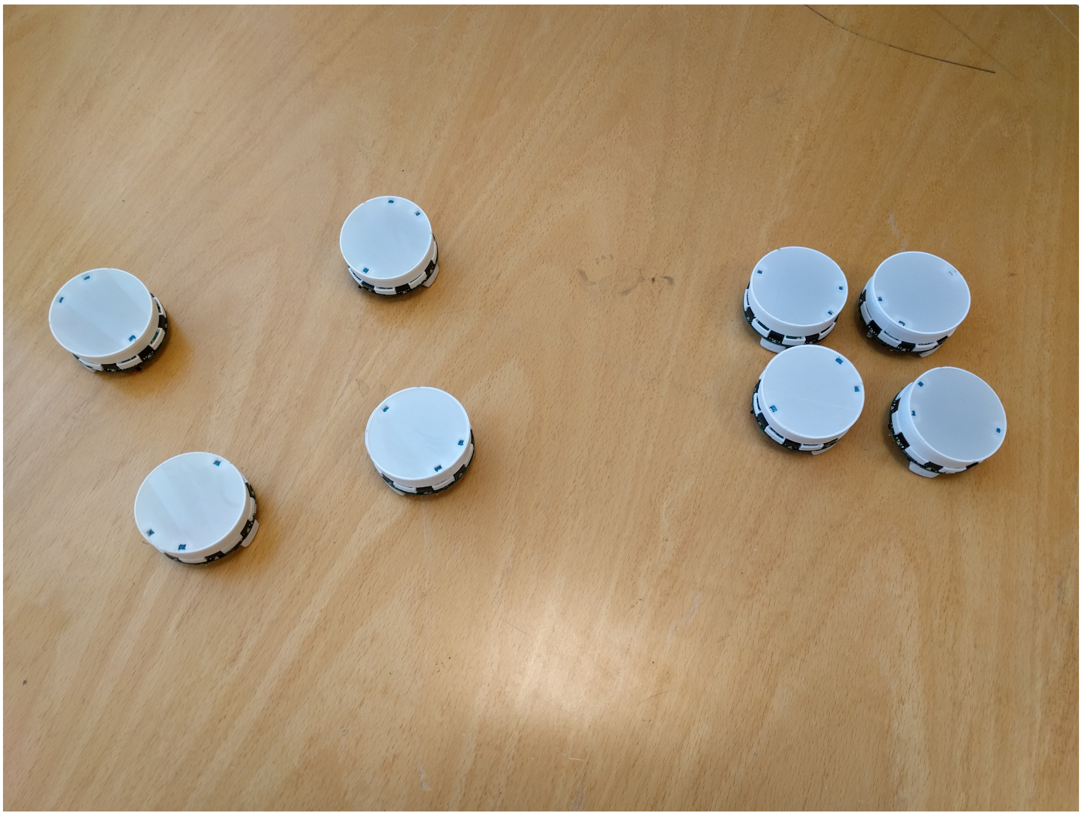
CHALMERS
UNIVERSITY OF TECHNOLOGY



Cooperative Robotics with Sensorial Delay

Master's thesis in Complex Adaptive Systems

FREDDIE OGEMARK
MAXIMILIAN LEYMAN

Department of Physics
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2018

MASTER'S THESIS IN COMPLEX ADAPTIVE SYSTEMS

Cooperative Robotics with Sensorial Delay

FREDDIE OGEMARK
MAXIMILIAN LEYMAN

Department of Physics
Soft Matter Lab
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2018

Cooperative Robotics with Sensorial Delay
FREDDIE OGEMARK
MAXIMILIAN LEYMAN

Supervisor: Giovanni Volpe, Department of Physics
Examiner: Giovanni Volpe, Department of Physics

Master's Thesis 2017:NN
Department of Physics
Soft Matter Lab
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: The Elisa 3 robots used in the study

Gothenburg, Sweden 2018

Abstract

The purpose of this work is to study how the behavior of robots changes when the data from their sensors is affected by a delay. Robots of the model Elisa 3 are therefore studied while performing Brownian motion and with certain parameters varying as a function of the intensity measured by their sensors. Introducing a delay and varying its sign is shown to have a significant effect on a robot's behavior. A single robot moving in an intensity field is either drawn to or avoiding higher intensities for a positive or a negative delay respectively. In this case experimental data shows good agreement with simulated behavior. Simulations also show that multiple robots should form clusters when interacting under the influence of a positive delay but the tendencies towards clustering that can be seen in the experiments are weaker. An increased detection range for the robots' sensors is proposed as a future improvement.

Keywords: autonomous robots, sensorial delay, swarming, clustering, SwisTrack, Brownian motion

Acknowledgements

We would like to thank Giovanni Volpe for guiding and helping us through our work, Jan Wehr for providing and increasing our knowledge about the theory.

FREDDIE OGEMARK, MAXIMILIAN LEYMAN, Gothenburg, June 2018

Contents

List of Figures

1

Introduction

1.1 Swarm behavior of agents interacting with each other

There are many examples in both nature and society of multiple agents interacting to produce emergent behaviors. Bacteria moving in groups [1], birds moving in a flock [2], people in crowds [3] or ants collecting food [4]. The collective behaviors produced can be very complex and often have properties that the individual agents do not have. However, these systems are all constituted by simple agents reacting to local information and this simplicity makes them both robust against changes and easy to program. Methods based on these collective behaviors are therefore being used in the field of autonomous robotics in order to make robots cooperate efficiently. Examples of potential application areas include disaster rescue [5] and cooperative transportation [6]. A relatively new way of making robots cooperate is to make use of a delay in the robots' sensors. Earlier work has shown how robots, performing Brownian motion and changing their speed in relation to a measured light intensity, have their behavior significantly changed when a delay is introduced [7]. A positive delay, corresponding to a delay in the time it takes to react to sensorial input, makes the robot stay near the intensity source. A negative delay, corresponding to a prediction of future inputs, makes the robot avoid the intensity source instead. Such a delay could therefore be used to promote clustering or segregation among interacting robots for positive and negative delays respectively. In our work we will continue studying how the behavior of robots performing Brownian motion is changed after introducing a delay in the system. Apart from robots whose speed vary as a function of the intensity we will also study how a varying rotational diffusion affects their behavior and later how the behavior is affected when both quantities vary at the same time.

1.2 Introducing the Elisa 3 robots

Autonomous robots of the type Elisa 3 are used for the experiments [8]. These are small robots that have a circular shape, measure 50 mm in diameter, 30 mm in height and weigh 39 g. They have one wheel on either side and a DC motor connected to each wheel with a 25:1 reduction gear. The robots have a 40.8 mm distance between both wheels and the wheels themselves have a diameter of 9 mm. In Figure 1.1 the Elisa 3 robot can be observed.

