Paper:

A Mobile Sensor Network Forming Concentric Circles Through Local Interaction and Consensus Building

Geunho Lee*, Seokhoon Yoon*, Nak Young Chong*, and Henrik Christensen**

*School of Information Science, Japan Advanced Institute of Science and Technology,
Nomi, Ishikawa 923-1292, Japan
E-mail: {geun-lee, seokhoon, nakyoung}@jaist.ac.jp
**College of Computing, Georgia Institute of Technology,
Atlanta, GA 30332, USA
E-mail: hic@cc.gatech.edu

[Received February 2, 2009; accepted May 25, 2009]

We address the problem of a swarm of autonomous mobile robotic sensors generating geometric shapes to build wireless ad hoc surveillance sensor networks. Robot swarms with limited observation are required to form different shapes under different task conditions. To do this, we propose decentralized coordination enabling a robot swarm dispersed across an area to form a desired shape. Our approach has *n* robots generate a circumscribed circle of a regular *n*-polygon based on local interaction with neighboring robots. The approach also enables a large robot swarm to form concentric circles through consensus. We mathematically demonstrate convergence confirming the feasibility using extensive simulation. Our results indicate that our approach is applicable to mobile sensor network surveillance and security networks.

Keywords: sensor network, robot swarm, concentric circle, local interaction, consensus

1. Introduction

The ever-widening variety of applications to which robot swarms are put in part is endlessly interesting and entertaining, because individual robots may be unaware of how their local movement affects swarm behavior overall [1]. We focused on how individual robot mobility may be controlled to collectively achieve a desired spatiotemporal swarm structure for a given task. Building on our earlier work for uniformly dispersing a sensor network [2], this paper presents decentralized coordination for a robot swarm generating concentric circles, originally inspired by cyclic insect pursuit [3] in which individuals maintain constant interval between themselves and others.

The problem may be better described by asking how to enable a swarm of robots to form concentric circles on a two-dimensional plane based solely on local information. Our main objective here is to provide a distributed coordination solution in which robots eventually generate circumscribed circles around regular polygons while making the centroids of individual circles coincides. This in turn may shed light on the implementation of other geometric formations of symmetry. In this sections that follow, the properties of proposed solutions are explained mathematically and their convergence analyzed. We show that a large swarm of mobile robots with limited visibility can establish concentric circles through extensive simulation. The encouraging results indicate that robot swarms can are be applied effectively to such problems as surveillance and security and contamination detection.

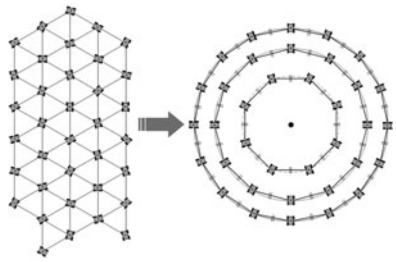


Fig. 1. Generating concentric circles through self-configuration.

This paper is organized as follows: Section 2 briefly describes related works and motivation. Section 3 formally defines the concentric circle generation problem. Section 4 details our decentralized coordination approaches and their convergence properties. Section 5 presents the overall algorithm for generating concentric circles, Section 6 summarizes simulation results of simulations, and Section 7 draws conclusions.

2. Background and Motivation

2.1. Related Work

Researchers in swarm robotics have proposed decentralized control schemes for self-configuration or geometric shape generation broadly divided into global and local strategies based on robot observation and/or communication range. Global strategies [4, 5] provide fast, accurate, efficient deployment but are technically unfeasible and lack scalability as the number of robots increases. Local strategies, based mainly on nature-inspired interactions between individual robots, may be further divided into biological emergence [6, 7], behavior-based [8, 9], and virtual physics-based [10–13].

Local strategies yield two different deployments depending on whether or not robots share the same a priori global information, e.g., the number of robots and the center point of the desired shape. Without global information, local interactions result mainly in lattice networks [2, 14]. While such configurations provide dense overage and multiple redundant connections ensuring maximum topological reliability and flexibility, they may not yield the desired overall geometric shape. When we are primarily interested in the overall geometric shape, both predefined geometric neighbor relations and a global reference should be provided that include a leader, common coordinates, or individual identifiers [15]. The context behind shape generation in [16] is that local strategies are used to solve organization and pattern generation problems at the group level.

Another important issue facing robot swarm coordination is consensus, or agreement, studied mainly based on graph theory [17]. When a large number of robots cooperates to conduct a specific mission, they must share available information resources. Relative positioning data, for example, enables a robot to construct its state structure for other robots [18], or robots directly exchange data mutually over a wireless network [19]. Theoretical information sharing techniques include time-invariant information exchange topology [20], dynamic information exchange topology [21], and communication delays [22]. Consensus techniques have also been used in such applications as pattern formation [23] and flocking [24].

In earlier work [2, 14], we presented the self-configuration of a robot swarm that configures itself on a two-dimensional (2-D) plane with geographic constraints. Robots are basically considered to be liquid particles that change location based on the shape of the container they occupy. Specifically, local interaction based on partially connected topology enables three neighboring robots to converge as an equilateral triangle. Accumulating such individual local robot behavior organizes uniformly spaced robot swarm to fill the environment. Our approach constructed uniformly spaced equilateral triangles conforming to the border of an unknown area, unlike in [13]. Here we assume that a swarm of robots disperses itself in an area with uniform spatial density.

In attempting to control a desired swarm shape, Suzuki and Yamashita [4] studied the generation of regular polygons based on a nonoblivious algorithm with unlimited memory capability. Defago and Konagaya [5] modified into an oblivious, or memoryless, algorithm, applying it to the algorithm of circles formation, decomposing the problem into two subproblems - first placing robots a circle and, second, arranging robots evenly along circles. Our step in forming concentric circles is far more challenging then a single circle and, to the best of our knowledge, no such research has been done previously.

