SpinBot: An Autonomous, Externally Actuated Robot for Swarm Applications

Andrew SaLoutos and Michael Rubenstein

Northwestern University, Evanston IL 60208, USA

andrew.saloutos@u.northwestern.edu
rubenstein@northwestern.edu

Abstract. Complexity, cost, and power requirements of individual robots are large factors in limiting the size of robotic swarms. In this paper, we present a prototype robot, the SpinBot, that is externally actuated via an orbital table and has only one infrared sensor pair for communication and sensing. The SpinBot can move autonomously in 2D by activating and de-activating its only onboard actuator, an electro-permanent magnet, and can communicate and sense the angles between its neighbors. The angle sensing is accomplished by adding a bearing to the robot chassis and offsetting the center of mass from the point of attachment between the SpinBot and the table surface, so that the upper part of SpinBot rotates about itself at the same frequency as the orbital table. We describe the design of the SpinBot in detail, and present results from our implementation of a centering algorithm for a group of four SpinBots which utilizes the SpinBot's unique sensing and locomotion abilities.

Keywords: Swarm robotics, external actuation, sensing

1 Introduction and Related Work

Traditionally, robots used for swarm applications are able to independently control their position in the environment using on-board power sources, such as batteries, coupled with multiple on-board actuators, such as electric motors. The use of multiple self-powered actuators significantly contributes to a robot's complexity, as approximated by part count, hardware cost, assembly time, and stored energy requirements. Furthermore, robots within swarms must have enough sensing hardware to intelligently interact with their environment and one another. Oftentimes, robots must have several specialized sensors, such as range-finders or encoders, as well as communication hardware and/or peripherals to be effective in accomplishing their range of objectives. Combined, these two factors greatly contribute to limiting the size of robotic swarms, which is currently on the order of 100-1000 robots [1-3].

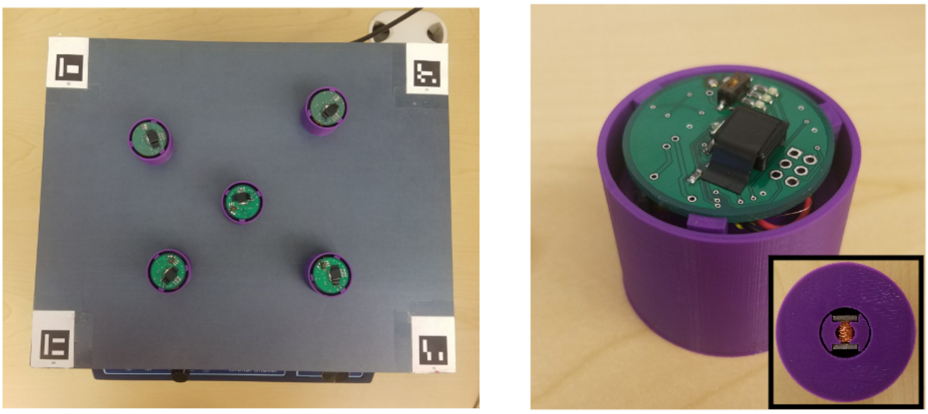

Fig. 1. Multiple SpinBots on the orbital table used for external actuation (left) and a close-up of a single SpinBot with an inset view of the bottom surface of the robot (right)

One approach that could reduce the complexity of a swarm robot, and therefore enable larger swarm sizes, is to use an apparatus that can create external forces which are utilized by robots for individual motion. While the apparatus to create these forces will contribute to the complexity of the overall system, where system complexity is (robot complexity $*$ number of robots $+$ apparatus complexity), it may be possible to scale this type of system to larger numbers as the robots may be less complex, and the apparatus complexity could be a constant or fixed cost if it is shared amongst all robots.

In past work, researchers have investigated using external forces to move and/or power robots, however, most are not able to scale to large numbers of independently controlled and autonomous robots. There are many approaches that move a passive robot, often on the micro-scale, through the use of external fields such as magnetic [4] or electro-static [5]. While this enables the robots to be very small, robot motion is controlled only by the external fields, so the robot itself cannot decide its own motion. Additionally, if multiple robots are present, they will all be exposed to the same field, and will react and move in nearly identical ways. To create a non-uniform external force field, some systems have utilized stochastic forces, such as thermal motion, random shaking, or fluid mixing [6-8] to power the movement of individuals. Generally, the apparatus for applying stochastic forces is scalable to larger numbers, and the motion of multiple robots will not be nearly identical. However, the individual robots are still unable to control the speed and direction of their own motion, and an individual cannot be guaranteed to reach a desired location in a bounded amount of time. In contrast, some approaches use an apparatus that can control many localized forces on its surface, such as isolated electromagnetic fields or vibration [9], [10]. As these apparatuses generally address each location to be controlled individually, the complexity of the apparatus will increase as more robots are used and therefore require more addressable locations.