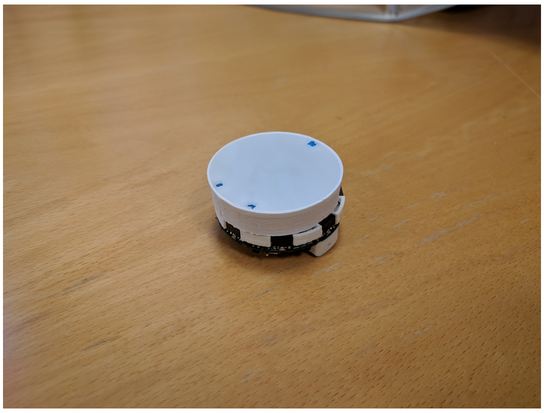


Figure 1.1: Elisa 3 robot

The maximum speed that can be reached is 60 cm/s. They are equipped with eight IR sensors that measure ambient light and the proximity of objects up to a distance of 6 cm from them. These eight IR sensors are pointing outwards from the robots with 45 degrees between them to create a detection field of 360 degrees altogether. Underneath the robots there are four ground sensors that can detect the proximity of the ground and the ambient light to allow the robots to detect if it is going towards a cliff or if it is in an area with IR light on the ground. These four ground sensors are placed in the front of the robot.
In Figure 1.2 the eight IR sensors and the four ground sensors can be highlighted.

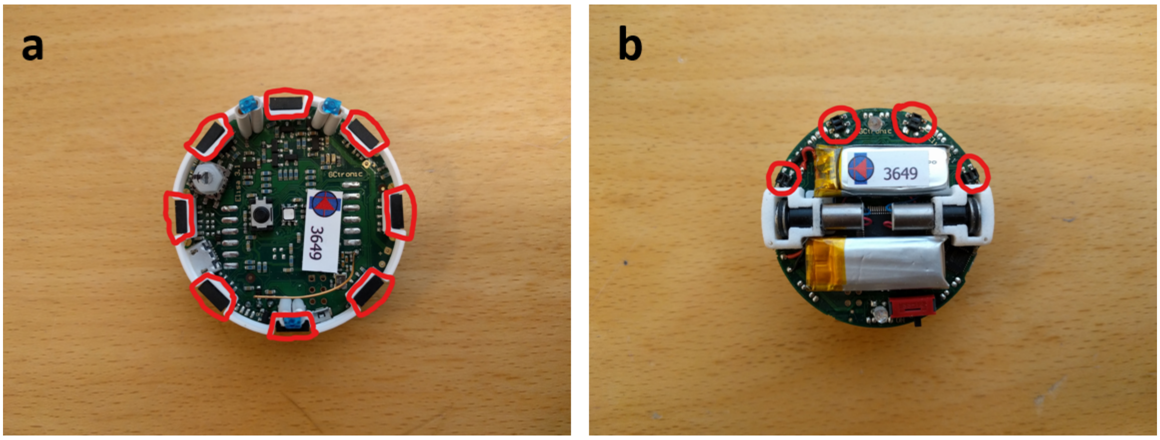


Figure 1.2: Image of the robots IR sensors and ground sensors. Figure **a** shows a view from the top of the robot without the white cover on where the IR sensors are marked with red. Figure **b** shows a view from the bottom of the robot and the ground sensors are marked with red.

To make the robots easier to track and allow them to communicate with other robots, they are equipped with three IR emitters with two being in the front and one being in the back. These IR emitters are capable of lighting up with IR light so that an IR camera can track, send and receive information from other robots. To allow them to know where they are located, they have an accelerometer that measures accelerations along the X, Y and Z axis.
There is an RGB LED in the center of the robots that allows them to change colors.

2

Methods

2.1 Simulations

Before implementing any behaviors on the robots themselves, these behaviors were tested using simulations in MATLAB. Besides finding relevant parameters for the robots, the simulations were also used to compare how well the robots followed the intended behaviors.

2.1.1 Brownian Motion

In the simulations the robots were modeled as agents moving in a 2D-plane with periodic boundary conditions. The basic behavior that they should perform in the absence of any external information was Brownian motion according to the following equations [9]:

$$\begin{aligned} x_i &= x_{i-1} + v\cos\phi_{i-1}\Delta t \\ y_i &= y_{i-1} + v\sin\phi_{i-1}\Delta t \\ \phi_i &= \phi_{i-1} + \sqrt{2D_R\Delta t}\eta_i \end{aligned} \tag{2.1}$$

Where x_i, y_i is the agent's location at time step i, ϕ_i is its orientation in the xy-plane in radians, v is its speed, Δt is the size of the time step, D_R is the rotational diffusion coefficient and η_i is a normally distributed white noise term. Three different cases were then tested in the simulations and later on the robots themselves: one in which the speed varies as a function of the intensity, one in which the diffusion coefficient varies and one in which both these quantities vary at the same time. Measuring an intensity I the speed changed according to the following equations:

$$v(I) = v_{min} + (v_{max} - v_{min})e^{-I} \tag{2.2}$$

Whereas the rotational diffusion coefficient varied in the following way:

$$D_R = D_{Rmax} - (D_{Rmax} - D_{Rmin})e^{-I} \tag{2.3}$$

The effect of varying these quantities can be seen in figures 2.1 a and b for the speed and the diffusion coefficient respectively. Both of these dependencies have the effect that the agent changes direction more often in regions where the intensity is high. When the rotational diffusion coefficient is higher the agent will rotate more and similarly, if the speed is lower then the white noise will have more time to affect its direction.

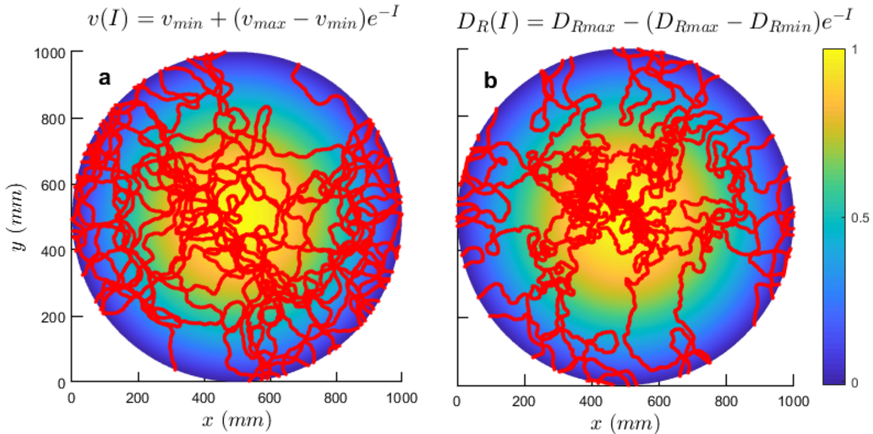


Figure 2.1: The effect of the intensity on an agent's behavior. In figure **a** the speed of an agent varies as a function of the intensity while figure **b** shows the trajectory of an agent whose rotational diffusion coefficient varies instead. In each case the agent is moving in the presence of a Gaussian intensity distribution, where the intensity is highest in the center. A video of these two simulations can be found at this link and in video 2 in appendix b.