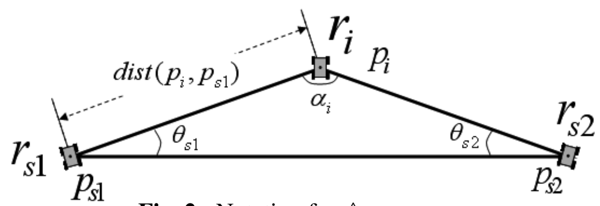


Fig. 2. Notation for $\triangle p_i, p_{s1}, p_{s2}$.

2.2. Why Concentric Circles?

Consider military surveillance and security in defending a territory against invaders. A security surveillance network could be built within and around the territory using mobile robotic sensors, but how to distribute the robotic sensors? Assuming the territory to be a point, a ring network would position robots the same distance from the point and provide omnidirectional coverage but any single sensor failing could cause the entire network to fail. Single robot movement or change thus affects the entire network performance. Hence, a multiple ring network, i.e., concentric circles, of several interconnected rings would overcome limitations while maintaining the ring network as shown in **Fig. 1**. Topologically, the multi-ring network has three advantages: (1) Rings are scalable because the robotic sensor workload is independent of total number. (2) Individually, sensors depend only on their neighbors, making rings easy to manage decentralized way. (3) Rings are robust against sensor failure or delay, thanks to the overall network's additional inner and outer rings.

The formation of concentric circles also yields three desirable effects: (1) Robots are enabled to reach consensus on a common origin, with the circle easily scaled up or down radially. for the common origin. (2) A very useful key is provided for self-positioning robots or sensors around to the origin. (3) Motion control required by shape formation is extended and changed straightforwardly to include the flocking problem.

3. Problem Statement

3.1. Robot Model and Notation

We consider a swarm of autonomous mobile robots denoted as $r_1, \ldots, r_i, r_j, \ldots, r_m$ on a plane, all within a single network constructed by our previously proposed self-configuration [2]. Robots have no leaders and identifiers, share no common coordinates, and retain no memory of past action. Due to a limited sensing boundary *SB*, they detect the locations of other robots only within a certain range. In addition, each robot is not allowed to communicate explicitly with other robots.

The distance between the robot r_i's position p_i and the robot r_j's position p_j is denoted as $dist(p_i, p_j)$, as shown in **Fig. 2**. Uniform interval d_u is defined as the desired distance between each robot in the formation. Each robot r_i detects the position $\{p_1, p_2, \cdots\}$ of other robots within its *SB* for its local coordinates. r_i can select two robots r_{s1}

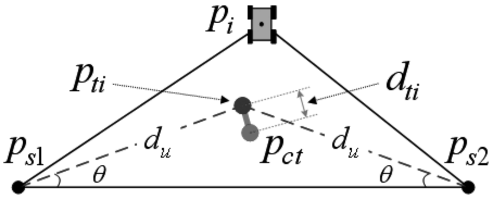


Fig. 3. Target calculation in the next time step through local interaction.

and r_{s2} called the neighbors of r_i, and the set of their positions is denoted as $\{p_{s1}, p_{s2}\}$. Given p_i and $\{p_{s1}, p_{s2}\}$, a *triangular configuration*, denoted by $\triangle p_i p_{s1} p_{s2}$, is defined as a set of three distinct positions $\{p_i, p_{s1}, p_{s2}\}$. Internal angle $\angle p_{s1} p_i p_{s2}$ of r_i is denoted by α_i in **Fig. 2**. Similarly, internal angles $\angle p_i p_{s1} p_{s2}$ and $\angle p_i p_{s2} p_{s1}$ are denoted as θ_{s1} and θ_{s2}. Local interaction is formally defined as follows: Given $\triangle p_i p_{s1} p_{s2}$, *local interaction* enables r_i to maintain d_u with $\{p_{s1}, p_{s2}\}$ at each time an isosceles triangle configuration is formed.

3.2. Problem Definition

We formally address the CONCENTRIC CIRCLE FORMATION problem for mobile robots with limited capabilities (mentioned above) as follows:

How should individual robot mobility be controlled to achieve concentric circles on a 2-D plane?

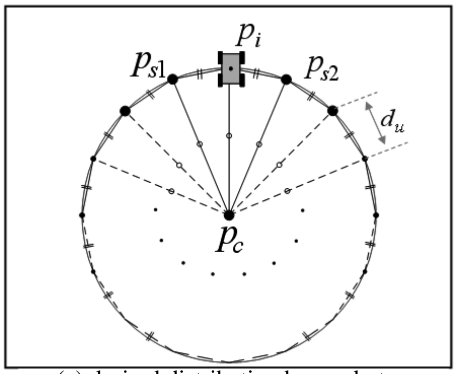
A circle is approximated by a regular n-polygon whose vertices correspond to the positions of n robots. Concentric circles share the same centroid. Circle generation requires that robots form a regular n-polygon while maintaining d_u between adjacent robots. Robots then must agree on the centroid of individual circles. The concentric circle formation problem is then decomposed into the following two subproblems:

- **Problem 1**, *Motion Control* How can n robots be made to converge into individual vertices of a regular n-polygon?
- **Problem 2**, *Observation Consensus* How can to make all robot circles be made to share the same centroid?

