The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Self-configuring Robot Swarms with Dual Rotating Infrared Sensors

Geunho Lee, Seokhoon Yoon, Nak Young Chong, and Henrik I. Christensen

Abstract— **This paper presents practical design and hardware implementation issues of self-configuring swarms of autonomous mobile robots. For the purpose, we develop a new low-cost position detection system that we call dual rotating infrared (DRIr) sensor. The DRIr sensor can provide robots with advanced sensing capabilities that give reliable information about the position and surface geometry of neighboring robots and obstacles. Special focus is placed on how to realize the observation and object identification of mobile robots through the use of DRIr sensors. We verify the functionality and performance of the DRIr sensors mounted on a commercial mobile robot. Experimental results show that a swarm of mobile robots equipped with the DRIr sensors can autonomously configure themselves into an area.**

I. INTRODUCTION

With recent advances in robotic and wireless networking technologies, much attention has been paid to potential applications for swarms of mobile robots. Robotic swarms are expected to be applied to various areas such as habitat or environmental monitoring, surveillance, and exploration [1]. In those applications, individual robots are assumed to be simple, cheap, and disposable. One of the technical challenge is how to develop large-scale swarms of robots at a reasonable cost, enabling us to deploy them effectively into a wide geographical area. While many different algorithms have been proposed to coordinate the actions of individual robots, few of them have been embodied in low cost yet full featured robots.

The main purpose of this paper is to present empirical results using our self-configuration algorithm [10][11] with physical robots dispersing themselves in a 2-D plane as illustrated in Fig. 1. What is important from the practical point of view is to provide robots with reliable information about the location of neighboring robots and obstacles. For the purpose, the dual rotating infrared (DRIr) proximity sensor is developed and mounted on the front and rear edge of each robot. This allows robots to detect the surface of neighboring robots and obstacles in all directions. Particularly, this paper presents how to realize each robot's observation function through the use of DRIr sensors. We performed experiments to demonstrate forming of equilateral triangle lattices by commercial robots, where the DRIr sensors allow robots to obtain effective relative position sensing. It is further expected that the DRIr sensor-equipped robot swarms can be deployed to achieve self-configuring robotic sensor networks in a real-world situation.

G. Lee, S. Yoon, and N.Y. Chong are with the School of Information Science, Japan Advanced Institute of Science and Technology, Ishikawa 923-1292, Japan {geun-lee, seokhoon, nakyoung}@jaist.ac.jp

H.I. Christensen is with the College of Computing, Georgia Institute of Technology, Atlanta, GA 30332, U.S.A. hic@cc.gatech.edu

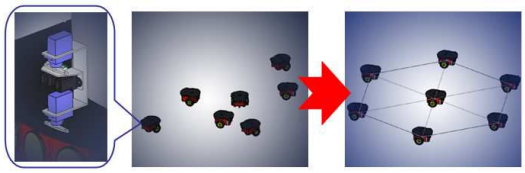
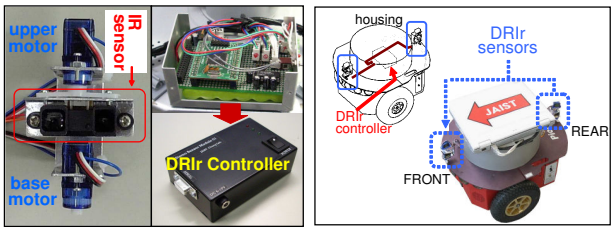


Fig. 1. The concept of self-configuration of robot swarms

II. BACKGROUND

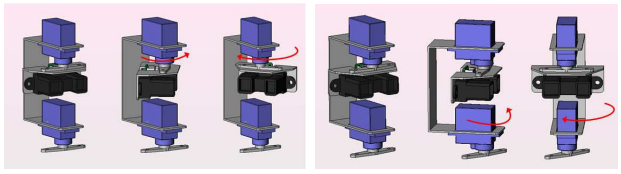
Self-configuration of large-scale robot swarms has been reported in the literature [2]-[11], mostly based on the intuition observed from a social organism or physical phenomena called behavior-based [2]-[3] or virtual physics-based [4]-[9] approaches. Many of these approaches used such physical phenomena as electric charges [3], gravitational forces [4], spring forces [5]-[7], potential fields [8], and other virtual models [9]. The configuration achieved resulted in lattice-type networks that offer high level coverage and redundant connections. The network can be classified into fully and partially connected topologies [13]. Fully connected topologies have each robot interact with all of other robots simultaneously. Thus, those approaches might over-constrain individual robots and lead to deadlocks. On the contrary, with partially connected topologies, robots interact selectively with other robots [6]. Therefore, robots are enabled to achieve faster formation and avoid getting stuck into deadlocks [7]. In our earlier work [10], we presented our self-configuration algorithm that enables robots to configure themselves into a 2-D plane with geographic constraints. The local nature of interaction based on a partially connected topology allowed robots to converge to an equilateral triangle with their two neighbors. By collecting such local behaviors, robot swarms was uniformly dispersed over an area.