2.1.2 Sensorial Delay

A sensorial delay δ was later introduced so that the agent would make an extrapolation based on earlier measured intensities rather than simply reacting to the current intensity. It would therefore base its behavior on the quantity $I(t-\delta)$ [7]. To calculate the extrapolated intensity an expansion is made to the first order:

$$I(t-\delta) = I(t) - \delta I'(t) = I(t) - \delta \frac{I(t) - I(t-\Delta t)}{\Delta t} \tag{2.4}$$

The effect of this delay on an agent's behavior varies depending on the sign of δ as seen in figures 2.2. Here, as in the rest of our work, the value of the delay will be expressed in terms of the characteristic reorientation time of the agent, τ. After this time has passed the standard deviation of the agent's rotation is one radian. In the cases where the rotational diffusion coefficient is constant it is set as the inverse of the reorientation time, $D_R = \tau^{-1}$. Henceforth we will also use the subscripts v and R to indicate whether it is the speed or the rotational diffusion that varies as a function of the intensity.

If the delay is positive, as in figure 2.2 a, the extrapolated intensity is highest when the agent is about to leave a region where the intensity is high. As mentioned above, a higher intensity means that the agent is more likely to change direction. This means that it will be likely to turn back and stay in the region with a high intensity if its delay is positive. Similarly, if the delay is negative, as in figure b, the extrapolated intensity will be highest when the actual one is increasing. Because of this the agent is likely to turn around when it is approaching a region with high intensity.

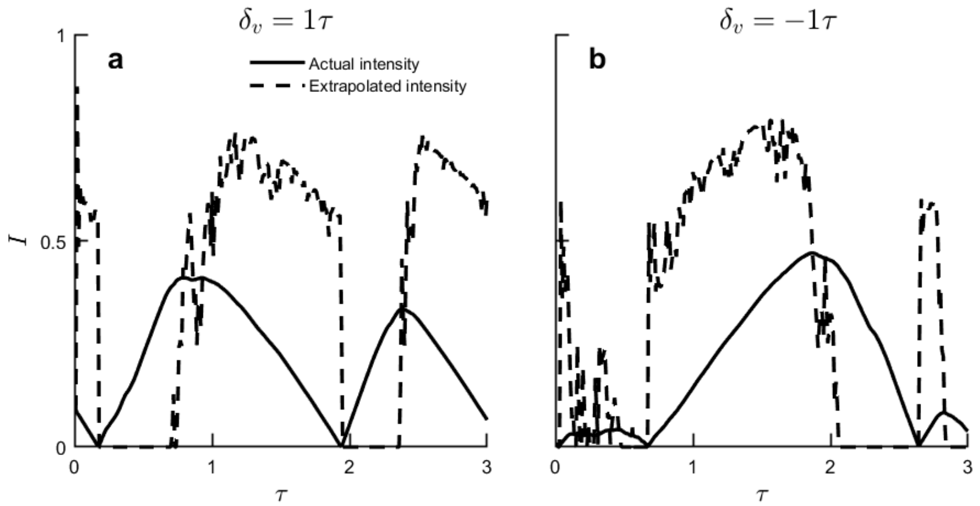


Figure 2.2: Comparison of the actual and the extrapolated intensities measured by a simulated agent. Figure **a** shows these intensities for an agent with a positive delay while figure **b** shows these intensities for a negative delay.

Figure 2.3 shows two examples of a trajectory of a simulated agent whose rotational diffusion coefficient varies as a function of the intensity. In figure 2.3 a, where the delay is positive, the agent gets stuck where the intensity is high and stays there. Whenever it is about to move out its rotational diffusion coefficient increases due to the delay and makes it turn back. The opposite then happens in figure 2.3 b where the coefficient increases as the agent approaches the center where the intensity is high. As a result the agent stays mostly near the edges in this case.

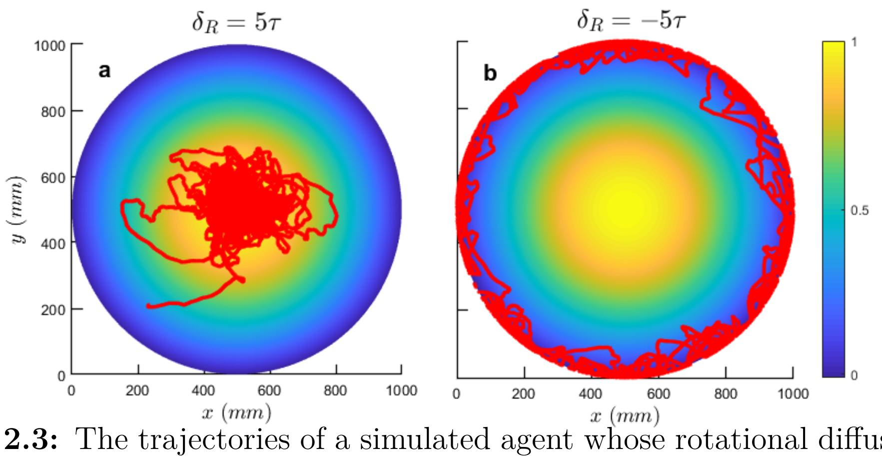


Figure 2.3: The trajectories of a simulated agent whose rotational diffusion coefficient varies as a function of the intensity. In figure **a** the agent has a positive delay whereas in figure **b** it has a negative delay. A video of these two simulations can be found at this link and in video 3 in appendix b.

2.1.3 Multiple Agents

Later simulations were performed with multiple agents interacting with each other instead of a static intensity source. In order to be a valid model for the robots later on each agent was equipped with a detection radius R within which it calculated the intensity as a linear function of the distance to all nearby agents. For an agent i the intensity was therefore calculated as:

$$I_i = \sum_{r_n \leq R, i \neq n} (1 - |r_n - r_i|/R) \tag{2.5}$$

2.2 Experiments

In the following sections the methods and settings used when performing the experiments will be described. First a method for making a robot perform Brownian motion will be described before moving on to the single- and multiple robot cases. Since the methods and settings used differ between these two cases they will be presented in separate subsections.

2.2.1 Brownian motion in robots

In order to get an understanding of what the robots are capable of doing they were tested using a software called Aseba studio [10]. This software allowed the user to program the robot and check all the sensor values in real time to see if any of the sensors were acting out of the ordinary. The testing was done by running some example programs where it was clear what the robots were supposed to do and which sensor values they were supposed to receive. After seeing that the robots behaved properly, the robots were given custom-made programs to check each of the sensors and learn how to program them in Aseba studio. Once it was clear how to program them, the task was to come up with a way to make the robots perform the Brownian motion.
Here they were programmed with two separate phases. In the first phase, called "Forward phase", the robot calculates a value for the speed and then moves forward. After moving forward it stops before going into the second phase, called "Rotation phase", during which the robot calculates a new direction and performs the rotation. After the robot has rotated in the desired direction, it stops before switching to "Forward phase" again.