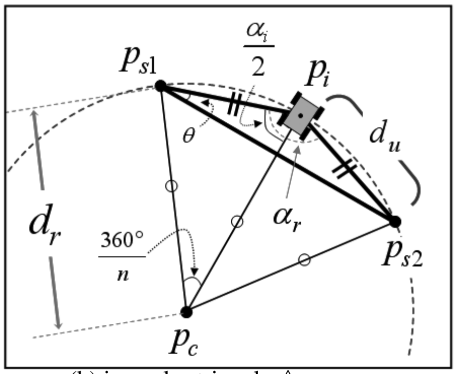
4. Decentralized Coordination

4.1. Motion Control

In controlling individual robot motion control, r_i interacts with its two neighboring robots r_{s1} and r_{s2} to calculate the target point p_{ti} at $t+1$, enabling the three robots to eventually form an isosceles triangle with side length d_u, as shown in **Fig. 3**. Once centroid p_{ct} in $\triangle p_i p_{s1} p_{s2}$ is obtained at t, r_i calculates p_{ti}, where line segment $\overline{p_{ct} p_{ti}}$ is $d_{ti}(= 2 d_u \sin\theta/3)$ long. Connecting individual isosceles triangles, as shown in **Fig. 4 (a)**, n robots are placed in the same interval d_u on a circumscribed regular n-polygon. Since all robots control motion in the same way, their p_{ti}



(a) desired distribution by n robots.



(b) isosceles triangle $\triangle p_i p_{s1} p_{s2}$.

Fig. 4. Geometric features under motion control.

at $t+1$ converges at an n-polygon vertex, thus yielding the n-polygon.

In considering a circumscribed circle around a regular n-polygon whose centroid is p_c and side length is d_u, triangle $\triangle p_i p_{s1} p_{s2}$ is an isosceles triangle with d_u and α_i, as shown in **Fig. 4 (b)**, so the internal angle $\angle p_i p_{s1} p_{s2}$ (=θ_{s1}) is identical to $\angle p_i p_{s2} p_{s1}$ (=θ_{s2}). Here, we denote θ_{s1} and θ_{s2} for simplicity as θ, the distance between p_c and each vertex is identical, and internal angles between the distance vectors $\vec{d_r}$ connecting p_c and each vertex are all $2\pi/n$. The distance vector length is denoted as d_r, the desired convergence distance, so two triangles $\triangle p_i p_{s1} p_c$ and $\triangle p_i p_{s2} p_c$ are congruent, both isosceles with side length d_r. In **Fig. 4 (b)**, angle $\angle p_{s1} p_i p_c$ is obtained as follows:

$$\frac{\alpha_i}{2} = \frac{\pi - (2\pi/n)}{2}. \qquad (1)$$

For convenience, α_r is used instead of α_i because α_i is a desired convergence angle identically applied to all robots located on each vertex of the n-polygon. Eq. (1) is rewritten as follows:

$$\alpha_r = \pi - \frac{2\pi}{n},\ n > 2. \qquad (2)$$

Similarly, in $\triangle p_i p_{s1} p_{s2}$, θ is given by $\theta = \pi/n$. If and only if these conditions are satisfied, n robots are considered placed on same circumference with the same interval d_u. Using the sine formula ($\frac{a}{\sin\alpha} = \frac{c}{\sin\gamma}$), d_r is rewritten as $d_r = d_u \frac{\sin(\frac{\pi-(2\pi/n)}{2})}{\sin(2\pi/n)}$. d_r is straightforwardly rewritten as

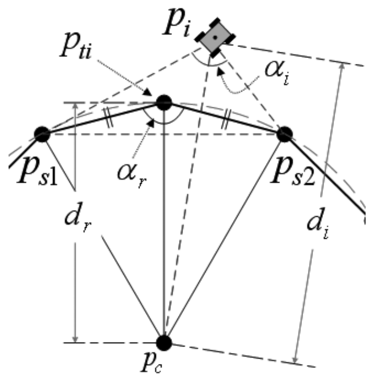


Fig. 5. Two control parameters (d_i and α_i) under motion control.

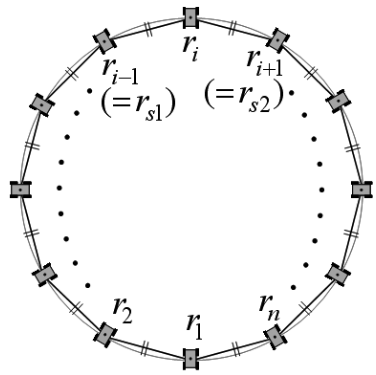


Fig. 6. Network model with ring topology.

follows:

$$d_r = \frac{d_u}{2\sin(\pi/n)}. \quad (3)$$

From the desired configuration above, r_i motion control is modeled by both $d_i(t)$ from p_c and $\alpha_i(t)$ as shown in **Fig. 5**. $d_i(t)$ is controlled as follows:

$$\dot{d}_i(t) = -a(d_i(t) - d_r) \quad (4)$$

where a is a positive constant. The solution of Eq. (4) is $d_i(t) = |d_i(0)|e^{-at} + d_r$ converging exponentially at d_r as t approaches infinity. $\alpha_i(t)$ is then controlled as follows:

$$\dot{\alpha}_i(t) = k(\alpha_r - \alpha_i(t)) \quad (5)$$

where k is a positive number. Eq. (5) is solved likewise with $\theta_i(t) = |\alpha_i(0)|e^{-kt} + \alpha_r$, which converges exponentially at α_r as t approaches infinity. Eqs. (4) and (5) imply that the trajectory of r_i converges at equilibrium state $x_e = [d_r \ \alpha_r]^T$. To show convergence at state $x_i(t) = [d_i(t) \ \alpha_i(t)]^T$, we use stability analysis based on Lyapunov's theory [25]. Convergence at the desired configuration is obtained by minimizing the energy level of the following scalar function:

$$v_i(x_i) = \frac{1}{2}(d_i - d_r)^2 + \frac{1}{2}(\alpha_r - \alpha_i)^2. \quad (6)$$

This scalar function is always positive definite except $d_i \neq d_r$ and $\alpha_i \neq \alpha_r$. The scalar function is derived by $\dot{v}_i = -(d_i - d_r)^2 - (\alpha_r - \alpha_i)^2$, which is negative definite. The scalar function is radially unbounded since it tends to $||V|| \to \infty$ as $||x_i(t)|| \to \infty$, so x_e is asymptotically stable, implying that r_i reaches the desired configuration.

