



Distributed Circumnavigation Control with Dynamic Spacing for a Heterogeneous Multi-robot System

Weijia Yao^(✉), Sha Luo, Huimin Lu, and Junhao Xiao

Department of Automation, National University of Defense Technology,
Changsha, China

weijia.yao.nudt@gmail.com, lhmnew@nudt.edu.cn

Abstract. Circumnavigation control is useful in real-world applications such as entrapping a hostile target. In this paper, we consider a heterogeneous multi-robot system where robots have different physical properties, such as maximum movement speeds. Instead of equal-spacing which is assumed in many existing studies, dynamic spacing according to robots' properties is proposed in this paper. For this purpose, two new concepts - *utility* and *formation guideline* - are presented. Then a distributed circumnavigation control algorithm based on utilities and formation guidelines is designed for any number of mobile robots from random 3D positions to circumnavigate a target. Theoretical analysis and experimental results are provided to prove the stability and effectiveness of the proposed control algorithm.

1 Introduction

One of the most prominent research topics on distributed multi-robot system is the formation control problem. Significant efforts have been made on the *circular formation control* and *circumnavigation control* problems. In circular formation control problem, robots remain in their positions after the formation is generated, while in circumnavigation control problem, they still encircle around the target. In this sense, circular formation control could be regarded as a special case of circumnavigation control when the circumnavigation speed equals to zero.

There are already many studies on circumnavigation control (or circular formation control) problems. Most of the existing studies only consider the case where robots are distributed evenly on the formation (i.e., equal spacing), such as [3, 7, 14]. In addition, the control algorithms proposed in these studies are only applicable on the 2D plane. Nevertheless, [1] proposes algorithms which are still effective in 3D space. The formation spacing, however, is fixed and equal. Although this is effective for a homogeneous multi-robot system, it may not be sufficient for a heterogeneous one where robots have different properties, such as maximum movement speeds. [9] and [10] propose a distributed control law for a multi-robot system to form a circular formation with any desired spacing among robots. However, it assumes that the robots are placed initially on a prescribed

circle and the control algorithm is not applicable in the 3D space. Another major disadvantage is that the desired spacing, which is a global quantity, should be specified for each robot beforehand. If the specified spacing does not sum up to 2π , or if robots are informed of inconsistent specified spacing, they will form an erroneous formation. Moreover, to the best of our knowledge, there are no studies concerning dynamic spacing for a heterogeneous multi-robot system.

In this study, we suppose that mobile robots are heterogeneous in terms of their kinematics abilities, such as maximum locomotion speeds, etc. In a scenario where these mobile robots need to entrap a hostile target, their inter-robot spacing should be different for better performance; those robots with lower mobility are supposed to gather together with smaller spacing than those with higher mobility, so the probability for the target to flee away from the formation is lower. We also consider the deterioration of individual performance due to physical worn-out or damage. Therefore, their spacing should be varied in a dynamic way during the circumnavigation process. Based on this, the goal of this paper is to propose a new distributed circumnavigation control algorithm which is able to control a group of heterogeneous mobile robots from any initial positions to circumnavigate a target with dynamic spacing in the 3D space.

The main contribution of this work is twofold. First, this paper proposes the concept of *utility* and *formation guideline*. Based on these two new concepts, we design a distributed circumnavigation control algorithm which enables robots to adjust their spacing dynamically according to the local variations of their utilities; a pre-specified desired spacing is not necessary (but it is necessary for studies such as [1, 9, 10]). The control algorithm is distributed and applicable for a heterogeneous multi-robot system of arbitrary size. Second, the distributed control algorithm does not require robots to be placed initially on a prescribed circular trajectory (but it is required in [9, 10]). Their initial positions can be arbitrarily chosen in the 3D space rather than being restricted on a 2D plane (which is the case in [9, 10]). In addition, the control algorithm can respond effectively to the situation where robots quit or join the circumnavigation process (but it is not studied in much literature such as [3, 8–10]).

The remainder of this paper is organized as follows. Section 2 introduces the circumnavigation control problem based on utilities and derives its corresponding mathematical formulation. Section 3 proposes the circumnavigation control algorithm. In Sect. 4, simulation and real-robot experiments are performed and results are analysed. Finally, Sect. 5 concludes the paper and summarizes the future work.

2 Problem Formulation

The research question is that a group of n ($n \geq 2$) mobile robots, denoted by r_i , $i = 1, \dots, n$, encircle a target in 3D space with dynamic spacing on a circular formation. Note that r_i is only used to represent the i th robot for convenience of narration; it does not correspond to any physical quantities. Suppose each mobile robot is modelled by a 3D kinematic point:

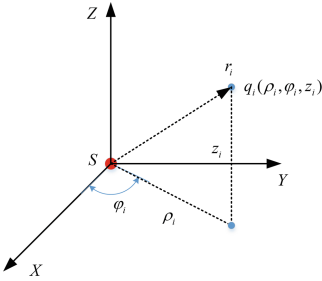


Fig. 1. The body reference frame with the target S as the origin.

