

A Decentralized Cluster Formation Containment Framework for Multirobot Systems

DOI:

[10.1109/TRO.2021.3071615](https://doi.org/10.1109/TRO.2021.3071615)

Document Version

Accepted author manuscript

[Link to publication record in Manchester Research Explorer](#)

Citation for published version (APA):

Hu, J., Bhowmick, P., Jang, I., Arvin, F., & Lanzon, A. (2021). A Decentralized Cluster Formation Containment Framework for Multirobot Systems. *IEEE Transactions on Robotics*, 37(6), 1936-1955. <https://doi.org/10.1109/TRO.2021.3071615>

Published in:

IEEE Transactions on Robotics

Citing this paper

Please note that where the full-text provided on Manchester Research Explorer is the Author Accepted Manuscript or Proof version this may differ from the final Published version. If citing, it is advised that you check and use the publisher's definitive version.

General rights

Copyright and moral rights for the publications made accessible in the Research Explorer are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Takedown policy

If you believe that this document breaches copyright please refer to the University of Manchester's Takedown Procedures [<http://man.ac.uk/04Y6Bo>] or contact uml.scholarlycommunications@manchester.ac.uk providing relevant details, so we can investigate your claim.



A Decentralized Cluster Formation Containment Framework for Multi-Robot Systems

Junyan Hu, *Member, IEEE*, Parijat Bhowmick, *Member, IEEE*, Inmo Jang, Farshad Arvin, *Senior Member, IEEE*, and Alexander Lanzon, *Senior Member, IEEE*

Abstract—Cooperative control of multi-robot systems (MRS) has earned significant research interest over the past two decades due to its potential applications in multi-disciplinary engineering problems. In contrast to a single specialized robot, MRS can be designed to offer flexibility, reconfigurability, robustness to faults and cost-effectiveness in solving complex and challenging tasks. In this paper, we aim to develop a unified cluster formation containment coordination framework for networked robots that can be decomposed into two layers containing the leaders and the followers. According to the proposed methodology, the leader robots are first distributed into a set of distinct and non-overlapping clusters depending on the positions and priorities of the targets exploiting a game-theoretic rule. Then they are steered to attain the desired formations around the corresponding targets. Subsequently, the follower robots are made to converge into the convex hull spanned by the leaders of the individual clusters. A prototype search and rescue operation is considered to highlight the usefulness of the proposed coordination framework. Furthermore, real-time hardware experiments were conducted on miniature mobile robots to validate the feasibility of the theoretical results.

Index Terms—Multi-robot coordination, decentralized decision making, autonomous clustering, formation tracking, containment control, task allocation, collision avoidance, search and rescue.

I. INTRODUCTION

With the recent advancements of computing, communication, sensing and control techniques, cooperative control of multi-robot systems (MRS) has established its worth in the recent years [1]. By using this technique, a large number of autonomous robots connected via a communication network can be coordinated to achieve a common objective [2]. Decentralized control protocols are mostly used in the coordination strategies, which depend only on the relative state information of the neighboring robots. Hence, the coordinated movements of the robots under the effect of the information flow can be analyzed via graph-theoretic approaches [1], [2].

When properly designed, MRS can offer flexibility, reconfigurability, robustness to faults and cost-effectiveness over a single sophisticated robot in solving complex and challenging tasks. For instance, MRS can be potentially utilized in military applications (such as entrapment/escorting mission [3], [4], target localization and mapping [5]–[7], security surveillance

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) [grant numbers EP/R008876/1, EP/P01366X/1 and EP/R026084/1] and EU Horizon 2020 [grant number 899520]. All research data supporting this publication are directly available within this publication.

The authors are with the Department of Electrical and Electronic Engineering, The University of Manchester, Manchester M13 9PL, UK. (e-mail: {junyan.hu, parijat.bhowmick, inmo.jang, farshad.arvin, alexander.lanzon}@manchester.ac.uk)



Fig. 1. Scenario of a smart agriculture farm that is facilitated with a proper coordination of (i) a team of aerial drones to survey the fields and to map weeds, yield and soil variation and (ii) a fleet of ground vehicles (specialized robots which consistently communicate with the survey drones) used for automated weeding, fertilizing and harvesting.

[8], [9], etc.); in search, rescue and retrieval operations especially in hazardous environments (e.g., in nuclear power plants, in earthquake-devastated areas, etc.) [10]; in autonomous shepherding missions [11]; in space and planetary exploration [12], [13]; in cooperative transportation [14]; in connected and autonomous vehicle platooning [15], etc. The key to the success of MRS in solving real-world robotics problems is to develop state-of-the-art cooperative control techniques and implementation mechanisms. These factors have urged the control researchers and robotics researchers to dedicate tireless effort in developing coordination techniques for MRS.

Among the popular cooperative control techniques, formation control, cluster formation tracking and containment control are the remarkable ones [16]. The basic form of the formation tracking control is the ‘leader-following’ case in which the followers attain the desired formation around the leader (or the target) and keep tracking the target [17], [18]. Formation control schemes can be implemented either in a centralized mode or via a decentralized manner. In [8], an optimal formation tracking scheme for a group of tri-rotor drones was designed utilizing a robust feedback linearization technique. However, the control law requires global information (i.e. the graph Laplacian matrix) of the network topology. In such a centralized method, the bandwidth requirement and the costs incurred in setting up an extensive communication infrastructure are significantly higher as compared to a decentralized system. Owing to the facts mentioned above, the idea of using decentralized cooperative control techniques which depends only on the local information (i.e. information of the immediate neighbors), has been widely accepted. In [19],

a decentralized observer-based formation control framework was developed for tracking the centroid of relative formations of MRS having first-order dynamics. In [20], a decentralized formation control scheme was proposed for a team of aerial and ground robots to be operated in a dynamic environment. The proposed method uses a constrained optimization algorithm to find the required parameters for achieving the desired formation in the presence of static and dynamic obstacles. However, only one leader has been considered in the literature mentioned above, even though the majority of the complex real-world robotics problems may require multiple leaders and multiple robot teams to accomplish the task.