Another approach makes use of external forces to reconfigure a group of modular robots [11], where all robots are on a table moving in an orbital manner, and robots can use this orbital force to rotate from one docking site to an adjacent one. This approach was modified to allow a robot to move autonomously using an electro-permanent magnet [12] as the only onboard actuator, without any notion of a required configuration, around the surface of an orbital table [13]. The work presented here utilizes this locomotion method, as it will be able to scale to large numbers of robots without significantly increasing the complexity of the apparatus or compromising the autonomy of the individual robots.

Aside from employing external actuation, reducing the complexity of a swarm robot, and therefore enabling larger swarm sizes, can be achieved by using sensors and communication hardware that are as simple and as versatile as possible, given the tradeoffs between the simplicity of a robot's peripherals, cost, ease-of-use, and the quality and quantity of information provided. For many robotic swarms, effective communication is extremely important. However, many communication methods require extra hardware that increases the part count, cost, and stored energy requirements of the robot, in a similar manner to on-board actuators. Furthermore, in addition to communicating with one another, robots used for swarm applications typically need to be able to determine distances and angles between themselves and neighboring robots.

Some approaches for communication in swarm robots are to send messages over radio [14-16], WiFi [17], or even Bluetooth [14]. These methods can be used for communication over much longer distances relative to the size of the robots, but the increased range does not allow for the construction of a communication neighborhood around a robot that is often required for swarm behaviors. These communication methods also require dedicated hardware, which increases the part count and cost of each individual robot. The more complicated protocols can create more software overhead and could necessitate a faster and more expensive controller for the robots as well.

To avoid the added cost and space requirements of dedicated communication hardware, some robots require physical connections [8] or specific orientations [7] to be able to communicate with one another. While this greatly reduces the necessary hardware and software for communication, it also severely limits the functionality of the robots, as the robots are only able to communicate in a limited set of directions defined by their lattice-style structure.

A simple approach for communicating and sensing in a larger number of directions is to add more sensors, such as pairs of infrared emitters and receivers, around the perimeter of a robot [14-17]. The increased number of sensors means that the robots have a finer sensor resolution and are less dependent on a rigid lattice structure for sensing and communication. However, the added components also increase the complexity of the robots, especially in terms of part count.

Another approach to increasing the resolution of sensing and communication is to take advantage of the geometry of the robots and the test surface [1]. The Kilobots in [1] use only one infrared pair to communicate in all radial directions by sending messages downward to reflect off the test surface. Based on the strength of received messages, they are also able to determine the distance between neighbors and themselves. The single infrared pair effectively takes the place of additional communication hardware and larger groups of sensors, while also greatly reducing the complexity of the Kilobots. However, since the orientation of the pair can only sense distance, precise localization is a challenge and requires many messages for a small group of robots.

Being able to sense the angle at which a message is received is preferable to distance sensing, such as that of the Kilobots, since distances from short-range communications do not contain as much information. For example, accurate angle sensing allows for a group of robots to use fewer leaders or beacon robots to form a coordinate system that does not suffer from the flip ambiguities that must otherwise be addressed [18].

The robot presented here, called the SpinBot, is shown in Figure 1.[1] The SpinBot uses the same locomotion method as the robot presented in [13], activating and deactivating an electro-permanent magnet at specific times within the rotational period of an orbital table. This allows each robot to move independently across the table surface of the orbital table without any traditional actuators. The SpinBot uses only one infrared emitter and receiver pair per robot, similar to the sensing employed by the Kilobots [1], but the SpinBot's sensor scans with the rotation of the table, recording the angle of received messages instead of the distance. The offset center of mass of the SpinBot, along with a ball bearing within the chassis of the robot, enable the scanning of the single sensor pair, which rotates in phase with the orbital table and has an angular resolution of less than 2°, depending on the rotation speed. There is no limit on the number of rotations that the SpinBot can make, and it is able to locomote in any direction on the table surface. The use of only one actuator and one sensor pair greatly reduces the complexity of the robot and will allow for high scalability in the future.

