The 6th Asian Control
July 18-21, 2006
Bali, Indonesia

Decentralized Formation Control for a Team of Anonymous Mobile Robots

Geunho Lee and **Nak Young Chong**

School of Information Science
Japan Advanced Institute of Science and Technology, Ishikawa 923-1292, Japan
e-mail: {geun-lee, nakyoung}@jaist.ac.jp

Abstract

We present a coordination framework for mobile robot teams performing a task through cooperation. In cooperative robotics applications, robot teams are basically required to generate and maintain a geometric pattern, and adapt to their environment. Moreover, keeping the team formation, they continue to strive toward achieving their mission even if a part of team is unable to work and function normally. Toward this end, a unified coordination strategy is proposed under the condition that robot teams are not allowed to have a direct inter-team communication, a pre-determined leader, and individual identification numbers. In this study, only by observing the other members, a robot team is enabled to reach the goal and keep its formation. In practice, image processing techniques are employed to estimate the center coordinates of the members with respect to each other. We discuss in detail the features of the proposed algorithm that includes self-organization, robustness, and flexibility. Experimental studies with four real robots demonstrate the validities of the proposed algorithm.

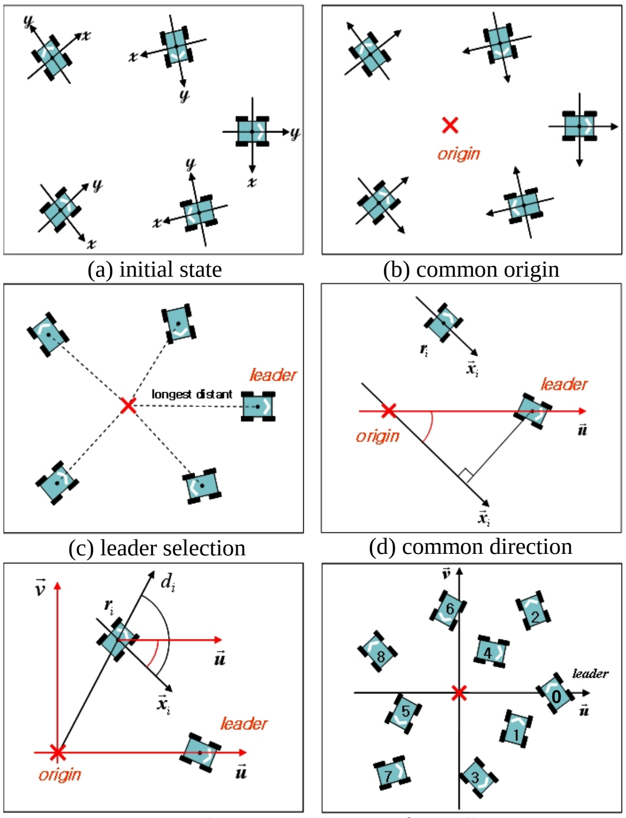
1 Introduction

Recently coordination of multi-robot teams is gaining increasing attention, because robots can perform important missions as a team in a wide variety of applications such as mine exploration, load carrying, surveillance-and-security, and search-and-rescue. In general robot teams offer many advantages over a single high performance robot in efficiency, costs per system, fault-tolerance, and generality. In order to enable a team of robots to perform a cooperative task, it is fundamentally required to coordinate team formation. Unfortunately, only a few researches have addressed a unified approach to the problem of coordinating robot teams that includes such problems as formation generation, formation keeping, and formation switching.