In large-scale multi-robot applications (e.g. cooperative searching, multi-target enclosing, cooperative hunting, etc.), the main task is, in general, divided into several subtasks and accordingly, the robots are distributed into many subgroups (or clusters) corresponding to each subtask. Note that the clustering can be done offline (i.e. before commencing the mission) or online (i.e. during an ongoing task) depending on the nature and priorities of the subtasks. In [21], a cluster consensus protocol was designed for linear multi-agent systems on a directed graph via distributed feedback controller. The proposed results were extended in [22] to deal with multiple cluster formation problems. However, the communication topology used in [21], [22] has significant restrictions, e.g., the subgroups must be connected via directed topology with acyclic partition and the topology is also required to satisfy the in-degree balanced condition for each subgroup. Besides, the agents cannot autonomously choose a subgroup to join, depending on the specific tasks and priorities. Often in some applications, the followers of different subgroups may need to be re-distributed into new subgroups during an ongoing mission depending on the changing priorities of the tasks [4]. Re-clustering is also required when a leader of one subgroup stops operating due to a fault and its followers need to be merged with other subgroups [4]. These are some of the essential concerns which advocate the requirement of autonomous clustering. A reliable and effective clustering algorithm seeks to know about: (i) how to decide the number of clusters, (ii) how to distribute the robots into the clusters, (iii) how many robots should be assigned to individual clusters to accomplish the given task, i.e., how to avoid over-crowdedness, (iv) when to initiate the re-clustering process, etc. The literature [23]–[27] laid significant contributions in developing clustering and group regulation algorithms that addressed some of the issues mentioned above. Lately, [28] has also proposed an improved autonomous clustering algorithm exploiting the concept of Nash-stable equilibrium and partitioning.

In the context of multi-robot navigation and control, maintaining coordination among the leaders and followers is of prime importance. The followers can either directly observe the leaders (referred to as ‘leader-following’ case) or can be surrounded by the leaders (known as ‘containment’). In some particular multi-robot applications, the containment strategy may be advantageous than the leader-following option. Consider a search and rescue operation to be performed by MRS in a cluttered environment, where the leader robots have sensors to detect the obstacles while the followers do

not. The leaders can navigate the followers safely towards the destination by keeping them inside a safe moving zone guarded by the leaders (i.e. the convex hull spanned by the leaders). This operation is known as the ‘containment control’ in the literature [16]. Pioneering research has been conducted in [29] and [30] on distributed containment control of multi-agent systems. In [29], a distributed containment control protocol was designed for a multi-agent system having double integrator dynamics in the presence of both static and dynamic leaders. [30] introduced a continuous containment control law for multi-agent systems having general linear dynamics contrary to the conventional discontinuous control action. However, these articles did not show experimental validation results. Besides, only a single group of agents was considered in the containment control problem.

So far, we have discussed the usefulness and applications of clustering, formation tracking and containment control individually. However, many real-world robotics problems need all three above techniques to be applied in a proper sequence. In this context, we refer to a practical example of an automated weed monitoring and control in precision agriculture, as shown in Fig. 1. This particular problem can be formulated as a cluster, formation containment control problem to be solved by a fleet of heterogeneous robots (i.e. a combination of aerial drones and ground vehicles). Firstly, the aerial drones are distributed into several subgroups, which will inspect the entire field and discern the areas full of weeds using formation tracking principles. The ground robots then directly interact with the survey-drones to approach and enter into the marked areas, following the containment algorithms, for removing the weeds. This case study necessitates thinking on a unified clustering, formation and containment methodology leading to a state-of-the-art cooperative control framework for decentralized multi-robot applications.

Motivated by the challenges stated above and by the increasing need to develop effective cooperative control frameworks for handling large-scale MRS, this paper aims to design a decentralized cluster formation containment (CFC) framework for a class of MRS. Below, we summarize the contributions of this paper along with mentioning the key features of the proposed CFC framework:

- A unified decentralized coordination framework is proposed for a class of networked MRS having multiple time-varying or stationary targets, which is comprised of autonomous clustering, formation tracking and containment control methodologies;
- The autonomous decision-making property facilitates re-clustering during an ongoing mission when two targets come too close or if some of the leaders become faulty. Furthermore, the proposed clustering algorithm helps to avoid over-crowdedness in the clusters;
- In virtue of the decentralized control mechanism, each robot needs the information of its neighbors only, which significantly reduces the bandwidth requirement. Furthermore, in contrast to many existing works on this topic, the CFC protocols do not need to calculate the eigenvalues of the graph Laplacian matrix;
- Reconfigurability of the proposed scheme enables a net-

worked MRS to withstand the situations when some of the robots stop working due to malfunctioning, or some new robots join the group;

- The proposed theory has been validated through affordable real robot experiments involving a group of autonomous mobile robots in a real-time laboratory environment equipped with low-cost sensors.

The remaining of the paper proceeds as follows: Section II provides important technical preliminaries which are invoked in deriving the main results. Section III contains the main contributions of the paper – Subsection III-A presents a game theory-based autonomous clustering algorithm for the leaders depending on the positions and priorities of the targets; Subsection III-B proposes a decentralized formation tracking scheme; Subsection III-C puts forward a decentralized containment control law; and Subsection III-D discusses a Decentralized Artificial Potential Field (DAPF) method for obstacle and collision avoidance. Section IV presents an exhaustive case study on a real-time search and rescue mission to demonstrate the usefulness of the proposed framework and also shows a prospective application of this framework. Both simulation results and experimental observations with real robots are provided to validate the efficacy and feasibility of the proposed CFC framework. Section V concludes the paper.