An important issue is how to coordinate a large number of robots without using costly hardware solutions [12]. Here we pay particular attention to the proximity sensor. The use of high accuracy sensors helps ensure accurate distance measurement, but can become costly. For instance, we have spent USD 100 on one set of DRIr sensor rotating 360 degrees. On the other hand, one of the most widely used laser sensors (Hokuyo Ltd.'s URG-04LX) with 240 degree range is 20 times expensive than our prototype. In most cases, robots need to be able to observe 360 degrees. The most common option is to place an adequate number of sensors at equal intervals around the circumference of the robot. Due



(a) DRIr sensor and its controller (b) Sensor installation

Fig. 2. Mobile robot equipped with DRIr sensors



(a) rotation of the upper motor (b) rotation of the base motor

Fig. 3. Combined motion of sensor rotation

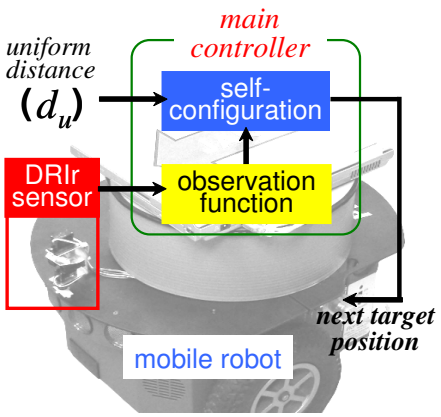


Fig. 4. Mobile robot control schematic

to technical difficulties, sensors may not be installed at equal intervals. Therefore, we develop a low-cost proximity sensor capable of 360 degree rotation.

III. Robot and Sensor System Setup

A. DRIr Sensor

Fig. 2 illustrates our DRIr sensors prototype. The overall system consists of a pair of DRIr sensors and their controller. A DRIr sensor has two MiniStudio MiniS RB90 servo motors and one Sharp GP2Y0A02YK infrared sensor. In detail, the Atmel ATmega128 microcontroller controls each servo motor rotating the infrared sensor and feeds the measured data to the main controller of the robot. The DRIr sensor controller forwards two-channel control signals to the front and rear DRIr sensors. One signal controls the rotation angle of each servo motor by pulse width modulation. The other signal is used for on-off control of the infrared sensor. Moreover, the analog output voltage representing the sensor-to-object distance is fed to the controller and converted to 10-bit digital values. From the relationship between the analog voltage level and the measured distance, the range from 12 *cm* to 180 *cm*, where the voltage level decreases with increasing distance in a unimodal fashion, can be used to estimate the distance. Each robot can be provided with a sensing range up to 400 *cm* including the size of the robot body.

B. Combined Motion of Sensor Rotation

One servo motor rotates up to 180 degrees, thus two identical motors can sweep 360 degrees. As illustrated in Fig. 3, the base motor enables the infrared sensor to be directed toward a specific direction, while the upper motor can rotate at higher speed with respect to the direction of the base motor. By the combination of the base and upper motors, a wide variety of emitting directions of infrared rays can be effectively controlled. In detail, the base motor rotates 180 degrees in azimuth with respect to the heading of the robot, and the upper motor adds another 60 degrees. The remaining 120 degree range cannot be observed since the line-of-sight path is blocked by the controller housing, but is covered by the rear DRIr sensor (that scans the same amount of range in the opposite direction). Therefore, a pair of DRIr sensors can cover a full 360 degrees.

