JAIST Repository
https://dspace.jaist.ac.jp/

Title	Adaptive Flocking of a Swarm of Robots Based on Local Interactions
Author(s)	Hanada, Y.; Lee, Geunho; Chong, Nak Young
Citation	IEEE Swarm Intelligence Symposium, 2007. SIS 2007.: 340-347
Issue Date	2007-04
Type	Conference Paper
Text version	publisher
URL	http://hdl.handle.net/10119/7793
Rights	Copyright (C) 2007 IEEE. Reprinted from IEEE Swarm Intelligence Symposium, 2007. SIS 2007. This material is posted here with permission of the IEEE. Such permission of the IEEE does not in any way imply IEEE endorsement of any of JAIST's products or services. Internal or personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution must be obtained from the IEEE by writing to pubs-permissions@ieee.org. By choosing to view this document, you agree to all provisions of the copyright laws protecting it.
Description	

Japan Advanced Institute of Science and Technology

Proceedings of the 2007 IEEE Swarm Intelligence Symposium (SIS 2007)

Adaptive Flocking of a Swarm of Robots Based on Local Interactions

Yosuke Hanada, Geunho Lee, and Nak Young Chong
School of Information Science, Japan Advanced Institute of Science and Technology, Nomi, Japan
{y-hanada, geun-lee, nakyoung}@jaist.ac.jp

Abstract— **This paper presents a novel flocking strategy for a large-scale swarm of robots that enables the robots to navigate autonomously in an environment populated with obstacles. Robot swarms are often required to move toward a goal while adapting to changes in environmental conditions in many applications. Based on the observation of the swimming behavior of a school of tunas, we apply their unique patterns of behavior to the autonomous adaptation of the shape of robot swarms. Specifically, each robot dynamically selects two neighboring robots within its sensing range and maintains a uniform distance with them. This enables three neighboring robots to form a regular triangle and remain stable in the presence of obstacles. Therefore, the swarm can be split into multiple groups or re-united into one according to environmental conditions. More specifically, assuming that robots are not allowed to have individual identification numbers, a pre-determined leader, memories of previous perceptions and actions, and direct communications to each other, we verify the validity of the proposed algorithm using the in-house simulator. The results show that a swarm of robots repeats the process of partition and maintenance passing through multiple narrow passageways.**

I. INTRODUCTION

A swarm of simple robots offers many advantages in terms of efficiency, fault-tolerance, and generality [1]. Therefore, robot swarms are expected to be deployed in a wide variety of applications such as odor localization, sensor networking, medical operation, surveillance, and search-and-rescue. In order to perform those tasks, robots need to flock autonomously in an unknown environment. However, few studies have been addressed adaptive flocking, aimed to cope with different geometric constraints in a given environment. Flocking algorithms can be divided into the centralized and decentralized approaches. The centralized approach [2-3] relies on a specific robot that supervises the movement of other robots. In general, a heavy computation burden is imposed on the supervising robot which requires a tight communication with other robots. On the other hand, the decentralized approach is achieved through only individual robot's decision. Specifically, this approach is often implemented through the use of the leader-follower method [4-6] and the leaderless method [8-14].

Fredslund and Mataric [4] addressed a neighbor-referenced method, where the ID and corresponding target point of the robots were pre-determined in a particular type of geometric pattern. When the robots flock, a leader robot navigates a path while the follower robots maintain the angle and distance to their neighbor. Parker *et al.* [5] proposed a tightly-coupled navigation assistance strategy by the leader with rich sensing capability, but such a strategy makes the leader more costly and the team becomes less robust to the failure of the leader. In the work done by Lee *et al.* [6], the leader selection and follower ID assignment was carried out in a decentralized, self-organizing way. The follower robots in a predetermined pattern kept pace with the leader robot that navigated toward a goal.

Balch and Hybinette [8] proposed a crystal generation process inspired technique for large-scale robot swarms. Each robot had local attachment sites attracted to other robots. When the robots encountered an environmental constraint, they could avoid the obstacle depending on the behavior-based rule combining the concept of an attractive force and a repulsive force. However, their technique required an effort to adjust parameters to perform a successful flocking. Based on an artificial physics framework [7] analogous to the gravitational force, Zarzhitsky *et al.* [9] reported a chemical plume tracing method. This method arranged the robots into a hexagonal formation, so that they could share real flow-field parameters of fluid dynamics with their six neighbors, and utilize these parameters to calculate the next target point. Shimizu *et al.* [10] introduced emergent behaviors for two-dimensional modular robots reconfiguring their geometric shape according to an environment. The technique was based on coupling between a connectivity control algorithm for connected neighbor robots and nonlinear oscillators generating locomotion, and then each robot locally interacted with their neighbor robots to generate the phase gradient. Kurabayashi *et al.* [11] presented an adaptive transition technique to enable a team of robots to change a formation by generating the nonlinear oscillators communicating with each other according to an environment. Esposito and Dunbar [12] coordinated the number of robots toward their respective goal while maintaining a range of wireless connectivity and line-of-sight under the presence of obstacles. To solve this problem, they presented a method for composing multiple potential functions, which indicated a set of possible input directions, into a single feasible movement direction from the condition that the state vector of a robot approaches the minima of the potential functions. Due to the existence of saddle points, this method is occasionally difficult for robot swarms to remain connected. Folino and Spezzano [13] introduced a parallel spatial clustering algorithm, termed SPARROW, stemmed from the field of artificial life. Their algorithm combined a smart exploratory method based on a flock of birds with a density-based cluster algorithm to discover clusters of arbitrary shape and size in spatial data.