The layout of this paper is as follows. First, we describe the design of the SpinBot and elaborate on its unique capabilities. Then, we present a simple centering algorithm that utilizes the capabilities of the SpinBot. Lastly, we discuss how the SpinBot platform can be improved for use in further development and evaluation of algorithms for swarm behavior.

2 Design of the SpinBot

As mentioned in the previous section, the SpinBot is an extension of the work in [13]. The robots are externally actuated by an orbital table, which produces a periodic circular translation of the entire flat table surface. Each robot can move in any direction by de-activating and re-activating an electro-permanent magnet (EPM) to detach and reattach to the table surface. The ball bearing in the center of each robot, shown in Figure 2, and an off-center weight allow for the robot to continuously rotate in-phase with the orbital table and simultaneously send messages and scan for messages from neighboring robots.

The robots are cylindrical, approximately 38 mm in diameter and 29 mm tall, and are 26 g each. They are externally actuated by the orbital table shown in Figure 1. The orbital table consists of a thin sheet of steel attached to a KJ-201BD orbital shaker typically used for mixing fluids. The table's translation diameter is about 22 mm, and it oscillates at frequencies up to 3.5 Hz. In total, the raw materials and components required to manufacture one SpinBot would cost less than $30.

---

[1] A supplemental video that shows the SpinBots during operation can be found at: https://youtu.be/4gB0pCXmMeA

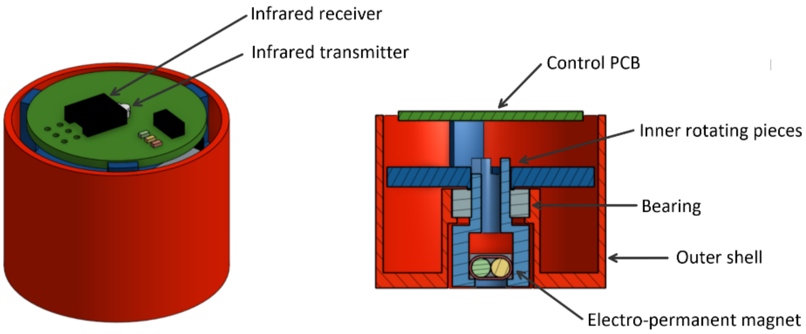

Fig. 2. Labeled CAD models of an assembled SpinBot (left) and a simple section view (right)

2.1 Mechanical Design

The chassis of the SpinBot consists of three 3D-printed pieces of PLA plastic and one ball bearing. The robot also uses one EPM, one battery, and two custom PCBS, one for the communication hardware and the controller and another for the capacitors to switch the EPM. An assembled CAD model and a section view of the main components of the SpinBot chassis are shown in Figure 2. The outer shell, shown in red, is in contact with the table and does not rotate. This piece also protects the wiring of the robot, allowing two robots to contact one another without risk of damaging one another. The inner pieces, shown in blue, connect through the ball bearing, shown in gray, which provides a low-friction interface for the inner pieces to rotate with the orbital table. A weight, which is not shown, hangs from the flange of the inner piece and offsets the center of mass from the center of rotation, causing the robot to rotate with the orbital table. The EPM is attached to the inner pieces and is suspended 0.13 mm above the table surface. This distance was made to be as small as possible, given the tolerances of the 3D-printed parts, so that the attractive force between the EPM and the table surface was strong enough to anchor the robot during rotation. If the EPM contacted the table surface as in [13], the attractive force would be much stronger and a smaller magnet could be used. However, this would require implementing a slip-ring or a similar device, to ensure that the communication and sensing hardware could still rotate with the table.

The ball bearing that supports the inner pieces of the robot provides an additional, unactuated rotational degree of freedom to the robot. As mentioned above, the SpinBot has an additional weight which displaces its center of mass relative to its center of rotation. This, combined with the extra rotational degree of freedom, allows the external actuation of the orbital table to spin the robots about themselves. The rotation of a single SpinBot is shown in Figure 3, where the left part of the figure shows the rotation, not to scale, from the point of view of a world frame and the right part of the figure shows the rotation from the point of view of the robot itself. From the world frame, the robot's center of rotation, which is also the point of attachment between the robot and the table, moves with the surface of the orbital table around the table's center of rotation. Due to the centripetal acceleration of the robot, the center of mass of the robot rotates around the same point, the table's own center of rotation, and at the same angular velocity as the table. From the point of view of the robot, its center of mass appears to be rotating around its attachment point at the same frequency as the rotation of the table.