C. Mobile Robot System

Our customized mobile robot largely consists of three parts: a pair of DRIr sensors, MobileRobots Pioneer 3-DX platform, and the main controller. The DRIr sensors are mounted on the front and rear edge of the robot (See Figs. 2-(b) and 4). This allows each robot to detect other robots in the front and rear direction simultaneously. The circular controller housing designed for the controller board represents the surface geometry whose center point is easy to detect irrespective of the robot's heading. A laptop PC is used as the main controller on top of the robot. The main control function consists of observation and self-configuration [10]. Fig. 4 shows the control schematic of the robot system. The inputs to the main controller include the measurement data obtained by the DRIr sensors and the predefined uniform interval between neighboring robots. Then, the robot's movement point at the next time instance can be computed through the self-configuration algorithm. Details on these functions will be further explained in the following sections.

IV. Observation Function

We consider a swarm of mobile robots denoted as $r_1, \cdots, r_n$. It is assumed that an initial distribution of all robots is arbitrary and distinct. Robots have no leader and no identifiers, and do not share any common coordinate system, and do not retain any memory of past actions and states. Due to limited sensing range, they can detect other robots only within a certain range. Notably, each robot is not allowed to communicate explicitly with other robots. In addition, each robot executes an identical algorithm, and acts independently of each other. At each time, each robot computes their movement position by the self-configuration algorithm [10] [11] (computation), based on the output of the observation function (sensing), and moves toward the computed positions (motion). A recursive series of these activations is controlled by the main controller. Notice that on top of each robot, the

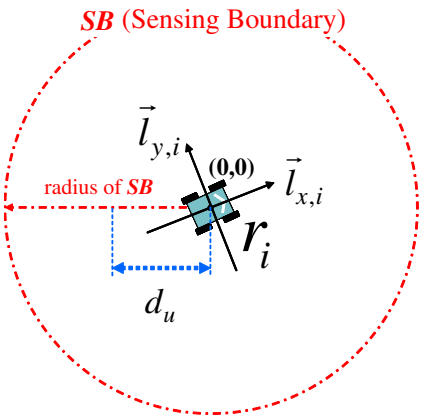


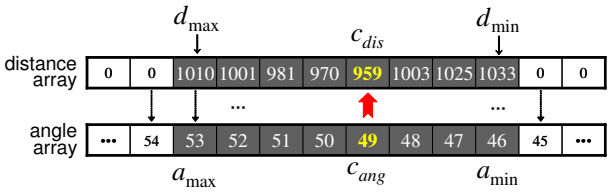
Fig. 5. Illustration of r_i's local coordinates

same circular controller housing with diameter d_r is mounted to detect and compute the robot's center position easily. In addition, all objects and obstacles are assumed to be larger than d_r.

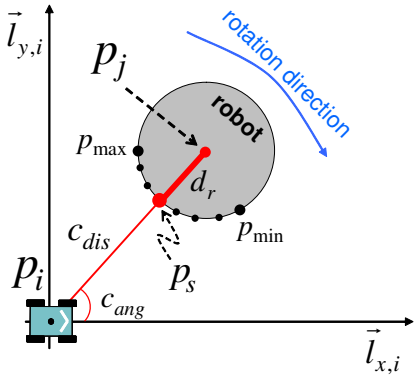
Let us consider a robot r_i with its local coordinates ($\vec{l}_{x,i}$ and $\vec{l}_{y,i}$) as shown in Fig. 5. Here, $\vec{l}_{x,i}$ defines the vertical axis of r_i's coordinate system as its heading direction. The horizontal axis $\vec{l}_{y,i}$ can be decided by rotating $\vec{l}_{x,i}$ 90 degrees counterclockwise. The center position of r_i is denoted as p_i. Accordingly, p_i is $(0, 0)$ with respect to r_i's local coordinates. The distance between p_i and p_j is denoted as $dist(p_i, p_j)$. We denote d_u as the desired interval between r_i and r_j. Next, r_i detects the center position $\{p_1, p_2, \cdots\}$ of other robots located within its sensing boundary *SB*, yielding a set of the positions O_i with respect to its local coordinates. Now, r_i can select the two robots r_{s1} and r_{s2} within its *SB* that we call the neighbors of r_i and denote the set of their positions, $\{p_{s1}, p_{s2}\}$, as N_i. Given p_i and N_i, the *Triangular Configuration*, denoted by $\mathbb{T}_i$, is defined as a set of three distinct positions $\{p_i, p_{s1}, p_{s2}\}$, where the internal angle $\angle p_{s1}p_ip_{s2}$ of r_i is denoted by α_i. We define the *Equilateral Configuration*, denote by $\mathbb{E}_i$, as a configuration that all the distance permutations of $\mathbb{T}_i$ are equal to d_u.