1-4244-0708-7/07/$20.00 ©2007 IEEE

Authorized licensed use limited to: Japan Advanced Institute of Science and Tech. Downloaded on November 10, 2008 at 19:46 from IEEE Xplore. Restrictions apply.

In their algorithm, robots are transformed into hunters with a foraging behavior that allow each robot to explore the spatial data while searching for clusters. Nembrini *et al.* [14] used local communications, but robots were not required to sense each other's position information. The aim of their approach is to maintain global connection while avoiding obstacles and adapting its shape to navigate toward a goal.

In contrast to most previous works, our approach begins with the following assumptions about the robots and their environment: no identification number, no pre-determined leader, no common coordinate system, no memory for the past perceptions/actions, no communication, and unknown task environments. In spite of these limitations, given any arbitrary initial positions, a large-scale swarm of robots is required to navigate toward a goal position while locally interacting with other robots in close proximity. According to environmental conditions, the robot swarm can be split into multiple groups or re-united into a single group while maintaining a uniform distance of each other. This paper is aimed to design a decentralized flocking algorithm by which a swarm of robots can navigate through unknown territory.

The rest of the paper is organized as follows. Section II presents the problem definition and the robot model from the computational point of view. Section III describes the local interaction among neighboring robots. Section IV and Section V explain the ideas behind the algorithms for the partition problem and the maintenance and unification problem, respectively. Section VI presents an integrated algorithm for adaptive flocking. The proposed algorithms are verified by the simulations. Finally, conclusions are drawn in Section VII.

II. PROBLEM STATEMENT

In nature, adaptive behavior can be easily seen in the swimming behavior of a school of fishes [15]. For example, when a school of fishes is faced with an obstacle such as a large rock or a predator, they can avoid collision by being divided into a plurality of smaller groups. Those groups can be reunited when they pass around the obstacle. It is notable that the schooling is based on the local interaction between members in close proximity. According to Stocker's work [16], a local geometric model of tuna school forms a diamond pattern. Our flocking approach for a swarm of robots is proposed by imitating the schooling of tunas. Here, we can define the problem of ***Adaptive Flocking*** for a large-scale swarm of robots as follows:

Given a swarm of robots $r_1, \cdots, r_n$ located at arbitrarily distinct positions, how to enable the robots to navigate through unknown territory and avoid obstacles in order to achieve an assigned mission.

We decompose the problem of Adaptive Flocking into 2 sub-problems as illustrated in Fig. 1.

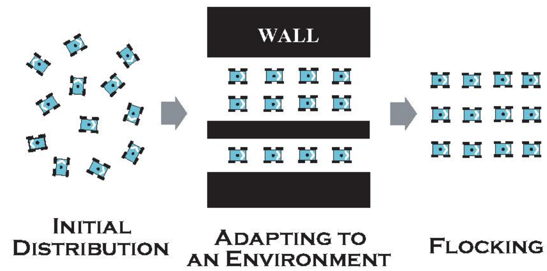


Fig. 1. Concept of Adaptive Flocking

Problem-1 (Partition) *Given that a robot observes an environmental constraint, how to enable the swarm be split into multiple smaller groups adapting to the constraint.*

Problem-2 (Maintenance/Unification) *Given that robots located at arbitrarily distinct positions, how to enable the robots maintain a group or unite separate groups.*

In this paper, we deal with an autonomous mobile robot modeled as a point with its local coordinate system that freely moves on the two-dimensional plane. The robots have no prior knowledge of their identification numbers, and do not share any common coordinate system nor central robot (leader). Furthermore, robots have a limited-range sensing capability and cannot communicate explicitly. They are only able to locally locate other robots.

Based on the robot model, each robot executes the same algorithm and then acts independently and asynchronously from other robots. Time is represented as an infinite sequence of discrete time instants $t_0, t_1, t_2, \cdots$. We assume that all robots repeat an endless activation cycle of sensing, computation, and motion. At each time instant t, a robot computes its target position (computation) by current observing the position of other robots (sensing), and moves toward the target (motion). Here, the time required in sensing t_s and computation t_c is negligibly compared with the time required in motion t_m. Let Δt denote the time interval between t_i and t_{i-1}, which is defined as follows: $t_s, t_c \ll t_m \cong t_i - t_{i-1} = \Delta t$. All robots are assumed not to move beyond the mobility limit of $d_{\max}$(= $d_u/\sqrt{3}$, where let d_u denote a uniform distance.) at each time. The maximum velocity $v_{\max}$ of r_i is resolved into $d_{\max}$ and the time instant t given by $v_{\max} = d_{\max}/t_m \cong d_{\max}/\Delta t$. Similarly, the maximum motion interval of the robot may be given by $t_{\max} = d_{\max}/v_{\max} = d_u/(\sqrt{3} \times v_{\max})$. This limitation of motion will prevent the robot from lagging behind the other robots.

Finally, the vector $\vec{I}$ from p_i occupied by r_i to the target point is termed the *local interaction vector*. Let $\vec{G}$ denote the *goal direction vector* which represents the heading of the swarm. The *maintenance vector* of each robot is denoted by $\vec{M}$. It can be defined by $\vec{I}$ generated with respect to $\vec{G}$. On the way toward $\vec{G}$, the direction is decided by the *partition vector* $\vec{P}$. Also let $\vec{U}$ denote the *unification vector*.

Authorized licensed use limited to: Japan Advanced Institute of Science and Tech. Downloaded on November 10, 2008 at 19:46 from IEEE Xplore. Restrictions apply.

