

Semi-centralized control for multi-robot formation and theoretical lower bound

Shuo Wan, Jiaxun Lu, Pingyi Fan, *Senior Member, IEEE* and Khaled B. Letaief*, *Fellow, IEEE*

Tsinghua National Laboratory for Information Science and Technology(TNList),

Department of Electronic Engineering, Tsinghua University, Beijing, P.R. China

E-mail: wan-s17@mails.tsinghua.edu.cn, luxj14@mails.tsinghua.edu.cn, fpy@tsinghua.edu.cn

*Department of Electronic Engineering, Hong Kong University of Science and Technology, Hong Kong

Email: eekhaled@ece.ust.hk

Abstract—Multi-robot formation control enables robots to cooperate as a working group in completing complex tasks, which has been widely used in both civilian and military scenarios. Before moving to reach a given formation, each robot should choose a position from the formation so that the whole system cost is minimized. To solve the problem, we formulate an optimization problem in terms of the total moving distance and give a solution by the Hungarian method. To analyze the deviation of the achieved formation from the ideal one, we obtain the lower bound of formation bias with respect to system’s parameters based on notions in information theory. As an extension, we discuss methods of transformation between different formations. Some theoretical results are obtained to give a guidance of the system design.

Index Terms—position arrangement, formation bias, leader-follower, formation conversion

I. INTRODUCTION

RECENT years, multi-robot moving in a formation are replacing complex single robots for both civilian and military applications. The multi-robot system has many advantages. By simply replacing broken robots, the overall system performance will not be degraded largely. In this way, it can reduce the complexity by decomposing complex tasks into small ones [1]. Furthermore, multi-robot system can complete more complex tasks with a lower cost. In military applications, robots can cooperate as a working group to complete surveillance tasks [2] or do spying work in adversarial areas based on techniques in [3][4][5]. In civilian applications, automatically driving car system may be helpful in the intelligent transportation systems.

Formation control is a critical problem for robots to coordinate. To complete a certain task, we often need robots to move by maintaining a sequential formations. For example, when cooperatively transporting a large item, certain formations are required, which enable them to complete the task [6]. In this process, the formation should be reached as fast as possible to improve the working efficiency. Furthermore, the deviation of achieved formation from the exactly required one should be constrained in a reasonable range to ensure the task’s completion. In practice, we also need to change formation according to different tasks and the process should be fast as well.

Multi-robot formation usually requires several basic conditions or assumptions. Here, it assumes that: 1) Robots are able

to set relative positioning of others around them by using the assistant essential information to guide themselves [7][8]; 2) Robots are able to control themselves to reach their desired places; 3) The basic communication capability is necessary, so that robots are able to cooperate with each other [9][10].

The multi-robot formation controlling approaches are mainly divided into three classes: decentralized control, centralized control and the combination of centralization and decentralization, refer to as semi-centralized control. In centralized control, there is a central agent monitoring the whole environment and controlling all robots’ motion to reach a formation. The agent can be either a computer or a robot. In decentralized control system, there is no central agent. Each robot makes their own decision. They monitor the environment by themselves and exchange their observations with others by communication. In the semi-centralized control method, there exists a central agent distributing tasks and sending commands to robots and each robot reaches its goal by decisions of their own.

To cope with different application scenarios, researchers presented several different methods based on different assumptions and requirements. In [11], it discussed the problem of planning collision-free paths for permutation-invariant multi-robot formations. In [12], it presented a formation controller for a scalable team of robots where communications are unavailable and sensor ranges are limited. Robots can reach a locally desired formation and change gradually to a globally desired one. In [13], it proposed a method of distributed estimation and control for preserving formation rigidity of mobile robot teams, which is a kind of virtual structure approach. In [14][15], they proposed a leader-follower controller for multi-robot teams. In [16], it developed a suboptimal controller for a leader-follower formation problem of quadrotors with the consideration of external disturbances. In [17], it combined centralization and decentralization methods and proposed a coordination architecture for spacecraft formation control. In [18], it presented an approach for formation control of autonomous vehicles traversing along a multi-lane road with obstacles and a certain degree of traffic.

However, to the best knowledge of us, the previous works didn’t discuss how to arrange each robot with a position in the formation. Instead, this is usually requirement from applications. Furthermore, they also didn’t present theoretical analysis of the achieved formation compared to the exactly required one.

To solve the problem of arranging robots in a formation, we transform the problem to a task distribution problem and solve it with the Hungarian method. Then we define the formation bias. To estimate the bias, we apply notions in information theory. We do an estimation of the Bayes risk by calculating the mutual information between measured value and true value to solve the problem.

In this paper, we focus on leader-follower method which is more similar to human's action mode when standing in a formation. Each robot has a reference and follows it until the robot arrives at the right position. Based on this, we first propose a method to obtain an optimal solution for arranging robots to positions in a given formation by minimizing the predicted total moving distance. Then we also consider the achieved formation's bias to the ideal one. Applying methods in decentralized estimation, we obtain a lower bound of the formation bias with respect to system parameters. This can guide us to design the system according to user's requirements. Finally, we discuss the transformation between different formations. We optimize the choice of new formation center and confirm it by simulation.

Our discussed system belongs to the third class. It has a central agent to send commands to robots. Each robot receives the commands and reaches its goal by itself. In the process, they detect possible collisions and avoid them. Each robot can measure the relative position of its reference and follow it to reach the formation.

The rest paper is organized as follows: A brief problem and notation statement are given in Section II. In Section III, we give method to arrange robots to positions in the formations by transforming it into a distribution optimization problem. In Section IV, we give a depiction of the leader-follower method in our considered system. A lower bound of the formation bias is derived in Section V. The proof is given in Section VI. Afterwards, we discuss formation conversion in Section VII. Some simulation results are presented in Section VIII. Finally, we give the conclusion.

II. PROBLEM STATEMENTS AND RELEVANT NOTATIONS

Consider a group of robots, each has a serial number ranging from 1 to n . Their initial positions $\{x_i(0)|i = 1, 2, \dots, n\}$ are randomly distributed. Their motion can be described as follows:

$$x_i(k+1) = x_i(k) + v_i(k) \quad (1)$$

where $x_i(k) \in R^2$ is the position of robot i at time slot k with $i \in \{1, 2, \dots, n\}$ and $\|v_i(k)\| \leq U_{\max}$. U_{\max} is the maximum of velocities. $v_i(k)$ is the speed of robot i . Given a formation, the robots need to move to reach it, during which there should not have any collision.

In the real world, there are many different demands for robot formation configuration. As for the square formation, we tend to find a leading robot and let other robots follow it to reach a formation. Sometimes, we want the center of the formation to be at a certain position. Thus, we may let robots keep up with the center to move. The center can be given previously by the planner or it can be decided by the system automatically.

The formation is represented by $F = \{f_1, f_2, \dots, f_n\}$, where f_i is the position of the i -th node in the formation. Considering the formation on 2D plane, we have $f_i \in R^2$. Besides, we also let origin to be the center of the formation, which means $\sum_{i=1}^n f_i = 0$. The formation F is predefined by users, which is used by the system as reference to produce controlling forces for reaching the formation.

Having those positions in the formation, the robots have to be arranged according to them. The arrangement is denoted by $D = \{d_1, d_2, \dots, d_n\}$, where $d_i \in \{1, 2, \dots, n\}$. d_i represents the serial number of the i -th robot's position in the formation.

When configuring a square formation, we assume there exists a leading robot which is chosen by the controlling center. Each robot keeps up with another robot according to the arrangement mentioned above, while some robots will follow the leading robot directly. The leading robot's serial number is denoted as i_{lead} . Besides, we define $H = \{h_1, h_2, \dots, h_n\}$, where $h_i \in \{1, 2, \dots, n\}$ represents the serial number of the robot followed by robot i . In addition, we also define $P = \{p_1, p_2, \dots, p_n\}$, where $p_i \in R^2$ represents the required relative position in the formation between robot i and the robot it's following. p_i can obviously be calculated by $p_i = f_{d_{h_i}} - f_{d_i}$. Assuming that robot i can measure the relative position of robot h_i to itself, the robot can move to adjust the relative position. If for all $i \neq i_{lead}$, there is $x_i(k) - x_{h_i}(k) = p_i$, the formation is reached at time slot k . Besides, if the robots collide with another, they may break down. So our controlling method should avoid such event happen.

When requiring the formation to be around a center, we may not require to find a leader. Assuming robots can know the relative position between the center and themselves, they can move to adjust the relative position to achieve the formation. Denoting the position of the center to be C_x , if at time slot k , there are $x_i(k) - C_x = f_i$ for all $i \in \{1, 2, \dots, n\}$, the formation is reached at time slot k .