Suzuki and Yamashita studied the problem for generating formation of a regular polygon based on a non-oblivious algorithm [2]-[3]. Using their result, Defago and Konagaya [1] studied a self-stabilizing algorithm for the circle formation. These are emerging issues on pattern generation by cooperative robotics in the field of distributed computing. Ikemoto *et al.* [4] proposed a bio-inspired algorithm that requires robots to be initialized in a line to make different formations. In [5], all robots with an external identification formed a variety of patterns from arbitrary initial positions. Many researches related to formation maintenance have employed the method of leader-followers. Among them, Gervasi and Prencipe [7] proposed a notable computational solution for the flocking problem based on CORDA [6] with weaker assumptions on scheduling but with the ability of detecting multiplicity. In their study, the team is initially divided into a leader robot and followers. Balch and Arkin [8] studied flocking problems for four formation patterns in the context of a behavior-based control paradigm. They demonstrated that robots with a unique ID successfully maintained an assigned pattern along a relatively long path and even in an obstacle rich environment. Carpin and Parker [9] introduced a cooperative assistive navigation method in order to flock heterogeneous multiple robot teams by a leader robot with a multitude of sensors. As for formation switching, Kurabayashi and Osagawa [10] proposed an intelligent adaptive formation transition algorithm. In their study, robots initially decomposed into a leader and followers were able to change formation patterns according to a task environment while flocking. A similar problem of coordinating small-scale robot teams with a synthetic functionality was addressed in [11] for creating and maintaining team formation. Fredslund and Mataric addressed the problem of achieving formation control in a robot team, where each robot was equipped with a color helmet indicating their ID. Before robots generate a formation, robot IDs and corresponding target point were pre-determined in a particular class of formation. The robots were required to be synchronized to some degree and position itself relative to its neighbor robot. In their study, two types of formations are addressed known as centered or non-centered formations. The leader may change according to the type and the followers must find a new friend in order to switch into another pattern.

In contrast to most previous works, our fundamental analysis is based on the assumption that robot team members might not have their external mark or ID. Likewise, a leader robot is not *a priori* selected. From this point of anonymity, we propose an algorithm that offers the advantage of robustness as well as flexibility. Each robot performs individual tasks under the same algorithm toward a unified mission. In practice, the robot team attempts to generate and maintain various geometric shapes, while moving toward their target. If some of the robot team members were not able to function normally, the team achieves the same or similar formation by reissuing the member IDs.

In this research, we aim to design a strong coordination framework for a team of multiple mobile robots under weak conditions. The team does not have a direct inter-team communication, a pre-determined leader, and a unique individual ID. The members can be considered disposable and therefore are equipped with a low-cost sensing unit. Only by measuring the position of anonymous members with respect to each other, the team can establish a common coordinate system to generate formations, flock[1], and adapt formations. This formation control approach features self-organizing, and fault-tolerant decentralized solution. Two fundamental contributions of our research are: 1) Formations can be made in a decentralized way adapting to an environment by only observing the position of other robots with no unique ID; 2)The same or similar formations can be recovered in spite of a lack of participating members resulted from individual failures.



(a) initial state (b) common origin
(c) leader selection (d) common direction
(e) common coordinates (f) ID allocation

Figure 1: Agreement and ID allocations

[1] The terminology is based on [7] implying that a team of robots follows a leader robot while maintaining its formation. Without the loss of generality, we used this term instead of keeping a formation pattern while navigating regardless of the existing of a leader robot.

In the following section, the system model and definitions of formation control problem are presented. Section 3 gives basic algorithms in the computational point of view. Observing algorithms for real robots are shown in Section 4. Experimental results are described in Section 5 and conclusions are drawn in Section 6.

2 Problem statements

All member robots are modeled as individual points with computational capabilities. It is initially assumed that they are located on distinct and arbitrary different positions without *a priori* coordinate system agreement as shown in Figure 1-(a). Each robot has no particular knowledge for the local coordinate system of other robots. In other words, no global coordinate system is given in advance. In addition, all anonymous members are unable to uniquely identify each others. Each robot executes the same algorithm. Therefore, robots act independently for each other incorporating only current sensing data in the execution. We assume that the selected leader robot can inform the team members of a target formation by some means.