The key to the observation function lies in obtaining reliable estimates of the surface of neighboring robots and obstacles or arena borders, which can be obtained through the following steps detailed below.

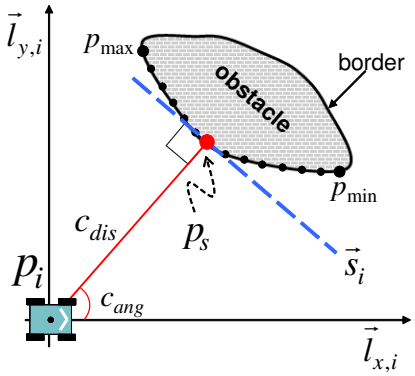
The measurement step constructs two one-dimensional arrays in the memory of each robot as illustrated in Fig. 6-(a). Here, the dimension of each array can be automatically adjusted according to the angular interval of the servo motor. When r_i scans its environment using its DRIr sensors at regular intervals, the distance to the surface of neighboring robots and obstacles is recorded in the corresponding cell of the first array. At the same time, the servo motor angle is recorded in the second array so that the distance array corresponds to the motor angle array. Next, r_i checks their distance array cells that contain a non-zero value (from the lower bound d_{min} to the upper bound d_{max}) and reads the corresponding angle array cells. By computing the average



(a) distance and angle arrays



(b) detecting the neighbor robot



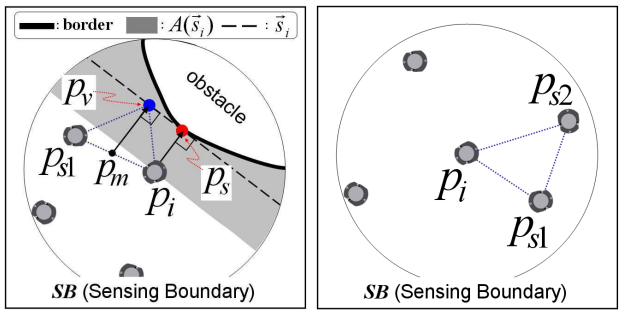
(c) detecting the obstacle

Fig. 6. Observing robots and obstacles using DRIr sensors

of a sequence of numeric values in the motor angle array, r_i selects the cell containing the value equal or closest to the average, and sets the center angle c_{ang} to this value. The distance cell corresponding c_{ang} is defined as the center distance c_{dis}.

The update step calibrates the measurement data with respect to a reference. For the purpose, a 100 $\times$ 100 2-D grid with 4 cm $\times$ 4 cm unit cells is built. While recording data in the distance and angle arrays, the estimated distance is simultaneously stored in the corresponding cell of the grid as an integer intensity value. Once a full 360 degree scanning is completed, the Sobel edge detection algorithm [15] improves the original surface detection data.

The recognition step distinguishes between robots and obstacles. r_i collects the cells with the non-zero value from d_{min} to d_{max} in the updated distance array. Then, the three feature points, p_{min}, p_{max}, and p_s, are specified using d_{min}, d_{max}, c_{dis}, and their corresponding cells in the angle array, respectively (See Figs. 6-(b) and (c)). p_s is computed



(a) an immovable point neighbor (b) two movable neighbors

Fig. 7. Two types of neighbors

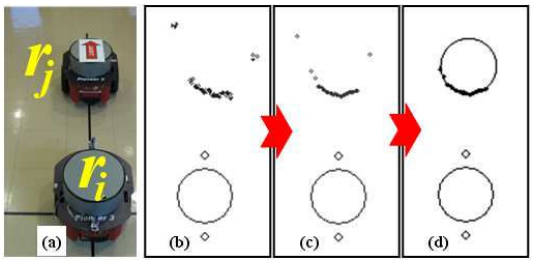


Fig. 8. Observation process of a single neighbor r_j by the robot r_i ((a) test scene, (b) measurement step, (c) update step, (d) recognition step)

based on c_{dis} with the minimum distance value among the cells and c_{ang}. Next, r_i computes $dist(p_{min}, p_{max})$ and checks whether $dist(p_{min}, p_{max})$ is shorter than the controller housing diameter. If $dist(p_{min}, p_{max})$ exceeds the diameter, the collected cells are considered as an obstacle. Otherwise, to identify the robot shape, the following two conditions should be further satisfied. The first condition tests whether the radius of curvature is reasonably similar to the controller housing. The second condition tests whether the two line segments $\overline{p_{min}p_s}$ and $\overline{p_{max}p_s}$ are symmetric with respect to r_i's local vertical coordinates, and the angle between them falls within a certain range.

