Implementing Collective Behaviors Using the Kilobot Platform

Franchesca Agatha V. Beltran, Hazel Anne P. Cruzat, Gia Nadine M. De Sagun and Elmer R. Magsino, *Member, IAENG*

***Abstract*—In this paper, a starter pack of 10 Kilobots are used to provide an initial study and implementation of collective behaviors that can be used in military operations, disaster response, and intelligent transportation systems. Basic target-surrounding and search swarm algorithms are demonstrated. For all these behaviors, there are three basic roles, namely, searcher/walker, target, and base, that define the movement and response of a Kilobot. We present the empirical performance results based on the average time of accomplishing a goal and distance traveled, while sharing our personal experiences when performing the experiments, specifically information to the capabilities and limitations of the Kilobots.**

***Index Terms*—Kilobots, Search-and-Rescue, Target-Surround, Collective Behaviors.**

I. Introduction

Social insects, e.g., ants, bees, demonstrate collective behaviors. Such behaviors allow a group of robots to accomplish a designated task in a decentralized manner [1] and operate as a distributed system. Typical collective behaviors are classified into marching, oscillation, wandering and swarming [2]. The inclusion of these collective behaviors into mobile robotics has provided a new avenue of research in using robots to accomplish a certain task.

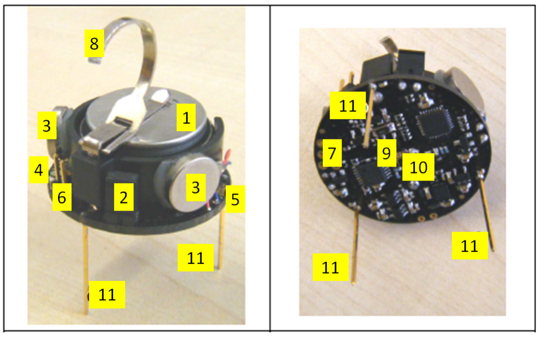

Fig. 1. The Kilobot. It is equipped with (1) 3.7V Battery (-up, +down), (2) Power Jumper, (3) Vibration motors, (4) LED (Red/Green/Blue), (5) Ambient light sensor, (6) Serial output header, (7) Direct programming socket, (8) Charging Tab, (9-10) IR transmitter/receiver, (11) Robot legs.

There are many robotic platforms developed for implementing collective behaviors among mobile robots. Some commercially available platforms are Spiderino [4], Colias [5] and Kilobots [3]. In this work, we use Kilobots to perform and evaluate our developed collective behavior algorithms.

Manuscript received August 1, 2017; revised October 6, 2017.

F. A. V. Beltran, H. A. P. Cruzat, G. N. M. De Sagun and E. R. Magsino are with the Department of Electronics and Communications Engineering, Gokongwei College of Engineering, De La Salle University, Manila, Philippines e-mail: elmer.magsino@dlsu.edu.ph.

The Kilobot, shown in Fig. 1, is a robotic platform designed by the Self-Organizing Systems Research Lab of Harvard University. It has a diameter of 3.3 cm and runs on a rechargeable 3.7V battery. This small framework contains various sensors and actuators. One thing unique about the Kilobots is the use of vibration motors for movement.

There are already some published research works using Kilobots to display different collective behaviors. In [6], a genetic algorithm on a lookup table is applied to demonstrate various Kilobot behaviors such as orbiting, navigation and dispersal. In [7], search and rescue algorithms are demonstrated. In [8], the simulation software of the Kilobot is used as an aid to teach elementary mathematics. These previous works, however, presented only simulation results.

There are still few published works that showcases empirical results using Kilobots as agents for collective behavior demonstrations. The work in [9] only presented Kilogrid. It is a platform used to simply extend the sensory system of a Kilobot and to create virtual Kilobot actuators. Experimental results from using Kilobot are shown in [10]. They used 100 Kilobots to evaluate a decision-making strategy that requires only agents with minimal capabilities and focused on the speed and accuracy tradeoff of their proposed strategy.