Given a robot r_i, p_i denotes its position at any time instant. Each robot r_i represented as (0,0) in its local coordinate system L_i measures the positions p_j of the other robots r_j, denoted by $(L_i[x_j], L_i[y_j])$. A ***configuration*** means a set of center points for a group of n robots $r_1,\cdots,r_n$ with distinct positions is arbitrarily located in the 2-dimensional plane. Namely, the configuration $C_i = \{L_i[p_k] \| 1 \le k \le n\}$ is the representation of all of the robots' distributions with respect to the local coordinate system of r_i. For convenience, we generally denote a configuration $C = \{p_i | 1 \le i \le n\}$ in this paper. Furthermore, we call the set of target positions a ***formation pattern*** and denote $FP = \{f_1,\cdots,f_i,\cdots,f_n\}$ where f_i indicates a target point occupied by each robot r_i. A ***distance*** between p_i and p_j is denoted as $dist(p_i, p_j)$. Given two arbitrary vectors $\vec{a}$ and $\vec{b}$, let $angle(\vec{a},\vec{b})$ be an ***angle*** between $\vec{a}$ and $\vec{b}$. A center point for a configuration C is obtained from dividing the sum of all points by the number n as shown Figure 1-(b), said to be a ***common origin*** O of the configuration and denoted by

$$O=(L_i[x_c],L_i[y_c])=\left(\frac{\sum L_i[x_j]}{n},\frac{\sum L_i[y_j]}{n}\right). \qquad (1)$$

In a configuration C, each robot of a team defines a ***leader robot*** r_L positioned most far away from the common origin O (Figure 1-(c)). A position of the leader p_L indicates the leader coordinates with respect to r_i. Next, a ***common direction*** is defined by connecting from O to p_L as shown in Figure 1-(d). We denote the common direction as $\vec{u}$ and define the angle between the local coordinate $\vec{x}_i$ -axis of each robot and $\vec{u}$ by

$$angle(\vec{x}_i,\vec{u})=\cos^{-1}\left(\frac{(L_i[x_L]-L_i[x_c])}{dist(p_L,O)}\right). \qquad (2)$$

ISBN 979-15017-0

The common direction $\vec{u}$ defines the horizontal axis of a common coordinate system. It is straightforward to decide the vertical axis $\vec{v}$ by rotating the horizontal axis 90 degrees counterclockwise. Straightly, every robots can appoint a ***common coordinate system*** which has $\vec{u}$ and $\vec{v}$ in a 2-dimensional plane. Using the common coordinate system, all robots can be assigned their ***coordinates*** given by

$$\begin{aligned} u_i &= dist(p_i, O) \times \cos\left(angle(\vec{x}_i, \vec{d}_i) - angle(\vec{x}_i, \vec{u})\right) \\ v_i &= dist(p_i, O) \times \sin\left(angle(\vec{x}_i, \vec{d}_i) - angle(\vec{x}_i, \vec{u})\right) \end{aligned}, \quad (3)$$

where $\vec{d}_i$ is a vector passing through p_i from O as shown in Figure 1-(e). Finally, given a common coordinate system, ***ID***s are assigned to all the members, starting from the leader numbered 0, by sorting their $\vec{u}$-coordinates in an increasing order. Specifically, they are assigned an odd ID if they have a negative $\vec{v}$-coordinate or an even ID if they have a positive $\vec{v}$-coordinate by turns until the numbering is completed in either half plane (See Figure 1-(f)). Remaining members in the other half plane are assigned consecutive numbers that follow after the last number.

Based on the agreement on the coordinate system and ID allocation, we define the formation control problem consisting of pattern generation, flocking, and pattern switching as follows:

Definition 1 (**Pattern Generation)** Given a common coordinate and robot IDs, each robot can compute their target position to form a pattern. We call the set of target positions a formation pattern and denote $FP = \{f_1, \cdots, f_{n-1}, f_L\}$ where f_i indicates the common coordinate located by each robot.

Definition 2 (**Flocking)** Given a formation pattern, follower robots maintain a distance to a leader r_L from each robot and an angle between $\vec{x}$-axis and $\vec{u}$-axis, denoted by $dist(p_i, p_L)$ and $angle(\vec{x}_i, \vec{u})$, respectively. If all robots keep pace with an arbitrary formation pattern while moving together, this is called flocking.