III. LOCAL INTERACTION

This section describes local interaction among neighboring robots. For all time instants, we assume that the number of neighbors that can be perceived is more than two. As shown in Fig. 2-(a), r_i selects the first neighbor located the shortest distance from p_i. The second neighbor is selected such that the total distance from p_{s1} occupied by the first neighbor to p_i through the position p_{s2} of the second neighbor is minimized. Then, the angles of the triangle become all acute angles in many cases. In Fig. 2-(b), r_i finds the center p_m of the triangle with respect to its local coordinates, and then measures the angle between the line connecting two neighbors and the horizontal axis of its local coordinate system. Using p_m and the angle, r_i calculates its target point p_t at a time instant. A geometric pattern formed by the local interaction is an isosceles triangle with distance d_u for two selected neighbor. Here, $\{p_{s1}, p_{s2}\}$ is defined as a set of positions of two neighbors selected by r_i.

Consider a triangle with the vertices A, B, and C that represent the position occupied by three robots as shown in Fig. 3-(a). Let α, β, and γ denote the angles of the triangle, respectively. Each robot located at the vertex of the triangle ABC may move to the new point A', B', and C'. The angles of the new triangle $A'B'C'$ are α', β', and γ'. Let O denote the center of ABC. Also let P on $\overline{AB}$ denote the point projected from O onto $\overline{AB}$. Similarly, let Q on $\overline{AC}$ indicate the point projected from O onto $\overline{AC}$. If we consider a quadrangle $APOQ$, the angles of P and Q are right angles. Therefore, $\angle POQ$ yields $180° - \alpha$. From the Fig. 3-(a), $\angle B'OC'$ is equal to $\angle POQ$. The triangle $B'OC'$ is an isosceles triangle since $\overline{OB'}$ and $\overline{OC'}$ is identical ($d_u/\sqrt{3}$ $(= \sqrt{3}/2 \times d_u \times 2/3)$). Hence, α of ABC is equal to $2a$. With the same manner, β and γ can obtain $2b$ and $2c$, respectively. Therefore, we see that α' of the triangle $A'B'C'$ is $(\beta+\gamma)/2$ as shown in 3-(b). Likewise, β' indicates $(\alpha+\gamma)/2$ and γ' does $(\alpha+\beta)/2$ with the same manner.

Accordingly, α' is given by

$$\alpha' = (\beta + \gamma)/2 \rightarrow \alpha(t+1) = (\beta(t) + \gamma(t))/2. \quad (1)$$

With the same manner, $\beta(t+1)$ and $\gamma(t+1)$ are yielded. Summarizing the relation equations between Fig. 3-(a) and -(b), we obtain the following expression for an internal angle:

$$\alpha(t+2) = \frac{\beta(t+1) + \gamma(t+1)}{2} = \frac{\alpha(t)}{2} + \frac{\beta(t) + \gamma(t)}{4}. \quad (2)$$

Likewise, $\beta(t+2)$ and $\gamma(t+2)$ can be yielded. Given (2), $\alpha(t+2)$ at the time $(t+2)$ can be summarized by the following equation:

$$\alpha(t+2) = \alpha(t)/2 + \alpha(t+1)/2. \quad (3)$$

With the same manner, $\beta(t+2)$ and $\gamma(t+2)$ are yielded. From (2) and (3), the simple relationship is given by $\alpha(t+1) = (\beta(t) + \gamma(t))/2$, $\beta(t+1) = (\gamma(t) + \alpha(t))/2$, and $\gamma(t+1) = (\alpha(t) + \beta(t))/2$. Equation (1) can be transformed using the following matrix formula:

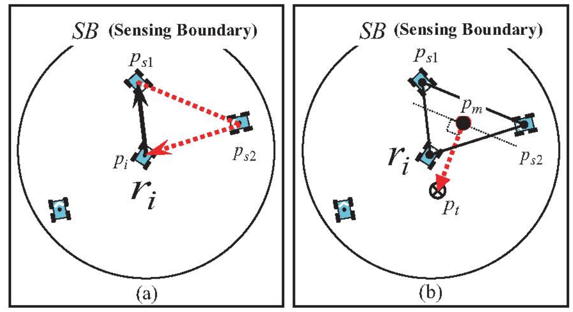
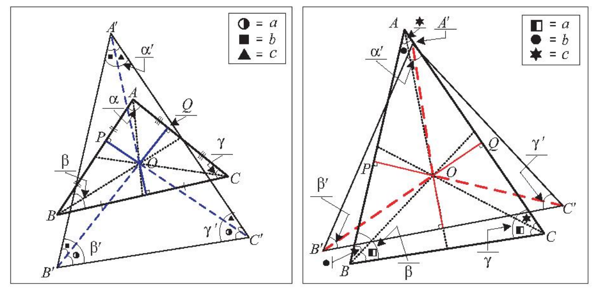


Fig. 2. Illustration of local interaction ((a) neighbor selection, (b) target computation)



(a) time instant : t and $t+1$ (b) time instant : $t+1$ and $t+2$

Fig. 3. Illustration of notations used in explanation of convergence by local interaction

$$\begin{bmatrix} \alpha(t+1) \\ \beta(t+1) \\ \gamma(t+1) \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha(t) \\ \beta(t) \\ \gamma(t) \end{bmatrix}. \quad (4)$$

Similarly, (3) can be re-written as follows:

$$\begin{bmatrix} \alpha(t+2) \\ \beta(t+2) \\ \gamma(t+2) \end{bmatrix} = \frac{1}{2} \left(\begin{bmatrix} \alpha(t) \\ \beta(t) \\ \gamma(t) \end{bmatrix} + \begin{bmatrix} \alpha(t+1) \\ \beta(t+1) \\ \gamma(t+1) \end{bmatrix} \right). \quad (5)$$