In this paper, we demonstrate collective behavior algorithms that can be applied in military operations, disaster response, and intelligent transportation systems by using the starter pack of 10 Kilobots. In general, we evaluate our experiments based on the duration of accomplishing the desired objective and how many Kilobots are able to perform a successful task. From these results, we are able to successfully demonstrate basic collective behaviors and most importantly, include in this paper our experiences regarding the capabilities and limitations when using Kilobots in any experiments.

The rest of the paper is organized as follows. Section II briefly discusses the Target-surrounding and Search-and-Rescue algorithms. Section III presents the experimental setup and discusses the results obtained from several experiment runs. We also present the difficulties we encountered during implementation in Section IV. Finally, Section V concludes the research.

II. Collective Behavior Algorithms for Kilobots

In this section, we briefly discuss the collective behaviors that we are going to implement in the starter pack of Kilobots. The detailed discussion about the target-surrounding and search-and-rescue collective simulation results using Kilobots is presented in [7].

ISBN: 978-988-14048-8-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

First, we define the following Kilobot behaviors. Searchers/walkers are moving Kilobots. They search and surround the target respectively. A Target is a mobile or immobile Kilobot that must be found and surrounded by searchers/walkers. Finally, the Base is a stationary Kilobot that indicates the area of return. This may not be present in a behavior being implemented.

Search-and-Rescue locates a specific target and returns to a designated part in the field. Once the desired Kilobot is found, the Target-surrounding method encloses a designated target until the desired return location is reached.

Target-surrounding performs gradient assignment for estimating the distance of one searcher to a target. The objective is that all Kilobots have the same estimated distance and are approximately 3.3 cm away from the target Kilobot.

There are three simple search behaviors to find a specific Kilobot. These are square formation, dispersal and sweep search.

The square formation maintains a square pattern of Kilobots while searching the field. This allows a group of Kilobots to have connections among the members of the group and only one elected leader is in contact with the other square-groups.

On the other hand, the dispersal algorithm randomly places all searchers in the field and allows them to elect only one leader based on the gradient assignment. If they all have the same values, a leader is chosen randomly.

Finally, the sweep search proceeds from a 4×2 formation of Kilobots. This formation is maintained throughout the sweep search.

Before proceeding to performing our experiments, we did initial tests to characterize the signal strength between two Kilobots separated by a certain distance D. This is shown in Fig. 2.

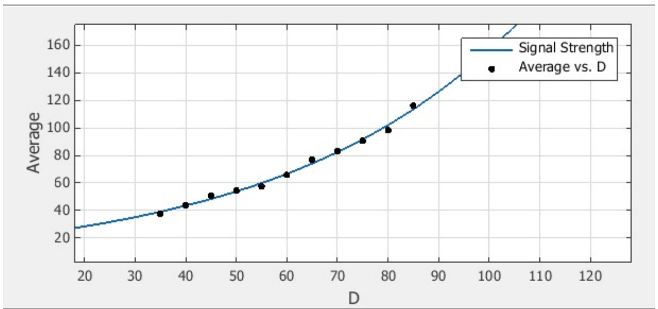

Fig. 2. Actual vs Measured Distance between two Kilobots.

The x-axis represents the physical distance (measured in mm) at which the two Kilobots are spaced. The y-axis on the other hand represents the average of the sensor readings of the distance between two Kilobots based on the signal strength of the IR message received. The signal strength is measured for 50 trials and the points of the plot represent the average of those trials. From these, we are able derive an analytical expression that we can use to determine the physical distance when we read the values of the proximity sensors.

III. Results and Discussion

The collective behavior algorithms are implemented by using 10 Kilobots on a whiteboard playing field that has a maximum dimension of 1.0 m $\times$1.0 m. As much as possible, the playing field must not be directly below a source of light.

TABLE I
Execution Time and Number of walkers reaching the target.