The weight used to move the center of mass of the SpinBot consisted of the battery and capacitor bank, which were placed on the same side of the chassis, and a column of #4 washers. The center of mass was shifted almost 6 mm away from the center of rotation and the mass of the rotating pieces of the robot increased from 7 g, including the external capacitors and the battery, to 10 g. In general, displacing the center of mass further away from the center of rotation and minimizing the rotational friction increases the bandwidth of possible rotation frequencies. In this application, the size of the SpinBot was somewhat limited by the dimensions of the surface of the orbital table and the effective communication range of the infrared sensors, so the center of mass could not be moved very far from the center of rotation. Despite this, the SpinBot could spin at frequencies (2.3 Hz to 3.6 Hz) comfortably within the range of the orbital table.

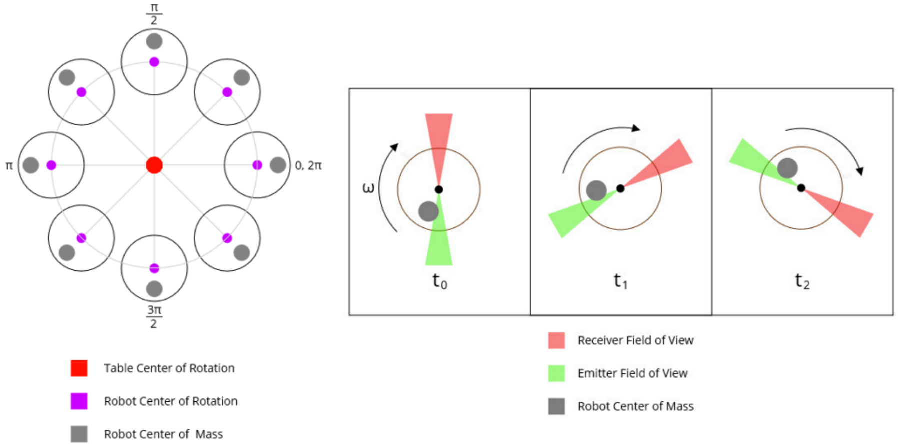

Fig. 3. Top down view of center of mass vs. center of rotation for one cycle from the world frame of reference (left) and the robot frame of reference (right)

2.2 Electronics Design

The SpinBot uses an ATmega328p microcontroller to control the communication hardware as well as the switching of the EPM. The communication hardware consists of a side-view surface-mounted infrared emitter and receiver. The analog signal from the receiver photodiode is amplified and compared to a reference voltage, resulting in a binary signal that is used as the basis for communication between robots. The length of time that the photodiode signal is above the reference voltage is inversely related to the distance between the sender and receiver of a message and could be used to approximate the distance between robots. However, that is not implemented here. Since the infrared receivers have an angle of half-sensitivity of 75°, a small hood was created to limit the field-of-view of the receiver to 35° and thus improve the robot's accuracy in determining the headings of its neighbors. This hood can be seen on the top surface of the SpinBot in Figure 2. The maximum communication range of the robots is roughly six body-lengths, but at this distance some message loss occurs. At distances under five body-lengths, or 190 mm, the communication is more reliable.

The infrared emitter and receiver were placed on the top surface of the control circuit board, facing away from each other. Because the SpinBots on the orbital table rotate in phase with the table and each other, this sensor orientation ensures that the field of view of the receiver of each robot will intersect with the field of view of the emitter of every other robot at some point during each period of rotation. This is shown in Figure 4, where three SpinBots are rotating clockwise on the table, with the same angular velocity. At time t_0, the SpinBots are in phase with one another but none of them are aligned. The emitters, shown as green cones, of the robots are in phase with one another, as are the receivers, shown as red cones. At time t_1, the robots have rotated 30°, and the bottom robot is able to receive messages from the upper right robot. At time t_2, the robots have rotated another 60°, and the two upper robots are now in alignment. After another 60° of rotation, the upper left robot and the bottom robot would align. This entire cycle would keep repeating for each full rotation.

A simple communication protocol was developed to ensure that robots only act on accurately received messages. In this protocol, each message consists of 8 total bits and the first and last bits must be 1, which leaves 6 bits to carry message data (or 64 unique messages). Due to the capacitance of the receiver's amplification circuit, the transmission rate of the infrared messages had to be limited at 5000 bits per second. At a rotation frequency of 3 Hz, this corresponds to 1.6 ms or approximately 1.73° per 8-bit message.