Substituting (4) into (5) gives

$$\begin{aligned} \begin{bmatrix} \alpha(t+2) \\ \beta(t+2) \\ \gamma(t+2) \end{bmatrix} &= \frac{1}{2} \begin{bmatrix} \alpha(t) \\ \beta(t) \\ \gamma(t) \end{bmatrix} + \frac{1}{2^2} \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha(t) \\ \beta(t) \\ \gamma(t) \end{bmatrix} \\ &= \frac{1}{2^2} \begin{bmatrix} \alpha(t) \\ \beta(t) \\ \gamma(t) \end{bmatrix} + \frac{3}{2^2} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \frac{\alpha(t) + \beta(t) + \gamma(t)}{3}. \quad (6) \end{aligned}$$

The relation equation for $(t+n)$ becomes the following generalized equation:

$$\begin{aligned} \begin{bmatrix} \alpha(t+n) \\ \beta(t+n) \\ \gamma(t+n) \end{bmatrix} &= \frac{1}{2^n} \begin{bmatrix} \alpha(t) \\ \beta(t) \\ \gamma(t) \end{bmatrix} \\ &+ \frac{\sum_{k=1}^{n} 3 \cdot 2^{k-2}}{2^n} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \frac{\alpha(t) + \beta(t) + \gamma(t)}{3}. \quad (7) \end{aligned}$$

Authorized licensed use limited to: Japan Advanced Institute of Science and Tech. Downloaded on November 10, 2008 at 19:46 from IEEE Xplore. Restrictions apply.

Taking an infinite value, the final value is given by,

$$
\begin{aligned}
\lim_{n\to\infty}\begin{bmatrix}\alpha(t+n)\\ \beta(t+n)\\ \gamma(t+n)\end{bmatrix} &= 0\times\begin{bmatrix}\alpha(t)\\ \beta(t)\\ \gamma(t)\end{bmatrix}+1\times\begin{bmatrix}1\\1\\1\end{bmatrix}\frac{\alpha(t)+\beta(t)+\gamma(t)}{3}\\
&= \begin{bmatrix}60^\circ\\ 60^\circ\\ 60^\circ\end{bmatrix}. \qquad (8)
\end{aligned}
$$

From (8), we see that each internal angle converges into the same angle after infinite activation steps. In summary, the robot r_i locally interacts with two neighbors in order to generate an isosceles triangle. By doing this every time instant, three robots are finally configured to have an equilateral triangle lattice. Likewise, considering the case with a large number of robots, we can use the Fermat point [17-18] to prove the convergence into the equilateral triangles with d_u.

IV. PARTITION PROBLEM

Given that an individual robot in a swarm detects an obstacle which hinders its movement, the swarm should be split into multiple groups. The problem is how to find each robot's heading toward the goal adapting to an environment.

A. Description of partition method

A swarm of robots is required to be divided into multiple smaller groups to pass through several passageways as shown in Fig. 4. Similar to Newton's law of Universal Gravitation [19], we can obtain a solution for the Partition Problem using the relative degree of attractive force, termed *favorite force*, that helps robots decide their direction in various environmental conditions. Based on the magnitude of the favorite vector $\vec{f}$, each robot determines where to move. In Fig. 4-(a), we denote the passageway as s_j with width w_j at distance d_j between the center of w_j and p_i of r_i. A favorite vector $\vec{f_j}$ for s_j is defined by

$$|\vec{f_j}| = w_j/d_j^2. \qquad (9)$$

A set of favorite vectors $\{\vec{f_j}|1 \le j \le n\}$ is the representation of the passageways. The robot selects the maximum magnitude of $\vec{f_j}$ denoted by $|\vec{f_j}|_{\max}$. Specifically, the direction vector toward $|\vec{f_j}|_{\max}$ is defined as the *partition vector* and is denoted by $\vec{P}$. If $|\vec{f_j}|_{\max}$ is not uniquely determined, $\vec{P}$ becomes the maximum value obtained from the vector sum $(\vec{f}_{j\_\max} + \vec{M})$ as illustrated in Fig. 4-(b).

B. Simulation result

In Fig. 5, there existed three passageways in the environment. Based on our proposed strategy, robots could be split into three smaller groups while maintaining the local geometric configuration. Through the local interactions, the rest of the robots could naturally adapt to an environment by just following their neighbors moving ahead.

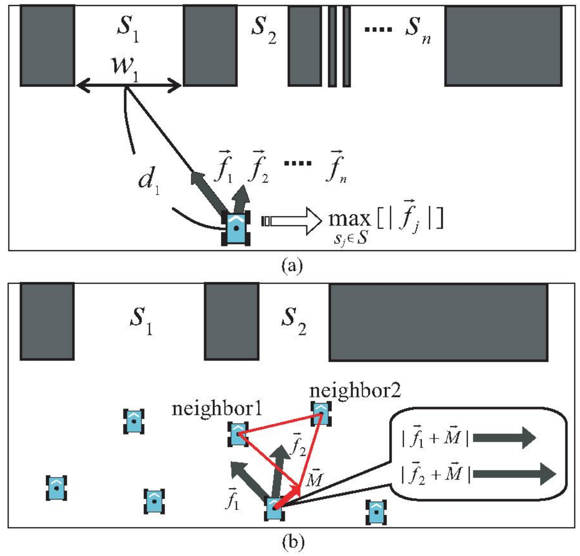
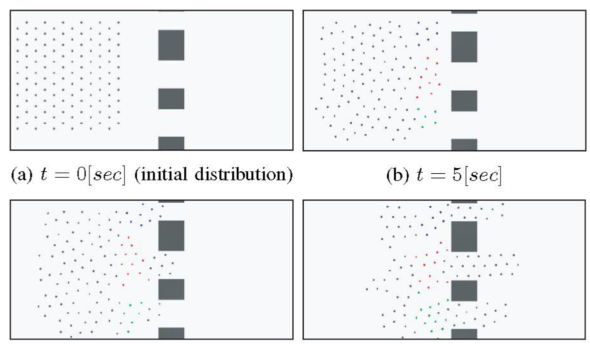