Definition 3 (**Pattern Switching)** Given a formation pattern, if the robots can form another pattern without changing current leader and their existing ID, it is called formation switching.

3 Algorithm Descriptions

3.1 Pattern generation

Formation pattern generation requires the robots to move to their individual target position pre-determined according to the ID and number of robots. The leader robot remains stationary and geometric patterns are arranged symmetrically with respect to $\vec{u}$-axis. Therefore, the robots with even IDs are located on the upper half plane of $\vec{u}$-axis and the remaining robots with odd IDs on the other side. Robots positioned closer to the leader are assigned lower IDs. In [11], robot IDs and corresponding target point are pre-determined in a particular class of formation. The robot requires being synchronized to some degree and positions itself relative to its neighbor robot. In contrast, the proposed ID allocation in this work is not intended for providing any formation dependant IDs. Table 1 gives an example of circle pattern generation.

Table 1: Example-Circle pattern generation

Input: Each robot with ID_i obtains a leader position (u_L, v_L). and is given a target pattern by the leader.

If { ID_i = even number} **Then**
  $mark := 1$
Else { ID_i = odd number}
  $mark := -1$
End If
$uniform :=$ uniform interval
$\theta := (360/n) \times [(ID_i + 1)/2]$
$factor := [(360 \times uniform)/(2\pi \times n)]$
$u_i' := \cos\theta \times factor + u_L$
$v_i' := \sin\theta \times factor \times mark$
$f_i := (u_i', v_i')$ //*Target point of* ID_i
Motion (f_i)

Output: Circle Pattern Generation

3.2 Flocking

A selected leader navigates the path toward achieving the mission while the followers keep pace with the leader. All followers uniformly maintain $dist(p_i, p_L)$ and $angle(\vec{x}_i, \vec{u})$ with the leader that remains unchanged during the mission. The followers do not move until their leader starts navigating after a formation is completely generated. They move independently and asynchronously to each other since their navigation is controlled by the leader position and angle.

3.3 Formation switching

When switching formation patterns, the leader and the existing robot IDs remain unchanged. The current coordinate system of the leader replaces the common coordinate system. Thus, robots update their coordinates with respect to the new common coordinate system $(\vec{u}', \vec{v}')$ and moves to the new target coordinates. Two types of formations are address in [11] named centered or non-centered formations, where the leader may change according to the type and the followers must find a new friend in order to switch into another pattern. However, our switching algorithm does not depend on the neighboring robots or the pattern type.

3.4 Multiple leader and robot failure

If two or more robots are located at the same distance from a common origin, the robot team can not decide a leader uniquely. In this case, the leader selection is repeated after the positions of the leader candidates are slightly perturbed off the circle of the same radius with the condition of $dist(p_i, O) < dist(p_i', O)$, where p_i indicates the initial point and p_i' means the new position of r_i after being perturbed. The other robots remain stationary until a single leader is finally selected. Also, the ID allocation varies according to the number of robots as well as the initial robot distribution. If there is the loss of the team members due to physical robot failure, the team attempts to achieve and maintain the same or similar pattern by just repeating the above mentioned algorithms with new ID.

4 Implementation

The key for the applicability of proposed approach lies in obtaining reliable estimates of the coordinates of the center point of the robots with respect to each other. One problem is that real robots might have an elliptic shape. According to the robot heading, the distance between an edge and the center point varies. Also, the robot might have an unequal interval of sonar sensors. Thus the blind range is not uniform and the observed edge of the real robot is not smoothly connected.