Through the above process, if a robot is recognized, its center point p_j can be obtained by adding c_{dis} to the radius d_r of the controller housing (see Fig. 6-(b)). Otherwise, as illustrated in Fig. 6-(c), r_i computes the tangent vector $\vec{s}_i$ to the surface at p_s. It is obvious that $\vec{s}_i$ is perpendicular to the vector $\overrightarrow{p_ip_s}$. Consequently, the outputs of the observation function are O_i of neighboring robots and p_s and $\vec{s}_i$ for either obstacles or arena borders.

V. SELF-CONFIGURATION

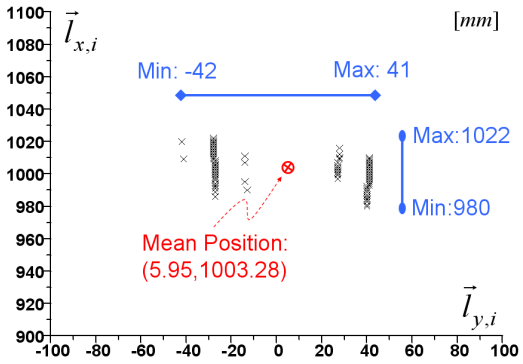
Using the outputs (O_i, p_s, and $\vec{s}_i$) of the observation function, r_i selects its two neighbors. As illustrated in Fig. 7-(a), let $A(\vec{s}_i)$ denote the area between the obstacle or arena border and the line passing through p_i and parallel to $\vec{s}_i$ within *SB*. r_i checks whether no neighbors exist in $A(\vec{s}_i)$ or $dist(p_i, p_s) \leq \frac{\sqrt{3}d_u}{2}$. If the condition is satisfied, r_i selects one neighbor and a virtual point p_v. Otherwise, r_i selects two neighbors. In detail, r_i selects and defines a neighbor located at the shortest distance from p_i as r_{s1}. With only one neighbor available, r_i computes the midpoint p_m of $\overline{p_ip_{s1}}$ and then defines the virtual point p_v projected onto $\vec{s}_i$ (See Fig. 7-(a)). Now, p_v is considered as p_{s2} of r_{s2}, and N_i is defined as $\{p_{s1}, p_v\}$. On the other hand, if the second neighbor r_{s2} is available, it is selected such that the total distance from p_{s1} to p_i passing through p_{s2} is minimized (See Fig. 7-(b)). Thus, in either case, r_i finally forms $\mathbb{T}_i$ with N_i. More details about this self-configuration algorithm and its convergence proof can be found in our prior papers [10] [11], where simulation results were provided to show how a large-scale swarm of robots converge into $\mathbb{E}_i$.

VI. EXPERIMENTAL RESULTS AND DISCUSSION

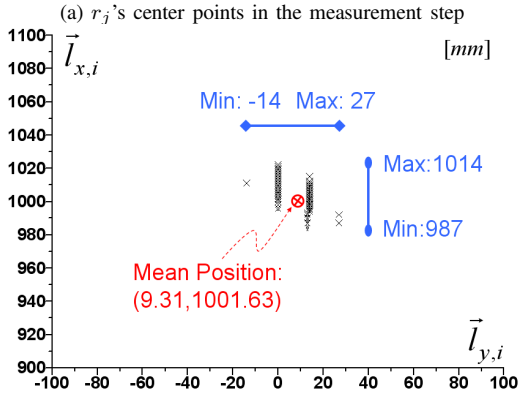
We use 5 Pioneer 3-DX robots, each of which is equipped with a pair of DRIr sensors, but unable to identify each other. Robots are initially located at arbitrary positions with different heading directions. They attempt to form a coordinated configuration starting from no *a priori* coordinate agreement, moving with a linear velocity of 150 *mm/s* and an angular velocity of 100 *deg/s*. d_u and the radius of *SB* are set to 80 cm and 150 cm, respectively. DRIr sensors emit an infrared ray every one degree while rotating with 308 *deg/s*.