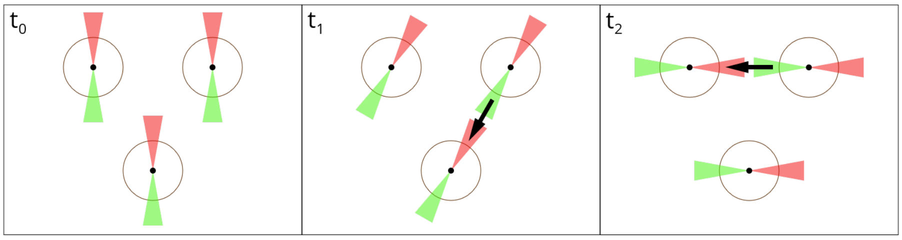

Fig. 4. Infrared sensor alignment of three SpinBots during rotation, where the green cones are the emitters' fields of view and the red cones are the receivers' fields of view

The SpinBot has three LEDs on the upper surface of the control PCB and a switch for the 40 mAh 4.2 V lithium-polymer battery used to power the robot. In previous work, the electro-permanent magnets used have been around 3.2 mm x 5.8 mm x 2.5 mm. The EPM used by the SpinBot was manufactured using the same process and materials as in [12] and [13], but with increased dimensions to produce the attraction force required to prevent the robot from sliding unnecessarily across the table during rotation. Both permanent magnets are 3.2 mm in diameter and 6.4 mm long, and the pole pieces are 6.9 mm x 5.6 mm x 1.5 mm. The EPM is wound with 34 AWG magnet wire, has a resistance of approximately 2.2 Ω, and is switched by applying 20 V for 120 μs in the desired direction using a boost regulator, three 100uF tantalum capacitors, and an h-bridge. The 120 μs switching time of the EPM did not have an adverse effect on the testing of the SpinBots described in Section 3, since each movement spanned around 80 ms. However, for shorter, quicker movements, or in future applications, the switching time may need to be specifically accounted for.

2.3 Operation of a Group of SpinBots

To use a group of SpinBots, the orbital table needs to be set on a level surface and the robots placed in the desired starting positions. The robots must be powered on before the table is turned on and set to the desired speed. One side of the table can be lifted slightly (by about 20°) to help kick-start the rotation of the robots, and then set back down to the level surface for the duration of the test or experiment. The batteries of the SpinBots are small and have limited capacity, typically lasting for an hour of testing. Recharging the batteries also takes about an hour, so for a small group of SpinBots it is feasible to have two sets of batteries, where one set is in use while the other is charging.

Simple operation tasks such as manually flipping power switches or changing batteries of each robot in a swarm can quickly become significant time commitments as the size of the swarm is increased. In the last section of this paper, we discuss some ways in which we could modify this iteration of the SpinBot design to further increase its scalability.

3 Centering Within a Convex Polygon

To showcase the basic functionality of the SpinBot, we implemented an algorithm in which one SpinBot navigates to the center of a triangle of SpinBots acting as “beacons”. This type of algorithm was chosen because it would require the robot to be able to localize itself within the beacons and then identify and locomote towards a desired location; these are two behaviors that are essential building blocks of more complex swarm algorithms. This algorithm could easily be modified for a convex polygon with any practical number of vertices, as long as the mobile SpinBot starts within the perimeter of the polygon. The motion planning stage of the algorithm was designed for the mobile SpinBot to use the unique IDs of the beacons in order to plan and time each step towards the center of the triangle.

3.1 Algorithm Overview

The starting position of the mobile SpinBot is constrained to be within the triangle formed by the beacons. From its initial position, the mobile SpinBot first calculates the period of its rotation by averaging the times between ten messages received from a single beacon.

After the rotation period has been calculated, the mobile robot begins the main loop of the algorithm. It waits to receive messages from all three beacon robots within the same rotation, and then calculates a movement direction. This calculation is based on the times between the received messages, which can be used to approximate the angles between two beacon robots and the mobile robot, since the rotation speed is constant and the period is known. Figure 5 shows the movement direction for the SpinBot at each location within the triangle of beacons. If the three times are all within a defined tolerance of one another, the robot is "centered" successfully. This is shown in Figure 5 as the light blue circle around the centroid of the triangle. If the angle between two beacons is larger than the other two angles by the defined tolerance, the robot will move towards the third beacon. This is shown in Figure 5 as the solid-colored triangles. Otherwise, if an angle between two beacons is less than the other two angles by the defined tolerance, the robot will move away from the third beacon. This is shown in Figure 5 as the light-colored bars. After the robot has selected a movement direction, it waits until it receives a message from the relevant beacon. Once it has received a message from the chosen beacon, the robot then initiates its movement routine. In this routine, the robot waits for 20% of its calculated rotation period, if it wants to move towards the beacon, or 70%, if it wants to move away from the beacon. This delay is necessary so that the velocity of the center of mass is in the direction of the desired movement at the time the robot detaches from the table surface. After the delay, the robot detaches and slides in the desired direction for a specified time and begins to wait for messages from the three beacons again. In this implementation, a fixed detachment time of 80 ms was used. The loop of collecting messages and executing a movement is performed until the times between the received messages meet the centering criteria.