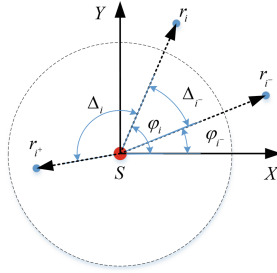


Fig. 2. Robots' projections and the target S on the XSY plane.

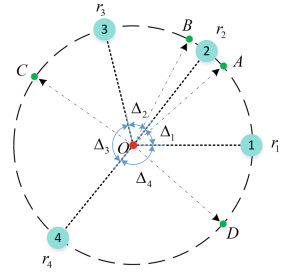


Fig. 3. The interpretation of the formation guideline.

$$\dot{p}_i(t) = u_i(t), \quad i = 1, \dots, n, \tag{1}$$

where $u_i(t)$ is the control input to the robot r_i and $p_i(t) \in \mathbb{R}^3$ is its position in the world reference frame \mathcal{W} . In this problem, robots are required to maintain on the same plane with the encircled target which is modelled by another kinematic point. Therefore, a (target) body reference frame \mathcal{B} centred at the target S is introduced (see Fig. 1). In addition, the cylindrical coordinate system is preferred to the commonly used Cartesian coordinate system since the former itself embodies three elements of interest: the distance between the projection of the robot on the XSY plane to the target (ρ), the height relative to the XSY plane (z) and the angle between the X -axis and the line joining the projection of the robot on the XSY plane with the target (φ). The cylindrical coordinates for r_i is denoted by $q_i = (\rho_i, \varphi_i, z_i)^T$ (see Fig. 1). To relate the cylindrical coordinates with the Cartesian coordinates, a vector function is defined as $q(p) = (\rho(p), \varphi(p), z(p))^T$, where $p \in \mathbb{R}^3$ is a generic vector with components p_x, p_y, p_z . $\rho(p) = \sqrt{p_x^2 + p_y^2}$, $\varphi(p) = \tan^{-1}(p_y/p_x)$ and $z(p) = p_z$. Note that $\varphi \in [0, 2\pi)$. The Jacobian matrix of the vector function will be used later, which

$$\text{is } J = \frac{\partial q}{\partial p^T} = \begin{bmatrix} \frac{p_x}{\sqrt{p_x^2 + p_y^2}} & \frac{p_y}{\sqrt{p_x^2 + p_y^2}} & 0 \\ \frac{-p_y}{p_x^2 + p_y^2} & \frac{p_x}{p_x^2 + p_y^2} & 0 \\ 0 & 0 & 1 \end{bmatrix}. \text{ For better analysis, we label the robots in}$$

the counter-clockwise direction according to their initial (angular) positions (φ_i) in \mathcal{B} as shown in Fig. 2. Note that the subscript i^- and i^+ represent the indices of the neighboring robots of r_i in the clockwise and counter-clockwise direction respectively. Especially, if $i = n$, $i^+ = 1$, and if $i = 1$, $i^- = n$. $\Delta_i > 0$ represents the difference between the angular positions of r_{i^+} and that of r_i . In particular,

$$\Delta_i = \begin{cases} \varphi_{i^+} - \varphi_i, & i = 1, \dots, n - 1, \\ \varphi_1 - \varphi_n + 2\pi, & i = n. \end{cases} \tag{2}$$

Also note that $\sum_{i=1}^n \Delta_i = 2\pi$,

Before giving the definition of the circumnavigation problem with dynamic spacing, we propose the concept of *utility*.

Definition 1 (Utility). *In a heterogeneous multi-robot system, given different kinds of robots, a robot's utility $\mu(t) \geq 0$ is determined by a given criterion (such as its maximum movement speed). The utility reflects the weight of the robot in the circumnavigation process at time t .*

For example, suppose a robot's maximum movement speed is the criterion. Let $\mu_i(t) = \frac{v_{mi}(t)}{v_M}$, $i = 1, \dots, n$, where $v_{mi}(t)$ is the maximum movement speed of r_i and v_M is the possible greatest movement speed in the heterogeneous multi-robot system. Then $\mu_i(t) \in [0, 1]$, $i = 1, \dots, n$. When $\mu_i(t) = 0$, the robot r_i cannot continue the circumnavigation process with other robots. In this case, its neighboring robots will neglect its role in the circumnavigation process. $\mu_i(t)$ will increase or decrease due to the enhancement or damage of the robot's locomotion capabilities. To explain directly how utilities are utilized to enable dynamic spacing among robots, we simply regard the utility of a robot to be proportional to its maximum movement speed. For simplicity of writing, the symbol t is neglected from μ unless it causes confusion. The circumnavigation control problem based on utilities is defined as follows:

Definition 2 (Circumnavigation Control Problem Based on Utilities).