Notations and acronyms are standard throughout. $\mathbb{R}_{>0}$ and $\mathbb{R}_{\geq 0}$ denote respectively the sets of all positive and all non-negative real numbers. Let I_n denote the identity matrix of dimension $n \times n$ and $\mathbf{1}_N$ be the vector with N number of entries all being 1. $\text{diag}\{a_1, \dots, a_N\}$ represents a diagonal matrix with the diagonal entries a_i . The Kronecker product of two matrices A and B is denoted by $A \otimes B$. $\|\cdot\|$ expresses the 2-norm of a vector or a matrix. A square matrix $A \in \mathbb{R}^{n \times n}$ is called a non-singular M -matrix if all its off-diagonal elements are non-positive and all eigenvalues of A have positive real parts [16].

II. TECHNICAL BACKGROUND

A. Communication networks

Consider a weighted and directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ with a non-empty set of nodes $\mathcal{V} = \{1, 2, \dots, N\}$, a set of edges $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$, and the associated adjacency matrix $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{N \times N}$. An edge rooted at the i^{th} node and ended at the j^{th} node is denoted by (i, j) , which means information can flow from node i to node j . a_{ij} is the weight of edge (j, i) and $a_{ij} \neq 0$ if $(j, i) \in \mathcal{E}$. Node j is called a neighbor of node i if $(j, i) \in \mathcal{E}$. Define the in-degree matrix as $D = \text{diag}\{d_i\} \in \mathbb{R}^{N \times N}$ with $d_i = \sum_{j=1}^N a_{ij}$. The Laplacian matrix $L \in \mathbb{R}^{N \times N}$ of \mathcal{G} is defined as $L = D - \mathcal{A}$.

Lemma 1: [31] Let $L \in \mathbb{R}^{n \times n}$ be an M -matrix with $\det[L] \neq 0$. Then, there exists a positive definite matrix $G = \text{diag}\{g_1, \dots, g_N\}$ such that $GL + L^T G > 0$.

B. Robot motions

Consider a team of multiple robots in which the position of the i^{th} robot is denoted by $x_i(t) \in \mathbb{R}^n$ and each robot is characterized by the single-integrator dynamics

$$\dot{x}_i(t) = u_i(t), \quad (1)$$

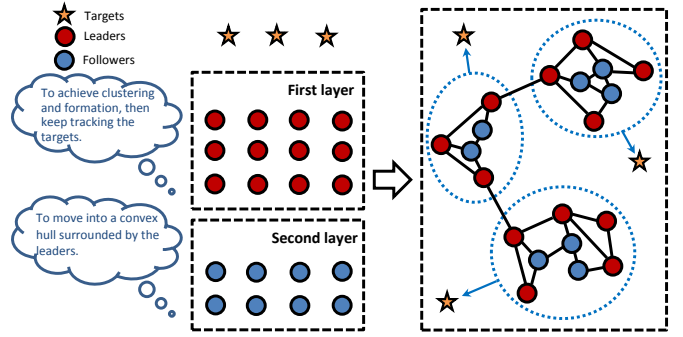


Fig. 2. Two-layer formation containment control framework proposed for multi-robot control problems.

where $u_i(t) \in \mathbb{R}^n$ is the bounded control input applied to the i^{th} robot and there exists a positive constant σ such that $\|u_i(t)\| \leq \sigma \forall t$.

Note that most of the robot platforms (e.g., UGVs and UAVs) are highly nonlinear and coupled [32], but the dynamics of the robot can be linearized to a single-integrator system at any operating point by exploiting the input-output feedback linearization technique [33]. The control inputs $u_i(t) \forall i \in \{1, \dots, N\}$ should be designed properly for the whole team based on the given tasks. These robots are required to complete the task successfully while avoiding environmental obstacles and collisions with each other along their trajectories.

C. Formation containment framework

A two-layer framework is adopted to handle the cluster, formation containment problem as shown in Fig. 2, which consists of the leaders' formation layer and the followers' containment layer. In the first layer, the objective is for the networked leaders to first divide into several separate clusters, then, in each cluster, they should converge to a predetermined formation and lastly, each clustered formation should move together following its target robot. In the second layer, the objective is for all the follower robots to converge into a convex hull spanned by the leaders in the neighboring clustered formation. The 'cluster, formation containment' objective is said to be achieved if the control objectives of both layers are achieved.

D. Problem statement

We now describe the problem statement of this paper in a precise form. Suppose a multi-robot coordination problem is given that can be formulated as a two-layer cluster, formation containment problem or its particular combinations. The main objective is to develop (i) a game theory-based clustering algorithm by which the leaders are first distributed into several clusters depending on the priorities and some distinguishing features (e.g., locations) of the targets and (ii) a decentralized and adaptive cluster, formation containment (CFC) protocol which steers the leaders within each cluster to attain a pre-specified formation around their target while tracking the corresponding target and also, enforces the followers to converge into the area (convex hull) spanned by the leaders

of a neighboring cluster. This is explained in Fig. 2 which shows that the leader robots (Red coloured) in the first layer are clustered into different groups according to the priorities and positions of the targets and then, cluster-formation is achieved. Subsequently, the follower robots (Blue coloured) in the second layer are attracted to enter into the regions surrounded by the leaders of each neighboring cluster.