To demonstrate the convergence of collective motions for n robots located on each vertex of the n-polygon, we define n-order scalar function V as follows:

$$V = \sum_{i=1}^{n} v_i(x_i). \quad (7)$$

Based on the convergence of Eq. (6), V is straightforwardly verified as positive definite and $\dot{V}$ as negative definite. V is radially unbounded because it tends to infinity as t approaches infinity. n robots consequently move toward equilibrium.

Convergence properties so far have been analyzed assuming that robots have information about the number of n robots to be located on the same circle. Under this assumption, our motion control enables n robots to achieve the desired configuration. Practically, due to limitations in *SB*, i.e., *locality*, in most cases, it is not possible to measure the number of n robots. Our discussion here is therefore only a necessary condition for the decentralized coordination of generating concentric circles. When no a priori information exists on the number of robots, we must determine how to reach agreement or how to share information about the number of n robots.

4.2. Observation Consensus

Consider a ring network shown in **Fig. 6**. This network consists of n ($\in R$) robots represented by corresponding point nodes numbered 1 through n in a 2-D plane. R is a real vector space in which each vector has a positive integer length. In the network, we use undirected graph $\mathcal{G} = \{V(\mathcal{G}), E(\mathcal{G})\}$ where $V(\mathcal{G})$ is a set of n vertices $V(\mathcal{G}) = \{v_1, v_2, \ldots, v_n\}$ and $E(\mathcal{G})$ is a set of edges between vertices $E(\mathcal{G}) = \{(v_i, v_j)|v_i, v_j \in V(\mathcal{G})\}$. We denote $V(\mathcal{G})$ and $E(\mathcal{G})$ for simplicity as $\mathcal{V}$ and $\mathcal{E}$. Let $A = (a_{ij})$ denote the symmetric adjacent matrix formalized as follows:

$$a_{ij} \triangleq \begin{cases} 1 & \text{if } (v_i, v_j) \in \mathcal{E} \\ 0 & \text{otherwise.} \end{cases} \quad (8)$$

Let $D = (d_{ij})$ denote the diagonal matrix formalized as follows:

$$d_{ij} \triangleq \begin{cases} d(v_i) & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \quad (9)$$

where $d(v_i)$ indicates the valance, or the number of edges, at vertex i. In a ring topology, $d(v_i)$ is *2*. The relation is defined as $\mathcal{N}_i \triangleq \{j \in \mathcal{V} : a_{ij} \neq 0\}$ and invariant while motion control is being executed. $\mathcal{N}_i$ means the relation between the two most adjacent neighbors on the same circle, so we denote this as *observation graph* $\mathcal{G}_o$. In $\mathcal{G}_o$, graph Laplacian [17] $L(\mathcal{G}_o)$ is defined as follows: $L \triangleq D - A$, where $L \in R^{n \times n}$.

With no a priori information about the number of robots n, the consensus solution enables state $x_i(t)$ (=$[d_i(t) \ \alpha_i(t)]^T$) to eventually reach agreement on equilibrium state $x_e = [d_r \ \alpha_r]^T$, for which see Eqs. (2) and (3). x_e depends on robot number n, so the problem becomes how to find n that enables x_i to converge at x_e. To reach consensus on n, which enables robots to form a regular n-polygon with p_c while maintaining d_u between adjacent robots, we consider r_i with the following network dynamics:

$$\dot{z}_i(t) = u_i(t) \quad \text{. . . . . (10)}$$

where $z_i \in R$ denotes r_i's state value and $u_i \in R$ r_i's input. Using $z_i = n$ as $t \to \infty$, where n exceeds 2, we obtain:

$$z = n\mathbf{1} \quad \text{. . . . . (11)}$$

where z is $[z_1 \ \cdots \ z_n]^T$ and $\mathbf{1}$ $[1 \ \cdots \ 1]^T$. To find n, the input of a consensus protocol in the network is defined as follows:

$$u_i(t) = \sum_{j \in \mathcal{N}_i} a_{ij}(z_j - z_i). \quad \text{. . . . . (12)}$$

Using Eqs. (10) and (12), the consensus protocol is summarized as

$$\dot{z}_i(t) = \sum_{j \in \mathcal{N}_i} a_{ij}(z_j - z_i). \quad \text{. . . . . (13)}$$

Collecting each consensus protocol of r_i into a matrix equation based on L, we obtain:

$$\dot{z}_i(t) = -Lz. \quad \text{. . . . . (14)}$$

If $\mathscr{G}_o$ in the n-polygon is connected, L has the properties that every row sums to 0, all diagonal elements are positive, and all off-diagonal elements are negative. When all nonzero eigenvalues of L have strictly positive real parts, the protocol solves the consensus problem [22], i.e., if n in (11) is found, x_i converge at x_e as $t \to \infty$.