Fig. 4. Illustration of a direction decision according to an environment ((a) computation of magnitudes for each favorite vector, (b) computation for the same favorite vector magnitudes using $\vec{M}$)



(a) $t = 0[sec]$ (initial distribution) (b) $t = 5[sec]$

(c) $t = 9[sec]$ (d) $t = 18[sec]$

Fig. 5. Simulation results for team partition with 120 robots

V. MAINTENANCE AND UNIFICATION PROBLEM

The second part of *Adaptive Flocking* is how to maintain a uniform interval among robots while navigating toward a goal. This enables the robots to generate a geometric configuration consisting of a multitude of equilateral triangles and, moreover, to unite separate groups into a single group.

A. Description of maintenance and unification method

Each robot adjusts $\vec{G}$ with respect to its local coordinates and computes its observation set O_i, which indicates a position set of other robots located within the sensing boundary of the observing robot at the time instant t. As shown in Fig. 6-(a), let $A(\vec{G})$ denote the goal direction area which is defined as a specific range of each robot's sensing boundary. r_i counts the number of robots existing in $A(\vec{G})$. If there exist more than one robot in $A(\vec{G})$, r_i defines p_{s1} as the one that has the shortest distance from p_i. Otherwise, robots spot a virtual point p_v located on $\vec{G}$ apart from d_u as the first neighbor p_{s1}. The second neighbor is selected by the

Authorized licensed use limited to: Japan Advanced Institute of Science and Tech. Downloaded on November 10, 2008 at 19:46 from IEEE Xplore. Restrictions apply.

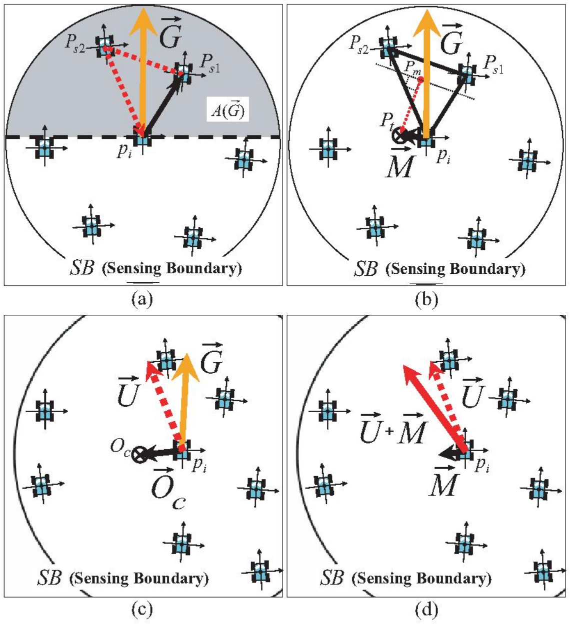


Fig. 6. Illustration of maintenance and unification method ((a) neighbor selection, (b) maintenance vector $\vec{M}$, (c) unification vector $\vec{U}$, (d) target computation)

method explained in Section III. r_i interacts with the two neighbors to generate an equilateral triangle. Then, the output of maintenance $\vec{M}$ can be obtained as illustrated in Fig. 6-(b).

After computing $\vec{M}$, each robot moves to the center of the polygon formed by the positions of robots observed within its sensing boundary. As presented in Fig. 6-(c), the center point O_c is obtained by dividing the sum of all positions of robots with the number of robots. Then, $\vec{O}_c$ is defined as the vector from p_i to O_c. In Fig. 6-(c), r_i computes a unification vector $\vec{U}$ from sum of $\vec{G}$ and $\vec{O}_c$. Finally, p_t is obtained by summating $\vec{U}$ and $\vec{M}$ as shown in Fig. 6-(d).

B. Simulation results

Fig. 7 illustrates how two separate groups of 120 robots merge into one while maintaining the local geometric configuration. The robots in two separate groups approach each other by moving toward the center of the polygon formed by the robots in the two groups. The rest of members just move by the interaction with their neighbors.

Fig. 8 shows the simulation results of flocking with 30 robots under no environmental constraints. Initially, robots are arbitrarily located on the two-dimensional plane. As shown in Fig. 8-(b) and (c), each robot generates its geometric configuration with their neighbors while moving toward a goal. Fig. 8-(d) illustrates that robots maintain the swarm while navigating. Through local interactions based on the geometric approach, the swarm could navigate toward a goal.

VI. ADAPTIVE FLOCKING

In this section, we describe the adaptive flocking algorithm that enables the robots to autonomously navigate in an environment populated with obstacles and present results of simulations that show the effectiveness of the proposed algorithm.