III. MAIN RESULTS

In this section, we propose a game theory-based decentralized and adaptive control framework for multi-robot applications. The proposed scheme has three main steps as described below:

- (Step 1:) An autonomous and target-specific clustering process is developed in Subsection III-A;
- (Step 2:) A decentralized and adaptive cluster, formation tracking (CFT) protocol has been proposed for the leader robots in Subsection III-B;
- (Step 3:) A decentralized and adaptive containment control law is designed for the follower robots in Subsection III-C.

Apart from these clustering, formation tracking and containment methodologies, the proposed framework also encompasses a set of obstacle and collision avoidance protocols because during real-time implementation, obstacle and/or collision avoidance may frequently be required.

Consider a group of N robots, which consists of M leaders and $N - M$ followers. Suppose we have p ($p \geq 1$) targets (either real robot or virtual robot) for the leaders to track. Let $\mathcal{T} = \{1, 2, \dots, p\}$, $\mathcal{L} = \{p + 1, p + 2, \dots, p + M\}$ and $\mathcal{F} = \{p + M + 1, p + M + 2, \dots, p + N\}$ be the sets of the targets, leaders and followers respectively. All the robots are connected by a communication network. The network topology of the entire MRS is denoted by the graph \mathcal{G} as shown in Section II-A. In formulating this problem, we assume that leaders do not consider followers and followers do not consider targets in their motions. Then the Laplacian matrix L associated with \mathcal{G} can be partitioned as

$$L = \begin{bmatrix} 0 & 0 & 0 \\ L_{lt} & L_{ll} & 0 \\ 0 & L_{fl} & L_{ff} \end{bmatrix}, \quad (2)$$

where $L_{lt} \in \mathbb{R}^{M \times p}$, $L_{ll} \in \mathbb{R}^{M \times M}$, $L_{fl} \in \mathbb{R}^{(N-M) \times M}$ and $L_{ff} \in \mathbb{R}^{(N-M) \times (N-M)}$.

A. Construction of target-specific clusters

In this subsection, we utilize a game theoretic decision making algorithm that enables multiple robots to cluster themselves considering not only their individual preferences but also the intra- and inter-cluster competitions (e.g., how to avoid over-crowdedness in the clusters without affecting the safe and reliable operation).

Given the set of leader robots \mathcal{L} , the associated network topology $\mathcal{G}_{\mathcal{L}} = \{\mathcal{V}_{\mathcal{L}}, \mathcal{E}_{\mathcal{L}}, \mathcal{A}_{\mathcal{L}}\}$ and the set of p targets \mathcal{T} , the leader robots are required to be able to distribute themselves autonomously into a set of p disjoint clusters $\Pi = \{\mathcal{V}_1, \mathcal{V}_2, \dots, \mathcal{V}_p\}$ called *partition* where $\mathcal{V}_k \subseteq \mathcal{V}_{\mathcal{L}}$ is

the robotic cluster assigned to target k such that $\mathcal{V}_k \neq \emptyset$, $\mathcal{V}_{\mathcal{L}} = \bigcup_{k=1}^p \mathcal{V}_k$ and $\mathcal{V}_k \cap \mathcal{V}_l = \emptyset$ for any $k, l \in \mathcal{T}$ and $k \neq l$. Let d_{ik} denote the distance from leader robot $i \in \mathcal{L}$ to target $k \in \mathcal{T}$ and r_k is the target's importance within the context of a given mission. In this context, a desirable collective behavior from the leader robots is such that they join clusters of targets in close proximity while forming different-sized clusters depending on the associated targets' importance. Here, the graph $\mathcal{G}_{\mathcal{L}}$ is assumed to be at least strongly connected so that the local information of any leader robot can somehow be transmitted to others in multi-hop fashion.

In order for the leader robots to construct such disjoint clusters in a decentralized manner without any inter-robot conflict, we utilize an anonymous hedonic game theoretic framework termed as 'Group Agent Partitioning and placing Event' (GRAPE) [28] which models the leaders as self-organized players willing to form robotic coalitions according to individual preferences. GRAPE offers the following advantages: (1) the leader robots can always converge to an agreed partition of disjoint clusters, called *Nash stable partition* (see Definition 1), where every robot is not willing to deviate; (2) the converging process is analytically proven to finish within polynomial time $O(|\mathcal{V}_{\mathcal{L}}|^2 d_{\mathcal{G}_{\mathcal{L}}})$, where $d_{\mathcal{G}_{\mathcal{L}}}$ denotes the diameter of the graph $\mathcal{G}_{\mathcal{L}}$; and (3) the resultant partition enjoys the suboptimality property defined with respect to maximizing the global utility and it is lower bounded.

The aforementioned benefits are guaranteed as long as a particular condition called 'Single-Peaked-At-One' (SPA0) [28] holds. Intuitively, SPA0 implies 'social inhibition' of multiple robots. In the present research problem, SPA0 can be satisfied if the following assumption holds.

Assumption 1: The individual preference of each leader robot within a cluster can be modelled as an individual utility function that monotonically decreases as the number of leaders in that cluster increases.

Now, we will define the utility function for i^{th} leader robot assigned to cluster $\mathcal{V}_k \in \Pi$ as

$$\Omega_{ik} = w_i^r \frac{r_k}{|\mathcal{V}_k \cup \{i\}|} - w_i^d d_{ik} \quad (3)$$

for all $i \in \mathcal{L}$ where r_k represents the importance of the k^{th} target; d_{ik} denotes the distance between the i^{th} robot and the k^{th} target; $w_i^r \in \mathbb{R}_{>0}$ and $w_i^d \in \mathbb{R}_{>0}$ are the weights used to scale the factors r_k and d_{ik} . This utility function implies that a leader robot would be more attracted towards a target having higher importance while avoiding a distant target. In addition, since task importance r_k is equally shared with the member of cluster \mathcal{V}_k , it is expected that the leader robots would be partitioned in accordance with the relative importance of the tasks. The utility function defined in (3) complies with Assumption 1 and therefore we can exploit all the advantages of GRAPE.