To evaluate the effectiveness of the observation function, we performed four kinds of tests as shown in Figs. 8 through 11. In the first test seen in Fig. 8-(a), the robot r_i observes its neighbor r_j located 100 cm away. Figs. 8-(b), (c), and (d) show the data processing results obtained through the measurement, update, and recognition steps, respectively. Compared with Fig. 8-(b), Fig. 8-(c) shows the enhanced surface detection by eliminating blurred and distorted edges. Secondly, Fig. 9 shows the results of 300 trials in the condition of Fig. 8-(a). Figs. 9-(a) and (b) show the estimation results of r_j's center points through each step of the process. Fig. 9-(c) shows the results of statistical analysis of r_j's center points estimated by the DRIr sensor and Hokuyo's URG laser scanner. The laser scanner outperforms the DRIr sensor in terms of accuracy, but the DRIr sensor also shows reasonably good accuracy. Thirdly, Fig. 10-(a) shows the data obtained from 3 sets of 300 trials performed in the same fashion with six neighboring robots. The position of neighboring robots changed in each set. Lastly, Fig. 10-(b) shows that the average error in estimating the center point of each robot resides within a 2.6 cm radius circle. From these results, the DRIr sensor scanning observation capability can be considered quite satisfactory for practical use. Moreover, through the measurement and update steps, the problem of mutual interference between opposing DRIr sensors can be overcome.

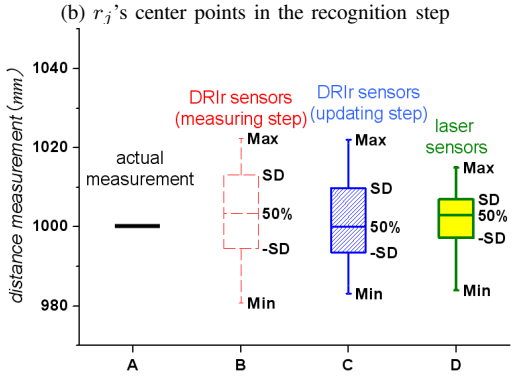
Fig. 11 presents the observation result with two neighboring robots and obstacles. Similarity, compared with Fig. 11-(b), Fig. 11-(c) shows the enhanced surface detection. Moreover, as shown in 11-(d), r_i could distinguish between robots and obstacles. From the results, we are convinced that the proposed DRIr sensor will provide robots with enhanced observation capabilities in an unknown environment. For obstacle-cluttered environments, a fusion of RFID and DRIr



(a) r_j's center points in the measurement step

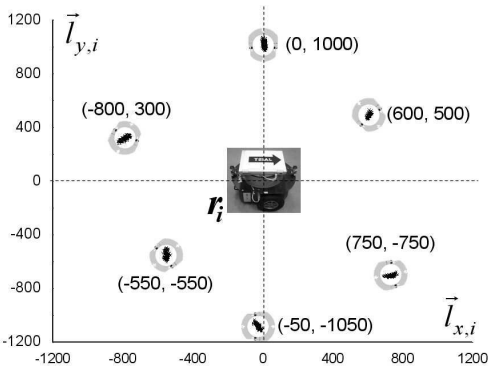


(b) r_j's center points in the recognition step

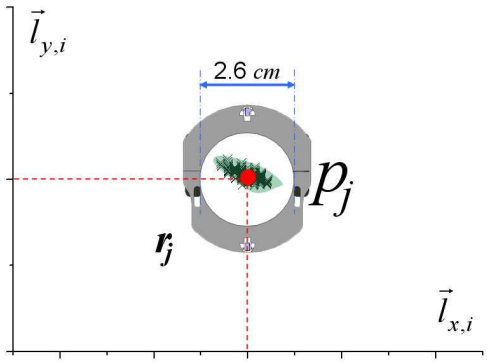


(c) comparing DRIr sensor and laser scanner

Fig. 9. Estimation of r_j's center points in Fig. 8-(a)