2.2.2 Single Robot

2.2.2.1 Arena and Tracking

The arena was a circular area with a diameter of 120 cm and whose circumference was constructed with black tape. Black tape was used because the robots could interpret the black tape as a cliff and would therefore stay inside of the circle. The IR lamp was placed 50 cm above the center of the arena to create an intensity field on the ground. It was later discovered that there was a circular area of weaker

intensity in the middle of the arena. For this reason, a circular object with a 15 cm diameter was placed in the middle to prevent the robot from going into the low-intensity circle.
To gather data from the trajectory, the robot needed to be tracked during the experiments. An RGB camera was therefore placed at an angle to capture the robots without being blocked by or blocking the IR lamp. The camera was used together with a software called SwisTrack [11] that allows the camera to track the robot's trajectory and save the data as a text file to later be used for visualization and analysis.
To make the camera able to track the robot, the robot's RGB LED was used to light up the robot with a bright white light to give the camera a light spot to track. This was done because the IR light made it impossible to track the robot with the IR camera once it entered the intensity field. In Figure 2.4 the arena with the object and the IR lamp can be observed.

Figure 2.4: Arena used for the single robot case

In figure 2.5 we see the intensity profile of the arena and how the speed and rotational diffusion coefficient vary as a function of the distance to the object without any delay. Figures 2.5 a and b show how the speed increases and the diffusion coefficient decreases with the distance which is due to their dependency on the intensity which is highest close to the object for the robot. The figures also show how the quantities will vary for the simulated agent that is used for the comparisons. One should note that neither the speed nor the diffusion coefficient reaches their respective minimum and maximum values in this case. When the delay is introduced, however, the extrapolations can increase or decrease the values of the measured intensity.

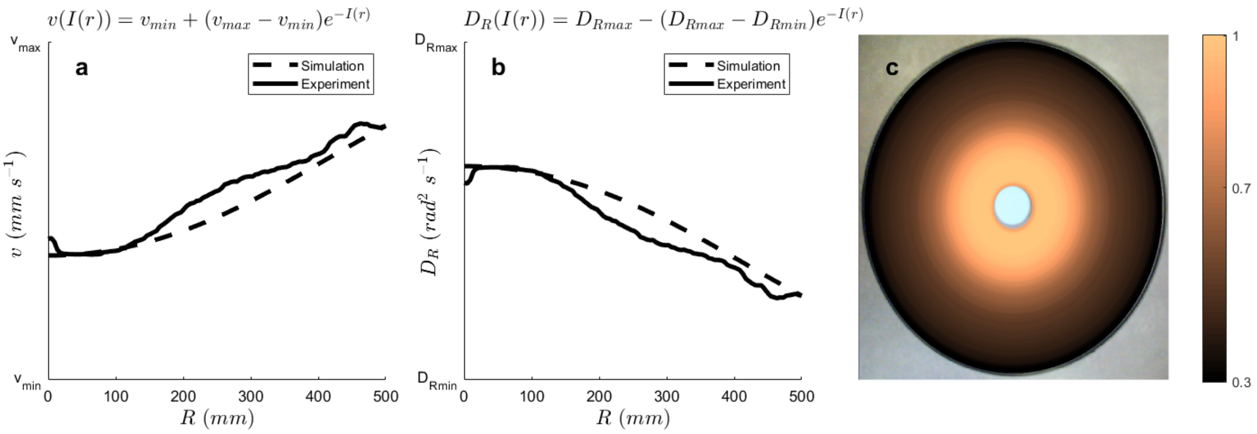


Figure 2.5: Figure **a** shows how the speed of the robot will vary as a function of the distance to the object compared with how the speed will vary with the distance to the center for a simulated agent. Figure **b** shows how the rotational diffusion coefficient varies as a function of this distance. Figure **c** shows the intensity profile of the area around the object.

2.2.2.2 Robot program

For the robot to capture the intensity of the IR light, the eight IR sensors on its circumference were used to measure the light in the surrounding area. To avoid hitting the object in the center of the arena, the same sensors were used to measure the proximity to objects in front of it and then turning around when this distance was small enough. To avoid going outside of the arena, the robot used the four ground sensors to detect when it was above the black tape and then turn around.

2.2.2.3 Analysis of the behavior

When examining the behavior of a single robot there are two quantities that we have focused on: the radial probability distribution of the robot's position and the radial drift of the robot. As we saw in figure 2.3, there will be a higher probability of finding the robot near the center of the arena when the delay is positive and a higher probability of finding it near the edges when the delay is negative. The radial probability distribution is therefore a good way of quantifying this behavior.
The radial drift is a way to measure how the robot moves relative to the center of the arena depending on its location. This is calculated using the following equation [7]:

$$D(r) = \frac{1}{\Delta t}\langle r_{n+1} - r_n | r_n \cong r \rangle \tag{2.6}$$

If the drift is positive this means that the robot tends to move away from the center from this location whereas a negative drift means that it moves towards the center. One can therefore expect the radial drift to be mostly positive when the delay is negative and mostly negative when the delay is positive.
After calculating these quantities for the robot's trajectories, they were compared with results obtained from simulations that were performed with the same parame-

ter values that the robots had as well as with theory. The theoretical formulas were obtained through personal communication with Jan Wehr (Department of Mathematics, University of Arizona, Tucson (AZ)).

2.2.3 Multiple Robots

2.2.3.1 Arena and Tracking

The arena was the same as the one in the single robot case (see section 2.2.2.1) except that there was no IR lamp above the arena and there was no object inside of the arena. The RGB camera was placed above the arena to give a top-down view. This perspective was necessary to avoid having the robots blocking each other in the image, which could be the case when the camera was looking at the arena from an angle.

2.2.3.2 Robot program

The difference between the single robot case (section 2.2.2.2) and this case is that the robots capture the intensity by using their eight IR sensors to detect the proximity to the other robots. The closer the other robots are, the higher the intensity.