The movement of the robots should be synchronized so that equation (1) can be meaningful to describe the robots' motion [19]. Assuming that the center or the leading robot can act as the synchronizer, we can achieve this. Besides, the speed should be adjusted from $v_i(k-1)$ to $v_i(k)$ at time slot k . We assume this can be achieved by a motion controller on the robot. As mentioned before, the controlling should be based on the ability of measuring the relative position of other robots [20]. We assume the on board sensor and communication system will help to achieve this.

III. OPTIMIZATION OF FORMATION MAPPING

As mentioned, the formation can be described by $F = \{f_1, f_2, \dots, f_n\}$. Besides, the initial positions of robots are randomly deployed as $X(0) = \{x_1(0), x_2(0), \dots, x_n(0)\}$. Given $X(0)$, we can optimise the arrangement of the robots to the position in the formation described by $D = \{d_1, d_2, \dots, d_n\}$. The aim of the optimization is to minimize the robots' total moving distance.

Assuming there is a center controller, it will collect the robots' initial position $X(0)$ and calculate the arrangement D

according to $X(0)$ and formation F . Then it will broadcast the arrangement to robots.

A. Cost Function

As previously stated, searching for the optimal arrangement D requires minimizing the total moving distance. Moreover, if all the robots choose a position ensuring the shortest path, the total consuming time will be smaller and the moving process seems more reasonable. Before the optimization, we should set up a cost function to describe the moving distance.

To get the cost function, one can estimate the distance between the initial position and the final position in the formation. However, in real system, the moving distance is actually very complex to describe, as the robots may get away the calculated path in the planning stage to avoid other robots or obstacles. For example, when it comes to square formation, each robot follows another one. If it is not following the leader, its destination will keep moving. So the actual path may not be straight, which will result in an extra length compared to the estimated distance. In this way, the real moving distance may be very difficult to be represented. As an alternative, we use the estimated distance to get a near optimal result. We assume that the robots are not very dense, and the collision does not frequently happen. Besides, considering the actual curve path will make the problem much more complex, which is not worth to do so. Therefore using the estimated distance may be a good selection in the initial planning stage.

Choosing the leader, others will move to the right position relative to it. Besides, $f_{d_{i_{lead}}}$ should also be the leading position in the square formation. Given the leading robot's position $x_{i_{lead}}$ and assuming that the formation is reached at time slot k , the i -th robot's current position is

$$x_i(k) = x_{i_{lead}} - f_{d_{i_{lead}}} + f_{d_i}$$

Therefore the estimated moving distance of robot i should be

$$C_i = \|x_i(0) - x_i(k)\| = \|(x_i(0) - f_{d_i}) - (x_{i_{lead}} - f_{d_{i_{lead}}})\|$$

So we can get the cost function for robot i .

$$C_t(i) = C_i^2 = \|(x_i(0) - f_{d_i}) - (x_{i_{lead}} - f_{d_{i_{lead}}})\|^2 \quad (2)$$

When it requires to reach a formation around a center, it is not need to choose a leading robot. Given a center with position C_x , each robot may move according to C_x . Assuming that collision does not happen frequently, the extra path length caused by avoiding possible collisions can be ignored first. Besides, each robot can move according to the fixed center, which means that without considering the few collisions, the path is straight with high possibility. In this way, if the formation is reached at time slot k , the right position for robot i will be

$$x_i(k) = C_x + f_{d_i}$$

Therefore the estimated moving distance of robot i is

$$C_i = \|x_i(0) - x_i(k)\| = \|(x_i(0) - f_{d_i}) - C_x\|$$

The corresponding cost function for robot i is defined as

$$C_t(i) = C_i^2 = \|(x_i(0) - f_{d_i}) - C_x\|^2 \quad (3)$$

B. Optimization problem and Hungarian method

1) *Problem Statement:* Considering the square formation with a leader, the optimization problem can be formulated as follows by using cost function (2).

$$D = \operatorname{argmin}_D \sum_{i=0, i \neq i_{lead}}^n \|(x_i(0) - f_{d_i}) - (x_{i_{lead}} - f_{d_{i_{lead}}})\|^2 \quad (4)$$

When it comes to reaching a formation around a center, the cost function is updated by equation (3) and the optimization problem is formulated as

$$D = \operatorname{argmin}_D \sum_{i=0}^n \|(x_i(0) - f_{d_i}) - C_x\|^2 \quad (5)$$

In fact, with semi-centralized control mode, the formation can be divided into the two scenarios above along the time. In the first phase, all robots are far from its required positions. So selecting a leader may be a good way to move close to the required positions. Then the second phase is triggered, one can select the mode where the center of formation is known. Now, we first introduce the Hungarian Method by Kuhn in [21][22].

2) *Review of Hungarian Method:* Considering a general assignment problem, there is an n by n cost matrix $A = [a_{ij}]$, where $a_{i,j}$ satisfies $a_{ij} \geq 0$. The aim is to find a set j_1, j_2, \dots, j_n , which is a permutation of $1, 2, \dots, n$, so that the sum $C_t = r_{1,j_1} + r_{2,j_2} + \dots + r_{n,j_n}$ is minimized.

As for matrix A , if there are zero elements existing both in different rows and columns, one can get the assignments easily, and the C_t mentioned above is obviously zero. However, A doesn't necessarily contain enough such zero elements. So the matrix have to be transformed by the Hungarian Method. Two basic theorems are introduced here, refer to Lemma 1 and 2, respectively.

Lemma 1. *Koning's theorem [23]: In any bipartite graph, the number of edges in a maximum matching equals the number of vertices in a minimum vertex cover.*

Remark 1. *In Lemma 1, bipartite graph refers to those whose vertices can be partitioned into two sets such that each edge has one endpoint in each set. A vertex cover in a graph is a set of vertices that includes at least one endpoint of every edge, and a vertex cover is minimum if no other vertex cover has fewer vertices. A matching is a set of edges none of which share an endpoint. If no other matching has more edges, a matching is maximum [24]. Based on Lemma 1, there is a conclusion that the minimum number of rows or columns needed to contain matrix A 's all zero elements is equal to the maximum number of zeros that can be chosen, in which none of two zero elements is on the same line.*

Lemma 2. *the distribution problem is unchanged if the matrix A is replaced by $A' = (a'_{ij})$, with $a'_{ij} = a_{ij} - u_i - v_j$ for constants u_i and v_j , and $i, j = 1, 2, \dots, n$ [22].*

With Lemma 2, one can transform matrix A to obtain some zero elements. With Lemma 1, one can find the independent zero elements and minimize C_t . If there are not enough zero elements, one can continue to transform A with Lemma 2 until success. The detailed procedure can refer to [21][22].

3) *Problem Transformation and Solution*: The formation mapping problem can be transformed to the assignment problem and solved by the Hungarian Method.

As for the optimization problem (4), given robot $i \neq i_{lead}$, its assigned position's serial number is d_i . Let $d_i = j$, we can have the following equation:

$$a_{ij} = \|(x_i(0) - f_{d_i}) - (x_{i_{lead}} - f_{d_{i_{lead}}})\|^2 \quad (6)$$

where a_{ij} is an element in the cost matrix A . It represents the cost of arranging robot i to the position with serial number j . In the considered system, the leading robot's position should be the explicit reference marker in the square formation. Therefore $d_{i_{lead}}$ can be fixed. However, to calculate a_{ij} , we should select the leading robot i_{lead} . The value of a_{ij} is then relevant to i_{lead} , which means we can calculate a cost matrix A given an i_{lead} . In this case, A is $(n-1)$ by $(n-1)$, for the leader is excluded. With A , we can use the Hungarian Method to get an optimal arrangement conditionally for a given i_{lead} . If we let each robot to be the leader and separately calculate the result, an optimal arrangement can be finally acquired by comparison of the conditional minimum cost results.

Note that in the optimization problem (5), given $d_i = j$ for robot i , the cost matrix A is calculated by the following equation:

$$a_{ij} = \|(x_i(0) - f_{d_i}) - C_x\|^2 \quad (7)$$

That is, given C_x , the cost matrix A can be acquired. Then the optimal arrangement can be directly achieved by the Hungarian Method. The result is correlated with C_x . In the considered system, C_x is usually predetermined.

IV. MOVING STRATEGIES

Given n robots' initial position $x(0)$, formation F and calculated formation mapping D , the robots can move to reach the formation. The moving strategy should not only achieve the goal, but also prevent possible collisions.