In a heterogeneous multi-robot system composed of n ($n \geq 2$) mobile robots, each of the robot's dynamics are modelled by (1). Suppose $f_i : \mathbb{R}^{n+1} \rightarrow (0, 2\pi)$, $i = 1, \dots, n$, is a smooth function of time and the utilities of robots, which maps utilities to the final holistic formation spacing. Assume $\lim_{t \rightarrow \infty} f_i(t, \mu_1, \dots, \mu_n)$ exists, the circumnavigation control problem based on utilities is to seek control laws satisfying the following asymptotic conditions:

$$\lim_{t \rightarrow \infty} \rho_i(t) = \rho^* \quad (3)$$

$$\lim_{t \rightarrow \infty} \Delta_i(t) = \lim_{t \rightarrow \infty} f_i \quad (4)$$

$$\lim_{t \rightarrow \infty} \dot{\varphi}_i(t) = \omega^* \quad (5)$$

$$\lim_{t \rightarrow \infty} z_i(t) = z^* \quad (6)$$

Here, $\mu_i > 0$, $\rho^* > 0$, $\omega^* \in \mathbb{R}$ and $z^* \in \mathbb{R}$. ρ^* , ω^* and z^* denote the circumnavigation radius, the angular speed and the circumnavigation height respectively.

In this paper, it is required that all robots and the target remain in the same plane in the end. Therefore, the default value of z^* is 0. However, z^* can be different for different robots. Equation (4) manifests that the final formation spacing is not specified manually as proposed in [9] or [13], but instead, it is determined by the f_i function, which will be referred to as f function for simplicity. Note that f_i function depends on the utilities of other robots instead of calculating by each robot alone. The advantage of eliciting the f function is that the spacing

among robots can be dynamically adjusted corresponding to the variations of robots' utilities.

The expression of the f function is determined by a *formation guideline*. It is proposed under specific physics background representing the relationship between the utilities of robots and the final formation spacing. In this paper, we suppose that multiple heterogeneous robots circumnavigate a target and try to prevent it from fleeing. In Fig. 3, four robots r_1, \dots, r_4 rotate around a target denoted by O . Suppose that the target is intelligent enough to determine the *best fleeing points* denoted by A, B, C and D in the figure. Obviously, the best fleeing points are related to the utility (i.e., the maximum movement speeds) of robots. The position of A , for instance, is calculated by $\angle AOr_2 = \frac{\mu_2}{\mu_1 + \mu_2}$. We also suppose that the probability of capturing the target by a robot is inversely proportional to the time spent on moving from its initial position along the circular trajectory at its maximum speed to the best fleeing point. Therefore, the formation guideline can be defined as

Formation Guideline 1. *In the final circumnavigation formation formed by robots, when the target tries to escape via any of the best fleeing point, the two robots adjacent to the best fleeing point have the same probability of capturing the target.*

To understand the above formation guideline, taking Fig. 3 for example, it means the travelling time for r_1 and r_2 to arrive at the best fleeing point A along the circular trajectory at their maximum speeds (i.e., μ_1 and μ_2 resp.) is the same, or the travelling time for r_2 to arrive at A or B along the circular trajectory at its maximum speed (i.e., μ_2) is identical, and hence, the probability of capturing the target is equal. Following this, it can be derived that $\frac{\mu_i}{\mu_i + \mu_{i+}} \Delta_i = \frac{\mu_i}{\mu_i + \mu_{i-}} \Delta_{i-}$. According to this equation, the relationship between the final desired formation spacing and the utilities is $\Delta_1 : \Delta_2 : \dots : \Delta_n = (\mu_1 + \mu_2) : (\mu_2 + \mu_3) : \dots : (\mu_n + \mu_1)$. Therefore, given μ_1, \dots, μ_n , the formation spacing can be determined, and the f function is expressed as follows:

$$f_i(t, \mu_1, \dots, \mu_n) = \frac{\mu_i + \mu_{i+}}{\sum_{k=1}^n \mu_k} \pi. \quad (7)$$

Other formation guidelines can be similarly defined¹.

Remark 1. Note that formation guidelines only reflect the relationship between the utilities of robots and the final formation spacing; it does not determine the utilities of robots.

3 Utility-Based Circumnavigation Control Algorithm

First we define a rotational matrix R_b , which is the representation of the body reference frame \mathcal{B} with respect to the world reference frame \mathcal{W} . Therefore, the

¹ Another example of the formation guideline can be found in the full version <http://arxiv.org/abs/1805.05395>.