3

Results

3.1 Single Robot

Figures 3.1, 3.3 and 3.5 show the trajectories of a single agent whose speed, rotational diffusion coefficient and both vary as a function of the intensity respectively. The trajectories were obtained using the same parameter values that were used on the robot and can be found in appendices A.1.1, A.1.2 and A.1.3. In subfigures a the delay is positive and we can see how the agent prefers to stay near the center of the arena where the intensity is highest. Subfigures b show a less clear preference where there is no delay and in subfigures c, where the delay is negative the agent clearly stays near the edge of the arena.
The same trends can be observed for the robot itself in figures 3.2, 3.4 and 3.6. In the subfigures a, b and c, the robot trajectories follow similar trends to those observed for the simulations. This behavior can also be seen when comparing the radial probability distributions of the robot and the simulated agent. In subfigures d, the delay is positive and the distributions show a higher probability of finding the robot near the center of the arena. In subfigures e, where there is no delay the probability is more evenly distributed with only a slightly higher probability near the center. When the delay is negative, as in subfigures f, probabilities show that there is a much higher probability of finding the agent or the robot near the edge of the arena. In each case we have compared the distributions for the robot and the agent with theoretical formulas. Simulations and experiments agree quite well but there is a certain discrepancy when it comes to the theory, particularly for the case with a negative delay. This can be explained by the fact that the parameters used in the experiments, and therefore the simulations, are not in the limit for which the theory holds. Subfigures g, h and i compare the radial drifts of the robot and the simulated agent with theory. The drifts that are obtained with a positive delay are mostly negative which indicates that the robot is moving towards the center in this case. With a negative delay, the drift is mostly positive, which shows that the robot is moving away from the center. Without any delay, the drift is almost zero. Simulations and experiments show significant agreement also when comparing the drifts but they do not match the theory very well. Again, this can be explained by the fact that the parameter values are not optimal but another factor is that in both the simulations and the experiments the arena is bounded, something that is not taken into account in the theory for the drifts. This can explain why, for the negative delay, the simulated and experimental drifts diverge from the theoretical ones near the edge of the arena.

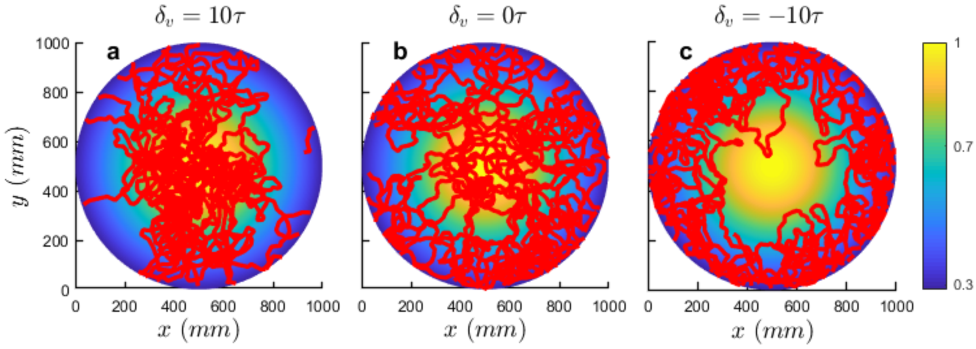


Figure 3.1: Simulated trajectories with a varying speed. In figure **a** the delay is positive, figure **b** shows a trajectory without any delay and in figure **c** the delay is negative.

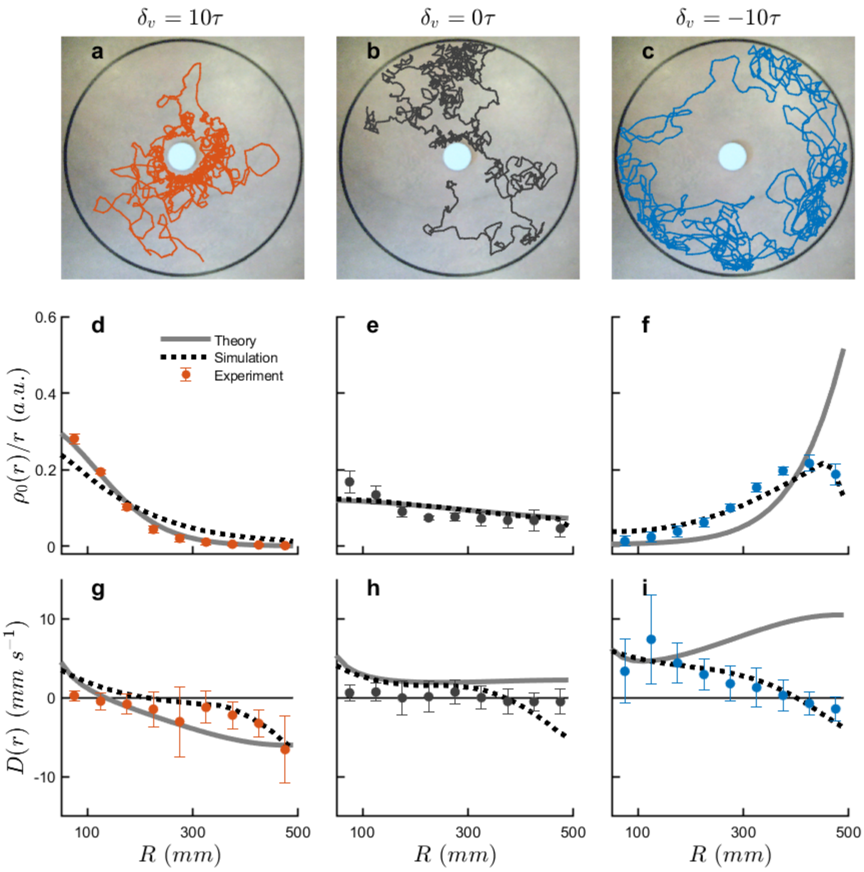


Figure 3.2: Figures **a**, **b** and **c** show the trajectory of the robot on the arena for a positive, zero and negative delay respectively. Figures **d**, **e** and **f** compare radial probability distributions of the robot with the theoretical distributions as well as distributions obtained with a simulated agent for the various delays. Figures **g**, **h** and **i** compare the radial drift of the robot and a simulated agent with theory. A video of the three experiments can be found at this link and in video 8 in appendix b.

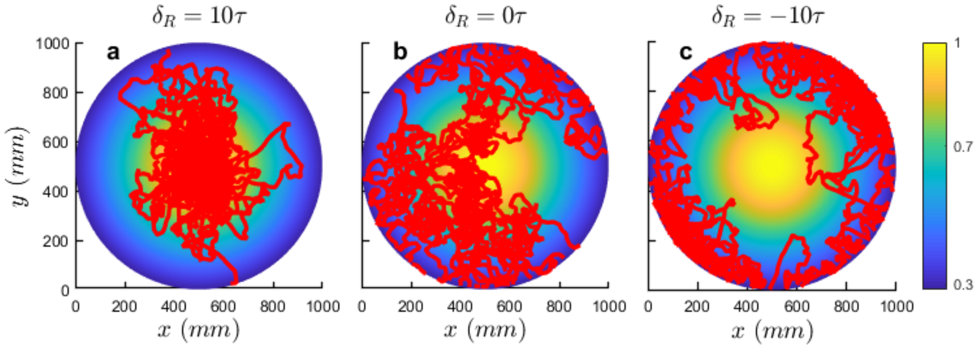


Figure 3.3: Simulated trajectories with a varying rotational diffusion coefficient. In figure **a** the delay is positive, figure **b** shows a trajectory without any delay and in figure **c** the delay is negative.