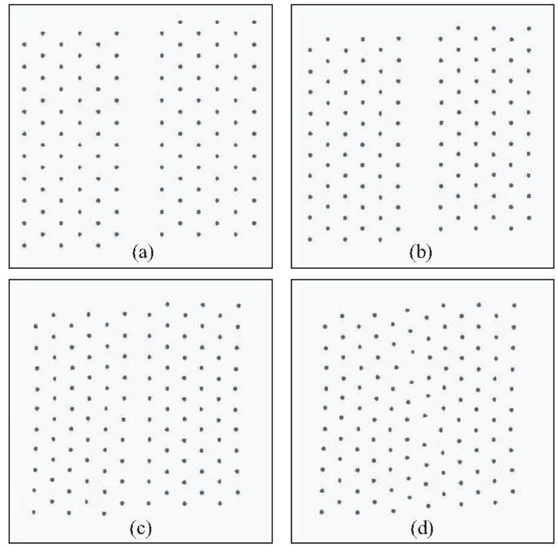


Fig. 7. Simulation results for maintenance and unification with 120 robots under no environmental constraint ((a) $t = 0[sec]$, (b) $t = 5[sec]$, (c) $t = 14[sec]$, (d) $t = 20[sec]$)

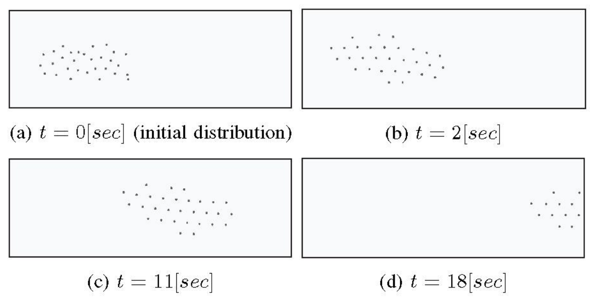


Fig. 8. Simulation results for flocking with 30 robots under no environmental constraint

A. Algorithm Intuition and Simulation Results

Those two methods proposed above are combined into Algorithm-1, where $\vec{P}$ is the partition vector, $\vec{M}$ is the maintenance vector, and $\vec{U}$ is the unification vector. Using $\vec{M}$ calculated by $\vec{G}$, a swarm of robots moves toward a goal while maintaining a uniform interval with each other. When observing an obstacle, robots are separated by $\vec{P}$ and $\vec{M}$, and follow $\vec{P}$. After being split into multiple teams, the robots maintain their team by $\vec{M}$.

In Fig. 9, the swarm navigates toward a stationary target located at the right side. On the way to the goal, some of the robots encounter with an obstacle that forces the swarm split into two groups. The rest of the robots can just follow their neighbors moving ahead. After being divided into two groups, each group maintains the geometric configuration while navigating. As shown in Fig. 9-(d), two groups are re-united by $\vec{U}$ or divided again into smaller groups due to another obstacle. Therefore, employing the proposed

Authorized licensed use limited to: Japan Advanced Institute of Science and Tech. Downloaded on November 10, 2008 at 19:46 from IEEE Xplore. Restrictions apply.

algorithm, it is possible for a swarm of robots to form local geometric patterns, flock, and adapt formations.

ALGORITHM-1 ADAPTIVE FLOCKING
(code executed by a robot r_i which occupies a point p_i)

INPUT: $\{p_1, \cdots, p_n\}$ are a set of positions for robots $\{r_1, \cdots, r_n\}$ perceived in the current sensing boundary.

constant d_u := uniform distance
$\vec{G}$:= goal direction vector
$A(\overrightarrow{G(t)})$:= goal directional area
IF{ $\exists p_k \in A(\overrightarrow{G(t)})$ } **THEN** /***Maintenance***/
  FOR $k_1 = 1, \cdots, m$ **DO** //where, m $\leq$ n
    $p_{s1} := \min[dist(p_i, p_{k_1})]$ //1st neighbor
  END FOR
ELSE $\{\forall p_k \notin A(\overrightarrow{G(t)})\}$
  p_{s1} := virtual point located on $\overrightarrow{G(t)}$ apart from d_u //1st neighbor
END IF
FOR $k_2 = 1, \cdots, n-1$ **DO**
  $p_{s2} := \min[dist(p_{s1}, p_{k_2}) + dist(p_{k_2}, p_i)]$ //2nd neighbor
END FOR
$p_m := (m_x, m_y)$ //center of (p_i, p_{s1}, p_{s2})
θ := angle between $\overline{p_{s1}p_{s2}}$ and r_i's local horizontal axis
$t_x := m_x + d_u \times \cos(\theta \pm \pi/2)/\sqrt{3}$
$t_y := m_y + d_u \times \sin(\theta \pm \pi/2)/\sqrt{3}$
$p_t := (t_x, t_y)$ //the target point
$\vec{M}$:= maintenance vector
IF {detecting n straits} **THEN** /***Partition***/
  FOR $j = 1, \cdots, n$ **DO**
    $\vec{f_j}$:= favorite vector
  END FOR
  k := the number of $|\vec{f_j}|_{\max}$
  IF $\{k = 1\}$
    ρ := favorite index //k-th $|\vec{f_j}|_{\max}$
  ELSE $\{k > 1\}$
    FOR $m = 1, \cdots, k$ **DO**
      $(\vec{f}_{j\_\max} + \vec{M})_m$:= sum of both $|\vec{f_j}|_{\max}$ and $\vec{M}$
    END FOR
    ρ := favorite index //m-th$|(\vec{f}_{j_\max} + \vec{M})_m|_{\max}$
  END IF
  $\vec{P} := \vec{f_\rho}$ //partition vector
  $target := \alpha \cdot \vec{P} + \vec{M}$
ELSE /***Unification***/
  O_c := center of O_i
  $\vec{O_c}$:= center vector to O_c from p_i
  $\vec{U} := \tau \cdot \vec{G} + \vec{O_c}$ //unification vector
  $target := \alpha \cdot \vec{U} + \vec{M}$
END IF

OUTPUT: r_i moves toward the target.

The second simulation results in Fig. 10 present the snapshots for adaptive flocking toward a moving goal represented as the red square. As the goal moves, the swarm starts to move. It can be observed that the snapshots of Fig. 9 differ from these of Fig. 10, because there exists a different $\vec{P}$ in accordance with $\vec{G}$ observed at each time instant.