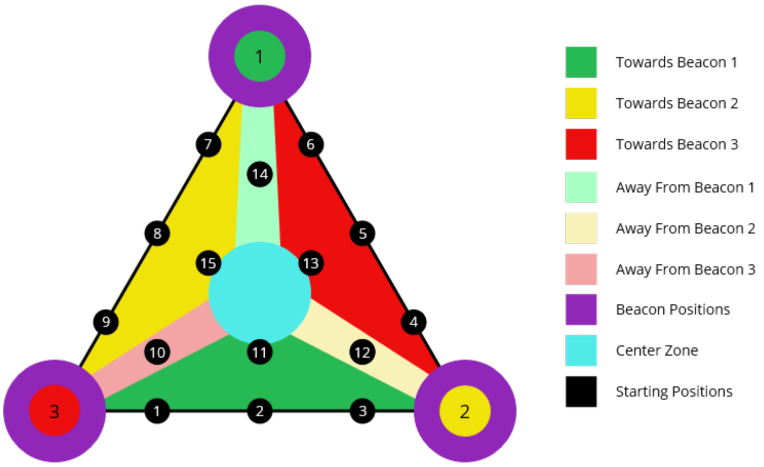

Fig. 5. Map of robot positions and movement directions for the mobile SpinBot and beacon robots 1, 2, and 3. The color of each beacon corresponds to the actual LED colors shown in the trial videos. The beacons, the center zone, and their spacing are shown to scale.

3.2 Experimental Setup

The three beacon robots were fixed in an equilateral triangle, 150 mm from each other, to ensure consistency across all the trials. These robots are shown as purple dots in Figure 5, above, and are labeled to correspond to the regions of the underlying movement direction map. The mobile SpinBot was placed at fifteen different starting positions, shown as black dots. As the robot moved, it used its three LEDs to indicate which beacon it had selected to utilize to begin its movement sequence. When the mobile SpinBot determined it had reached the center zone of the triangle, shown in light blue, it turned on all three of its LEDs and the trial was stopped. If it did not reach the center by 2 minutes after its first movement, the trial was ended, and the final position was recorded. A trial was considered a "success" if the true center of the beacon triangle was underneath the footprint of the mobile SpinBot after it had declared itself "centered", corresponding to an accuracy of one body length, or 38mm.

3.3 Performance Results

Each of the fifteen trials was recorded, and fiducial markers on the orbital table were used to isolate the translating table surface and extract the positions of the four robots.[2] Figure 6 shows several frames of one of the isolated videos for the trial starting from position 4, including the robot at the starting position (Fig. 6A), the selection of a beacon to move towards (Fig. 6B), the moment after the movement (Fig. 6C), and the final "centered" position (Fig. 6D). For the same trial, Figure 7 shows the path of the robot from the starting position to the final position as well as the distance between the mobile robot and the center of the beacons at each time step. In the trajectory plot of Figure 7, the sharp change in direction of the SpinBot clearly coincides with the preferred direction map shown in Figure 5. In the error plot of Figure 7, each steep change in error is during a movement of the robot, and a decrease in error indicates that the movement was successfully directed towards the center. The flat areas of the error graph indicate times during which the robot had difficulty collecting three successive messages or receiving the message from the beacon corresponding to its selected movement direction.