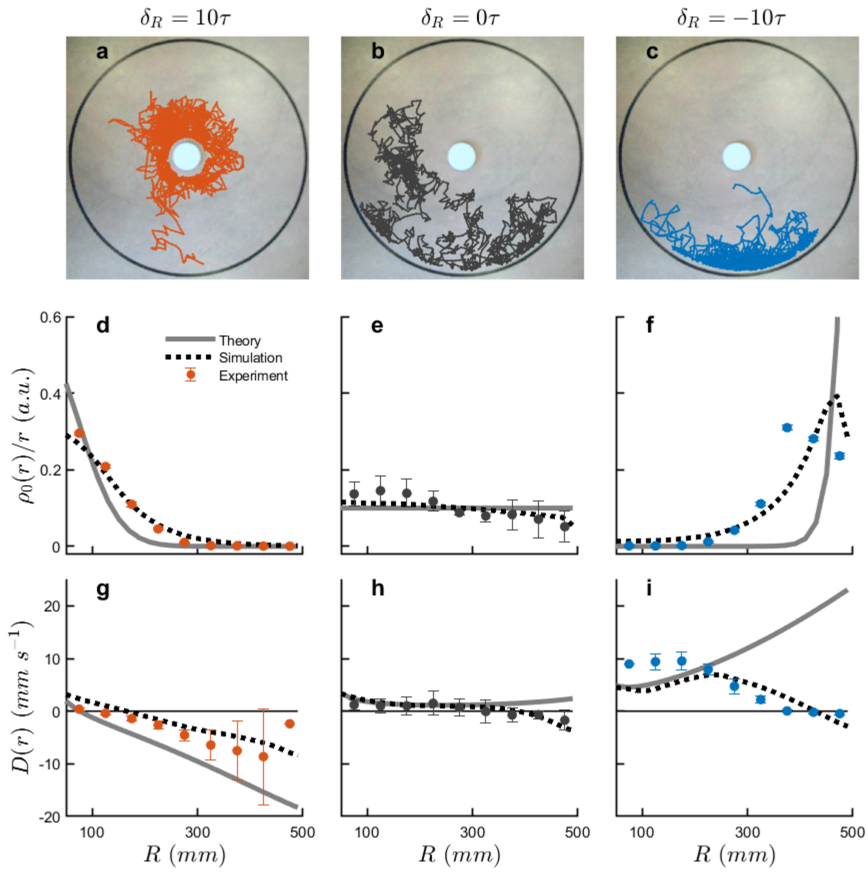


Figure 3.4: Figures **a**, **b** and **c** show the trajectory of the robot on the arena for a positive, zero and negative delay respectively. Figures **d**, **e** and **f** compare radial probability distributions of the robot with the theoretical distributions as well as distributions obtained with a simulated agent for the various delays. Figures **g**, **h** and **i** compare the radial drift of the robot and a simulated agent with theory. A video of the three experiments can be found at this link and in video 12 in appendix b.

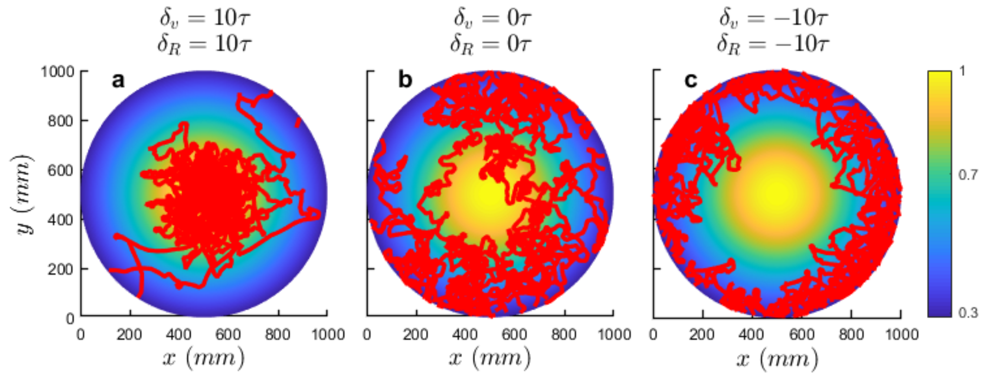


Figure 3.5: Simulated trajectories with speed and rotational diffusion coefficient varying at the same time. In figure **a** the delay is positive, figure **b** shows a trajectory without any delay and in figure **c** the delay is negative.

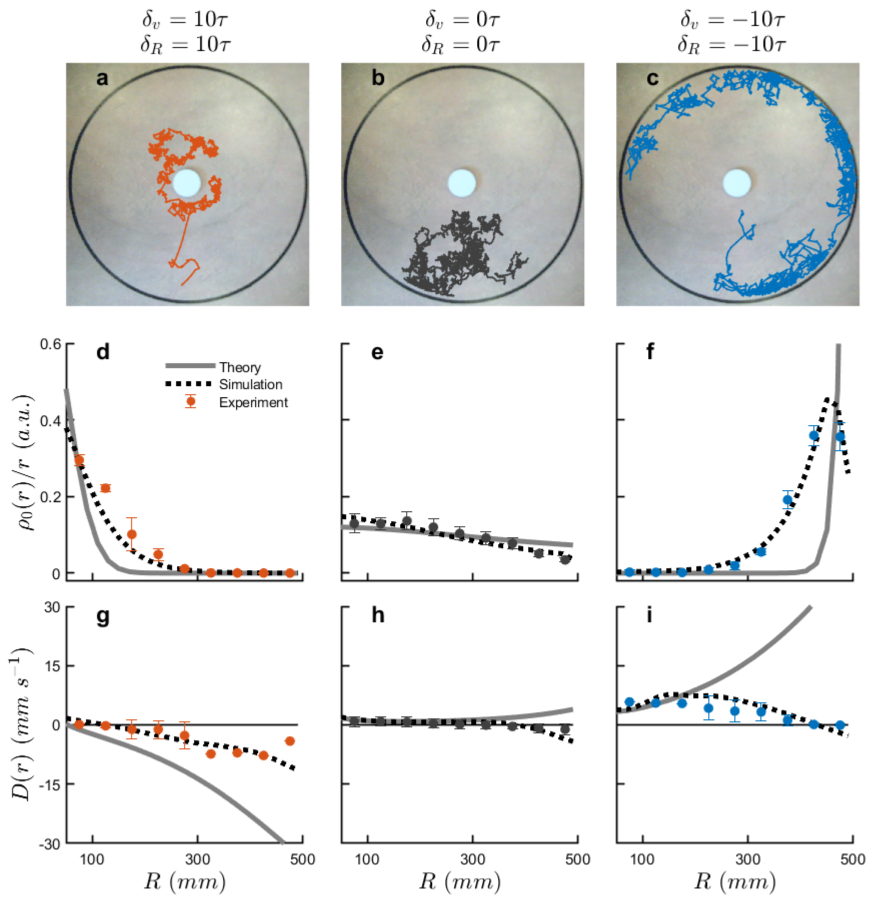


Figure 3.6: Figures **a**, **b** and **c** show the trajectory of the robot on the arena for a positive, zero and negative delay respectively. Figures **d**, **e** and **f** compare radial probability distributions of the robot with the theoretical distributions as well as distributions obtained with a simulated agent for the various delays. Figures **g**, **h** and **i** compare the radial drift of the robot and a simulated agent with theory. A video of the three experiments can be found at this link and in video 16 in appendix b.