Definition 1 (Nash stable partition): Given the set of leader robots \mathcal{L} , the set of targets \mathcal{T} and the utility functions of the leader robots $\{\Omega_{ik}, \forall i \in \mathcal{L}, \forall k \in \mathcal{T}\}$, a partition Π is said to be Nash stable if for each leader $i \in \mathcal{L}$, $\Omega_{i\Pi(i)} \geq \Omega_{ik}$ for all $k \in \mathcal{T}$ where $\Pi(i)$ represents the index of the cluster to which i^{th} leader robot belongs.

1) *Algorithm*: The decentralized and autonomous clusterization process is presented as Algorithm 1. Here, all the superscripts are used to refer to information locally owned by each leader robot, for example, Π^i being the partition locally known to the i^{th} leader robot. The clustering process for each leader robot consists of two parts: (1) local decision making based on given Π^i and (2) making consensus with neighbor leader robots for all k such that $(k, i) \in \mathcal{E}_{\mathcal{L}}$.

Algorithm 1 Decentralized and autonomous clustering algorithm for each leader robot $i \in \mathcal{L}$

```

// Initialize
1:  $ok^i \leftarrow false$ ;  $\sigma^i \leftarrow 0$ ;  $\tau^i \leftarrow 0$ 
2:  $\Pi^i \leftarrow \{\mathcal{V}_1, \dots, \mathcal{V}_p\}$  where  $\forall \mathcal{V}_k = \emptyset$ 
3: while true do
  // Make a new decision if necessary
4:   if  $ok^i = false$  then
5:      $k^* \leftarrow \arg \max_{\mathcal{V}_k \in \Pi^i} \Omega_{ik}$ 
6:     if  $k^*$  is not the current cluster index then
7:       Join  $\mathcal{V}_{k^*}$  and update  $\Pi^i$ 
8:        $\sigma^i \leftarrow \sigma^i + 1$ ;  $\tau^i \in \text{unif}[0, 1]$ 
9:       Update  $M^i = \{\Pi^i, \sigma^i, \tau^i\}$ 
10:    end if
11:     $ok^i = true$ 
12:  end if
  // Make a local consensus
13:  Broadcast  $M^i$  to neighbor leaders
14:  Receive  $\exists M^k$  from any leader  $k$  such that  $(k, i) \in \mathcal{E}_{\mathcal{L}}$ 
15:  for each  $M^j \in \{\forall M^k\}$  do
16:    if  $(\sigma^j > \sigma^i)$  or  $(\sigma^j = \sigma^i \ \& \ \tau^j > \tau^i)$  then
17:       $M^i \leftarrow M^j$ 
18:       $ok^i \leftarrow false$ 
19:    end if
20:  end for
21: end while

```

Firstly, given Π^i , each leader robot $i \in \mathcal{L}$ looks for its best cluster according to its utility functions $\Omega_{ik} \forall k \in \mathcal{T}$ (Line 5) unless the robot is satisfied with Π^i (Line 4). This local satisfaction is represented by ok^i , which becomes *true* as soon as the robot has finished examining Π^i no matter whatever decision was made based on it (Lines 6–11). If a new decision was made, the leader robot unilaterally joins its best cluster, denoted by \mathcal{V}_{k^*} , and updates Π^i accordingly (Line 7). The header information of Π^i is also updated wherein σ^i indicates the number of evolution of the partition and τ^i is a randomized time stamp. This header facilitates decentralization of this algorithm, the details of which will be described in the following paragraph.

Thereafter, the leader robot broadcasts to its neighboring leaders a message M^i consisting of the header (i.e., σ^i, τ^i) and the partition information Π^i (Line 13). Meanwhile, the robot could also receive other messages from its neighbor leaders (Line 14). Given multiple messages, the robot selects only one message among them ruling out the others by using each header information (Lines 15–20). This process allows each robot to eventually select a common partition information in a decentralized manner as long as the communication topology

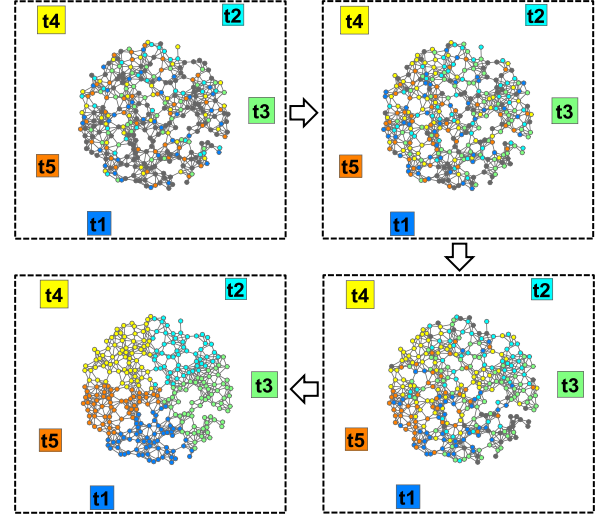


Fig. 3. Example of the clustering algorithm with 5 targets and 320 leaders, where color of each circle implies that the corresponding leader is assigned to the same colored target.

$\mathcal{G}_{\mathcal{L}}$ is connected but not necessarily fully connected. From the viewpoint of each local partition Π^i , its evolution process is equivalent to the one with the assumption of a mutual exclusion process [28], in which the partition can always converge to be Nash stable unless it is ruled out. Therefore, in this decentralized algorithm, the local partition with the highest evolution time and/or the highest time stamp value will eventually be a consensus to every leader robot. An example of the proposed clusterization algorithm is illustrated in Fig. 3.