Trial #	Dispersal Time (min)	Total Execution Time (min)	# of walkers reaching the target	Successful Trial?
1	00:10.96	07:46.38	5	Yes
2	00:17.78	08:14:59	4	No
3	00:08.53	09:29.58	4	No
4	00:05:57	07:58.56	6	Yes
5	00:07.08	06:50.35	3	No
6	00:22.25	13:25.69	6	Yes
7	00:14.32	12:53.11	6	Yes
8	00:23.46	09:28.32	4	No
9	00:28.69	06:24.89	4	No
10	00:16.98	07:46.36	3	No
11	00:15.77	12:34.44	6	Yes
12	00:15.56	06:03.59	5	Yes
13	00:18.88	11:30.36	5	Yes
14	00:22.28	11:15.29	6	Yes
15	00:20.65	10:27.26	6	Yes

A. Target-Surrounding Algorithm

The walkers' objective is to surround the target Kilobot (dark pink/purple led) regardless of the formation. The Dispersal algorithm is executed first before to allow the Kilobots walkers to spread apart. When at least one walker is more than 70 mm away from the group, the target Kilobot is placed less than 3 cm away from one of the walkers. Fig. 3 shows ordered images of the algorithm during one of the experiments.

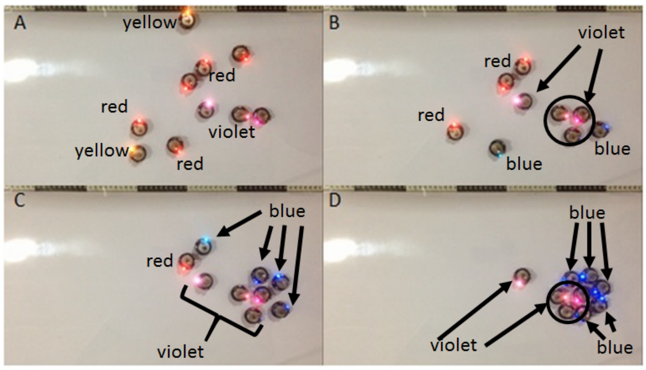

Fig. 3. Target-surround experiment results. (A) Start of the algorithm, (B-C) Slowly converging to the target Kilobot, and (D) Target has been surrounded.

The algorithm's performance is evaluated based on the number of walkers surrounding the target and the time duration to complete the algorithm. We measure the time duration from the point when the target Kilobot is placed inside the arena until it is surrounded by enough beacons, such that the remaining walkers can no longer move as close as 50 mm to the target. Therefore, the target Kilobot is considered totally surrounded when there are at least five (5) encircling Kilobots.

Out of the 15 trials, only nine are successful. As shown in Table I, the slowest successful execution time is seen on Trial 6, 13 min and 25 sec, while the fastest completion time is 6 min and 3 sec, Trial 12. The average time for the successful trials is equal to 10 min and 21 sec.

ISBN: 978-988-14048-8-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

For the nine successful trials that we achieved, we observed how fast each successful walker is able to reach the target Kilobot. Fig. 4 illustrates these findings. Given these data, we observed that the average time increased as more walkers have reached the target. Ideally, this time should be decreasing. We attribute this inaccuracy to the election between walkers. We further study this shortcoming in one of our future studies.

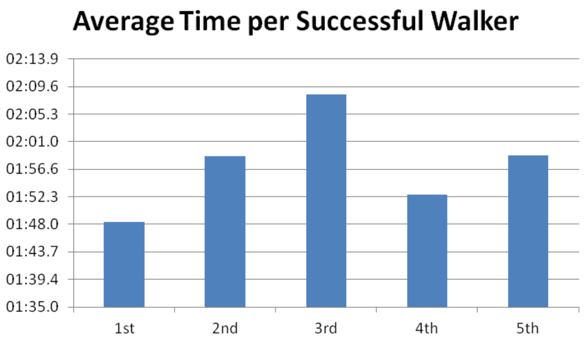

Fig. 4. Average time it takes for a successful walker to surround a target Kilobot.

B. Square Formation Search Algorithm

Formation control is an important aspect in the Square Formation algorithm. The Square Formation Search Algorithm uses two groups to search for a Target Kilobot. The search is considered to be completed once these two groups and the Target Kilobot are all in contact. Target Surrounding Algorithm is then performed. Fig. 5 shows how a square group moves. The average travel distance while maintaining the square formation is 40.4 cm at an average time of 3 min and 24 sec.

Fig. 5. Actual experimental results for maintaining the Square Formation.

We also noticed that in this algorithm, the Kilobots spent more time waiting to reassemble, (See Fig. 6). This happens in the event that the remaining group of searchers move in the opposite direction of the waiting group, e.g., Trial 4. This caused the average time until both groups meet to be higher than the average time until the target is found, which expresses that the algorithm spends more time on rescuing than on searching for the target.