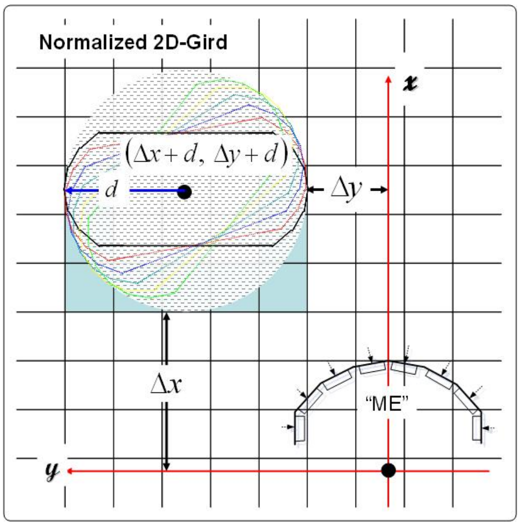
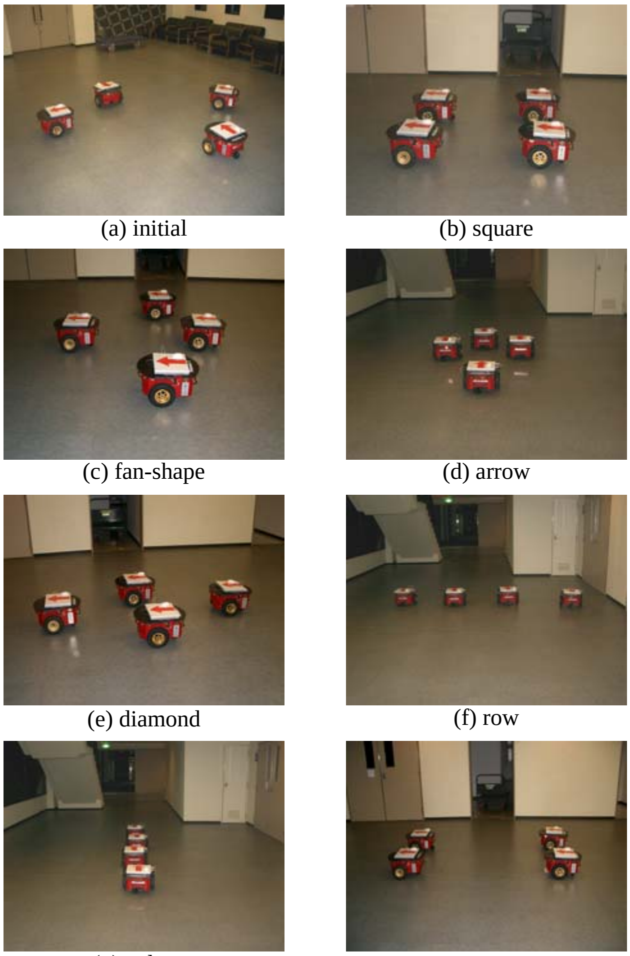


Figure 2: Estimating other coordinates from edge trajectories

In this paper, the image processing techniques [12] are employed to recognize the center coordinates of the other robots using only sonar sensors. To begin with, we made a 5000 mm × 5000 mm 2D-grid with a 50 mm × 50 mm unit cell. In the searching step, robots detect other robots using 16 sonar sensors by rotating 180 degrees at an interval of 10 degrees. Robots read data from all sonar sensors three times consecutively at an interval. These distance data are recorded and updated as an integer intensity value in the corresponding cell that represents the relative distance from the observing robot. Specifically, the Canny algorithm [12] eliminates a low intensity cell within the grid which is then run through the Sobel algorithm [12]. These methods are applied to find the edge of a robot using the gradient of discrete information appeared in the boundary of a robot. Finally, each robot executes the histogram equalization processing [12] that generates a histogram with a uniform intensity distribution to improve the edge detection. By the equalization, the grid can overcome the distortion problem resulted from an unequal interval of sensors.