To show convergence at n based on Lyapunov's theory [25], we first consider the following scalar function:

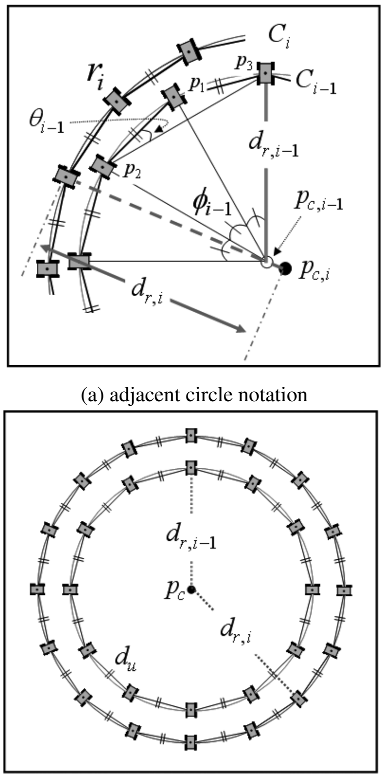
$$V(z) = \frac{1}{2} z^T Lz. \quad \text{. . . . . (15)}$$

This scalar function is positive semidefinite due to the positive semidefiniteness of L ($L1 = 0$ and *rank* $L = n-1$: nonfull-rank). The derivative of the scalar function is given by

$$\dot{V}(z) = \frac{1}{2}(\dot{z}^T Lz + z^T L\dot{z}) = -z^T L^T Lz \leq 0. \quad \text{. (16)}$$

Note that the scalar function derivative is negative semidefinite. Here, the convergence cannot be shown only using Lyapunov's theory. Instead, we attempt to show convergence based on LaSalle's theory [26]. If the set of points where $\dot{V}(z)$ in Eq. (16) was 0 is small and, at most of the points, the vector field is not parallel to the set, then LaSalle's invariance principle enables us to use the scalar function of Eq. (15) to prove asymptotic stability.

We analyze the convergence property. Since Eq. (14) is a locally Lipschitz map [26] from R^n into R^n, we define set $\Omega \triangleq \{z \in R^n | V(z) \leq c\}$ where $c \in [0 \ \ \infty)$. Next, we check whether Ω is a close, compact, positive invariant set. If $V(z) \to \infty$ as $||z|| \to \infty$, Ω is bounded for all values of c as the compact set. $V(z)$ is, moreover, a decreasing function because $\dot{V}(z) \leq 0$ in Ω, from which Ω is easily seen to be positive invariant set. We therefore define set $E = \{z \in \Omega | \dot{V}(z) = 0\}$, and set M as the largest set in E. Since E itself is the positive invariant set within our case, i.e., $z(0) \in M \Longrightarrow z(t) \in M, \forall t \geq 0$, $M = E$. Our interest now lies in showing that $z(t) \to M$ as $t \to \infty$, i.e., our consensus problem is that the only solution that remains identically in E is $z = n\mathbf{1}$ in Eq. (11). Eq. (16) is therefore rewritten as $\dot{V}(z) = 0 \Rightarrow -z^T L^T Lz = -(Lz)^T(Lz) = 0$. Since $\dot{V}(z) = 0 \Rightarrow Lz = 0$, using the properties of L, we obtain the following:



(a) adjacent circle notation

(b) two coincided at circles

Fig. 7. Reaching an agreement on the same centroid in individual circles.

$$\dot{V}(z) = 0 \Rightarrow z = n\mathbf{1}. \quad \text{. . . . . (17)}$$

From Eq. (17), $z(t)$ approaches $n\mathbf{1}$ as $t \to \infty$, i.e., the consensus protocol in Eqs. (12) and (13) enables r_i to achieve x_e as $t \to \infty$ through convergence $n\mathbf{1}$.

5. Algorithm for Generating Concentric Circles

We now discuss our algorithm for generating concentric circles by forming individual circumscribed circles around regular n-polygons. Robots first determine where they are after dispersing themselves uniformly in an area, as shown in **Fig. 1**, by measuring the number of robots located at uniform distance d_u. If the number of robots is less than 6, r_i is in the outermost, or boundary, layer, and starts to interact with its neighbors to expand out-

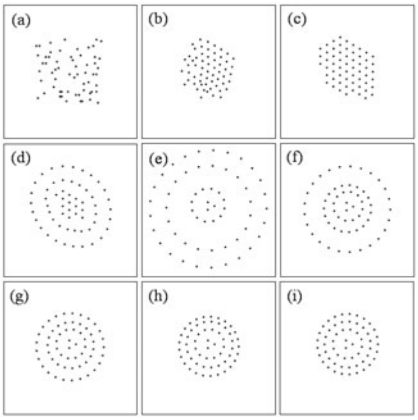


Fig. 8. Concentric circles generated by 60 robots

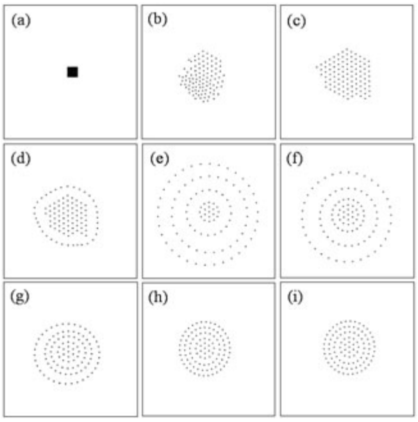


Fig. 9. Concentric circles generated by 100 robots

ward. Otherwise, it remains idle. Specifically, r_i determines the center point of the robots within its *SB*, then, r_i moves away from the center point in radially along a vector connecting the center point and p_i, and defines the interaction range by rotating the radial direction vector 90 degrees clockwise and counterclockwise. r_i finally selects the two neighbors that have the smallest angle between each boundary of interaction range and vectors connecting p_i and robots located within d_u. After neighbors are selected, the circle starts to be generated, with the same process done sequentially for the subsequent layers.