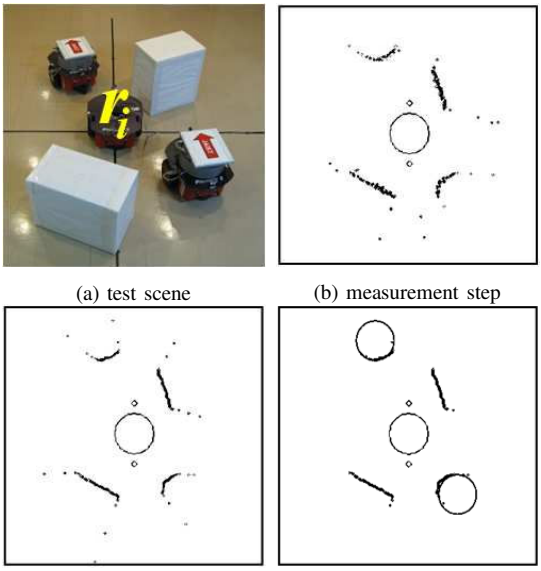


(a) Observation test of six neighboring robots (mm)



(b) average boundary of the computed center points

Fig. 10. Observation results of randomly positioned robots



(a) test scene (b) measurement step

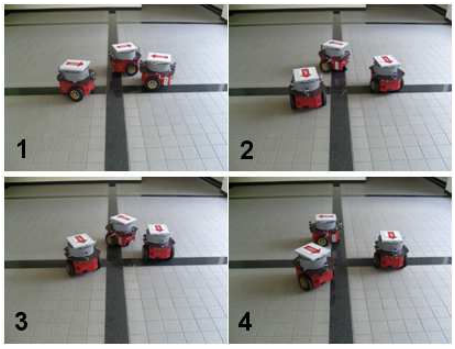
(c) update step (d) recognition step

Fig. 11. Observation process of multiple robots and obstacles

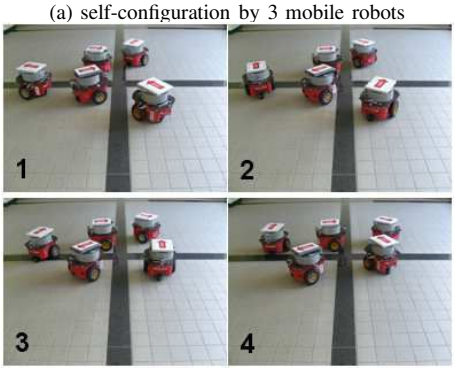
sensors could be positively advantageous [14].

For further verification, we performed self-configuration experiments as presented in Figs. 12 and 13. First, Figs. 12-(a) and (b) show the snapshots of the self-configuration process in an open space. In Fig. 12-(a), three robot could generate the desired $\mathbb{E}_i$ with d_u. In this case, robots did not need to change their neighbors throughout the process, since there existed only two robots available. However, in Fig. 12-(b), each robot changed their neighbors at each time, based on the output of their observation function. Whether or not robots should change their neighbors, each robot converged toward forming $\mathbb{E}_i$ with d_u.

Secondly, Fig. 13 shows self-configuration in a geographically constrained space. The left hand sides show the snapshots of the self-configuration process. The output of the robot r_i's observation function at each corresponding process step is shown on the right hand sides. Similarly, the output (O_i, p_s, and $\vec{s}_i$) of the observation function can be obtained by each robot. If any of the robots detects an arena border in their output, they determine whether they use the border (as a virtual immovable robot) for local interactions. In Fig. 13, r_i remained close to the border until the self-configuration was completed. Its behavior was determined through interacting with the closest neighbor and



(a) self-configuration by 3 mobile robots



(b) self-configuration by 5 mobile robots

Fig. 12. Experiment results of self-configuration in a free open space

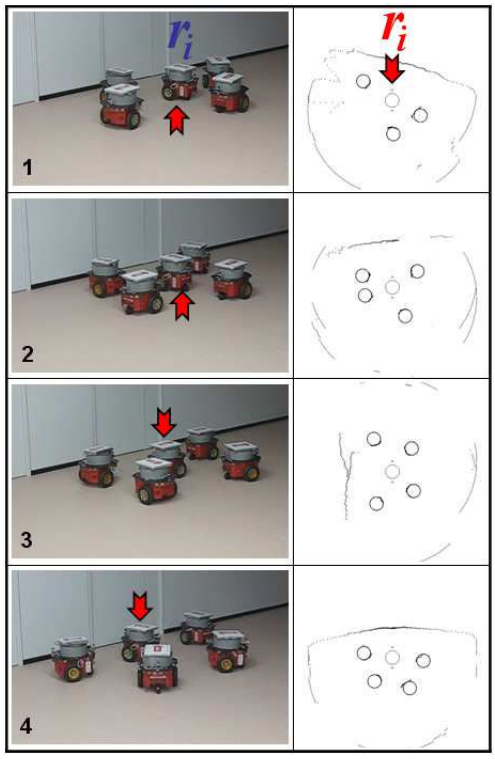


Fig. 13. Experiment results of self-configuration conforming to a flat border

the virtual robot. Consequently, these results verify that our robots equipped with DRIr sensors work satisfactorily under our laboratory conditions.