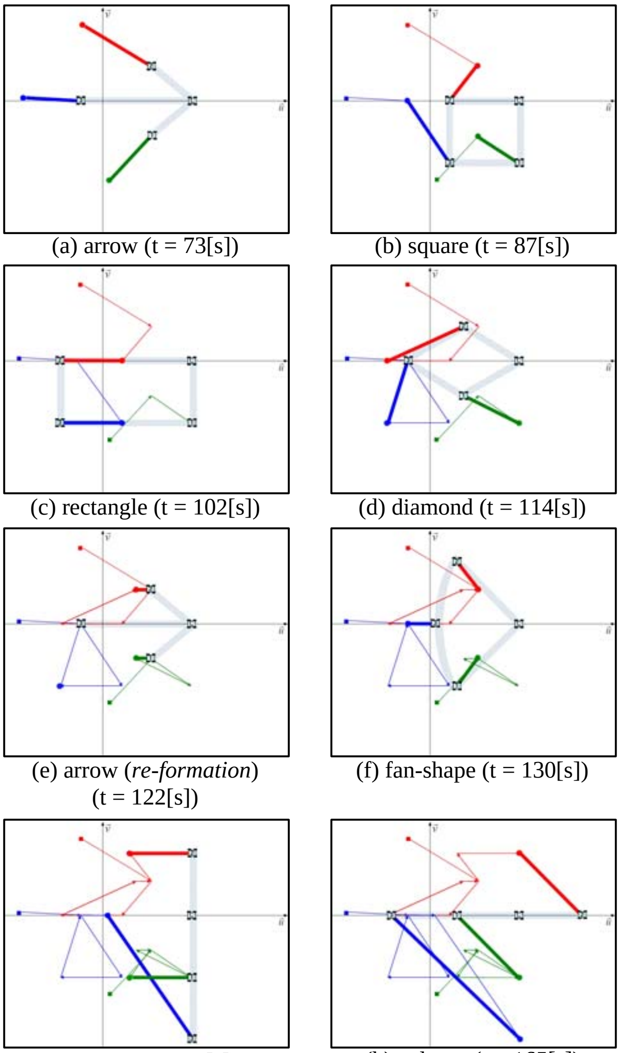
In the checking step, robots compare the normalized grid with a 500 mm × 500 mm checking mask around the estimated center point while turning 30 degrees. As illustrated in Figure 2, robots collect a cell with a maximum intensity value in the checking mask. Each robot finally puts the adjacent cell together and makes a virtual half circle on the grid from which they compute the center coordinates of the other robots. Each robot finds the minimum distance of Δx and Δy to the half circle with respect to their local coordinate system. Then, the center coordinates are easily obtained by adding the distance of semimajor axis d of the elliptic robot edge to Δx and Δy, respectively. Using this estimation, each robot establishes a common coordinate system within an acceptable error range. Note that, however, this method requires robots to be initially positioned apart a minimum distance of 600 mm, with a line of sight. Practically, the time required for recognizing the position of the robots with respect to each other is 58 seconds. The studies performed from the computational standpoint [1-3] [6-7] assumed that the robots are modeled as a circle or even points and equipped with unlimited sensors. In contrast, this observing algorithm can overcome the problem of the elliptic geometry of the robots with arbitrary heading directions.



(a) initial (b) square
(c) fan-shape (d) arrow
(e) diamond (f) row
(g) column (h) rectangle

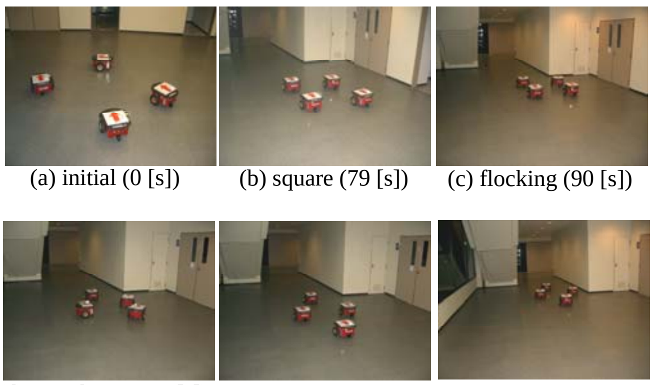
Figure 3: Seven Formation Patterns Using four Pioneer3-DX Robots

5 Experimental evaluations



(a) arrow (t = 73[s]) (b) square (t = 87[s])
(c) rectangle (t = 102[s]) (d) diamond (t = 114[s])
(e) arrow (*re-formation*) (t = 122[s]) (f) fan-shape (t = 130[s])
(g) row (t = 146[s]) (h) column (t = 165[s])

Figure 4: Trajectories for formation switching
(The elapsed time indicates the pattern generation time including the recognition process.)