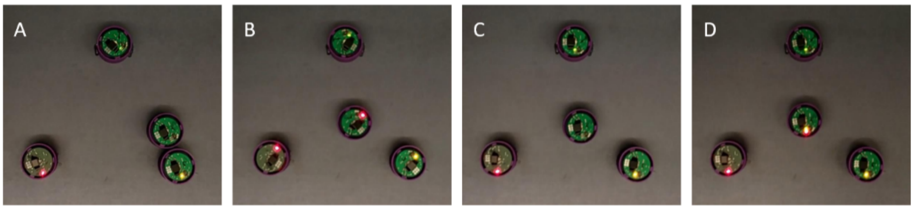

Fig. 6. Video frames from the trial starting at position 4: starting position (A), before a movement (B), after a movement (C), final position in the center (D)

---

[2] Videos of each trial can be found at: https://www.youtube.com/playlist?list=PLIDlvzD-4U-lDHa7PLRJNOSd9wE1bmXpd

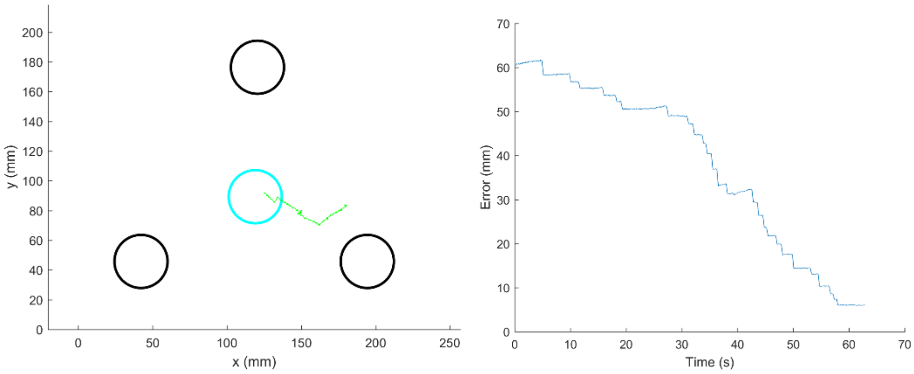

Fig. 7. Path (left) and distance to the center (right) for the SpinBot starting from position 4

Figure 8 shows the paths taken by the robot from each starting position relative to the beacon positions. The center zone, shown in light blue, is the geometric center of the beacons. From thirteen of the fifteen starting positions, the mobile SpinBot successfully reached the center zone. For the two unsuccessful trials, from position 8 and position 12, the robot's final distances to the center of the beacons were 24.4 mm and 19.4 mm, respectively, compared to the success threshold of 19 mm (the radius of the SpinBot). The path from position 8 is shown in pink in Figure 8, and the path from position 12 is shown in orange. The trajectories of each trial were qualitatively similar to what was expected, and the final position was not far from the success threshold.

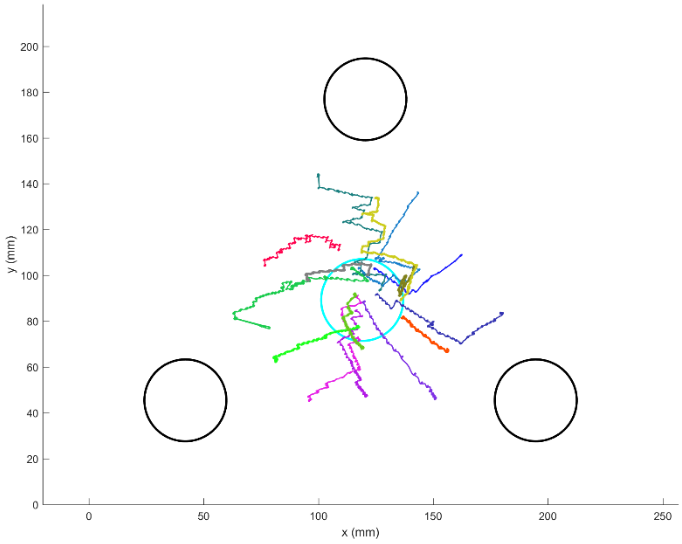

Fig. 8. Paths of the mobile SpinBot from each of the fifteen starting positions, with the center zone shown in light blue. Paths from positions 8 and 12 were unsuccessful, and are shown in pink and orange, respectively. All of the other paths were successful.

4 Conclusions and Future Work

In this paper, we described the design and functionality of the SpinBot, a novel robot for implementing swarm robotics algorithms. We showed that the SpinBot can accurately sense the angles between itself and other robots using one infrared emitter/receiver pair, and that it could reliably use this bearing information to locomote to the center of an equilateral triangle of other SpinBots. The sensing and locomotion strategies presented here provide a foundation that could be used to implement more complex algorithms or behaviors with a larger swarm of SpinBots, such as parallel self-assembly or formation of more complex hollow shapes.