following equation calculates the cylindrical coordinates of r_i in the frame \mathcal{B} : $q_i = q(R_b^T(p_i - p_b))$, where p_i and p_b are the Cartesian coordinates of r_i and the target in the frame \mathcal{W} respectively. Then the derivative of q_i is the dynamics of robots in the cylindrical coordinates, which is $\dot{q}_i = J_i[\dot{R}_b^T(p_i - p_b) + R_b^T(\dot{p}_i - \dot{p}_b)]$, where J_i is the Jacobian matrix as shown in Sect. 2, i.e., $J_i = \frac{\partial q}{\partial p^T} \Big|_{p = R_b^T(p_i - p_b)}$. Note that $\det(J) = \frac{1}{\sqrt{p_x^2 + p_y^2}}$ as long as $p_x^2 + p_y^2 \neq 0$. This means J_i is invertible as long as the distance between r_i and the target is non-zero. This condition can always be guaranteed since the initial positions of the robots and the target do not coincide, and by designing appropriate control algorithms, the distance can be guaranteed to be non-zero all the time. By letting

$$u_i = \dot{p}_i = \dot{p}_b + R_b(J_i^{-1}v_i - \dot{R}_b^T(p_i - p_b)), \quad (8)$$

we can switch our focus to the new control input in the cylindrical coordinates $v_i = \dot{q}_i = (\dot{\rho}_i, \dot{\varphi}_i, \dot{z}_i)^T$ [1]. The advantage of transforming to this control input is that we can control ρ_i , φ_i and z_i separately, which are the three main variables in the circumnavigation problem.

Notations. For positive integers m and n , M_n and $M_{m \times n}$ are a set of all $n \times n$ and $m \times n$ real matrices. If all the entries in a matrix is nonnegative, this matrix is called nonnegative. We denote I_d as the $d \times d$ identity matrix. $\mathbf{1}$ and $\mathbf{0}$ are vectors of all 1's or 0's of suitable dimensions respectively. The underlying directed graph (or digraph) of a nonnegative matrix $M \in M_n$, denoted by $\mathcal{G}(M)$, is the directed graph with the vertex set $\{v_i\}, i \in \{1, \dots, n\}$, such that there is a directed edge in $\mathcal{G}(M)$ from v_j to v_i if and only if $m_{ij} \neq 0$. A directed graph is called strongly connected if for every pair of vertices, there is a directed path between them [4]. The following is a preliminary result related to any strongly connected digraph.

Lemma 1 (Theorem 3 of [6]). *Assume G is a strongly connected digraph with Laplacian L satisfying $Lw_r = \mathbf{0}$, $w_1^T L = \mathbf{0}$ and $w_1^T w_r = 1$. Then $R = \lim_{t \rightarrow \infty} \exp(-Lt) = w_r w_1^T \in M_n$.*

Theorem 1. *Consider a multi-robot system with robot dynamics described by (1) and (8), by introducing the control input $v_i = \dot{q}_i = (\dot{\rho}_i, \dot{\varphi}_i, \dot{z}_i)^T$ into (8), where*

$$\dot{\rho}_i = k_\rho(\rho^* - \rho_i), \quad (9)$$

$$\dot{z}_i = -k_z z_i, \quad (10)$$

$$\dot{\varphi}_i = \omega^* + k_\varphi(\bar{\varphi}_i - \varphi_i). \quad (11)$$

Note that k_ρ , k_z and k_φ are positive gains, and

$$\bar{\varphi}_i = \begin{cases} \varphi_{i-} + \frac{\mu_{i-} + \mu_i}{\mu_{i+} + 2\mu_i + \mu_{i-}}(\Delta_i + \Delta_{i-}), & i = 2, 3, \dots, n, \\ \varphi_{i-} + \frac{\mu_{i-} + \mu_i}{\mu_{i+} + 2\mu_i + \mu_{i-}}(\Delta_i + \Delta_{i-}) - 2\pi, & i = 1, \end{cases} \quad (12)$$

where μ_i is the utility of the robot r_i and it is piecewise constant. If the f function is shown as (7) (Formation Guideline 1), the circumnavigation control problem based on utilities encoded by (3), (4), (5) and (6) can be solved with exponential convergence speed.

Proof. It is obvious that (9) and (10) do not rely on the states of other robots, and they are basically P control laws with reference input ρ^* and 0 respectively. So according to the classical control theory, ρ_i and z_i will converge exponentially to ρ^* and 0 respectively.

Since $\mu_i, i = 1, \dots, n$, is piecewise constant, it is obvious that $\lim_{t \rightarrow \infty} f_i$ exists. We define $\bar{\varphi} = [\bar{\varphi}_1 \dots \bar{\varphi}_n]^T$ and $\varphi = [\varphi_1 \dots \varphi_n]^T$, so (11) and (12) can be written into compact forms respectively as follows:

$$\dot{\varphi} = \omega^* \mathbf{1} + k_\varphi(\bar{\varphi} - \varphi), \tag{13}$$

$$\bar{\varphi} = A\varphi + b, \tag{14}$$

$$A = \begin{bmatrix} 0 & \frac{\mu_n + \mu_1}{\mu_2 + 2\mu_1 + \mu_n} & 0 & \dots & 0 & 0 & \frac{\mu_1 + \mu_2}{\mu_2 + 2\mu_1 + \mu_n} \\ \frac{\mu_2 + \mu_3}{\mu_3 + 2\mu_2 + \mu_1} & 0 & \frac{\mu_1 + \mu_2}{\mu_3 + 2\mu_2 + \mu_1} & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\mu_{n-1} + \mu_n}{\mu_1 + 2\mu_n + \mu_{n-1}} & 0 & 0 & \dots & 0 & \frac{\mu_n + \mu_1}{\mu_1 + 2\mu_n + \mu_{n-1}} & 0 \end{bmatrix} \tag{15}$$

where $A \in M_n$ is shown as (15), and $b = 2\pi \left[\frac{-(\mu_1 + \mu_2)}{\mu_2 + 2\mu_1 + \mu_n} \ 0 \ \dots \ 0 \ \frac{\mu_{n-1} + \mu_n}{\mu_1 + 2\mu_n + \mu_{n-1}} \right]^T$.

