (p_{j_y} - p_{i_y})^2} \right) \leq 0 \\
& \frac{\partial}{\partial \delta} \left(\sqrt{(\delta - b)^2 + a} - \sqrt{(\delta - 0)^2 + (p_{j_y} - p_{i_y})^2} \right) \leq 0 \\
& \frac{\partial}{\partial \delta} \left(\sqrt{(\delta - b)^2 + a} - \sqrt{\delta^2 + (p_{j_y} - p_{i_y})^2} \right) \leq 0 \\
& \frac{\partial}{\partial \delta} \left(\sqrt{(\delta - b)^2 + a} - \sqrt{\delta^2 + (-(R_1 + r) - R_1)^2} \right) \leq 0 \\
& \frac{\partial}{\partial \delta} \left(\sqrt{(\delta - b)^2 + a} - \sqrt{\delta^2 + r^2} \right) \leq 0 \\
& \frac{1}{2} \left[((\delta - b)^2 + a)^{-\frac{1}{2}} (2\delta - 2b) - (\delta^2 + r^2)^{-\frac{1}{2}} (2\delta) \right] \leq 0 \\
& ((\delta - b)^2 + a)^{-\frac{1}{2}} (\delta - b) - (\delta^2 + r^2)^{-\frac{1}{2}} (\delta) \leq 0 \\
& ((\delta - b)^2 + a)^{-\frac{1}{2}} (\delta - b) \leq (\delta^2 + r^2)^{-\frac{1}{2}} (\delta) \\
& (\delta^2 + r^2)^{\frac{1}{2}} (\delta - b) \leq (\delta) ((\delta - b)^2 + a)^{\frac{1}{2}} \quad \text{all terms are positive, see Theorem 5} \\
& (\delta^2 + r^2)(\delta - b)^2 \leq (\delta)^2 ((\delta - b)^2 + a) \quad \text{squaring preserves inequality for positive terms} \\
& \cancel{\delta^2(\delta - b)^2} + r^2(\delta - b)^2 \leq \cancel{\delta^2(\delta - b)^2} + a\delta^2 \\
& r^2(\delta - b)^2 \leq a\delta^2 \\
& r^2(\delta^2 - 2b\delta + b^2) \leq a\delta^2 \\
& r^2\delta^2 - r^22b\delta + r^2b^2 \leq a\delta^2 \\
& (r^2 - a)\delta^2 - r^22b\delta + r^2b^2 \leq 0 \quad \text{If parabola opens down} \\
& \frac{\partial}{\partial \delta} ((r^2 - a)\delta^2 - r^22b\delta + r^2b^2) \leq 0 \\
& 2(r^2 - a)\delta - r^22b \leq 0 \\
& 2(r^2 - a) \leq 0 \\
& \text{where } a = (-R_1 - r - R_1 \cos(\theta_i)), b = -R_1 \cos(\theta_i)
\end{aligned}$$

If $2(r^2 - a) \leq 0$ and $(r^2 - a)2r^2 - r^22b\sqrt{2}r + r^2b^2 \leq 0$, then the change in distance $d' - d$ strictly decreases as δ increases. In other words, the amount that the two robots get closer is smallest when they are as close as possible. Because of this property, we can guarantee aggregation by showing that $d' - d < s$ when d is at its minimum of $\sqrt{2}r$. In other words, if these two simple analytic conditions are satisfied, then aggregation between two kin robots is guaranteed. For convenience, we show the expanded forms of $d' - d$ for the two possible cases. To apply this, substitute the parameters of your robot into the two equations below and see if the conditions hold true. □

Theorem 3. $d' - d < s$ in state $(1, 0)$ when one robot sees the other.