A good approach to solve the problem is to imitate human. When people need to stand in a formation, they tend to find a reference and move to the exactly desired position relative to it. During the process, if two or more people approach the same position, they may slow down or deviate to avoid possible collisions. The reference can be selected as the leader or other people beside them in the formation. When people want to reach a formation around a center, they can also choose the center as the reference. To conclude, this is actually a leader-follower mode. As for robots, they can also follow such mode. Therefore a polar partitioning model [14][15] can be applied and extended to our scenario.

Fig. 1 represents the coordinate system. The follower can measure the leader's position. Together with the relative position in the formation, it can calculate a currently desired destination, which is the origin of the polar coordinates. The follower can measure its destination's position and locate itself in the polar coordinates.

Considering a follower located in circle R_m with the radius R , the center of R_m is the follower's current destination. Fig. 2 shows that R_m is divided by curves

$$\left\{ r = r_h | r_h = \frac{R}{n_r - 1} (h - 1), h = 1, 2, \dots, n_r \right\} \quad (8)$$

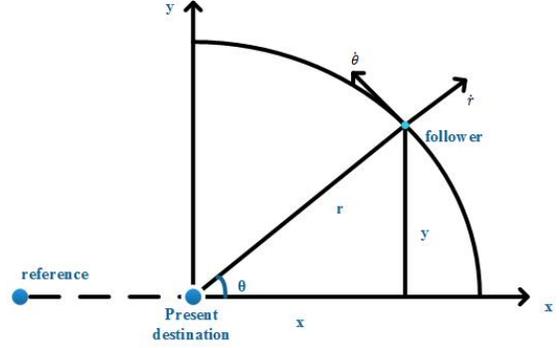


Fig. 1. leader follower model and Circle coordinate system

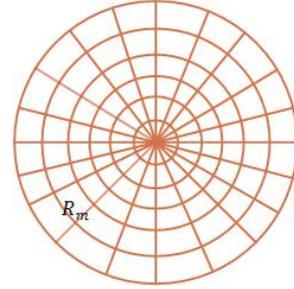


Fig. 2. A partitioned circle whose center is the present destination

$$\left\{ \theta = \theta_j | \theta_j = \frac{2\pi}{n_\theta - 1} (j - 1), j = 1, 2, \dots, n_\theta \right\} \quad (9)$$

A region is denoted by

$$R_{hj} = \{x = (r, \theta) | r_h \leq r \leq r_{h+1}, \theta(j) \leq \theta \leq \theta(j+1)\} \quad (10)$$

whose boundaries are given in (8) and (9).

Supposing a follower is denoted by robot i , at time slot k , let $x_i(k) \in R_{hj}$. The motion of robot i is described by (1). Then the moving strategies should be

$$x_i(k+1) \in R_{(h-1)j} \quad (h \neq 1) \quad (11)$$

$$v_i(k) = 0 \quad (h = 1) \quad (12)$$

However, if it detects a possible collision in $R_{(h-1)j}$, it will choose to move to $R_{h(j+1)}$ at time slot $(k+1)$. If $R_{h(j+1)}$ is also possible for collision, it will choose $R_{h(j-1)}$. If no possible moving is safe, it will stop for a time slot. When robot i detects possible collision with robot j and $i > j$, only robot i will deviate from the path and robot j will not be affected. In this way, it can improve the system's efficiency intuitively. Therefore the formation can be reached if each robot follows this procedure to find its destination.

V. THEORETICAL ANALYSIS OF FORMATION BIAS

Based on the procedure described previously, the formation can be finally reached. However, the measurement of position may not be accurate. Besides, the moving strategies need to partition the space, which will cause inaccuracy. So robots' final position may have a bias compared to the exactly required one, which causes a formation bias.

In this section, we provide a theoretical analysis of the formation's bias. We first introduce the notion of the formation bias and analyze possible issues causing it. Then we give a lower bound of it, which is related to the system's parameters. Finally we use simulations to check the theoretical results.

A. Formation bias definition

In the system, each robot should measure the position of its destination, including relative distance r and relative angle θ . Then the robot need to quantify r and θ to complete the space partition. The measurement of r and θ may bring some bias. Besides, the quantification will also bring bias to the final formation.

Assuming that robot can move from one region to another accurately, the bias mainly results from measurement and quantification. Besides, the angle θ represents the direction from which the robot approaches its destination. This does not contribute to the final formation bias. So we mainly consider bias caused by r here.

Supposing the formation is reached at time slot k , the robots' destinations are denoted by $DS = \{ds_1, ds_2, \dots, ds_n\}$. For robot i , its position bias $Er(i)$ is

$$Er(i) = |x_i(k) - ds_i| \quad (13)$$

Therefore the formation bias Er can be defined as

$$Er = \frac{1}{n} \sum_{i=1}^n Er(i) \quad (14)$$

which represents the mean deviation of robots. The bias is mainly from the measurement of distance r . We will analyze the bias of r to give an estimation of Er .

B. Problem Modeling

A model of decentralized estimation with single processor is first considered. In this process, the estimator can't gain direct access to the parameter of interest. It can receive the measurement from a local sensor. The sensor can make several measurements about the parameter. After acquiring the samples, it will quantize them and send them to the estimator for analysis. Finally, it makes an estimation about the parameter.

The estimation performance can be evaluated by the distortion, which is represented by a function of the parameter value and its estimated value. The minimum possible distortion is defined as Bayes risk. In [25], it gives the lower bounds of the Bayes risk for the estimator problems.

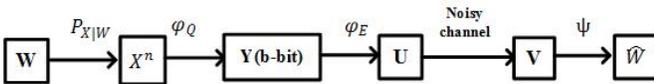


Fig. 3. The model of decentralized estimation with single processor

Fig. 3 shows the decentralized estimation system. W is the parameter of interest, which is measured by local processor. W is derived from a prior distribution P_W . Given $W = w$, the sensor can get a sample X generated from the

distribution $P_{X|W=w}$. Typically, the sensor gets n samples independently according to the distribution $P_{X^n|W=w}$, where $X^n = \{X_1, X_2, \dots, X_n\}$. After that, the system uses quantification function φ_Q to map the observed message X^n into a b -bit message

$$Y = \varphi_Q(X^n) \quad (15)$$

Then the encoder uses the coding function φ_E to transform Y into codeword

$$U = \varphi_E(Y) \quad (16)$$

U is transmitted through a noisy channel and received as V . Finally, the estimator calculates

$$\hat{W} = \psi(V) \quad (17)$$

as the estimates of W . Acquiring this estimation model, we can use it to estimate our formation bias. Before this, we should fit our problem to this model.

- * W is defined as the robot's distance variable from its present destination and w is its value. We consider robots in the circle R_m . Assuming w satisfies $w \in [0, l_0]$ in practical system, we have $l_0 < R$.
- * X^n are n independently drawn samples of W . The sensor on the robot measures the distance r for n times. The n independent measures are denoted as $\{x_1, x_2, \dots, x_n\}$.
- * After acquiring the measurement, the system will use curves (8) to divide R and locate the measured distance into one of the regions. This is a process of quantification. The number of divisions n_r represents the quantization precision. The result is b -bit length Y in the system.
- * In our system, the controller can gain direct access to Y . There is not process through the noisy channel. So we don't need to make the channel coding and transmit it to the receiver. Therefore we have $V = U = Y$.

Before the analysis, the relative distributions and equations should also be defined.

P_W is a prior distribution of W . In this system, W represents the distance r . As mentioned above, robot i is in circle R_m , whose center is the robot's present destination. As mentioned above, we assume $W \in [0, l_0]$. l_0 is an empirical constant of the system. To ensure the quantification being effective, there is $R > l_0$. To calculate P_W , we assume that robot i has an equal possibility to be located at each point in the circle with the radius l_0 . Therefore

$$P(W < r) = \frac{S_r}{S_{l_0}} = \frac{\pi r^2}{\pi l_0^2} = \frac{r^2}{l_0^2} \quad (18)$$

where $P(W < r)$ represents the probability that the robot's distance to its destination is smaller than r . S_r and S_{l_0} respectively represents the acreage of the circle with the radius r and l_0 .

Acquiring the distribution function, we need to calculate the density function.

$$P(W = r) = \frac{dP(W < r)}{dr} = \frac{2r}{l_0^2} \quad (19)$$

$P_{X|W}$ represents the distribution of measurement X conditioned on W . Given $W = w$, the measurement random

variable X is generated according to the distribution $P_{X|W=w}$. In this system we choose Gaussian distribution as $P_{X|W=w}$.

$$P_{X|W=w} = \frac{1}{\sqrt{2\pi\sigma^2}} \times e^{-\frac{(x-w)^2}{2\sigma^2}} \quad (20)$$

where the real value of the parameter is w , which is also the mean value of the Gaussian distribution. σ is the measurement variance.