During each time period where μ_i is constant, A and b are constant matrix and vector respectively. Note that matrix A is a row stochastic matrix and furthermore, it could be considered as the adjacency matrix [4] corresponding to a weighted directed ring denoted by $\mathcal{G}(A)$. It can be readily verified that $\mathcal{G}(A)$ is strongly connected. Next we define the *error signal* as

$$e_\varphi = \bar{\varphi} - \varphi = (A - I_n)\varphi + b = -L_p\varphi + b, \tag{16}$$

where $L_p = I_n - A$, which is the Laplacian matrix of $\mathcal{G}(A)$. Since L_p is constant at each time period, the derivative of e_φ is $\dot{e}_\varphi = -L_p\dot{\varphi}$. By substituting (13) and (16) into this equation, we further obtain the *error dynamics* as

$$\dot{e}_\varphi = -\omega^* L_p \mathbf{1} - k_\varphi L_p e_\varphi = -k_\varphi L_p e_\varphi. \tag{17}$$

Note that $\mathbf{1}$ is the right eigenvector associated with the zero eigenvalue of L_p , so $-\omega^* L_p \mathbf{1} = 0$. The solution to (17) is $e_\varphi(t) = \exp(-k_\varphi L_p t) e_\varphi(0)$. According to Lemma 1 and also note that $k_\varphi > 0$ only affects the convergence speed but not the convergence value, we have $\lim_{t \rightarrow \infty} e_\varphi(t) = w_r w_l^T e_\varphi(0)$, where $L_p w_r = \mathbf{0}$, $w_l^T L_p = \mathbf{0}$ and $w_l^T w_r = 1$. By substituting (16) into this equation, we obtain the following:

$$\lim_{t \rightarrow \infty} e_\varphi(t) = w_r (-w_l^T L_p \varphi + w_l^T b) = w_l^T b w_r. \tag{18}$$

Let $w_r = \mathbf{1}$ and $w_l = \frac{w_L}{\sum w_L}$, where the i^{th} entry of w_L is

$$\left[w_{L_i} = (\mu_{i^+} + 2\mu_i + \mu_{i^-}) \prod_{j=1, j \neq i, i^-}^n (\mu_j + \mu_{j^+}) \right],$$

and $\sum w_L = \sum_{i=1}^n w_{L_i}$. It can be easily verified that w_l^T and w_r are the left and right eigenvector of the Laplacian matrix L_p associated with the zero eigenvalue respectively, and $w_l^T w_r = 1$. Therefore, (18) becomes $\lim_{t \rightarrow \infty} e_\varphi(t) = \mathbf{0}$, or $\lim_{t \rightarrow \infty} \varphi(t) = \lim_{t \rightarrow \infty} \bar{\varphi}(t)$. According to (13), the circumnavigation speed of each robot converges to the desired angular speed ω^* . In addition, under this condition, $\bar{\varphi}_i$ is replaced by φ_i in (12) and therefore, for robots with indices $i = 2, \dots, n$, the equation $\varphi_i = \varphi_{i^-} + \frac{\mu_{i^-} + \mu_i}{\mu_{i^+} + 2\mu_i + \mu_{i^-}} (\Delta_i + \Delta_{i^-})$ further becomes $\frac{\Delta_i}{\Delta_{i^-}} = \frac{\mu_i + \mu_{i^+}}{\mu_i + \mu_{i^-}}$. This means a sequence of equations $\frac{\Delta_n}{\Delta_{n-1}} = \frac{\mu_n + \mu_1}{\mu_{n-1} + \mu_n}, \dots, \frac{\Delta_2}{\Delta_1} = \frac{\mu_2 + \mu_3}{\mu_1 + \mu_2}$. Assuming $\Delta_1 = k(\mu_1 + \mu_2)$, $k \neq 0$, we have $\Delta_i = k(\mu_i + \mu_{i^+})$, $i = 2, \dots, n$. According to $\sum_{i=1}^n \Delta_i = 2\pi$, it follows that $2k \sum_{i=1}^n \mu_i = 2\pi$, and hence $k = \pi / \sum_{i=1}^n \mu_i$. Therefore, $\Delta_i = (\mu_i + \mu_{i^+})\pi / \sum_{i=1}^n \mu_i = f_i(t, \mu_1, \dots, \mu_n)$. So the formation spacing expressed by (4) and (7) can be achieved.