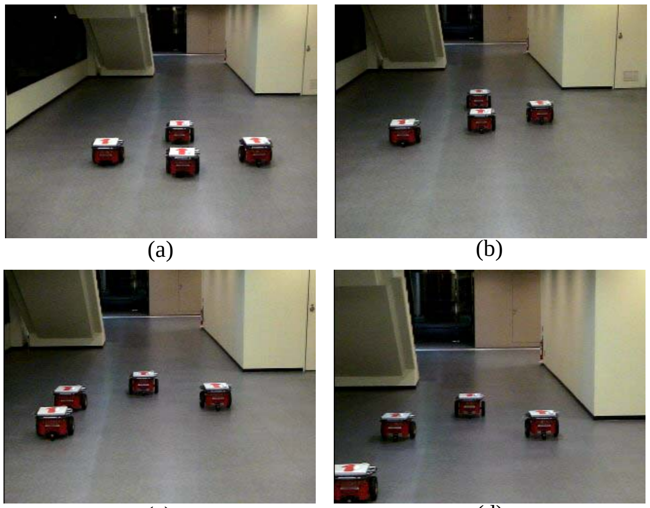


(a) initial (0 [s]) (b) square (79 [s]) (c) flocking (90 [s])
(d) switching (105 [s]) (e) diamond (114 [s]) (f) re-flocking (115 [s])

Figure 5: Formation control adapting to an environment

To evaluate the validity and effectiveness of the proposed formation control strategy, we carried out three kinds of experiments with a team of four Pioneer 3-DX robots. Robots are equipped with 16 ultrasonic range sensors and control programs run on a laptop computer that sits on top of the robots. Robots do not communicate with each other, but are only allowed to receive a desired formation pattern from a leader. In these experimental tests, the robot moved with a linear velocity of 200 mm/sec and an angular velocity of 100 deg/sec.

In the first experiment, the robot team generates and adapts formation patterns from an arbitrary position and heading direction. Robots are aware of their target positions according to the formation pattern, but do not know who goes where. As shown in Figure 3, the robot team generated seven different formation patterns. Moreover, formations can be switched continuously from one pattern to another with the same leader as illustrate in Figure 4. This figure depicts the trajectories of robots switching formations, where the bold lines indicate the trajectory approaching the current target point and the thin lines represent the history of previous trajectories. The leader remains stationary to help the followers generate a pattern by sending messages for target patterns consecutively in the following order: arrow, square, rectangle, diamond, arrow, fan-shape, row, and column. The team generates the arrow pattern twice, which demonstrates the reliability of formation switching from any given formations.



(a) (b)
(c) (d)

Figure 6: Robustness against the Loss in Team Population
(a) flock with the fan-shape by 4 robots; (b) loss of a team member; (c) regenerating a triangle pattern with the same team size; (d) flocking with the triangle pattern by 3 robots

The second experiment demonstrates how the robot team flocks flexibly adapting to an environment. After forming a square pattern, the robot team navigates toward a stationary target located 8 m away. On the way to the target, the team encounters an obstacle forcing them to switch into a diamond pattern that shifts the common origin of the team away form the obstacle. Then the team re-flocks to the

ISBN 979-15017-0

target point while maintaining the diamond pattern. Figure 5 shows the snapshots of this experiment. The leader decided an appropriate formation, played as a stationary post for formation switching, and guided the team.

In the third experiment, the robustness is verified against the accidental failure of robot members. While flocking in a fan-shape pattern, one robot stops and immediately the remaining robots re-form similar triangle pattern to continue the mission. The replacement pattern is generated by reissuing IDs before the team re-flocks toward the target. Figure 6 shows the snapshots of this formation recovering.