In the considered system, it assumes that the sensor measures the distance for n times independently. The mean and variance value are with the same w and σ . In this case, a joint Gaussian distribution is applied here to represent $P_{X^n|W=w}$.

$$P_{X^n|W=w} = \frac{1}{(\sqrt{2\pi\sigma^2})^n} \times e^{-\frac{\sum_1^n (x_i-w)^2}{2\sigma^2}} \quad (21)$$

The measured distance is not directly used by the controller. It has to be quantified first. The distance r satisfies $r \in [0, l_0]$ and there is $l_0 < R$, then $[0, R]$ is the quantitative range. The range is divided into n_r regions, so the quantization bit b satisfies

$$b = \log_2(n_r) \quad (22)$$

In this way, the distributed estimation system is applied to the formation bias estimation problem. Then we can use theories in distributed estimation to give a lower bound on the Bayes risk of the estimation. This will guide us to choose the system's parameter.

C. Lower Bounds Estimation

As shown in Fig. 3, the parameter of interest is W , which is derived from a prior distribution P_W . After sampling and quantification, the parameter is estimated as $\hat{W} = \psi(V)$. By using a non-negative distortion function $l : W \times W \rightarrow R^+$, we can define the Bayes risk of estimating \hat{W} as

$$R_B = \inf_{\psi} E(l(W, \psi(V))) \quad (23)$$

where the distortion function l is of the form $l(w, \hat{w}) = \|w - \hat{w}\|^q$ and $q \geq 1$. In this paper, W represents the distance r . Therefore function l is defined on R^1 . In [25], it provides lower bounds on the Bayes risk for estimation. Then based on the above model and distributions, we can obtain a lower bound of R_B and derive the formation's bias.

Theorem 1. *In a multi-robot formation control system, we measure the distance n times with a measuring variance σ . The radius of controlling circle R_m is R and the quantification rate is set as b . If the measured distance satisfies $r \in [0, l_0]$, we can derive the lower bound of R_B :*

$$R_B \geq \frac{l_0}{2e} \max \left\{ 2^{-(\log \frac{nl_0}{2\sigma} + \frac{1}{2} - \frac{1}{2} \log(2\pi e))}, 2^{-\left(1 - e^{-\frac{nl_0^2}{2\sigma^2}}\right)b} \right\} \quad (24)$$

where $l(w, \hat{w}) = \|w - \hat{w}\|$.

Remark 2. *The above R_B describes measuring bias of the distance. This bias describes the robot's deviation from its right position in the formation. The whole formation bias is*

the mean value of each robot's deviation as defined in (14). Therefore the lower bound of R_B can be viewed as the lower bound of formation bias.

VI. PROOF OF LOWER BOUND

In this section, we give proof of the lower bound in Theorem 1. In [25], the author gives theorems to estimate lower bound of Bayes risk. We introduce the theorems as lemmas to analyze the formation bias.

Lemma 3. *For any arbitrary norm $\|\cdot\|$ and any $q \geq 1$, when the parameter of interest $W \in R^d$ and W is distributed in $[0, 1]$. The Bayes for estimating the parameter based on the sample X with respect to the distortion function $l(w, \hat{w}) = \|w - \hat{w}\|^q$ satisfies*

$$R_B \geq \sup_{P_{U|W,X}} \frac{d}{qe} \left(V_d \Gamma \left(1 + \frac{d}{q} \right) \right)^{-\frac{q}{d}} 2^{-(I(W;X|U) - h(W|U)) \frac{q}{d}} \quad (25)$$

To make problems simple, when we are estimating a real-valued W with respect to $l(w, \hat{w}) = \|w - \hat{w}\|$,

$$R_B \geq \sup_{P_{U|W,X}} \frac{1}{2e} 2^{-(I(W;X|U) - h(W|U))} \quad (26)$$

Considering the unconditional version, there's a simpler form

$$R_B \geq \frac{1}{2e} 2^{-I(W;X)} \quad (27)$$

In fact, (27) is an unconditional version of lemma 3, which can give us a lower bound of R_B . This lower bound is for the case $W \in [0, 1]$. If the distribution range is changed, we can multiply a parameter to correct it.

In the single processor estimating system displayed by Fig. 3, the estimator can't gain direct access to the measurement X^n . X^n is quantified and transmitted to the estimator. In fact, the estimation is based on V . So we can use unconditional version of lemma 3 to obtain a lower bound of Bayes risk R_B by replacing $I(W; X)$ with $I(W; V)$. Now, we need to calculate an upper bound of $I(W; V)$.

Lemma 4. *In decentralized estimation with a single processor, for any choice of ϕ_Q and ϕ_E , there is*

$$I(W, V) \leq \min \{ I(W, X^n) \eta_T, \eta(P_{X^n}, P_{W|X^n})(H(X^n) \wedge b) \eta_T, \eta(P_{X^n}, P_{W|X^n}) C T \} \quad (28)$$

where C is the Shannon Capacity of the noisy channel $P_{V|U}$, and

$$\eta_T = \begin{cases} 1 - (1 - \eta(P_{V|U}))^T & \text{with feedback} \\ \eta(P_{V|U}^{\otimes T}) & \text{without feedback} \end{cases} \quad (29)$$

In (29), T is the time spent transmitting a message Y through the channel $P_{V|U}$, and η is a constant.

On the calculation of η , it can be described as follows. Given a channel K whose input alphabet is X and output alphabet is Y . There is a reference input distribution μ on X . If for a constant $c \in [0, 1)$ and any other input distribution ν on X , there is $D(\nu K \| \mu K) \leq c D(\nu \| \mu)$, we say K satisfies an SDPI at μ .

The SDPI constant of K at input distribution μ is defined as

$$\eta(\mu, K) = \sup_{\nu: \nu \neq \mu} \frac{D(\nu K || \mu K)}{D(\nu || \mu)} \quad (30)$$

The SDPI constant of K is defined as

$$\eta(K) = \sup_{\mu} \eta(\mu, K) \quad (31)$$

With Lemma 3, one can obtain a lower bound of Bayes risk R_B relative to the mutual information $I(W, V)$. With Lemma 4, one can calculate an upper bound of $I(W, V)$ in our formation bias estimating system based on the model in section V.B, and then derive the result of Lemma 3.

In our considered formation bias estimation model, the quantified result Y is directly used for estimation. So the transmitting channel can be viewed as lossless. From the definition of η (30) (31), we can obtain

$$\eta(P_{V|U}) = 1 \quad (32)$$

Then according to the definition (29), there is

$$\eta_T = 1 \quad (33)$$

Besides, our model considers no channel loss. That is, the channel capacity C is infinite. In (28), the upper bound of $I(W, V)$ is the minimum value of three parts. The last part $\eta(P_{X^n}, P_{W|X^n})CT$ is related with C . Therefore it can be ignored. Then the upper bound (28) can be simplified to the following

$$I(W, V) \leq \min\{I(W, X^n), \eta(P_{X^n}, P_{W|X^n})(H(X^n) \wedge b)\} \quad (34)$$

In this considered system, X^n is the measurement of the distance between the robot and its present destination. It contains n samples taken from the sensor. Its element X_i satisfies $X_i \in [0, l_0]$. This is a continuous variable. When quantifying it with b bit, it will certainly result in some bias. So $H(X^n)$ is larger than b . That is,

$$H(X^n) \wedge b = b \quad (35)$$

Therefore the upper bound (34) can be simplified further as

$$I(W, V) \leq \min\{I(W, X^n), \eta(P_{X^n}, P_{W|X^n})b\} \quad (36)$$

Next we should separately estimate $I(W, X^n)$ and $\eta(P_{X^n}, P_{W|X^n})b$ to finally obtain the upper bound.

Clarke [26] shows that

$$I(W, X^n) = \frac{d}{2} \log \left(\frac{n}{2\pi e} \right) + h(W) + \frac{1}{2} \mathbb{E} [\log \det J_{X|W}(W)] + o(1) \quad (37)$$

where $h(W)$ is the differential entropy of W , and $J_{X|W}(W)$ is the Fisher information matrix about w contained in X .