3.2 Multiple Robots

When multiple robots interacted at the same time, the goal was to observe clustering among the robots when they were under the influence of a positive delay and segregation while under the influence of a negative delay. The only case when they showed a clear difference between positive and negative delays, however, was when both their speed and rotational diffusion coefficients varied as a function of the intensity as in figure 3.7. Here, when the delay is positive, the robots show a tendency to form small clusters which last for some period of time before disintegrating. With a negative delay, the robots mainly avoid each other. They do, however, manage to get really close to each other sometimes before turning away. When only the speed or the rotational diffusion coefficient varied as a function of the intensity, the different behaviors for a positive compared to a negative delay were not clearly distinguishable. Also, even though there are differences between the cases with a positive and a negative delay when both quantities vary at the same time, this difference is more obvious in the simulations, as seen in figure 3.8. Here the agents form one cluster that remains stable over a long period of time when the delay is positive. When the delay is negative they turn away from each other almost immediately. One possible reason as to why it is harder to obtain the desired behavior from the robots themselves could be that their sensors are affected by noise in their surroundings. Due to this noise, the sensors occasionally make false readings and to deal with this problem we had to limit the range of sensor values used for detection. This effectively reduced the detection range of the robots. A suggested improvement is therefore to increase the detection range of the robots which would make the system more robust against noise in the sensors. The detection range could be increased by using the same method as Mijalkov, et al. did in 2016 [7], where they attached IR LEDs to each robot and had the robots measuring the intensity of the IR light that their neighbors emitted. Another crucial difference between the simulations and the experiments is that it is possible for robots to block the view from other robots that are inside of each other's detection radius. This means that even though multiple robots might be inside the detection radius some of them will not contribute to the intensity that is measured.

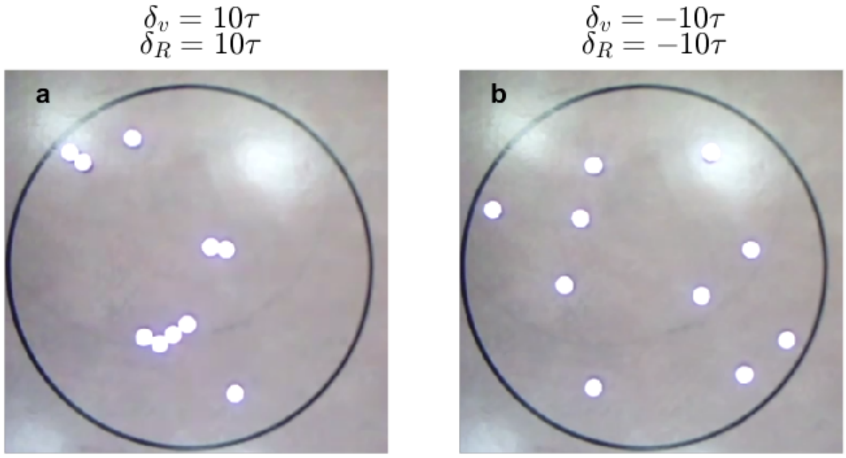


Figure 3.7: 10 robots interacting with each other. Figures **a** and **b** show the results obtained when robots have their speed and rotational diffusion coefficients varying as a function of the intensity for a positive and a negative delay respectively. A video of these two experiments can be found at this link and in video 19 in appendix b.

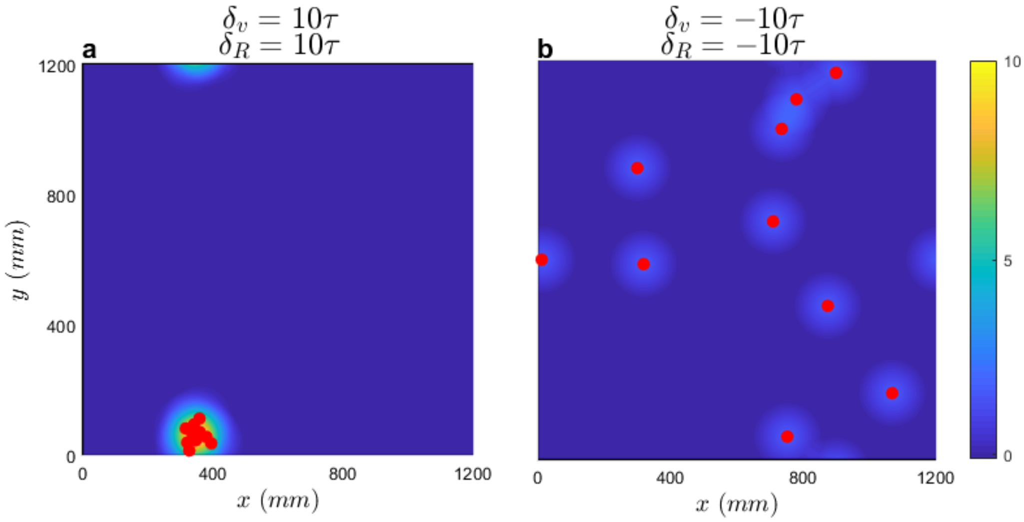


Figure 3.8: 10 simulated agents interacting with each other. Figures **a** and **b** show the results obtained when the agents have their speed and rotational diffusion coefficients varying as a function of the intensity for a positive and a negative delay respectively. A video of these simulations can be found at this link and in video 4 in appendix b.

4

Conclusion

In this study it has been shown that introducing a delay for a robot's sensorial data can significantly alter its behavior. When a single robot is moving in an intensity field, changing the sign of the delay can either make the robot get stuck in or avoid regions where the intensity is high. Comparisons between experimental data from the robot and simulations show good agreement, implying that the hardware limitations do not play such a significant role in this case. When multiple robots interact the same agreement cannot be seen. In the simulations the agents show a clear tendency towards clustering when the delay is positive whereas for the robots this tendency is much weaker. At the same time, the robots do not show the same tendency to avoid each other as the agents do in the simulations. This suggests that the system is more sensitive to disturbances to sensorial values and that limitations in hardware play a more significant role. However, as there is an obvious difference in the robots' overall behavior for the different delays it is believed that improved hardware or sensors with longer detection ranges could improve the results. It would be worth looking into such improvements as the delay could potentially be used to control the large-scale behaviors of multiple robots [12].

Bibliography

[1] A. Sokolov, I. Aranson, J. Kessler and R. Goldstein, "Concentration Dependence of the Collective Dynamics of Swimming Bacteria", Physical Review Letters, vol. 98, no. 15, 2007.