Remark 2. Since (9), (10), (11) and (17) typically admit a linear system, the convergence is global and exponential. In fact, for the convergence of e_φ , a Lyapunov function can be defined as $V(e_\varphi) = e_\varphi^T P e_\varphi$, where $P = \text{diag}\{w_l\}$, so the global and exponential convergence can also be proved using the Lyapunov theorem.

Remark 3. In the definition of circumnavigation control problem based on utilities, (4) contains the utilities of all robots. However, it can be seen from (12) that each robot only needs to obtain the utilities of its two neighboring robots. Moreover, it should be noted that robots do not know what the holistic expected formation is; the actual formation (or spacing) among robots adapt dynamically to the variations of the local utilities of neighboring robots. In addition, when a robot joins or leaves the formation, according to (11) and (12), the spacing among robots will adjust dynamically through local update of the utilities of neighboring robots. To sum up, the utility-based circumnavigation control algorithm does not rely on the number of robots, and it is able to dynamically adjust the formation spacing dependent on the change of utilities.

Remark 4. When $\mu_\theta = 0$, the robot r_θ has quitted from the circumnavigation process, and therefore the communication topology has changed. The change of communication topology means the indices of the neighboring robots alter accordingly. When $\mu_2 = 0$, for example, the neighboring robots of r_3 change from r_2 and r_4 to r_1 and r_4 . In this way, the circumnavigation control algorithm based on utilities can well adapt to the cases where there are local variations on utilities or where robots join or quit from the formation. The formation spacing can adjust dynamically based on the selected formation guideline, achieving distributed formation reconfiguration.

Another problem that is worth considering is whether robots preserve their initial orders during the whole circumnavigation process. For the next theorem, the definition of a Metzler matrix [5] is given. For a real matrix $M = [m_{ij}] \in M_n$, if all its off-diagonal elements are non-negative, i.e., $m_{ij} \geq 0, i \neq j$, M is a Metzler matrix.

Theorem 2. *During the circumnavigation process, robots always keep their initial orders in the formation. In other words, $\Delta_i(t) > 0, i = 1, \dots, n$, for $t \geq 0$.*

Proof. According to (2), (11) and (12), for $i = 1, \dots, n$, it follows that

$$\begin{aligned} \dot{\Delta}_i = k_\varphi & \left[\frac{\mu_i + \mu_{i^+}}{\mu_{i^*} + 2\mu_{i^+} + \mu_i} \Delta_{i^+} - \left(\frac{\mu_{i^+} + \mu_{i^*}}{\mu_{i^*} + 2\mu_{i^+} + \mu_i} \right. \right. \\ & \left. \left. + \frac{\mu_{i^-} + \mu_i}{\mu_{i^+} + 2\mu_i + \mu_{i^-}} \right) \Delta_i + \frac{\mu_i + \mu_{i^+}}{\mu_{i^+} + 2\mu_i + \mu_{i^-}} \Delta_{i^-} \right], \end{aligned} \tag{19}$$

where i^* represents $(i^+)^+$, which is the index of the second adjacent robot for the robot r_i in the counter-clockwise direction. Let $\Delta = [\Delta_1 \dots \Delta_n]^T$, then (19) can be rewritten as $\dot{\Delta} = k_\varphi M_\Delta \Delta$, where M_Δ is shown in (20).

$$M_\Delta = \begin{bmatrix} \frac{-(\mu_2 + \mu_3)}{\mu_3 + 2\mu_2 + \mu_1} + \frac{-(\mu_n + \mu_1)}{\mu_2 + 2\mu_1 + \mu_n} & \frac{\mu_1 + \mu_2}{\mu_3 + 2\mu_2 + \mu_1} & \dots & \frac{\mu_1 + \mu_2}{\mu_2 + 2\mu_1 + \mu_n} \\ \frac{\mu_2 + \mu_3}{\mu_3 + 2\mu_2 + \mu_1} & \frac{-(\mu_3 + \mu_4)}{\mu_4 + 2\mu_3 + \mu_2} + \frac{-(\mu_1 + \mu_2)}{\mu_3 + 2\mu_2 + \mu_1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\mu_n + \mu_1}{\mu_2 + 2\mu_1 + \mu_n} & 0 & \dots & \frac{-(\mu_1 + \mu_2)}{\mu_2 + 2\mu_1 + \mu_n} + \frac{-(\mu_{n-1} + \mu_n)}{\mu_1 + 2\mu_n + \mu_{n-1}} \end{bmatrix}. \tag{20}$$

Therefore, the solution of $\Delta(t)$ is $\Delta(t) = \exp(k_\varphi M_\Delta t) \Delta(0)$. Since M_Δ is a Metzler matrix, it has been proved that $\exp(k_\varphi M_\Delta t)$ is a non-negative matrix. In addition, due to $\Delta(0) > 0$, it follows that $\Delta(t) > 0, t \geq 0$, which means that robots always keep their initial orders in the formation.

Remark 5. The significance of this theorem is that it provides a preliminary result for collision avoidance. In other words, if robots are treated as mass points, then collision will not happen since they always keep their initial orders. For real robots with geometric shape, given sufficiently large spacing, the collision will not happen, but this will need further investigation.