From (19), we get the density function of W

$$p(W) = \frac{2W}{l_0^2} \quad (38)$$

The differential entropy of W is

$$\begin{aligned} h(W) &= \mathbb{E}(-\log(p(W))) \\ &= \int_0^{l_0} -\frac{2W}{l_0^2} \log\left(\frac{2W}{l_0^2}\right) dW \\ &= \frac{1}{2} - \log \frac{2}{l_0} \end{aligned} \quad (39)$$

Now, we need to calculate the fisher information. The fisher information can be written as

$$\det J_{X|W}(W) = -\mathbb{E}\left[\frac{\partial^2}{\partial W^2} \log(P_{X|W})\right] \quad (40)$$

From (21), $P(X|W)$ is a joint Gaussian distribution of n independent samples. Then we have

$$\log(P_{X|W}) = -\frac{n}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - W)^2 \quad (41)$$

Then

$$\begin{aligned} \det J_{X|W}(W) &= -\mathbb{E}\left[\frac{\partial^2}{\partial W^2} \left(-\frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - W)^2\right)\right] \\ &= \frac{1}{2\sigma^2} \mathbb{E}\left[\frac{\partial}{\partial W} \left(-2 \sum_{i=1}^n (x_i - W)\right)\right] \\ &= \frac{1}{2\sigma^2} \mathbb{E}[2n] \\ &= \frac{n}{\sigma^2} \end{aligned} \quad (42)$$

In this system, since $W \in R^1$, we have $d = 1$. By using (39) and (42), one can obtain an estimation of $I(W, X^n)$.

$$\begin{aligned} I(W, X^n) &= \frac{1}{2} \log \left(\frac{n}{2\pi e} \right) + \frac{1}{2} - \log \left(\frac{2}{R} \right) + \frac{1}{2} \log \left(\frac{n}{\sigma^2} \right) + o(1) \\ &= \log \frac{nR}{2\sigma} + \frac{1}{2} - \log 2\pi e + o(1) \end{aligned} \quad (43)$$

The first part $I(W, X^n)$ is estimated. We shall estimate the second part of (36). The critical part of this is to estimate $\eta(P_{X^n}, P_{W|X^n})$. Here, a lemma in [25] can help us achieve it.

Lemma 5. For a joint distribution $P_{W,X}$, suppose there is a constant $\alpha \in (0, 1]$ such that the forward channel $P_{X|W}$ satisfies

$$\frac{dP_{X|W=w}}{dP_{X|W=w'}}(x) \geq \alpha \quad (44)$$

for all $x \in X$ and $w, w' \in W$.

Then the SDPI constants of the forward channel $P_{X|W}$ and the backward channel $P_{W|X}$ satisfy

$$\eta(P_{X|W}) \leq 1 - \alpha \quad (45)$$

and

$$\eta(P_{W|X}) \leq 1 - \alpha \quad (46)$$

In the considered system, we measure the distance independently for n times and get X^n . So we replace $\eta(P_{W|X})$ with $\eta(P_{W|X^n})$. From (31), we have

$$\eta(P_{W|X^n}) \geq \eta(P_{X^n}, P_{W|X^n}) \quad (47)$$

Then if (46) holds, we have

$$\begin{aligned} \eta(P_{X^n}, P_{W|X^n}) &\leq \eta(P_{W|X^n}) \\ &\leq 1 - \alpha \end{aligned} \quad (48)$$

Therefore the critical step of this part is to estimate α .

From (21), $P_{X|W=w}$ is a joint Gaussian distribution, with w as its mean. Then we can use the density function (21) to calculate

$$\begin{aligned} \frac{dP_{X|W=w}}{dP_{X|W=w'}}(x) &= \frac{\frac{1}{(\sqrt{2\pi}\sigma^2)^n} e^{-\sum_{i=1}^n \frac{(x_i-w)^2}{2\sigma^2}}}{\frac{1}{(\sqrt{2\pi}\sigma^2)^n} e^{-\sum_{i=1}^n \frac{(x_i-w')^2}{2\sigma^2}}} \\ &= e^{\sum_{i=1}^n \frac{(x_i-w')^2}{2\sigma^2} - \sum_{i=1}^n \frac{(x_i-w)^2}{2\sigma^2}} \\ &= e^{\frac{\sum_{i=1}^n (w-w')(2x_i-w-w')}{2\sigma^2}} \end{aligned} \quad (49)$$

As each measurement x_i is obtained independently. Therefore finding the minimum value of $(w-w')(2x_i-w-w')$ is equivalent to the calculation of minimum value of (49).

$$h = \min \left\{ (w-w')(2x-w-w') \right\} \quad (50)$$

where $x \in [0, l_0]$ and $w, w' \in [0, l_0]$.

It is easy to see that

$$(w-w')(2x-w-w') = 2(w-w')x - (w^2 - w'^2) \quad (51)$$

When $w > w'$, (51) increases as x increases. So the minimum value is obtained as $x = 0$. Then we have

$$h = -(w^2 - w'^2) \quad (w > w') \quad (52)$$

For $w, w' \in [0, l_0]$, we have

$$h = -l_0^2 \quad (w = 0, w' = l_0) \quad (53)$$

Similarly, when $w < w'$, (51) decreases as x increases. So the minimum value is obtained as $x = l_0$. Then we have

$$h = 2(w-w')l_0 - (w^2 - w'^2) \quad (54)$$

To calculate the minimum value, we transform (54) as the following.

$$h = w'^2 - 2w'l_0 - w^2 + 2wl_0 \quad (55)$$

(55) can be viewed as a quadratic function of w' . The minimum value can be reached as $w' = l_0$, for $w < w'$. That is, we have

$$h = -l_0^2 - w^2 + 2wl_0 \quad (w' = l_0) \quad (56)$$

(56) can be further viewed as a quadratic function of w . The minimum value can be reached as $w = 0$. Therefore

$$h = -l_0^2 \quad (w = 0) \quad (57)$$

Together with (53) and (57), we have

$$\min \left\{ (w-w')(2x-w-w') \right\} = -l_0^2 \quad (58)$$

Then there is

$$\frac{\sum_{i=1}^n (w-w')(2x_i-w-w')}{2\sigma^2} \geq -\frac{nl_0^2}{2\sigma^2} \quad (59)$$

Together with (49), we have

$$\frac{dP_{X|W=w}}{dP_{X|W=w'}}(x) \geq e^{-\frac{nl_0^2}{2\sigma^2}} \quad (60)$$

That is, (see (44))

$$\alpha = e^{-\frac{nl_0^2}{2\sigma^2}} \quad (61)$$

Finally, from Lemma 5 and (48), we have

$$\eta(P_{X^n}, P_{W|X^n}) \leq 1 - e^{-\frac{nl_0^2}{2\sigma^2}} \quad (62)$$

Now we have finished estimating the second part of (36). We can get a conclusion

$$I(W, V) \leq \min \left\{ \log \frac{nl_0}{2\sigma} + \frac{1}{2} - \frac{1}{2} \log(2\pi e), \left(1 - e^{-\frac{nl_0^2}{2\sigma^2}} \right) b \right\} \quad (63)$$

In (27), we replace $I(W, X)$ with $I(W, V)$. The lower bound in (27) should be corrected by multiplying l_0 . According to the upper bound given in (63), the final lower bound of the Bayes risk can be obtained

$$R_B \geq \frac{l_0}{2e} \max \left\{ 2^{-(\log \frac{nl_0}{2\sigma} + \frac{1}{2} - \frac{1}{2} \log(2\pi e))}, 2^{-\left(1 - e^{-\frac{nl_0^2}{2\sigma^2}} \right) b} \right\} \quad (64)$$

Therefore the proof of theorem 1 is completed.

VII. FORMATION CONVERSION CONTROL

In the practical use, robots should not only form a formation, but also change to another one to accomplish a different task. The formations are usually planned according to the routine requirements. This includes several different cases.

- * Robots reach a formation with a leader chosen by the user.
- * Robots reach a formation with a leader at the leading place. The leader is chosen by the system. The leading place is determined by the formation.
- * Robots reach a formation around a center chosen by the planner according to its requirement.
- * Robots reach a formation around a center chosen by the system to optimize the process.

In the following discussion, we assume the new formation's area should be the same as the former one.

In this part, we mainly discuss the transformation between square formation, triangle formation and circle formation. The method can also be applied to other transformations. Besides, we also give a moving strategy for a given center by the planner and a method to adaptively choose a center. We also confirm that the chosen center can minimize the total moving distance.

A. Triangle Formation

Fig. 4 shows the structure of a triangle formation with its center on the origin. This is a isosceles triangle, with equal numbers of robots on its two waist edges. From geometry, we know $AO = 2OD$. Excluding robots on the three vertices, we assume there are x robots on one waist edge and y robots on

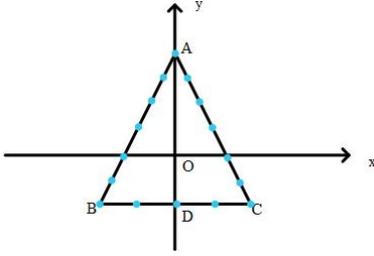


Fig. 4. a triangle formation with robots stand on the black points along the edges

the bottom edge. The total number of robots is n , then we have

$$3 + 2x + y = n \quad (65)$$

Therefore given y , we can have the arrangement of robots.