VII. CONCLUSION

In this paper, we presented practical design and hardware implementation issues for formations of mobile robots, enabling them to form equilateral triangle lattices. In particular, the proximity sensor prototype that we call DRIr sensor was developed to provide robots with full 360 degree azimuth scanning capability. By employing a pair of DRIr sensors, robots could obtain relative positioning information as well as the surface geometry of neighboring robots and obstacles in their vicinity. Robots and obstacles could be distinguished from each other based on the geometric features. We successfully demonstrated the self-configuration process of a swarm of mobile robots by exploiting each robot's capability of scanning observation. A major contribution of this work can be summarized as follows: 1) The DRIr sensor is low-cost, reliable, and easily integratable into commercial mobile robots. 2) Robots equipped with the DRIr sensors can be dispersed into unknown areas without explicit inter-robot communication.

REFERENCES

[1] H. Choset, "Coverage for robotics-a survey of recent results," *Annals of Math. and Artificial Intelligence*, vol.31, no.1-4, pp.113-126, 2001

[2] T. Balch and M. Hybinette, "Social potentials for scalable multi-robot formations," *IEEE Int. Conf. Robotics and Automation*, pp.73-80, 2000

[3] A. Howard, M. J. Mataric, and G. S. Sukhatme, "Mobile sensor network deployment using potential fields: a distributed, scalable solution to the area coverage problem," *Int. Sym. Distr. Autono. Robotic Systems*, pp.299-308, 2002

[4] W. Spears, D. Spears, J. Hamann, and R. Heil, "Distributed, physics-based control of swarms of vehicles," *Autono. Robots*, vol.17, no.2-3, pp.137-162, 2004

[5] K. Fujibayashi, S. Murata, K. Sugawara, and M. Yamamura, "Self-organizing formation algorithm for active elements," *IEEE Sym. Reliable Distr. Systems*, pp.416-421, 2002

[6] J. McLurkin and J. Smith, "Distributed algorithms for dispersion in indoor environments using a swarm of autonomous mobile robots," *Int. Sym. Distr. Autono. Robotic Systems*, pp.831-890, 2004

[7] B. Shucker, T. Murphey, and J. K. Bennett, "A method of cooperative control using occasional non-local interactions," *IEEE Int. Conf. Robotics and Automation*, pp.1324-1329, 2006

[8] J. Reif and H. Wang, "Social potential fields: a distributed behavioral control for autonomous robots," *Robotics and Autono. Systems*, vol.27, no.3, pp.171-194, 1999

[9] Y. F. Zheng and W. Chen, "Mobile robot team forming for crystallization of protein," *Autono. Robots*, vol.23, no.1, pp.69-78, 2007

[10] G. Lee and N. Y. Chong, "Self-configurable mobile robot swarms with hole repair capability," *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp.1403-1408, 2008

[11] G. Lee and N. Y. Chong, "A geometric approach to deploying robot swarms," *Annals of Math. and Artificial Intelligence*, vol.52, no.2-4, pp.257-280, 2008

[12] E. Sahin, "Swarm robotics: from sources of inspiration to domains of application," *8th International Conference on the Simulation of Adaptive Behavior* (LNCS), vol.3342, pp.10-20, 2005.

[13] S. Ghosh, K. Basu, and S. K. Das, "An architecture for next-generation radio access networks," *IEEE Network*, vol.19, no.5, pp.35-42, 2005

[14] M. Kim and N. Y. Chong, "Direction sensing RFID reader for mobile robot navigation," *IEEE Transactions on Automation Science and Engineering*, vol.6, no.1, pp.44-54, 2009.

[15] R. C. Gonzalez and R. E. Woods, *Digital image processing*, 2nd ed. Prentice Hall, 2002