2) *Suboptimality of a partition*: Given a partition Π generated by Algorithm 1, suboptimality of the partition is lower bounded and can be expressed in terms of the aggregate utility of the leader robots defined as

$$J = \sum_{\forall i \in \mathcal{L}} \Omega_{i\Pi(i)}, \quad (4)$$

where $\Pi(i)$ indicates the index of the target selected by the i^{th} robot according to the game theoretic rules listed in Algorithm 1. In this process, each leader robot attempts to maximize its own utility without considering the global utility and hence, it does not achieve global maximum even though the result is agreed by all the leader robots. In this paper, suboptimality of a Nash stable partition is obtained via Algorithm 1 and is denoted by

$$\rho = \hat{J}/J^*, \quad (5)$$

where \hat{J} represents the aggregate utility and J^* denotes the optimal value of J . Suboptimality of a Nash stable partition is shown to be lower bounded as follows:

Theorem 1: Given a Nash stable partition Π , its suboptimality is lower bounded by the value

$$\rho = \frac{\hat{J}}{\hat{J} + \sum_{\forall \mathcal{V}_k \in \Pi} \max_{i \in \mathcal{L}, q \leq |\mathcal{L}|} q \cdot (\tilde{\Omega}_{ik}[q] - \Omega_{ik})} \quad (6)$$

where

$$\tilde{\Omega}_{ik}[q] = w_i^r \frac{r_k}{q} - w_i^d d_{ik} \quad \forall q \in \mathbb{N}$$

and Ω_{ik} is as defined in (3).

Proof: Given a Nash stable partition Π , it follows from Definition 1 that

$$\Omega_{i\Pi(i)} \geq \Omega_{ik^*} \quad \forall i \in \mathcal{L} \quad (7)$$

where k^* indicates the index of the cluster that the i^{th} leader robot should have individually selected so as to construct an optimal partition collectively. Note that Ω_{ik^*} does not correspond to the optimal value (denoted by $\Omega_{ik^*}^*$) because the given partition Π is not optimal rather suboptimal. The term Ω_{ik^*} in (7), can be expressed as

$$\Omega_{ik^*} = \Omega_{ik^*}^* - (\Omega_{ik^*}^* - \Omega_{ik^*}). \quad (8)$$

Substituting (8) into (7) and then summing both sides of (7) over all $i \in \mathcal{L}$, we get

$$\sum_{\forall i \in \mathcal{L}} \Omega_{i\Pi(i)} \geq \sum_{\forall i \in \mathcal{L}} \Omega_{ik^*}^* - \sum_{\forall i \in \mathcal{L}} (\Omega_{ik^*}^* - \Omega_{ik^*}). \quad (9)$$

The left-hand side of (9) indicates \hat{J} , the first term in the right-hand side represents J^* while the second term is upper bounded by

$$\sum_{\forall k \in \mathcal{T}} |\mathcal{V}_k^*| \cdot \max_{\forall i \in \mathcal{V}_k^*} (\Omega_{ik^*}^* - \Omega_{ik^*}), \quad (10)$$

where \mathcal{V}_k^* denotes the cluster corresponding to k^{th} target, given the optimal partition Π . Although $\Omega_{ik^*}^*$ is unknown, its upper bound can be obtained from the following relation derived from (3)

$$\tilde{\Omega}_{ik}[q] = w_i^r \frac{r_k}{q} - w_i^d d_{ik} \quad (11)$$

for all $q \in \{1, 2, \dots, |\mathcal{L}|\}$ and for all $i \in \mathcal{L}$. Using (11), an upper bound of (10) can be calculated as

$$\sum_{\forall \mathcal{V}_k \in \Pi} \max_{i \in \mathcal{L}, q \leq |\mathcal{L}|} q \cdot (\tilde{\Omega}_{ik}[q] - \Omega_{ik}) \quad (12)$$

Now, (12) implies from (9)

$$\hat{J} \geq J^* - \sum_{\forall \mathcal{V}_k \in \Pi} \max_{i \in \mathcal{L}, q \leq |\mathcal{L}|} q \cdot (\tilde{\Omega}_{ik}[q] - \Omega_{ik}).$$

The above inequality is equivalent to

$$\rho \geq \frac{\hat{J}}{\hat{J} + \sum_{\forall \mathcal{V}_k \in \Pi} \max_{i \in \mathcal{L}, q \leq |\mathcal{L}|} q \cdot (\tilde{\Omega}_{ik}[q] - \Omega_{ik})} \equiv \rho$$

where ρ is the lower bound of the suboptimality of the Nash stable partition Π . ■

While developing the technical results of this subsection, we have encountered several important issues which deserve further discussion. We have brought them into limelight through following remarks.

Remark 1: Although the autonomous clustering strategy is useful in multi-robot applications, it may face difficulties when applied in a highly cluttered environment. This happens because the utility function (3) relies on the point to point relative distance d_{ik} between a robot and a target without considering practical constraints (e.g. long wall). To handle such cases, the robots may be equipped with proximity sensor modules to detect obstacles and to read also their approximate