Based on the coordination in Section 4, concentric circles are generated in two stages. Stage 1 forms individual circles. Reaching agreement on the number of robots located on the same circle, n robots form a regular n-polygon considered a circumscribed circle around the polygon with p_c and d_r. After individual circle generation, r_i continues with another attempt at consensus enabling agreement to be reached on the centroid of individual circles. To consider notation and definitions, as shown in **Fig. 7 (a)**, we denote the circle generated by r_i as C_i with its centroid $p_{c,i}$. C_i denotes a set of points $\{r_1,\ldots,r_n\}$ occupied by n robots. Note that C_{i-1} whose centroid is $p_{c,i-1}$ exists inside outermost circle C_i. Each angle between two adjacent robots and $p_{c,i}$ is denoted as ϕ_i. Similarly, each angle between two adjacent robots and $p_{c,i-1}$ is denoted as ϕ_{i-1}. Radius $d_{r,i}$ from $p_{c,i}$ to the vertex in C_i, and radius $d_{r,i-1}$ from $p_{c,i-1}$ to the vertex in C_{i-1} are shown in **Fig. 7 (a)**. Angle $\angle p_1p_2p_3$ is denoted as θ_{i-1}. After forming C_i with $d_{r,i}$, r_i attempts to agree on $p_{c,i-1}$ of C_{i-1}. $p_{c,i}$ is made coincident with $p_{c,i-1}$ by having each of the robots in C_i conduct the following four sequential processes: (1) r_i finds the number of robots located on C_{i-1} by calculating θ_{i-1}. (2) Using θ_{i-1}, r_i obtains ϕ_{i-1}. (3) $d_{r,i-1}$ is calculated using the obtained θ_{i-1} and ϕ_{i-1}. (4) r_i defines $p_{c,i-1}$ for with respect to its local coordinates.

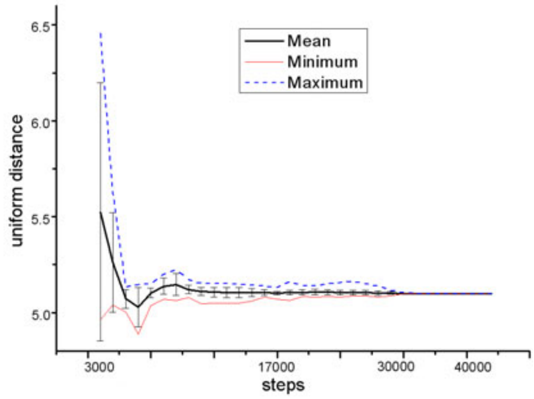
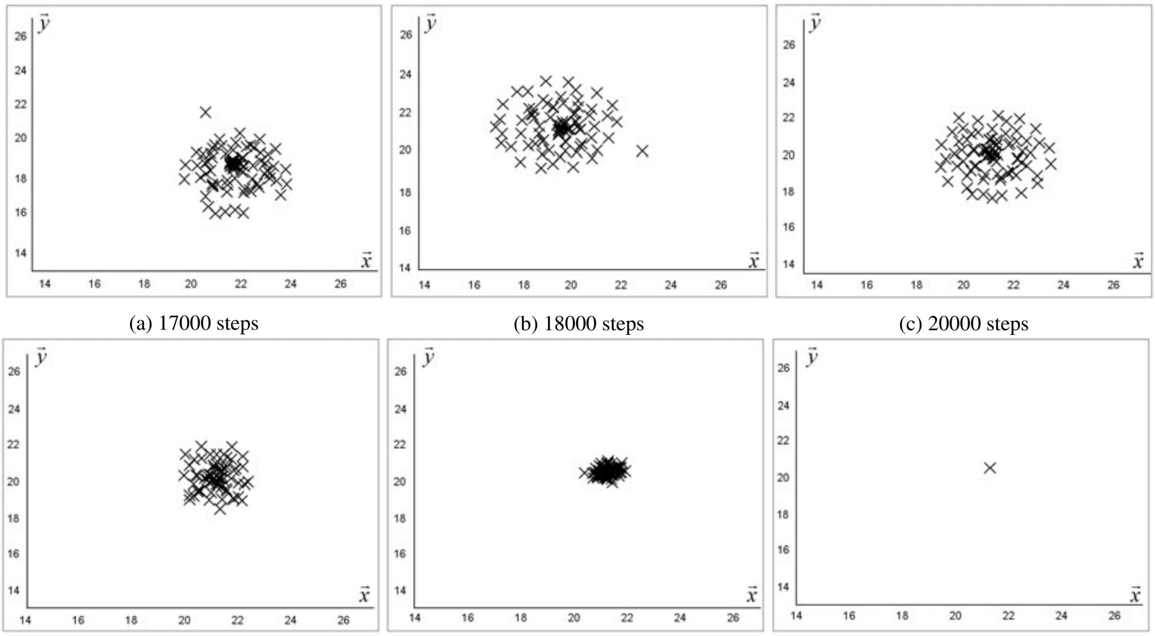


Fig. 10. Distance variation between p_i and two neighboring points during concentric circle generation.

After setting the new r_i's $p_{c,i-1}$, robots in C_i moves to reach agreement on $p_{c,i-1}$ while maintaining d_u. At this very moment, d_r in Eq. (3) is newly set to calculated $d_{r,i-1}$. Repeating motion control in Eqs. (2) and (3), the overall robot swarm generates concentric circles having p_c, *i.e.*, $p_{c,i-1} = p_{c,i}$, and maintaining d_u with neighboring robots on the same circle as shown in **Fig. 7 (b)**.