[2] I. Bajec and F. Heppner, "Organized flight in birds", Animal Behaviour, vol. 78, no. 4, pp. 777-789, 2009.

[3] J. Faria, J. Dyer, C. Tosh and J. Krause, "Leadership and social information use in human crowds", Animal Behaviour, vol. 79, no. 4, pp. 895-901, 2010.

[4] M. Beekman, D. Sumpter and F. Ratnieks, "Phase transition between disordered and ordered foraging in Pharaoh's ants", Proceedings of the National Academy of Sciences, vol. 98, no. 17, pp. 9703-9706, 2001.

[5] J. Casper and R. Murphy, "Human-robot interactions during the robot-assisted urban search and rescue response at the World Trade Center", IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 33, no. 3, pp. 367-385, 2003.

[6] A. YAMASHITA, M. FUKUCHI, J. OTA, T. ARAI and H. ASAMA, "Motion Planning for Cooperative Transportation of a Large Object by Multiple Mobile Robots.", Transactions of the Japan Society of Mechanical Engineers Series C, vol. 68, no. 665, pp. 165-172, 2002.

[7] Mijalkov, M., McDaniel, A., Wehr, J. and Volpe, G. (2016). Engineering Sensorial Delay to Control Phototaxis and Emergent Collective Behaviors. Physical Review X, 6(1).

[8] "Elisa-3 - GCtronic wiki", Gctronic.com, 2018. [Online]. Available: http://www.gctronic.com/doc/index.php/Elisa-3. [Accessed: 06- Apr- 2018].

[9] G. Volpe, S. Gigan and G. Volpe, "Simulation of the active Brownian motion of a microswimmer", American Journal of Physics, vol. 82, no. 7, pp. 659-664, 2014.

[10] "Aseba Studio - Thymio & Aseba", Thymio.org, 2018. [Online]. Available: https://www.thymio.org/en:asebastudio. [Accessed: 27- Apr- 2018].

[11] "SwisTrack - Wikibooks, open books for an open world", En.wikibooks.org, 2018. [Online]. Available: https://en.wikibooks.org/wiki/SwisTrack. [Accessed: 27- Apr- 2018].

[12] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo, Swarm robotics: a review from the swarm engineering perspective, Swarm Intelligence 7, 1 (2013).

A
Appendix A

A.1 Single robot

A.1.1 Varying speed

Minimum speed: $v_{min} = 4.3$ mm s^{-1}
Maximum speed: $v_{max} = 25.7$ mm s^{-1}
Characteristic reorientation time: $\tau = 3.5$ s
Rotational diffusion coefficient: $D_R = \tau^{-1} = 0.29$ rad^2 s^{-1}
Time step: $\Delta t = 1.4$ s

A.1.2 Varying rotational diffusion coefficient

Speed: $v = 17.1$ mm s^{-1}
Characteristic reorientation time: $\tau = 3.5$ s
Minimum rotational diffusion coefficient: $D_{Rmin} = 0.014$ rad^2 s^{-1}
Maximum rotational diffusion coefficient: $D_{Rmax} = 1.4$ rad^2 s^{-1}
Time step: $\Delta t = 1.4$ s

A.1.3 Speed and diffusion varying at the same time

Minimum speed: $v_{min} = 4.3$ mm s^{-1}
Maximum speed: $v_{max} = 25.7$ mm s^{-1}
Characteristic reorientation time: $\tau = 3.5$ s
Minimum rotational diffusion coefficient: $D_{Rmin} = 0.014$ rad^2 s^{-1}
Maximum rotational diffusion coefficient: $D_{Rmax} = 1.4$ rad^2 s^{-1}
Time step: $\Delta t = 1.4$ s

A.2 Multiple Robots

A.2.1 Speed and diffusion varying at the same time

Minimum speed: $v_{min} = 1.7$ mm s^{-1}
Maximum speed: $v_{max} = 4.3$ mm s^{-1}
Characteristic reorientation time: $\tau = 3.5$ s
Minimum rotational diffusion coefficient: $D_{Rmin} = 0.014$ rad^2 s^{-1}
Maximum rotational diffusion coefficient: $D_{Rmax} = 1.4$ rad^2 s^{-1}

Time step: $\Delta t = 1.4$ s
Detection radius: $R = 60$ mm

B

Appendix B

B.1 Video links

Here one can find the links to the videos captured during the simulations and experiments in this work:

1. Simulated Brownian motion:
   https://youtu.be/jGYJbGj5uSg

2. Intensity demonstration:
   https://youtu.be/7js2fEsz7Dc

3. Delay demonstration:
   https://youtu.be/M0p-jRmMrCs

4. Two videos of multiple robots simulated. Speed, rotation varies and the delay is positive for one and negative for the other:
   https://youtu.be/-sBbZD0RvBA

5. One robot. Speed varies and the delay is positive:
   https://youtu.be/9U7teRzP6kk

6. One robot. Speed varies and there is no delay:
   https://youtu.be/d61yYCj0Y3c

7. One robot. Speed varies and the delay is negative:
   https://youtu.be/hdNEI8T4YyU

8. Three videos of one robot. Speed varies:
   https://youtu.be/XBx5XqnkSPo

9. One robot. Rotation varies and the delay is positive:
   https://youtu.be/rMiJDk98z8s

10. One robot. Rotation varies and there is no delay:
    https://youtu.be/Er8xiAe0lm4

11. One robot. Rotation varies and the delay is negative:
https://youtu.be/81JNAGWIZFs

12. Three videos of one robot. Rotation varies:
https://youtu.be/7T2rDnDGGIM

13. One robot. Speed, rotation varies and the delay is positive for both:
https://youtu.be/mxE-8fCBWCg

14. One robot. Speed, rotation varies and there is no delay:
https://youtu.be/O3B9ibv6f0I

15. One robot. Speed, rotation varies and the delay is negative for both:
https://youtu.be/XGbcGwff5I8

16. Three videos of one robot. Speed and rotation varies:
https://youtu.be/TogBCc2xZC4

17. Multiple robots. Speed, rotation varies and the delay is positive:
https://youtu.be/eRox4Pw6gCo

18. Multiple robots. Speed, rotation varies and the delay is negative:
https://youtu.be/Z9Ps30gKAxc

19. Two videos of multiple robots. Speed, rotation varies and the delay is positive for one and negative for the other:
https://youtu.be/WxL5PrwiZz4

20. Multiple robots. The speed varies with intensity from an IR lamp:
https://youtu.be/oJbS_aSDhiA

21. Multiple robots, The rotational diffusion coefficient varies with intensity from an IR lamp:
https://youtu.be/4dLGVDj5HFE

22. Multiple robots, Speed and the rotational diffusion coefficient varies with intensity from an IR lamp:
https://youtu.be/P94Z30cyPM8