dimensions to aid the clustering algorithm to resolve such conflicting cases. □

Remark 2: Recently, for clustering multiple robots, algorithms based on bipartite matching [34] and genetic algorithms [35] were proposed. However, these methods rely on a centralized architecture (i.e. global information is required in the iterative process), which could only be used well for a small number of robots with an ideal communication network as pointed out in [36]. In order to avoid such limitations, distributed k-means algorithms [26], [27] have been popularly used in MRS. Although the agents using k-means algorithms can be made self-organizing in a way that chooses the most ‘self-interested’ cluster, it is not straightforward to form the social utility functions. Compared to k-means algorithms, the proposed clustering algorithm has a different algorithmic process, which utilizes the game-theoretic findings that are useful for MRS (e.g. ability to accommodate social utilities, convergence towards an agreed outcome, decentralization and sub-optimality guarantee). For instance, proportional clustering with regard to the importance of tasks is desired from the global perspective. In our work, this desirable feature is encoded as the first term (i.e. $w_i^r \frac{r_k}{|\mathcal{V}_k \cup \{i\}|}$) in (3), but it may not be accommodated in k-means algorithms because the inter-agent conflicts may cause divergence of the k-means based clustering process (e.g. repeated back-and-forth decisions). Besides, auction-based algorithms for multi-robot task allocation problems are addressed in [37], [38]. However, in this case, a hierarchical architecture over the given robots should be introduced and hence global level of information sharing and high computation resources may be required. Different from the aforementioned works, GRAPE is a self-organizing approach where each robot’s decision-making relies only on local information and local interaction, which facilitates scalability. Moreover, even if each robot does not consider its social utility in the algorithmic process, the performance in terms of global utility can still be guaranteed, which is another different benefit. □

Remark 3: A Nash-stable partition may not be always optimal similar to the characteristic of a Nash equilibrium. For a multi-robot clustering problem, it may not always be feasible to find an optimal solution in a reasonable time specially in applications which involve many robots. As pointed out in [39], there is a trade-off between the optimality of a solution and the search efficiency for obtaining that solution. Apart from the optimality issue, scalability is also desirable to accommodate as many robots as possible in order to deal with a variety of real-world robotic problems. To ensure scalability, the notion of a Nash equilibrium is beneficial because, in the process of finding it, the required inter-robot communication can be reduced as the robots do not need to have permission protocol usually required in the other frameworks. Although the proposed method is fully decentralized, it may suffer from suboptimal outcomes. Therefore, a lower bound of the possible suboptimality should be analyzed. □

Remark 4: The cluster construction algorithm proposed in this paper enables the clustering process adaptable to any dynamic changes in the structural configuration of the robots. For instance, in case of loss of some robots from a multi-

robot system or addition of new robots into that system during an ongoing mission, a new decision is made to redistribute the robots into different clusters corresponding to the new configuration of the MRS. It can be shown that the number of iterations required for the algorithm to converge into a new Nash stable partition is at most the number of the total robots and hence, for MRS with finite number of robots, the clustering algorithm is guaranteed to converge into a new Nash stable partition in case a re-clustering is required any time during the process [28]. \square

B. Cluster formation tracking of the leader robots

In this subsection, we introduce a *cluster formation tracking* (CFT) protocol for the leader robots in each cluster. Once the clustering process is done (discussed in the previous subsection) based on the priorities and locations of the targets, the next goal is to achieve cluster formation with respect to each target. Let $\mathcal{V}_{\bar{i}}$ denote the clusters where the subscript $\bar{i} \in \{1, 2, \dots, p\}$ and $\mathcal{G}_{\bar{i}}$ represents the portion of the entire graph which corresponds to the interaction among the leader robots in the \bar{i} th cluster $\mathcal{V}_{\bar{i}}$ and the target assigned to $\mathcal{V}_{\bar{i}}$. Let $n_{\bar{i}}$ be the number of leaders in cluster $\mathcal{V}_{\bar{i}}$. Without loss of generality, let $x_{\bar{i}}$ be the position vector of the target assigned to $\mathcal{V}_{\bar{i}}$. Below, we mention some assumptions required to develop the main results.

Assumption 2: The leader robots in each cluster use only the neighboring state information to achieve the cluster formation tracking.

Assumption 3: In each cluster, the interaction topology among the leaders is connected with the assigned target being the root node.

Based on Assumptions 2 and 3, the sub-Laplacian matrix L_{ll} which corresponds to the portion of \mathcal{G} involving leaders and the targets can be partitioned in the following pattern [21]:

$$L_{ll} = \begin{bmatrix} L_{11} & 0 & \cdots & 0 \\ 0 & L_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & L_{pp} \end{bmatrix}, \quad (13)$$

where $L_{\bar{i}\bar{i}}$ for $\bar{i} \in \{1, 2, \dots, p\}$ is associated with $\mathcal{G}_{\bar{i}}$. Note that L_{ll} has been extracted from the complete graph Laplacian matrix L shown in (2).

The desired formation may be time-varying or time-invariant and it is specified by the vector $h_L(t) = [h_{p+1}^T(t), h_{p+2}^T(t), \dots, h_{p+M}^T(t)]^T$ where $h_i(t) \in \mathbb{R}^n \forall t \in \mathbb{R}_{\geq 0}$ and $\forall i \in \mathcal{L}$ is a preset vector known to the i th leader. For the \bar{i} th cluster, the desired cluster formation is specified by the vector $\bar{h}_{\bar{i}}(t) = [h_{\phi_{\bar{i}+1}}^T(t), h_{\phi_{\bar{i}+2}}^T(t), \dots, h_{\phi_{\bar{i}+n_{\bar{i}}}}^T(t)]^T$ where $\phi_{\bar{i}} = \sum_{k=0}^{\bar{i}-1} n_k$ with $n_0 = p$ and each element in $\bar{h}_{\bar{i}}(t)$ is piece-wise continuously differentiable. It is easy to verify that $h_L(t) = [\bar{h}_1^T(t), \bar{h}_2^T(t), \dots, \bar{h}_p^T(t)]^T$ reflects the formation vector for the entire MRS. $\bar{x}_{\bar{i}} = [x_{\phi_{\bar{i}+1}}^T(t), x_{\phi_{\bar{i}+2}}^T(t), \dots, x_{\phi_{\bar{i}+n_{\bar{i}}}}^T(t)]^T$ represents the position vector of the leaders in the \bar{i} th cluster.