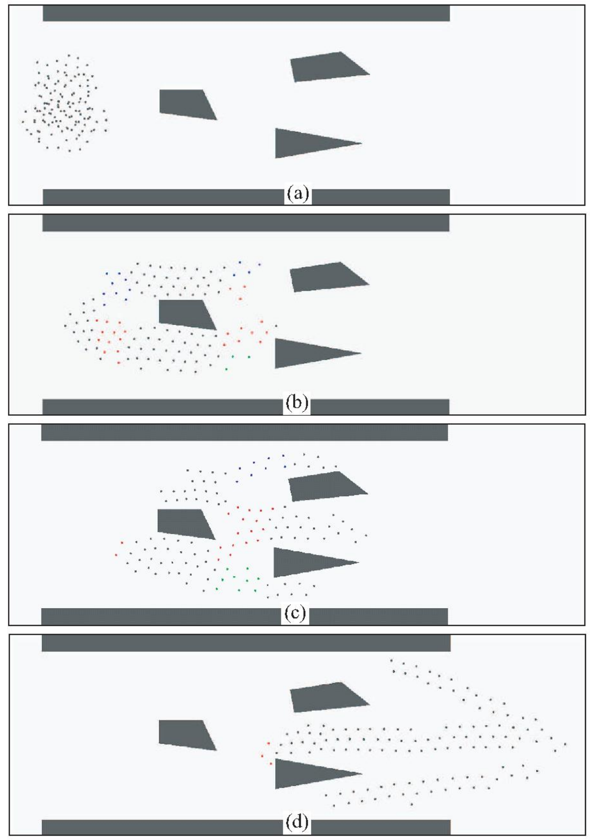


Fig. 9. Simulation results for adaptive flocking with 120 robots toward a stationary goal ((a) $t = 0[sec]$(initial distribution), (b) $t = 18[sec]$, (c) $t = 28[sec]$, (d) $t = 50[sec]$)

B. Discussions

The partition strategy in Section IV is that each robot can find its heading based on $\vec{f_j}$ and $\vec{M}$ under unknown environments. In Fig. 11, we investigate the behaviors of a swarm of robots without the partition capability. It took about 150 seconds to pass through the passageway. In the simulation result of Fig. 5, it took about 50 seconds with the same velocity and interval distance. From this, it is evident that the partition provides a swarm with an efficient navigation capability in an obstacle-cluttered environment. Likewise, unless the robots have the unification capability, they may separately perform a task after being divided as presented in Fig. 12. The capability of unification is essential for a task which can not be completed by an insufficient number of members.

Finally, Fig. 13 shows the changes in the number of neighbor robots shown in Fig. 9 that travel at a distance of d_u from each robot. During the first 10 sec., each robot generated an equilateral triangle of a side length of d_u with its neighbors, which resulted in a significant increase of the number of neighbor robots at a distance of d_u. From 10 sec. to 30 sec., the number of robots accompanied by 6 neighbors decreased, while the number of robots accompanied by 4 neighbors increased. If we take a close look at Fig. 9-(c),

Authorized licensed use limited to: Japan Advanced Institute of Science and Tech. Downloaded on November 10, 2008 at 19:46 from IEEE Xplore. Restrictions apply.

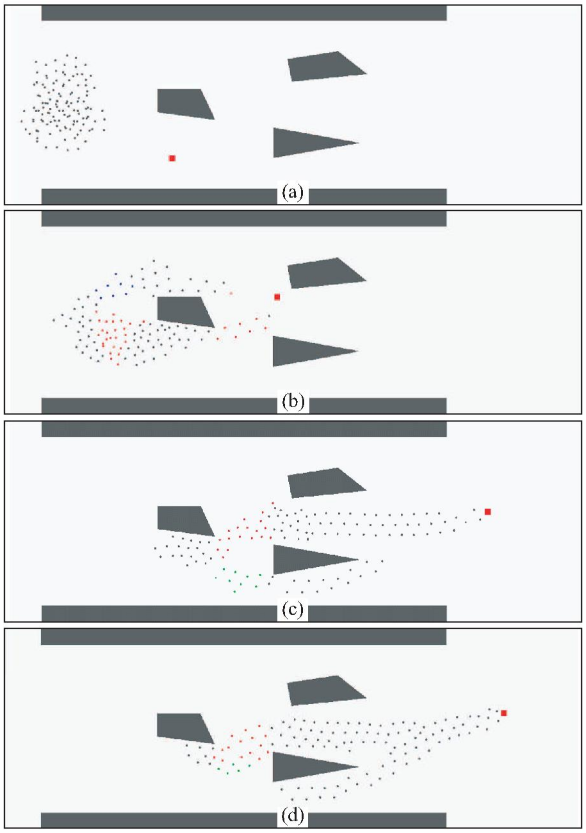


Fig. 10. Simulation results for adaptive flocking with 120 robots toward a moving goal ((a) $t = 0[sec]$(initial distribution), (b) $t = 28[sec]$, (c) $t = 67[sec]$, (d) $t = 78[sec]$)

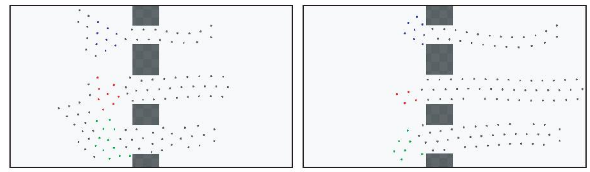
during this period, the swarm was split into multiple units due to the obstacles in its path. After the period, the multiple units were re-united and the number of neighbors gradually increased as expected.