4 Experimental Results and Analysis

Although it is claimed that formation guidelines correspond to specific physics backgrounds, in the experiment, we do not try to reproduce the specific scenarios. This is because the emphasis here is the stability of the circumnavigation control algorithm based on utilities, and how the global formation spacing reacts dynamically to the variation of the utilities. In the experiments, robots' utilities are supposed to be proportional to its maximum movement speed. However, how the utilities are calculated from the maximum movement speeds is not the

interest of the study. Instead, the variation of the utilities are manually specified. Readers can think of an increase in the utilities as an update of robots' locomotion capabilities, while the decrease means the deterioration of performance due to worn-out or damage of robots².

4.1 Experiment with Soccer Robots

In this experiment, four soccer-playing robots [11, 12] are used and Formation Guideline 1 is adopted. Since the soccer-playing robots have omnidirectional movement abilities and they can reach any given velocity instantly, their dynamics can be regarded as the first-integrator model given in (1). In addition, an omnidirectional vision system is equipped on each robot with algorithms for self-localization and the recognition of a yellow football [2]. The position and velocity of the robot itself and the position and velocity of the football are obtained by its own omnidirectional vision system. Moreover, robots are only allowed to receive information from its neighboring robots and the information is transmitted using wireless communication.

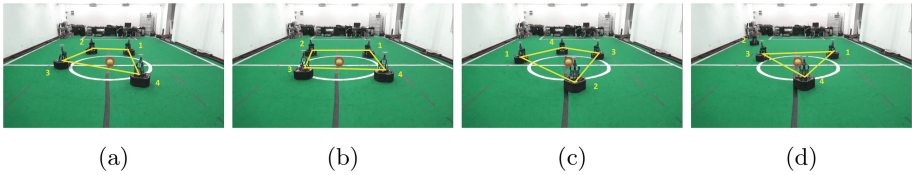


Fig. 4. The real robot experiment. (a)–(d) illustrate the positions of robots at 4 s (Stage 1), 19 s (Stage 2), 39 s (Stage 3) and 61 s (Stage 4) respectively. (Color figure online)

The utilities of robots r_1 , r_3 and r_4 remain 20 throughout the whole circumnavigation process, while the utility of the robot r_2 varies according to a piecewise constant function. That is, $\mu_2 = 1$, ($0 \leq t < 15$); $\mu_2 = 20$, ($15 \leq t < 30$); $\mu_2 = 50$, ($30 \leq t < 45$); $\mu_2 = 0$, ($t \geq 45$). For convenience, the four time ranges are denoted by Stage 1, 2, 3 and 4 respectively. Following Formation Guideline 1, it can be calculated the final expected spacing for the four stages is $[62 \ 62 \ 118 \ 118]^T$, $[90 \ 90 \ 90 \ 90]^T$, $[114 \ 114 \ 66 \ 66]^T$ and $[120 \ 120 \ 120]^T$ (unit: degree) correspondingly. Note that at Stage 4, the robot r_2 quits from the circumnavigation process as its utility becomes zero. In this experiment, robots' initial positions are randomly chosen. The experiment parameters are $\rho^* = 2$ m, $w^* = 0.5$ rad/s, $k_\varphi = 2.5$ and $k_\rho = 2$.

The circumnavigation process is shown in Fig. 4. It demonstrates the positions of robots at different stages. The yellow lines connecting each robot's center

² The simulation results are illustrated in the full version <http://arxiv.org/abs/1805.05395>.

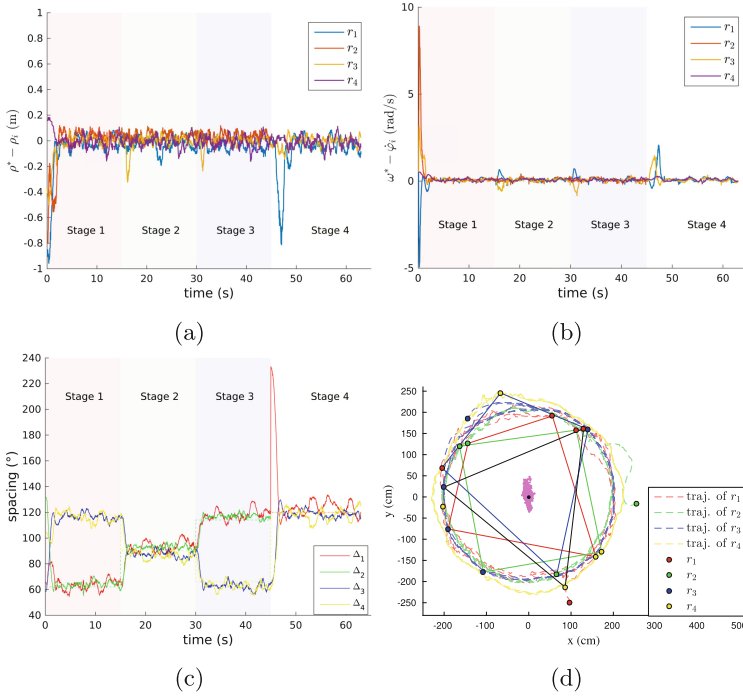


Fig. 5. The data plots of the real robot experiment. (Color figure online)