C. Dispersal Search Algorithm

The Dispersal Search Algorithm finds the Target Kilobot and returns to the base once all the searchers are complete,

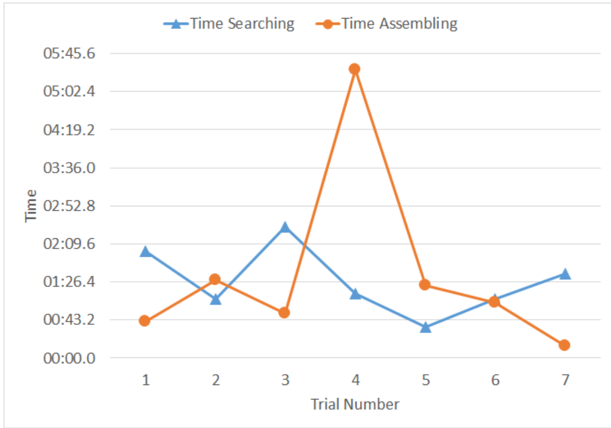

Fig. 6. Searching Time vs. Reassembling Time for Square Formation search algorithm. Average time until target is found = 1 min, 27 sec. Average time until both groups meet = 1 min, 35 sec. Average execution time for Target-Surrounding algorithm = 11 min, 42 sec. Average total execution time for Square Formation algorithm = 14 min, 44 sec.

including the target. Fig. 7 shows the snapshots taken during one of the experiments. Fig. 7 (A) illustrates the starting formation of the dispersal search. In Fig. 7 (B), the violet colored Kilobot located at the middle of the arena serves as the base while the other one serves as the target. The red colored Kilobots are the searchers. If these searchers found the target, then, they turn to color blue or orange (Fig. 7 (C)). A blue color means the Kilobot is in wait state. Orange denotes a moving state for the leader. At this point the target becomes a follower Kilobot. This action is depicted by a green light (Fig. 7 (D)). Once a searcher learns that the target is already found, it then becomes a follower and changes its color to green (Fig. 7 (E)). If the searchers are already complete as seen in Fig. 7 (F), all Kilobots turn to the same color as the base, signaling that they already reached the base.

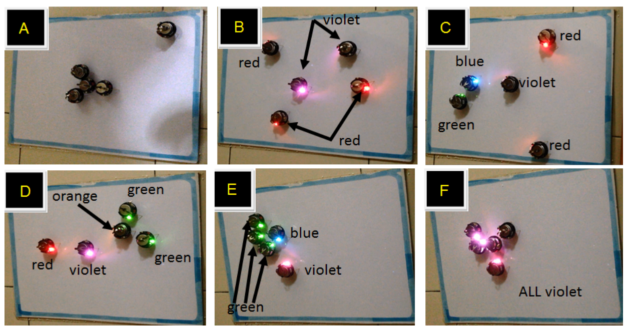

Fig. 7. Dispersal Search Algorithm Experimental Results.

This algorithm is implemented in four different scenarios as enumerated below.

1) An arena having 40 cm by 30 cm in dimension and having 3 searchers to find the target Kilobot.
2) An arena having 40 cm by 30 cm in dimension and having 5 searchers to find the target Kilobot.
3) An arena having 40 cm by 30 cm in dimension and having 8 searchers to find the target Kilobot.
4) An arena having 1 m by 1 m in dimension and having 8 searchers to find the target Kilobot.

ISBN: 978-988-14048-8-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

We performed this experiment to provide an empirical result to support the fact that the more the searchers are, the faster it finds the target. Obviously, the Kilobots are able to cover more area for searching, thus the faster it finds the target. This is shown in Fig. 8. We are also able to find that the maximum number of Kilobots that can be used in locating a target given this arena size of 40 cm by 30 cm is equal to five (5). However, we can no longer determine such number when we increased our arena size due to the small number of Kilobots that we have.