6. Simulation Results and Discussion

Having developed our swarm robot simulator to verify the validity of the proposed algorithm, uniform distance d_u was set to 5.1 throughout the simulation. **Figs. 8** and **9** show the results of how a robot swarm discriminates itself to generate concentric circles. Note that concentric circles were formed after self-configuration shown in **Figs. 8 (a)**-**(c)** and **9 (a)**-**(c)**, so dispersed robots have a uniform spatial density. After dispersion, each robot determines for itself whether its current location is at the outermost layer of the swarm. Robots in the outermost layer started generating a circle, as shown in **Figs. 8 (d)** and **9 (d)**. Using the consensus on the number of robots, each robot relocates to form an isosceles triangle and, by repeating this process, all robots generate circular patterns. After a circle is generated by n robots, robots attempt to have the centroid of the circle agree with that of the adjacent circle radially inward, as shown in **Figs. 8 (h)** and **9 (h)**. **Fig. 8 (i)** shows that a swarm of 60 robots has generated 4 concentric circles and **Fig. 9 (i)** shows that a swarm of 100 robots has generated 6 concentric circles while maintaining d_u with neighboring robots along the same circle.



(a) 17000 steps (b) 18000 steps (c) 20000 steps

(d) 24000 steps (e) 28000 steps (f) 32000 steps

Fig. 11. Agreement on concentric circle centroid.

Figure 10 shows the simulation results for 100 robots with 15 initial uniform dispersion states, where distance variations between each robot and their adjacent robots are plotted based on the activation cycle step. The bold solid line denotes the mean value, the solid line the minimum value, and dashed line the maximum value. Error bars represent 95% confidence intervals. Note that each robot converged at predetermined uniform distance d_u. After 17,000 steps, the mean converged quite significantly at d_u. A closer look at the overall process shows that the generation of individual circles was completed at this step. Robots thereafter relocated to reach agreement on the centroid of individual circles, so the mean changes only negligibly.

Figure 11 shows the process of reaching agreement on the centroid of concentric circles. Note in **Fig. 11 (a)** through **(c)** that the centroids of circles are not coincident and continue changing for a while, during which individual circles move toward a new centroid to make them coincident with each other. The robot swarm, thus, converged at the same centroid and the same d_u.

Three main features highlight our approach: (1) Our algorithm enables a robot swarm with limited sensing capabilities to form concentric circles. We proposed local interaction-based motion control to form an isosceles triangle. Robots generate a circle circumscribed by a regular n-polygon while reaching agreement on the number of n robot located on the same circle, then further agree on the centroid of individual circles. (2) Isosceles triangles are built on the circumscribed circles of regular n-polygons. The triangle element is easy to construct and highly scalable as the number of robots increases. (3) Our approach eliminates major assumptions such as robot identifiers, common coordinates, global orientation, specific leaders, and direct communication. Robots calculate their target position without having to require past actions or states, which makes it easier to cope with transient error.

Practically speaking, three issues remain: (1) Our proposal relies on the fact that robots sense the positions of neighboring robots precisely using infrared [27, 28] or sonar sensors [16], which require improvement. (2) Individual robots were not allowed to communicate explicitly with other robots. Using direct communications enables different shapes to be formed arbitrarily based on role assignment. Robots still require a priori knowledge, however, such as individual identifiers or global coordinates. Direct communication may also involve difficulties as limited bandwidth, range, and interference. (3) Certain applications require robot swarms to form concentric shapes while adapting to environmental borders such as holes, walls, or obstacles. To further facilitate implementation of our proposal in such real environments, a border-conforming approach must be developed.

7. Conclusion

We have presented a distributed shape generation algorithm enabling large robot swarms to form concentric circles on a two-dimensional plane. In being recognized as a generalized approach, this work sheds light on the generation of other symmetrical shapes. The key competitive advantage is that robots cooperatively form a shape under minimal conditions such as no leader, no unique identifiers, no common coordinates, and no direct mutual communication. Based on local interaction among neighboring robots that form an isosceles triangle alone, robots generate circles circumscribed by regular n-polygons while reaching consensus on the number of robots located in the same circle, then further agreeing on the centroid of individual circles. The properties of this approach have been discussed mathematically and verified through extensive simulation. The results indicate that swarms of robots carrying minimal capability sensors can be applied effectively to such deployment as surveillance sensor networks and contamination detection.

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] 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.

[3] N. Boeddeker and M. Egelhaaf, "Steering a virtual blowfly: simulation of visual pursuit," Proc. Biological Sciences, Vol.270, No.1527, pp. 1971-1978, 2003.

[4] I. Suzuki and M. Yamashita, "Distributed anonymous mobile robots: formation of geometric patterns," SIAM Jour. of Computing, Vol.28, No.4, pp. 1347-1363, 1999.

[5] X. Defago and A. Konagaya, "Circle formation for oblivious anonymous mobile robots with no common sense of orientation," Proc. 2nd ACM Int. Work. Principles of Mobile Computing, pp. 97-104, 2002.

[6] G. Baldassarre, V. Trianni, M. Bonani, F. Mondada, M. Dorigo, and S. Nolfi, "Self-organized coordinated motion in groups of physically connected robots," IEEE Trans. on Systems Man and Cybernetics - Part B, Vol.37, No.1, pp. 224-239, 2007.

[7] Y. Ikemoto, Y. Hasegawa, T. Fukuda, and K. Matsuda, "Graduated spatial pattern formation of robot group," Information Sciences, Vol.171, No.4, pp. 431-445, 2005.

[8] B. Werger and M. J. Mataric, "From insect to internet: situated control for networked robot teams," Annals of Mathematics and Artificial Intelligence, Vol.31, pp. 173-198, 2001.