Given the area of the formation as $a \times b$, we can set

$$AD = a \quad BC = 2b$$

or

$$AD = b \quad BC = 2a$$

Then the position of A, B and C can be determined to locate every point in the formation.

B. Circle Formation

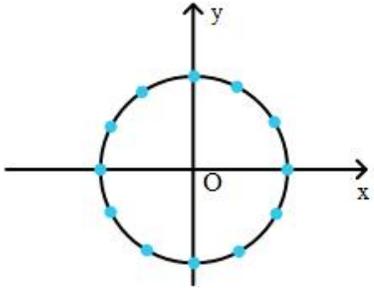


Fig. 5. a circle formation with robots stand on the arc evenly

Fig. 5 shows a circle formation with its center at the origin. Given the formation's area S , we can calculate the radius by $r = \sqrt{\frac{S}{\pi}}$. Then the positions of all points can be determined by partitioning the whole circle equally into n sections.

C. Transformation Strategies

Given a new formation, the robots need to move from the present one to it.

In Section III and Section IV, we have discussed methods to arrange robots in a given formation for arbitrary initial positions for robots and move to reach the formation.

1) *Center given by planner:* If the center is given and is not far, the robots can move to the proper positions around the center. But if the center is too far, this may be very costly. We should first reduce the distance between the desired center and those robots.

Assuming robots' position at time slot k is

$$X(k) = \{x_1(k), x_2(k), \dots, x_n(k)\}$$

Their present center is

$$C_{xp} = \frac{1}{n} \sum_{i=1}^n x_i(k) \quad (66)$$

while the desired center given by planner is denoted as C_x . Then we let all robots move towards the direction of $C_x - C_{xp}$ at a given speed. Given a threshold d_0 , if there is

$$\|C_x - C_{xp}\| \leq d_0 \quad (67)$$

the robots are close enough to the center. Then They follow method in section III and section IV to reach the formation around the center.

2) *Center determined by the system:* If the planner don't give the position of the center, the system need first find a center. From (3), the expected moving cost for reaching the formation is

$$C = \sum_{i=1}^n \|(x_i(0) - f_{d_i}) - C_x\|^2 \quad (68)$$

We need to choose the center to minimize C .

Theorem 2. For n robots with the initial position $X(0) = \{x_1(0), x_2(0), \dots, x_n(0)\}$ and a formation $F = \{f_1, f_2, \dots, f_n\}$ which satisfies $\sum_{i=1}^n f_i = 0$, when choosing the center as

$$C_x = \frac{1}{n} \sum_{i=1}^n x_i(0) \quad (69)$$

the moving cost C is minimized

Proof: Defining a function F of C_x as

$$F(C_x) = C = \sum_{i=1}^n \|(x_i(0) - f_{d_i}) - C_x\|^2 \quad (70)$$

then

$$\begin{aligned} \frac{dF}{dC_x} &= \sum_{i=1}^n -2(x_i(0) - f_{d_i} - C_x) \\ &= -2\left[\sum_{i=1}^n x_i(0) - \sum_{i=1}^n f_{d_i} - nC_x\right] \end{aligned} \quad (71)$$

For $\sum_{i=1}^n f_i = 0$, we have

$$\frac{dF}{dC_x} = -2\left[\sum_{i=1}^n x_i(0) - nC_x\right] \quad (72)$$

Let $\frac{dF}{dC_x} = 0$, we have

$$C_x = \frac{1}{n} \sum_{i=1}^n x_i(0) \quad (73)$$

and it's obviously that this is the minimum point.

According to Theorem 2, one can choose the optimal center and follow the methods in Section III and Section IV to reach the formation.

VIII. SIMULATIONS

In this section, we shall present some simulations for the multi-robot formation control. At the beginning, we present a group of robots reaching a square formation with a leader. Afterwards, we provide a display of robots in square formation changing to circle formation, then to triangle formation, and finally to a circle formation with a far away center. Then the moving cost of our position arrangement strategy in Section III is compared with other position arrangement strategies by simulations. At last, we discuss the relations between formation bias and system parameter. All simulations are run on MATLAB.

A. Formation Illustration

In this simulation, all robots are in a 1100×1100 square area. The number of robots n is set to be 15, and their initial positions are generated randomly in a 300×300 area.

We set the radius of R_m as $R = 300$, the quantification rate as $b = 7$, the area of the formation as $S = 28800$ and the variance of measuring the distance as $\sigma = 1$. The estimated total moving cost is calculated by definitions in (2) and (3), while the practical moving cost is calculated with simulated data.

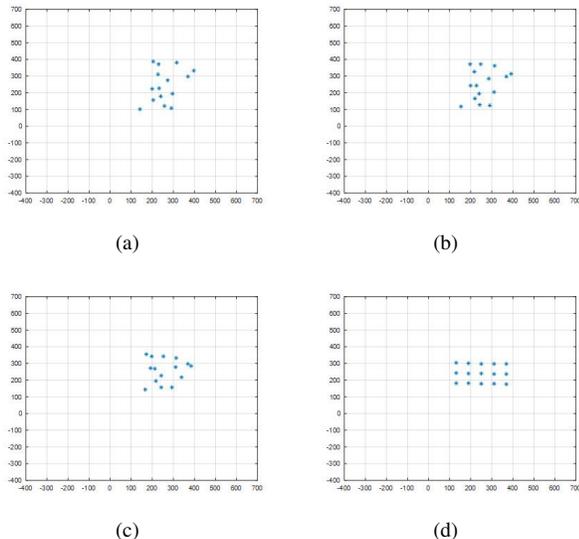


Fig. 6. Sub-figures (a) shows a group of robots in random initial position. (b)-(c) show the robots begin to move according to the calculated mapping arrangement. (d) shows robots having reached a square formation.

1) *Reaching a square formation from random initial positions:* In Fig. 6(a), robots' initial positions are randomly distributed. By applying methods in section III, the system selects a leading robot and arranges each robot in a position of the formation. Then robots move and reach the square formation. Finally, the square formation is reached as shown in Fig. 6(d).

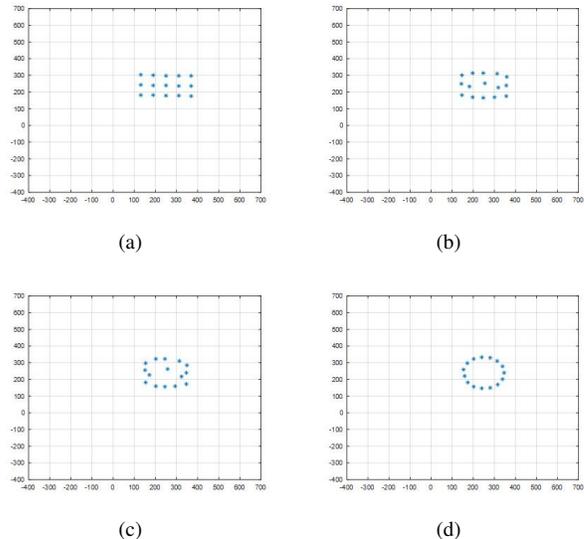


Fig. 7. Sub-figures (a) shows a group of robots in a square formation. (b)-(c) show the robots begin to move according to new mapping in circle formation. (d) shows robots having reached a circle formation.

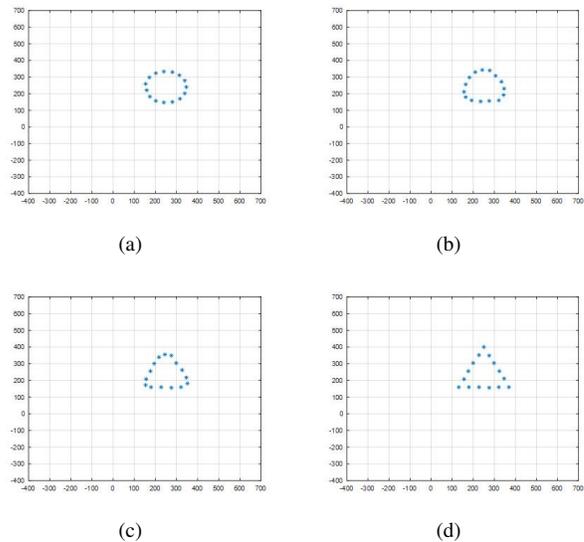


Fig. 8. Sub-figures (a) shows a group of robots in a circle formation. (b)-(c) show the robots begin to move according to new mapping in triangle formation. (d) shows robots having reached a triangle formation.