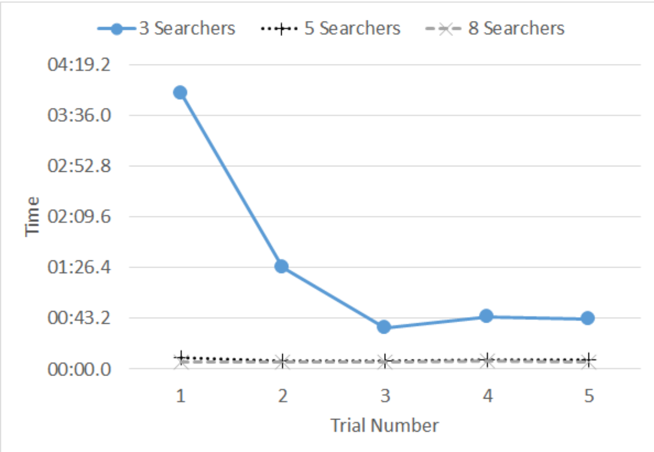

Fig. 8. Average time until target is found when using the Dispersal Search algorithm.

We also observe that there are instances wherein the searchers cannot easily find the target. When the searcher's distance from the target is approximately equal to 6 cm, sometimes, the searcher still changes its direction due to its random movements. The Kilobot's movement is not easily controllable because its motion comes from the vibration of the motors.

Another obvious observation is that the more the searchers are, the longer time it takes for these to return to the base. One factor is the overflow of data in wireless communication. Since there is a number of searchers trying to maintain its communication with the leader, there are instances where the followers lose their connections to the leader because the message being sent by the leader cannot reach the other followers anymore. This also affects the message being sent by the leader to the other searchers informing them that the target is already found, thus, the searchers do not respond to the leader even though they are near to each other. Some followers cannot get near the leader because the other followers are already blocking the way.

D. Sweep Search Algorithm

In sweep search, we performed 15 trials and noted the point where the formation shown in Fig. 9 (a) will break. Vertical and horizontal separations between two adjacent Kilobots are equal to 4.5 cm and 9 cm respectively. We pointed out earlier that controlling the movements of the Kilobot is very hard to achieve since we only depend on the vibrations. We also add the fact that when the energy of the battery is slowly decreasing, the motion of the Kilobot is also affected.

Also, when two Kilobots fail to communicate properly, shade is immediately applied to prevent ambient light from interfering.

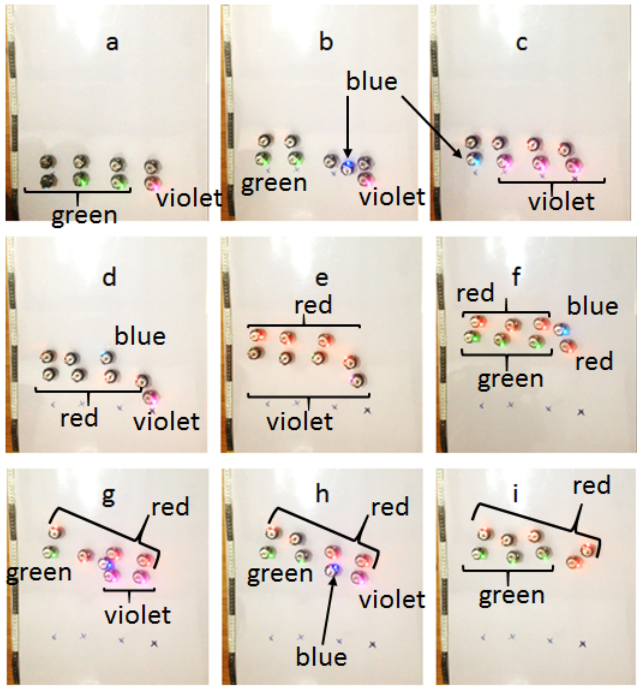

Fig. 9. Actual results for the sweep search algorithm.

Fig. 10 presents the Time vs. Accumulated Distance of every iteration for each trial. During the first iteration of all trials, the distance values converge to 20 cm. However, as more iterations progress, the distances begin to vary from one trial to another. This indicates how deviations occur more often as more iterations take place. Most of the trials end after the third iteration and only two out of the 15 trials are able to reach the fifth iteration. The slope of the graph lessens as time progresses indicating how the formation is no longer able to move forth at the same rate as it is during the first iteration. We define an iteration as an event when one group of 4 Kilobots have overtaken the other group.