Proof. In this state, robot i will arc left with speeds 0.06667, 0.2 and robot j will rotate right with speeds 0.2, -0.1333. Robot i advances towards j along a slight arc, whereas j rotates mostly in place. Because j is not stationary, there is an additional amount which it could move away from i , so in addition to showing that $d' - d < s$ we must add another approximate value to s which we'll call s' . This maximum distance j can move in one time step is the length of the chord defined by the radius R_0 (radius of turn for sensor $I = 0$) and angle θ_j . Equation (9) shows the full definition of this bound.

$$\begin{aligned}
\theta_j &= \Delta t \omega = \Delta t \frac{v_{l_1} - v_{r_1}}{W} = \Delta t \frac{0.2 - (-0.1333)}{W} = \frac{\Delta t}{3W} \\
s' &= 2R_0 \sin\left(\frac{\theta_j}{2}\right) = 2(0.8W) \sin\left(\frac{\Delta t}{6W}\right) = 1.6W \sin\left(\frac{\Delta t}{6W}\right)
\end{aligned} \tag{9}$$

$$\begin{aligned}
d' - d &< s + s' \\
\|p_j - p'_i\| - \|p_j - p_i\| &< s + s' \\
\sqrt{(p_{j_x} - p'_{i_x})^2 + (p_{j_y} - p'_{i_y})^2} - \sqrt{(p_{j_x} - p_{i_x})^2 + (p_{j_y} - p_{i_y})^2} &< s + s' \\
\sqrt{(\sqrt{2}r_j - R_1 \sin(\theta_i))^2 + (-(R_1 + r_j) - (-R_1 \cos(\theta_i)))^2} - \sqrt{(\sqrt{2}r_j - 0)^2 + (-(R_1 + r_j) - (-R_1))^2} &< s + s' \\
\sqrt{(\sqrt{2}r_j - R_1 \sin(\theta_i))^2 + (-R_1 - r_j + R_1 \cos(\theta_i))^2} - \sqrt{2r_j^2 + r_j^2} &< s + s' \\
\sqrt{(\sqrt{2}r_j - W \sin(\theta_i))^2 + (-W - r_j + W \cos(\theta_i))^2} - \sqrt{3}r_j &< \frac{\Delta t}{15W} + 1.6W \sin\left(\frac{\Delta t}{6W}\right)
\end{aligned}$$

□

Theorem 4. $d' - d < s$ in state $(1, 1)$ when both robots see each other.

Proof.

$$\begin{aligned}
d' - d &< s \\
\|p'_j - p'_i\| - \|p_j - p_i\| &< s \\
\sqrt{(p'_{j_x} - p'_{i_x})^2 + (p'_{j_y} - p'_{i_y})^2} - \sqrt{(p_{j_x} - p_{i_x})^2 + (p_{j_y} - p_{i_y})^2} &< s \\
\sqrt{((\sqrt{2}r_j - 2R_1 \sin(\theta_i))^2 + ((-(R_1 + r_j) - (R_1 - R_1 \cos(\theta_i))) - (-R_1 \cos(\theta_i)))^2} - \sqrt{2r_j^2 + (-(R_1 + r_j) - (-R_1))^2} &< s \\
\sqrt{(\sqrt{2}r_j - 2R_1 \sin(\theta_i))^2 + (-r_j - 2R_1 + 2R_1 \cos(\theta_i))^2} - \sqrt{2r_j^2 + r_j^2} &< s \\
\sqrt{(\sqrt{2}r_j - 2W \sin(\theta_i))^2 + (-r_j - 2W + 2W \cos(\theta_i))^2} - \sqrt{3}r_j &< \frac{\Delta t}{15W}
\end{aligned}$$

□

Theorem 5. *Proof that $(\delta - b) > 0$*

Proof.

$$\begin{aligned}
\delta - b &> 0 \\
\delta - W \sin(\theta_i) &> 0 \\
\sqrt{2}r - W \sin(\theta_i) &> 0 \\
\sqrt{2}r - W \sin(\theta_i) &> 0 \\
\frac{\sqrt{2}}{2}W - W \sin(\theta_i) &> 0 \\
\frac{\sqrt{2}}{2} - \sin(\theta_i) &> 0 \\
\frac{\sqrt{2}}{2} &> \sin(\theta_i) \\
\frac{\pi}{4} &> \theta_i
\end{aligned}$$

lower bound of δ

The robot must be wider than its track width, $r \geq \frac{W}{2}$

□

Theorem 6. *An isolated kin robot will aggregate to a ring of kin.*

Proof.