VII. CONCLUSION

Adaptive flocking is a first step toward real-world implementations of robot swarms with a limited capability. In this paper, we presented a novel algorithm of adaptive flocking, enabling a swarm of robots to navigate toward achieving a mission while adapting to an unknown environment. We assumed that robots navigated around the obstacle under such weak conditions as limited ranges of sensing, no identification number, no pre-determined leader, no common coordinates, no memory, and no communication with each other. Through only local interactions, the swarm could adapt to an environment while maintaining a uniform distance between members. We verified the effectiveness of the proposed strategy using the in-house simulator. The simulation results show that the adaptive flocking is a simple and efficient approach to autonomous navigation in a changing environment. In practice, this approach is expected to be used in a variety of applications such as odor localization and search-and-rescue.

(a) $t = 0[sec]$ (initial distribution) (b) $t = 52[sec]$

Fig. 11. Simulation results for flocking without partition capability



(a) $t = 28[sec]$ (b) $t = 40[sec]$

Fig. 12. Simulation results for flocking without the unification capability

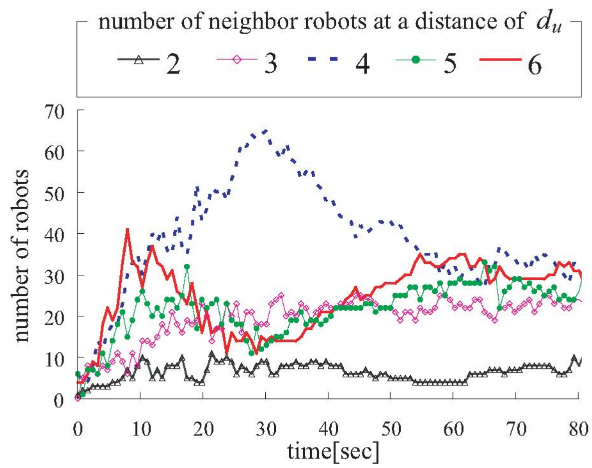


Fig. 13. Changes in the number of neighbor robots at a uniform distance in Fig. 9

ACKNOWLEDGMENTS

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

REFERENCES

[1] E. Sahin. "Swarm robotics: from sources of inspiration to domains of application," Swarm Robotics, SAB 2004 International Workshop (Revised Selected Papers) E. Sahin and W. M. Spears (Eds.), Lecture Notes in Computer Science 3342, Springer, 2005

[2] M. Egerstedt and X. Hu. "Formation constrained multi-agent control," IEEE Transactions on Robotics and Automation, Vol.17, No.6, pp.947-951, 2001

[3] C. Belta and V. Kumar. "Trajectory design for formations of robots by kinetic energy shaping," in Proc. IEEE International Conference on Robotics and Automation, 2002

[4] J. Fredslund and M. J. Mataric. "A general algorithm for robot formations using local sensing and minimal communication," IEEE Transactions on Robotics and Automation, Vol.18, No.5, pp. 837-846, 2002

Authorized licensed use limited to: Japan Advanced Institute of Science and Tech. Downloaded on November 10, 2008 at 19:46 from IEEE Xplore. Restrictions apply.

[5] L. E. Parker, B. Kannan, F. Tang, and M. Bailey. ”Tightly-coupled navigation assistance in heterogeneous multi-robot teams,” in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, 2004
[6] G. Lee, Y. Komatsu, S. Sone, and N. Y. Chong. ”Formation navigation of a team of mobile robots adapting an environment,” in Proc. 36th International Symposium on Robotics, 2005
[7] W. Spears, D. Spears, J. Hamann, and R. Heil. ”Distributed, physics-based control of swarms of vehicles,” Autonomous Robots, Vol.17, No.2-3, pp.137-162, 2004
[8] T. Balch and M. Hybinette. ”Social potentials for scalable multi-robot formations,” in Proc. IEEE International Conference on Robotics and Automation, 2000
[9] D. Zarzhitsky, D. F. Spears, and W. M. Spears. ”Distributed robotics approach to chemical plume tracing,” in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, 2005
[10] M. Shimizu, T. Mori, and A. Ishiguro. ”A development of a modular robot that enables adaptive reconfiguration,” in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, 2006
[11] D. Kurabayashi, K. Okita, and T. Funato. ”Obstacle avoidance of a mobile robot group a nonlinear oscillator network,” in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, 2006
[12] J. M. Esposito and T. W. Dunbar. ”Maintaining wireless connectivity constriants for swarms in the presence of obstacles,” in Proc. IEEE International Conference on Robotics and Automation, 2006
[13] G. Folino and G. Spezzano, ”An adaptive flocking algorithm for spatial clustering,” in Proc. 7th International Conference on Parallel Problem Solving from Nature, 2002
[14] J. Nembrini, A. Winfield, and C. Melhuish. ”Minimalist coherent swarming of wireless networked autonomous mobile robots,” in Proc. International Conference on Simulation of Adaptive Behavior on From Animals to Animats, 2002
[15] E. O. Wilson. ”Sociobiology: The new synthesis,” Harvard University Press, 1976
[16] S. Stocker. ”Models for tuna school formation,” Mathematical Biosciences, Vol. 156, pp. 167-190, 1999
[17] A. Johnson. ”Famous problems and their mathematicians,” Teacher Ideas Press, 1999
[18] A. Johnson. ”Classic math,” Dale Seymour Publications, 1994
[19] D. Halliday, R. Resnick, and J. Walker. ”Fundamentals of physics,” Wiley, 5th Ed., 1997

Authorized licensed use limited to: Japan Advanced Institute of Science and Tech. Downloaded on November 10, 2008 at 19:46 from IEEE Xplore. Restrictions apply.