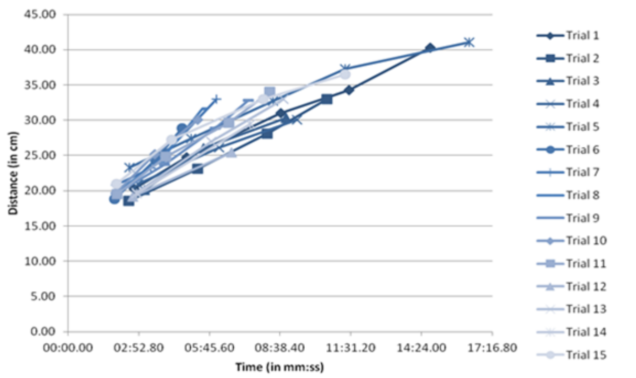

Fig. 10. Time vs. Accumulated Distance Plot for Sweep Search algorithm distance test.

After multiple tests of the sweep algorithm, the target and base are incorporated. The target is usually placed in areas that are reachable by the searchers in a maintained formation. The base Kilobot is placed at 40 cm in all of the trials. The

ISBN: 978-988-14048-8-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

searchers will execute their normal sweep algorithm. Once the target is found, they temporarily act as beacons until the target gets to the leftmost searcher, then, the searchers resume their states and continue the iterations until the base is encountered. Fig. 11 illustrates that the addition of a target poses no significant differences in the sweep algorithm alone.

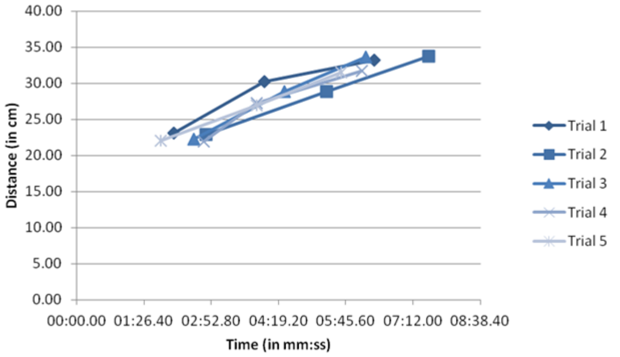

Fig. 11. Time vs. Accumulated Distance Plot for Sweep Search algorithm.

IV. Experiences when Implementing Collective Behaviors on Kilobots

Months of experience in handling the Kilobots directed us towards a certain way of implementing our algorithms. We also have realizations regarding the Kilobot's capabilities and how can we manipulate these to our advantage. We enumerate these below.

1) It is more reliable to use an alternate sequence of counter-clockwise and clockwise movements to achieve a straight walking pattern. Motor calibration does not provide enough precision for a straight walking pattern.
2) Although the manufacturer already provided the APIs that contain checksum for error detection, few erroneous messages are still prevalent. The countermeasure we did is to use multiple conditions when checking a message.
3) Kilobots are very delicate. It is advised that the Kilobot's legs be handled with utmost care even when returning the Kilobots to their packages.
4) The limited number of Kilobots also provided constraints in the algorithm design and implementation. Acquiring Kilobots are quite expensive.
5) The choice of a playing field is important. It should be not too sensitive to the ambient light and is smooth to allow seamless movements for the Kilobots.

V. Conclusion and Future Work

We have demonstrated in this work target-surround and search-and-rescue algorithms implemented on a starter pack of 10 Kilobots. The Target-Surrounding algorithm is only successful 60% of the time mostly because of the Kilobot's hardware problems during locomotion. The square formation algorithm failed to achieve 100% success rate because of its heavily dependence on the success of the Target-Surrounding algorithm. The Dispersal Search algorithm is always successful when locating the target and returning to the base, but at the expense of a longer execution time. Finally, the sweep search algorithm is also successful on all trials because the base is strategically placed on the average distance traveled by the sweep searchers determined during the distance test.