indicate the formation shape. The ball in the middle of the field is the target to be encircled, which is marked by a red circle. The corresponding data plots are shown in Fig. 5. Since robots move on the ground, the error plot of $-z$ is omitted. In Fig. 5d, the red, green, blue and black solid lines connecting the centres of robots represent the formation shapes at Stage 1, 2, 3 and 4 respectively. The dashed lines originated from robots are their trajectories. Note that since r_2 quits from the formation at Stage 4 ($\mu_2 = 0$), the data related to r_2 is not plotted after 45 s. The circumnavigation radii, angular speeds and formation spacing converge to but fluctuate around the desired values at each stage (see Fig. 5a, b and c respectively). Noticeably, at the last intersection (45 s), the circumnavigation radius and spacing for the robot r_1 deviate significantly from the desired values due to the absence of the robot r_2 in the formation, but the variations diminish rapidly subsequently (see Fig. 5a and c). The robot r_4 is hardly affected as it is not a neighboring robot of r_2 . In Fig. 5d, the black dot at the center is the real position of the target and the cluster of pink dots are the perceived positions of the target by r_1 . This manifests that information noise increases the uncertainty of the perceived information. Although there are fluctuations due to the information noise, the real spacing converges to the expected spacing at each stage (see Fig. 5c).

5 Concluding Remarks and Future Work

This paper proposes a distributed control law for a multi-robot system to realize circumnavigation process with dynamic spacing based on utilities. Unlike most of the existing study, in this paper, the spacing is not fixed and equal but they are dynamic, which is useful if robots are heterogeneous (e.g. with different kinematics capabilities). The theoretical analysis using graph theory along with the experiments prove the effectiveness of the proposed circumnavigation control algorithm based on utilities.

Although Theorem 2 implies that robots will not collide with each other since their orders are unchanged during the circumnavigation process, this claim is based on the assumption that robots are considered as mass points. The collision avoidance problem taking into account the physical dimensions of robots will be studied in the future.

Acknowledgements. Our work is supported by National Science Foundation of China (NO. 61503401 and NO. 61773393), and graduate school of National University of Defense Technology.

References

1. Franchi, A., Stegagno, P., Oriolo, G.: Decentralized multi-robot encirclement of a 3D target with guaranteed collision avoidance. *Auton. Robots* **40**(2), 245–265 (2016)
2. Lu, H., Yang, S., Zhang, H., Zheng, Z.: A robust omnidirectional vision sensor for soccer robots. *Mechatronics* **21**(2), 373–389 (2011)
3. Marshall, B.J.A., Broucke, M.E., Francis, B.A.: Formation of vehicles in cyclic pursuit. *IEEE Trans. Autom. Control* **49**(11), 1963–1974 (2015)
4. Mesbahi, M., Egerstedt, M.: *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, Princeton (2010)
5. Minc, H.: *Nonnegative Matrices*. Wiley, Hoboken (1988)
6. Olfati-Saber, R., Murray, R.M.: Consensus problems in networks of agents with switching topology and time-delays. *IEEE Trans. Autom. Control* **49**(9), 1520–1533 (2004)
7. Pavone, M., Frazzoli, E.: Decentralized policies for geometric pattern formation and path coverage. *J. Dyn. Syst. Meas. Control* **129**(5), 633–643 (2007)
8. Tang, S., Shinzaki, D., Lowe, C.G., Clark, C.M.: Multi-robot control for circumnavigation of particle distributions. In: Ani Hsieh, M., Chirikjian, G. (eds.) *Distributed Autonomous Robotic Systems*. STAR, vol. 104, pp. 149–162. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-55146-8_11
9. Wang, C., Xie, G., Cao, M.: Forming circle formations of anonymous mobile agents with order preservation. *IEEE Trans. Autom. Control* **58**(12), 3248–3254 (2013)
10. Wang, C., Xie, G., Cao, M.: Controlling anonymous mobile agents with unidirectional locomotion to form formations on a circle. *Automatica* **50**(4), 1100–1108 (2014)
11. Xiong, D., et al.: The design of an intelligent soccer-playing robot. *Ind. Robot* **43**, 91–102 (2016)

12. Yao, W., Dai, W., Xiao, J., Lu, H., Zheng, Z.: A simulation system based on ROS and Gazebo for RoboCup middle size league. In: 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 54–59. IEEE (2015)
13. Yao, W., Zeng, Z., Wang, X., Lu, H., Zheng, Z.: Distributed encirclement control with arbitrary spacing for multiple anonymous mobile robots. In: 2017 Chinese Control Conference (CCC), Dalian, China, July 2017
14. Yu, X., Liu, L.: Distributed circular formation control of ring-networked nonholonomic vehicles. *Automatica* **68**, 92–99 (2016)