The proposed approach did not require a pre-determined leader, an individual ID, a common coordinates, or optical sensors. Moreover, the proposed formation control method is verified to be robust against the robot failure. Finally, we demonstrated that the team accomplished the assigned mission without having to have high quality sensors and equipments. This allows us to organize a team with a simple, economical unit which we can easily deploy even in hazardous environments.

6 Conclusion

Formation control is the first step toward real-world implementation of cooperative robotics. In this paper we presented a unified formation control approach, enabling a team of multiple robots to cooperate in a common task. In this study a team of anonymous robots was assumed to participate in the formation process under weak conditions: no pre-determined leader, no unique individual IDs, and no common coordinates. Only by measuring the center point of the other robots, the robots agreed on a common coordinate system and were assigned their ID. With the obtained coordination system and IDs, the robot team was able to generate various formation patterns, flock, and change from one formation to another adapting to an environment or tasks. The proposed algorithm featuring decentralized, self-organized, and robust design was successfully implemented and verified using a team of 4 real robots. Specifically, to recognize the positions of other robots using a limited number of sonar sensors, image processing techniques were employed. We could therefore demonstrate that the team accomplished the assigned mission without having to have high quality sensors and equipments. This will allow us to organize a robot team with simple, economical units which we can easily deploy even in hazardous environments.

Acknowledgment

This research is conducted as a program for the “Fostering Talent in Emergent Research Fields” in Special Coordination Funds for Promoting Science and Technology by Ministry of Education, Culture, Sports, Science and Technology.

References

[1] X. Defago and A. Konagaya, *Circle formation for oblivious anonymous mobile robots with no common sense of orientation*, Proc. 2nd ACM International Workshop on Principles of Mobile Computing, 2002.

[2] I. Suzuki and M. Yamashita, *Distributed anonymous mobile robot: Formation of geometric patterns*, SIAM Journal of Computing, Vol. 28, No. 4, pp.1347-1363, 1999.

[3] I. Suzuki and M. Yamashita, *Agreement on a common x-y coordination system by a group of mobile robots*, Proc. Dagstuhl Seminar on Modelling and Planning for Sensor-Based Intelligent Robots, 1996.

[4] Y. Ikemoto, Y. Hasegawa, T. Fukuda, and K. Matsuda, *Graduated spatial pattern formation of robot group*, Journal of the Robotics Society of Japan, Vol.22, No.7, pp.911-919, 2004. (in Japanese)

[5] M. Lemay, F. Michaud, D. Letourneau, and J.-M. Valin, *Autonomous initialization of robot formations*, Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, 2004.

[6] G. Prencipe, *CORDA: Distributed coordination of a set of autonomous mobile robots*, Proc. 4th European Research Seminar on Advances in Distributed Systems, 2001.

[7] V. Gervasi and G. Prencipe, *Coordination without communication: The case of the flocking problem*, Discrete Applied Mathematics, Vol.143, No.3, pp.203-223, 2003.

[8] T. Balch and R. C. Arkin, *Behavior-based formation control for multi-robot teams*, IEEE Transactions on Robotics and Automation, Vol.14, No.6, pp.926-939, 1998.

[9] S. Carpin and L. E. Parker, *Cooperative Leader Following in a Distributed Multi-robot System*, Proc. IEEE International Conference on Robotics and Automation, 2002.

[10] D. Kurabayashi and K. Osagawa, *Formation transition based on geometric features for multiple autonomous mobile robot*, Journal of the Robotics Society of Japan, Vol.23, No.3, pp.376-382, 2005. (in Japanese)

[11] J. Fredslund and M. J. Mataric, *A general algorithm for robot formations using local sensing and minimal communication*, IEEE Transactions on Robotics and Automation, Special issue on Advances in Multi-Robot Systems, Vol.18, No.5, pp. 837-846, 2002.

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