In this example, the estimated cost is 72000 and the practical cost is 150860. In fact, when reaching the square formation, each robot chooses another one in the nearby position as the reference. It adjusts to stand at the right position relative to the reference. For the robot referred to is also moving, the destination keeps moving as well. Then the robot's practical path is far from straight. That is why the practical path causes an extra distance compared to the estimated one resulting from calculating by using straight line distance.

2) *Conversion from square formation to circle formation:* The process of transformation from square formation to circle formation is shown in Fig. 7. The user doesn't give the center. The system determines the center's position according to

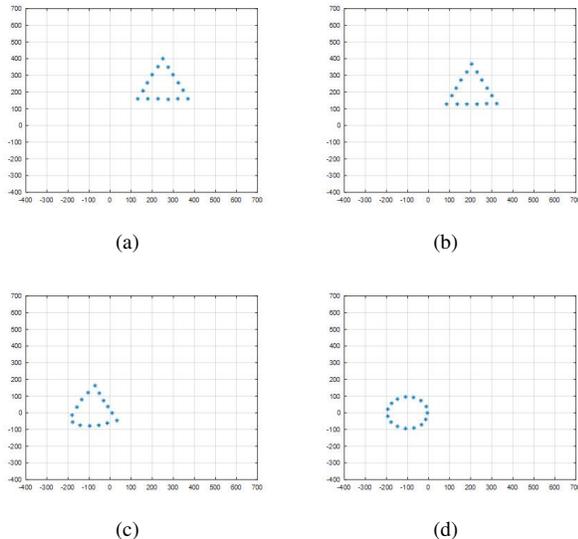


Fig. 9. Sub-figures (a) shows a group of robots in a triangle formation preparing to reach a formation around a center given by the user. (b) shows the robots is moving together near to the given center. (c) shows robots are moving to reach a formation around the given center. (d) shows robots having reached a circle formation around the center.

Theorem 2. Then the system arranges each robot to a position in the circle formation. Finally, robots move and reach the circle formation as shown in Fig. 7(d).

In the example, the estimated cost is 27211, while the practical moving cost is 27470. In the process, each robot moves referring to the fixed center. Regardless of the unfrequent collision avoidance, the practical path is nearly straight. Therefore the estimated cost is a little larger than the practical one. The collision avoidance also results in an extra moving cost.

3) *Conversion from circle formation to triangle formation:* In Fig. 8, robots transform gradually from circle formation to triangle formation. There is no center given by the user. The system first determines the center of the formation and obtains the triangle formation according to section VII.A. Then the system completes arrangement of robots in the formation. Finally, robots move and reach the triangle formation as shown in Fig. 8(d).

In this example, the estimated cost is 15642, while the practical cost is 15652. During this transformation, there is nearly no collision avoidance. Therefore the practical cost is nearly the same as the estimated one.

4) *Reaching a circle formation around a faraway center:* In Fig. 9(a), robots are in the triangle formation. User gives a center (-100,0). Robots first move together towards the center as shown in Fig. 9(b). When they are near the desired positions, the system calculates the arrangement of robots. Then robots move to reach the circle formation as shown in Fig. 9(c). Finally they reach the circle formation in Fig. 9(d).

In this example, the estimated cost is 2.71×10^6 , while the practical moving cost is 2.76×10^6 . This includes the extra cost caused by collision avoidance. Furthermore, robots move near the center rather than move directly towards the right position relative to the center. This causes an extra cost compared with

the straight distance.

B. Moving Cost comparison

To confirm the advantage of our proposed position arrangement strategy in reducing the moving cost, we compare it with other strategies. One strategy is called fixed position moving mode, where each robot is arranged in a fixed position in the formation. Another one is referred to as random position moving mode, where robots are arranged randomly in the formation. We do simulations and obtain the moving cost for each method. The comparison is done for two different cases. We first do experiments of reaching a square formation with a leader from random initial positions. Then we simulate the case of reaching a circle formation with a center. For each case, we do 200 experiments and compare the moving cost of the three strategies. In this simulation, our system parameter is the same as the one in Section VIII.A.

1) *moving cost for square formation:* As shown in Fig. 10, the moving cost of our proposed position arrangement is lower than the other two methods. Furthermore, its moving cost is more stable. Therefore the proposed position arrangement based on the Hungarian method can improve the system performance both in moving cost and in stability.

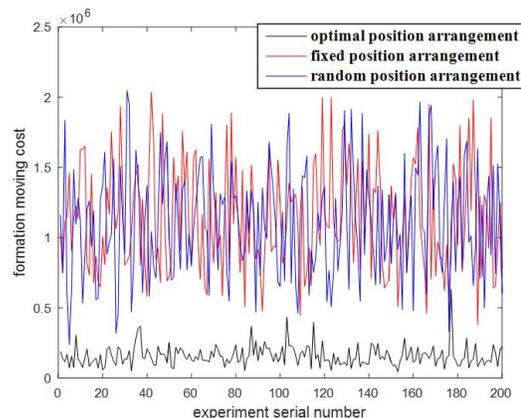


Fig. 10. Comparison of the moving cost of different position arrangement strategies for reaching the square formation. The graph represents moving cost of the three methods in different experiments.

2) *moving cost for circle formation:* As shown in Fig. 11, in the case of reaching the circle formation with a center, the result is similar as that presented in Fig. 10. Our arrangement strategy based on the Hungarian method can improve the system performance compared to other two methods.

C. Formation bias analysis

To investigate the offsets of the theoretical lower bound on the formation bias for the circle formation process, we carry out simulations to investigate the change of formation with respect to three different system parameters. The experimental data is fitted with the least squares method. The calculated lower bound is also depicted.

In each simulation, we adjust the conditioning parameter in a previously set range. In each parameter value, we carry out

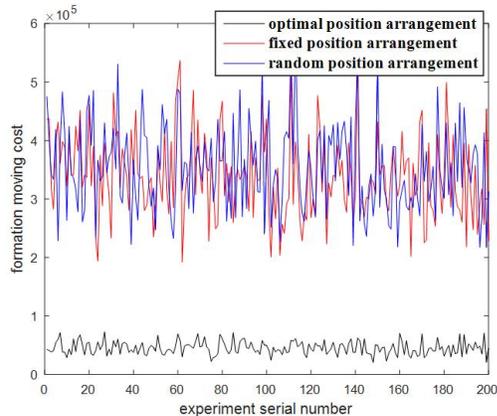


Fig. 11. Comparison of the moving cost for different position arrangement strategies for reaching a formation around a center. The graph represents moving cost of the three methods in different experiments.

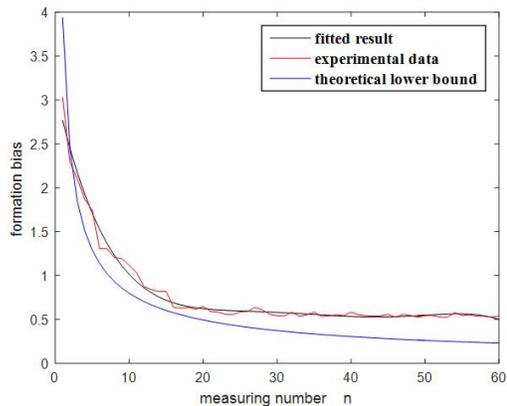
20 experiments. We let robots be in random initial positions and then complete reaching the circle formation to obtain the formation bias for each experiment.

1) *Measuring times n* : In this simulation, the formation area is $S = 28800$ and the measuring variance of angle is 0.05. There are 15 robots whose measuring variance of distance is 2. The radius of the system's controlling circle R_m is 200. The distance's partition number is 200. Converting it to the quantification rate, we have $b = 7.64$. The measuring number ranges from 1 to 60. In theoretical calculation, the distance's empirical ranging threshold is $l_0 = 120$.

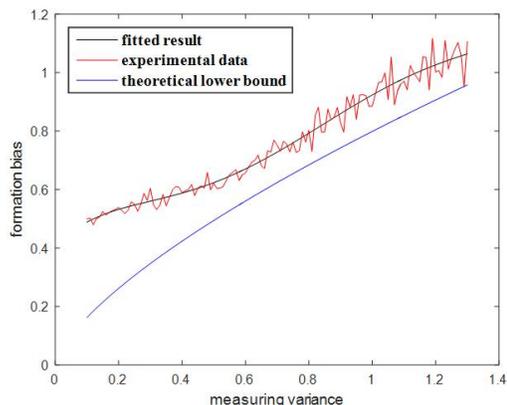
Fig. 12(a) shows the change of formation bias with respect to measuring times n . All curves decrease as n increases. From the point of common sense, when n is small, increasing the measuring times can reduce the formation bias apparently. When n is large enough, increasing n can't obtain obvious evolution. The tendency of curves is consistent with common sense. The red curve represents experimental data and the black curve is the fitting result with least square method. The blue curve is the calculated lower bound according to Theorem 1. In the beginning, the theoretical result is a little larger than experimental data. This is due to less statistics be done. However, in the rest majority part, the lower bound is below the experimental data and the tendency is the same.