The Dispersal Search Algorithm is 95% faster than the Square Formation Algorithm in finding the target. However, the Square Formation algorithm is 84% faster in reassembling with the rest of the searchers. This goes to show that the speed of the rescue operation is sacrificed in order to allow a faster and more effective search operation. The advantage of the Dispersal Search Algorithm is that it can immediately determine the location of a victim during Search-and-Rescue operations. However, it takes too long to finish the rescue operation and bring the target to safety. The Square Formation, on the other hand, spends more time searching for the target, but less in waiting for all searchers to reassemble. Through further observation of our results, the Square Formation algorithm may be considered to be more effective as it has a shorter execution time as compared to the Dispersal Search algorithm. We will investigate the hybrid of the algorithms in the future taking note the advantages of the demonstrated collective behaviors.

We further explore the use of some of these algorithms in underwater vehicles [11] and aerial drone platforms [12], [13], since these are much cheaper and faster to acquire than the Kilobots.

References

[1] C. R. Kube and H. Zhang, "Collective robotics: From social insects to robots," *Adaptive behavior*, vol. 2, no. 2, pp. 189–218, 1993.
[2] K. Sugawara and T. Watanabe, "Swarming robots-collective behavior of interacting robots," in *SICE 2002. Proceedings of the 41st SICE Annual Conference*, vol. 5. IEEE, 2002, pp. 3214–3215.
[3] M. Rubenstein, C. Ahler, N. Hoff, A. Cabrera, and R. Nagpal, "Kilobot: A low cost robot with scalable operations designed for collective behaviors," *Robotics and Autonomous Systems*, vol. 62, no. 7, pp. 966–975, 2014.
[4] M. Jdeed, S. Zhevzhyk, F. Steinkellner, and W. Elmenreich, "Spiderino-a low-cost robot for swarm research and educational purposes," in *Intelligent Solutions in Embedded Systems (WISES), 2017 13th Workshop on*. IEEE, 2017, pp. 35–39.
[5] F. Arvin, J. Murray, C. Zhang, and S. Yue, "Colias: An autonomous micro robot for swarm robotic applications," *International Journal of Advanced Robotic Systems*, vol. 11, no. 7, p. 113, 2014.
[6] M. Beckerleg and C. Zhang, "Evolving individual and collective behaviours for the kilobot robot," in *Advanced Motion Control (AMC), 2016 IEEE 14th International Workshop on*. IEEE, 2016, pp. 263–268.
[7] E. R. Magsino, F. A. V. Beltran, H. A. P. Cruzat, and G. N. M. De Sagun, "Simulation of search-and-rescue and target surrounding algorithm techniques using kilobots," in *Control, Automation and Robotics (ICCAR), 2016 2nd International Conference on*. IEEE, 2016, pp. 70–74.
[8] Z. Hui, J. Rojas, L. Lin, H. Ting, and C. Zhao, "A swarm framework for teaching elementary addition operations," in *Robotics and Biomimetics (ROBIO), 2014 IEEE International Conference on*. IEEE, 2014, pp. 1027–1032.
[9] A. Antoun, G. Valentini, E. Hocquard, B. Wiandt, V. Trianni, and M. Dorigo, "Kilogrid: A modular virtualization environment for the kilobot robot," in *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*. IEEE, 2016, pp. 3809–3814.
[10] G. Valentini, E. Ferrante, H. Hamann, and M. Dorigo, "Collective decision with 100 kilobots: speed versus accuracy in binary discrimination problems," *Autonomous Agents and Multi-Agent Systems*, vol. 30, no. 3, pp. 553–580, 2016.
[11] E. R. Magsino and R. C. Gustilo, "Visual surveying control of an autonomous underwater vehicle," in *Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM), 2014 International Conference on*. IEEE, 2014, pp. 1–5.

[12] E. R. Magsino, J. R. B. Chua, L. S. Chua, C. M. de Guzman, and J. V. L. Gepaya, "A rapid screening algorithm using a quadrotor for crack detection on bridges," in *Region 10 Conference (TENCON), 2016 IEEE*. IEEE, 2016, pp. 1829–1833.
[13] J. R. M. Cornejo, L. M. A. Mapili, R. Pesit, E. N. Young, and E. R. Magsino, "Visual simultaneous localization and mapping for co-operating unmanned aerial vehicles," in *21st Mechatronics and Machine Vision in Practice, M2VIP 2015*. Mechatronics and Machine Vision in Practice, 2015.