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

[10] Y. F. Zheng and W. Chen, "Mobile robot team forming for crystallization of protein," Autonomous Robots, Vol.23, No.1, pp. 69-78, 2007.

[11] 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.

[12] J. Reif and H. Wang, "Social potential fields: a distributed behavioral control for autonomous robots," Robotics and Autonomous Systems, Vol.27, No.3, pp. 171-194, 1999.

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

[14] 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.

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

[16] G. Lee and N. Y. Chong, "Decentralized formation control for small-scale robot teams with anonymity," Mechtronics, Vol.19, No.1, pp. 85-105, 2009.

[17] J. L. Gross and J. Yellen, "Graph theory and its applications," CRC Press, 1999.

[18] J. R. Carpenter, "Decentralized control of satellite formations," Int. Journal of Robust and Nonlinear Control, Vol.12, No.2-3, pp. 141-161, 2002.

[19] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," IEEE Trans. on Automatic Control, Vol.49, No.9, pp. 1465-1476, 2004.

[20] W. Ren and R. W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," IEEE Trans. on Automatic Control, Vol.50, No.5, pp. 655-661, 2005.

[21] T. Vicsek, A. Czirok, E. Ben-Jacob, I. Cohen, and O. Schochet, "Novel type of phase transitions in a system of self-driven particles," Physical Review Letters, Vol.75, No.6, pp. 1226-1229, 1995.

[22] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," IEEE Trans. on Automatic Control, Vol.49, No.9, pp. 1520-1533, 2004.

[23] Z. Lin, B. Francis, and M. Maggiore, "Necessary and sufficient graphical conditions for formation control of unicycles," IEEE Trans. on Automatic Control, Vol.50, No.1, pp. 121-127, 2005.

[24] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, "Stable flocking of mobile agents, part i: fixed topology," Proc. 42nd IEEE Conf. Decision and Control, pp. 2010-2015, 2003.

[25] J. E. Slotine and W. Li, "Applied nonlinear control," Prentice-Hall, 1991.

[26] H. K. Khalil, "Nonlinear systems," 2nd ed., Prentice-Hall, 1996.

[27] S. Yoon, G. Lee, N. Y. Chong, and H. Christensen "Multi-robot formation generation with dual rotating infrared sensors," Proc. 39th Int. Symposium on Robotics, pp. 867-872, 2008.

[28] G. Lee, S. Yoon, N. Y. Chong, and H. Christensen, "Self-configuring robot swarms with dual rotating infrared sensors," Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, October, 2009. (to be appeared)

Name:
Geunho Lee

Affiliation:
Postdoctoral Researcher, School of Information Science, Japan Advanced Institute of Science and Technology (JAIST)

Address:
1-1 Asahidai, Nomi, Ishikawa 923-1292, Japan

Brief Biographical History:

1999 Electro. Eng., Seoul Nat. Univ. of Tech.(B.S.)

2002 Electri. & Electro. Eng., Yonsei Univ.(M.S.)

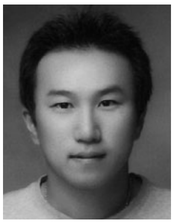
2008- Sch. Info. Sci., JAIST (Ph.D.)

Main Works:

• "Adaptive flocking of robot swarms: algorithms and properties," IEICE Trans. Communications, Vol.E91-B, No.9, pp. 2848-2855, 2008.

Membership in Academic Societies:

• The Institute of Electrical and Electronics Engineers, Inc. (IEEE)



Name:
Seokhoon Yoon

Affiliation:
Master Student, School of Information Science, Japan Advanced Institute of Science and Technology (JAIST)

Address:
1-1 Asahidai, Nomi, Ishikawa 923-1292, Japan

Brief Biographical History:

2006 Control & Instrumentation Eng., Korea Univ.(B.S.)

Membership in Academic Societies:

• Student Member, The Institute of Electrical and Electronics Engineers, Inc. (IEEE)

Name:
Nak Young Chong

Affiliation:
Associate Professor, School of Information Science, Japan Advanced Institute of Science and Technology (JAIST)

Address:
1-1 Asahidai, Nomi, Ishikawa 923-1292, Japan

Brief Biographical History:

1998-2007 Research Fellow, National Institute of Advanced Industrial Science and Technology
2003-Present Associate Professor, JAIST
2008-2009 Visiting Professor, Georgia Institute of Technology, USA

Main Works:

- "Direction Sensing RFID Reader for Mobile Robot Navigation," IEEE Trans. Auto. Sci. and Eng., Vol.6, No.1, pp. 44-54, 2009.

Membership in Academic Societies:

- The Institute of Electrical and Electronics Engineers, Inc. (IEEE)
- The Robotics Society of Japan (RSJ)
- The Society of Instrument and Control Engineers (SICE)

Name:
Henrik Christensen

Affiliation:
Georgia Institute of Technology

Address:
RIM@GT, 801 Atlantic Dr, Atlanta, GA 30308, USA

Brief Biographical History:

1987 Received M.Sc. from Aalborg University
1989 Received Ph.D. degree in EE from Aalborg University
1992-1998 Associated Professor, Aalborg University, Denmark
1998-2006 Chaired Professor, Royal Institute of Technology, Sweden
2006-Present Professor and Director, Georgia Institute of Technology, USA

Main Works:

- He was contributed novel new methods to SLAM, visual servoing and planning. He has published more than 250 contribution in conferences and journals. His current research focuses on systems integration and design across robotics and applied perception.

Membership in Academic Societies:

- Senior Member, The Institute of Electrical and Electronics Engineers, Inc. (IEEE)
- Association for the Advancement of Artificial Intelligence (AAAI)