2) *Distance measuring variance σ* : In this simulation, the formation area is $S = 28800$ and the measuring variance of angle is 0.05. There are 15 robots. Each robot measures the distance for 10 times. The radius of the system's controlling circle R_m is 200. The distance's partition number is 200. Converting it to the quantification rate, we have $b = 7.64$. The distance's measuring variance ranges from 0.1 to 1.3. In theoretical calculation, the distance's empirical ranging threshold is $l_0 = 120$.

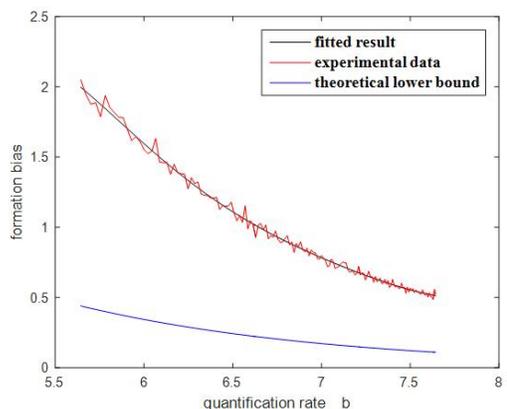
Fig. 12(b) shows the change of formation bias with respect to distance measuring variance σ . In general, all curves are increasing with σ . From the point of common sense, when the measuring variance increases, the measuring bias increases and causes the formation bias to increase. The tendency of curves is consistent with common sense. The theoretical lower bound



(a)



(b)



(c)

Fig. 12. Sub-figures (a) shows the formation bias decreases as the measuring number n increases. It depicts the experimental data, the fitted curve and the calculated lower bound. (b) shows the formation bias increases as the distance's measuring variance increases. (c) shows the formation bias decreases as the quantification rate b increases.

is below the experimental data and the tendency is the same.

3) *Quantification rate b* : In this simulation, the formation area is $S = 28800$ and the measuring variance of angle is 0.05. There are 15 robots. Each robot measures the distance for 10 times. The radius of the system's controlling circle R_m is 200. The measuring variance of distance is 0.01. The distance's partition number ranges from 50 to 200. Converting it to the quantification rate, we have $b \in [5.64, 7.64]$. In theoretical calculation, the distance's empirical ranging threshold is $l_0 = 120$.

Fig. 12(c) shows the change of formation bias with respect to the quantification rate b . All curves increase with b . From the point of common sense, when b increases, the measuring bias decreases, causing the formation bias to decrease. When b is large enough, the decreasing tendency slows down. The tendency of curves is consistent with common sense. The lower bound is below the experimental data. Though there is a gap, the tendency is the same.

IX. CONCLUSION

In this paper, we discussed the control of multi-robot formation and analyzed the system theoretically. First, we presented a method to arrange robots to positions in a formation by optimizing the total moving distance. We formulated the problem into a task distribution optimization problem and solved it in Hungarian method. Then we employed a leader-follower method to achieve the formation. Afterwards, we defined the formation bias and presented a lower bound of it, which can help us determine the parameter of the system. Besides, we also discussed control in formation changing. We also carried out various simulations to investigate the effects of system parameters on the formation bias.

REFERENCES

- [1] Maitreyi Nanjanath and Maria Gini. Repeated auctions for robust task execution by a robot team. *Robotics and Autonomous Systems*, 58(7):900–909, 2010.
- [2] Stuart Young and Alexander Kott. A survey of research on control of teams of small robots in military operations. *arXiv preprint arXiv:1606.01288*, 2016.
- [3] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: a versatile and accurate monocular slam system. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015.
- [4] Isaac Deutsch, Ming Liu, and Roland Siegwart. A framework for multi-robot pose graph slam. In *Real-time Computing and Robotics (RCAR), IEEE International Conference on*, pages 567–572. IEEE, 2016.
- [5] G. Diaz-Arango, H. Vzquez-Leal, L. Hernandez-Martinez, M. T. Sanz Pascual, and M. Sandoval-Hernandez. Homotopy path planning for terrestrial robots using spherical algorithm. *IEEE Transactions on Automation Science and Engineering*, PP(99):1–19, 2017.
- [6] Luiz Chaimowicz, Thomas Sugar, Vijay Kumar, and Mario Fernando Montenegro Campos. An architecture for tightly coupled multi-robot cooperation. In *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, volume 3, pages 2992–2997. IEEE, 2001.
- [7] Tomáš Krajník, Matías Nitsche, Jan Faigl, Petr Vaněk, Martin Saska, Libor Přeučil, Tom Duckett, and Marta Mejail. A practical multirobot localization system. *Journal of Intelligent & Robotic Systems*, 76(3-4):539–562, 2014.
- [8] Marco Todescato, Andrea Carron, Ruggero Carli, Antonio Franchi, and Luca Schenato. Multi-robot localization via gps and relative measurements in the presence of asynchronous and lossy communication. In *Control Conference (ECC), 2016 European*, pages 2527–2532. IEEE, 2016.
- [9] Li Wang, Aaron D Ames, and Magnus Egerstedt. Multi-objective compositions for collision-free connectivity maintenance in teams of mobile robots. In *Decision and Control (CDC), 2016 IEEE 55th Conference on*, pages 2659–2664. IEEE, 2016.
- [10] Manos Koutsoubelias and Spyros Lalis. Coordinated broadcast-based request-reply and group management for tightly-coupled wireless systems. *arXiv preprint arXiv:1606.08277*, 2016.
- [11] Stephen Kloder and Seth Hutchinson. Path planning for permutation-invariant multirobot formations. *IEEE Transactions on Robotics*, 22(4):650–665, 2006.
- [12] Hongjun Yu, Peng Shi, and Cheng-Chew Lim. Robot formation control in stealth mode with scalable team size. *International Journal of Control*, 89(11):2155–2168, 2016.
- [13] Zhiyong Sun, Changbin Yu, and Brian Anderson. Distributed estimation and control for preserving formation rigidity for mobile robot teams. *arXiv preprint arXiv:1309.4850*, 2013.
- [14] Ali Karimodini, Hai Lin, Ben M Chen, and Tong Heng Lee. Hybrid three-dimensional formation control for unmanned helicopters. *Automatica*, 49(2):424–433, 2013.
- [15] Ali Karimodini, Hai Lin, Ben M Chen, and Tong Heng Lee. Hybrid formation control of the unmanned aerial vehicles. *Mechatronics*, 21(5):886–898, 2011.
- [16] W. Jasim and D. Gu. Robust team formation control for quadrotors. *IEEE Transactions on Control Systems Technology*, PP(99):1–8, 2017.
- [17] Randal W Beard, Jonathan Lawton, and Fred Y Hadaegh. A coordination architecture for spacecraft formation control. *IEEE Transactions on control systems technology*, 9(6):777–790, 2001.
- [18] Xiangjun Qian, Arnaud De La Fortelle, and Fabien Moutarde. A hierarchical model predictive control framework for on-road formation control of autonomous vehicles. In *Intelligent Vehicles Symposium (IV), 2016 IEEE*, pages 376–381. IEEE, 2016.
- [19] Wenteng Zeng and Mo-Yuen Chow. Resilient distributed control in the presence of misbehaving agents in networked control systems. *IEEE transactions on cybernetics*, 44(11):2038–2049, 2014.
- [20] Dimitra Panagou and Vijay Kumar. Cooperative visibility maintenance for leader–follower formations in obstacle environments. *IEEE Transactions on Robotics*, 30(4):831–844, 2014.
- [21] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.
- [22] Harold W Kuhn. Variants of the hungarian method for assignment problems. *Naval Research Logistics Quarterly*, 3(4):253–258, 1956.
- [23] Denés König. Gráfok és mátrixok. *matematikai és fizikai lapok*, 38: 116–119, 1931.
- [24] John Adrian Bondy and Uppaluri Siva Ramachandra Murty. *Graph theory with applications*, volume 290. Macmillan London, 1976.
- [25] Aolin Xu and Maxim Raginsky. Information-theoretic lower bounds on bayes risk in decentralized estimation. *IEEE Transactions on Information Theory*, 2016.
- [26] Bertrand S Clarke and Andrew R Barron. Jeffreys' prior is asymptotically least favorable under entropy risk. *Journal of Statistical planning and Inference*, 41(1):37–60, 1994.