A key direction for future work on the SpinBot is further development of the hardware capabilities to enable the creation of a true "swarm" of SpinBots, and thus the implementation of more complex behaviors. For example, charging the individual batteries of a swarm of robots can be a significant time commitment, which can easily be reduced by integrating a scalable charging mechanism into the design of the SpinBot chassis. Another possible improvement would be to implement a scalable power-on sequence, such as a very low-power hibernation mode that the robots could stay in between testing and charging sessions. Furthermore, there are several ways to modify the hardware for more robust control of the robot's locomotion across the table surface, such as adding an accelerometer or gyroscope or improving the speed and range of the communication and sensing. A swarm of SpinBots would also require a larger orbital table, which could be designed to have more precise speed control than the table currently being used.

References

1. Rubenstein, M., Cornejo, A., Nagpal, R.: Programmable self-assembly in a thousand-robot swarm. Science 345(6198), 795-799 (2014).
2. Konolige, K., et al: Centibots: Very large scale distributed robotic teams. Experimental Robotics IX 21, 131-140 (2006).
3. Wurman, P., D'Andrea, R., Mountz, M.: Coordinating hundreds of cooperative, autonomous vehicles in warehouses. AI magazine 29(1), 9 (2008).
4. Kummer, M., et al: OctoMag: An electromagnetic system for 5-DOF wireless micromanipulation. Robotics, IEEE Transactions on 26(6), 1006-1017 (2010).
5. Donald, B., Levey, C., McGray, C., Paprotny, I., Rus, D.: An untethered, electrostatic, globally controllable MEMS micro-robot. Microelectromechanical Systems, Journal of 15(1), 1-15 (2006).
6. White, P., Zykov, V., Bongard, J., Lipson, H.: Three Dimensional Stochastic Reconfiguration of Modular Robots. In: Robotics: Science and Systems, pp. 161-168. MIT Press, Cambridge (2005).
7. Bishop, J., et al: "Self-organizing programmable parts." In: International Conference on Intelligent Robots and Systems, pp. 3684-3691. IEEE (2005).
8. Gilpin, K., Knaian, A., Rus, D.: Robot pebbles: One centimeter modules for programmable matter through self-disassembly. In: Robotics and Automation (ICRA), 2010 IEEE International Conference on, pp. 2485-2492. IEEE (2010).

9. Pelrine, R., et al: Diamagnetically levitated robots: An approach to massively parallel robotic systems with unusual motion properties. In: Robotics and Automation (ICRA), 2012 IEEE International Conference on, pp. 739-744. IEEE (2012).
10. Coutinho, M., Will, P.: Using dynamic vector force fields to manipulate parts on an intelligent motion surface. In: Assembly and Task Planning (ISATP 97), 1997 IEEE International Symposium on, pp. 200-205. IEEE (1997).
11. White, P., Yim, M.: Scalable modular self-reconfigurable robots using external actuation. In: Intelligent Robots and Systems (IROS 2007), 2007 IEEE/RSJ International Conference on, pp. 2773-2778. IEEE (2007).
12. Knaian, A.: Electropermanent magnetic connectors and actuators: devices and their application in programmable matter. PhD dissertation. Massachusetts Institute of Technology (2010).
13. Wang, H., Rubenstein, M.: Autonomous mobile robot with independent control and externally driven actuation. In: Intelligent Robots and Systems (IROS 2016), 2016 IEEE/RSJ International Conference on, pp. 3647-3652. IEEE (2016).
14. Hilder J., Horsfield A., Millard A.G., Timmis J.: The Psi Swarm: A Low-Cost Robotics Platform and Its Use in an Education Setting. In: Alboul L., Damian D., Aitken J. (eds) Towards Autonomous Robotic Systems. TAROS 2016. Lecture Notes in Computer Science, vol 9716. Springer, Heidelberg (2016).
15. Bonani, M., Raemy, X., Pugh, J., Mondana, F., Cianci, C., Klaptocz, A., Magnenat, S., Zufferey, J.C., Floreano, D., Martinoli, A.: The e-puck, a robot designed for education in engineering. In: Proc. of the 9th Conference on Autnomous Robot Systems and Competitions, vol. 1, pp. 59–65 (2009).
16. McLurkin, J. et al: A Robot System Design for Low-Cost Multi-Robot Manipulation. In: Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on, pp 912-918. IEEE (2014).
17. Duvallet, F., et al: Developing a Low-Cost Colony. In: Distributed Intelligent Systems, AAAI Fall Symposium 2007 on (2007).
18. Moore, D., Leonard, J., Rus, D., Teller, S.: Robust distributed network localization with noisy range measurement. In: Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, pp 50-61. ACM (2004).