□

Theorem 7. *Robots aggregate by less the closer they are, in the non-kin case.*

Proof. We show that $d' - d$ always decreases as δ decreases (until its minimum of $\sqrt{2}r$). This is true if the derivative $\frac{\partial}{\partial \delta}(d' - d) \leq 0, \forall \delta > \sqrt{2}r$.

$$\begin{aligned}
& \frac{\partial}{\partial \delta}(d' - d) < 0 \\
& \frac{\partial}{\partial \delta}(\|p_j - p'_i\| - \|p_j - p_i\|) \leq 0 \\
& \frac{\partial}{\partial \delta} \left(\sqrt{(p_{j_x} - p'_{i_x})^2 + (p_{j_y} - p'_{i_y})^2} - \sqrt{(p_{j_x} - p_{i_x})^2 + (p_{j_y} - p_{i_y})^2} \right) \leq 0 \\
& \frac{\partial}{\partial \delta} \left(\sqrt{(\delta - b)^2 + a} - \sqrt{(\delta - 0)^2 + (p_{j_y} - p_{i_y})^2} \right) \leq 0 \\
& \frac{\partial}{\partial \delta} \left(\sqrt{(\delta - b)^2 + a} - \sqrt{\delta^2 + (p_{j_y} - p_{i_y})^2} \right) \leq 0 \\
& \frac{\partial}{\partial \delta} \left(\sqrt{(\delta - b)^2 + a} - \sqrt{\delta^2 + (r - 0)^2} \right) \leq 0 \\
& \frac{\partial}{\partial \delta} \left(\sqrt{(\delta - b)^2 + a} - \sqrt{\delta^2 + r^2} \right) \leq 0 \\
& \frac{1}{2} \left[((\delta - b)^2 + a)^{-\frac{1}{2}}(2\delta - 2b) - (\delta^2 + r^2)^{-\frac{1}{2}}(2\delta) \right] \leq 0 \\
& ((\delta - b)^2 + a)^{-\frac{1}{2}}(\delta - b) - (\delta^2 + r^2)^{-\frac{1}{2}}(\delta) \leq 0 \\
& ((\delta - b)^2 + a)^{-\frac{1}{2}}(\delta - b) \leq (\delta^2 + r^2)^{-\frac{1}{2}}(\delta) \\
& (\delta^2 + r^2)^{\frac{1}{2}}(\delta - b) \leq (\delta)((\delta - b)^2 + a)^{\frac{1}{2}} \quad \text{all terms are positive, see Theorem 5} \\
& (\delta^2 + r^2)(\delta - b)^2 \leq (\delta)^2((\delta - b)^2 + a) \quad \text{squaring preserves inequality for positive terms} \\
& \cancel{\delta^2(\delta - b)^2} + r^2(\delta - b)^2 \leq \cancel{\delta^2(\delta - b)^2} + a\delta^2 \\
& r^2(\delta - b)^2 \leq a\delta^2 \\
& r^2(\delta^2 - 2b\delta + b^2) \leq a\delta^2 \\
& r^2\delta^2 - r^22b\delta + r^2b^2 \leq a\delta^2 \\
& (r^2 - a)\delta^2 - r^22b\delta + r^2b^2 \leq 0 \quad \text{If parabola opens down} \\
& \frac{\partial}{\partial \delta}((r^2 - a)\delta^2 - r^22b\delta + r^2b^2) \leq 0 \\
& 2(r^2 - a)\delta - r^22b \leq 0 \\
& 2(r^2 - a) \leq 0 \\
& \text{where } a = (-R_1 - r - R_1 \cos(\theta_i)), b = -R_1 \cos(\theta_i)
\end{aligned}$$

If $2(r^2 - a) \leq 0$ and $(r^2 - a)2r^2 - r^22b\sqrt{2}r + r^2b^2 \leq 0$, then the change in distance $d' - d$ strictly decreases as δ increases. \square