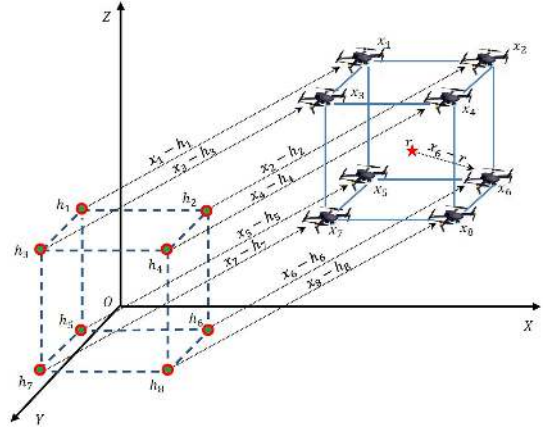


Fig. 4. An example shows a static cube formation in a 3D space with eight quadrotor UAVs tracking a fixed target.

The leader robots in the \bar{i} th cluster for all $\bar{i} \in \{1, 2, \dots, p\}$ are said to achieve cluster formation tracking (CFT) if for any given bounded initial positions

$$\lim_{t \rightarrow \infty} (\bar{x}_{\bar{i}}(t) - \bar{h}_{\bar{i}}(t) - (\mathbf{1}_{n_{\bar{i}}} \otimes I_n)x_{\bar{i}}(t)) = 0 \quad (14)$$

for all \bar{i} .

h_i is the formation configuration vector used to characterize the shape or pattern of a formation. A graphical interpretation of the formation configuration vector h_i and the geometric relationships between h_i and x_i is provided via Fig. 4. Suppose eight quadrotor UAVs are required to form a static cube formation in a 3D space along with a given target to track where $r(t)$ represents the position of the target. Fig. 4 shows two cubes which are congruent to each other – one is constituted by h_i and the other one is comprised of x_i for all $i \in \{1, 2, \dots, 8\}$. For a static formation, the relative distance vector $x_i - h_i$ should maintain constant value for all i such that the shape of the formation remains preserved. Note that if the target moves, the whole formation also moves accordingly.

Remark 5: If, at any point of time during the cluster formation tracking, two or more targets come closer and the distance between them falls below a given threshold (this situation is likely to occur in case of moving targets), then those few particular targets are considered as a whole target and consequently, the associated clusters are merged to form a single bigger cluster. In this case, all the leaders in the new cluster gradually attain a new formation and keep tracking the centroid of the targets. Note that this threshold distance (to quantify the closeness of two adjacent targets) is not at all unique. It needs to be fixed by the engineer depending on the specific task, required number of clusters, and the environmental conditions. \square

We now define the formation tracking error ξ_i for each of

the leader robots

$$\begin{aligned} \xi_i = & \sum_{j=p+1}^{p+M} a_{ij}((x_i - h_i) - (x_j - h_j)) \\ & + \sum_{k=1}^p a_{ik}((x_i - h_i) - x_k) \quad \forall i \in \mathcal{L}. \end{aligned} \quad (15)$$

Lemma 2: Let L_{ll} and L_{lt} be the components of the graph Laplacian matrix L defined in (2). Suppose Assumptions 2 and 3 hold. Then, $L_{ll}^{-1}L_{lt}$ can be expressed as

$$L_{ll}^{-1}L_{lt} = - \begin{bmatrix} \mathbf{1}_{n_1} & 0 & \cdots & 0 \\ 0 & \mathbf{1}_{n_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{1}_{n_p} \end{bmatrix}. \quad (16)$$

Proof: For the cluster $\mathcal{V}_{\bar{i}}$ for any $\bar{i} \in \{1, 2, \dots, p\}$, let the vector $e_{\bar{i}} \in \mathbb{R}^{n_{\bar{i}}}$ have $a_{(s_{\bar{i}+i})(\bar{i})}$ as its i^{th} component if the i^{th} leader in this cluster observes the target, and all the other entries are 0. Then, we have

$$\begin{aligned} L_{ll} & \begin{bmatrix} \mathbf{1}_{n_1} & 0_{n_1} & \cdots & 0_{n_1} \\ 0_{n_2} & \mathbf{1}_{n_2} & \cdots & 0_{n_2} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{n_p} & 0_{n_p} & \cdots & \mathbf{1}_{n_p} \end{bmatrix} \\ & = \begin{bmatrix} L_{11}\mathbf{1}_{n_1} & 0_{n_1} & \cdots & 0_{n_1} \\ 0_{n_2} & L_{22}\mathbf{1}_{n_2} & \cdots & 0_{n_2} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{n_p} & 0_{n_p} & \cdots & L_{pp}\mathbf{1}_{n_p} \end{bmatrix} \\ & = \begin{bmatrix} e_1 & 0_{n_1} & \cdots & 0_{n_1} \\ 0_{n_2} & e_2 & \cdots & 0_{n_2} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{n_p} & 0_{n_p} & \cdots & e_p \end{bmatrix} = -L_{lt}. \end{aligned} \quad (17)$$

Since L_{ll} is non-singular via Assumptions 2 and 3, the desired result can be drawn immediately from (17). ■

Theorem 2 establishes a decentralized and adaptive cluster formation tracking scheme for the leader robots in each cluster, which is one of the main contributions of this paper.

Theorem 2: Given a set of leader robots $\mathcal{L} = \{p+1, p+2, \dots, p+M\}$ which is distributed into p number of non-overlapping and disjoint clusters denoted by $\mathcal{V}_{\bar{i}}$ for $\bar{i} \in \{1, 2, \dots, p\}$. Let $\mu > \sigma$ for a given $\sigma \geq 0$ and also let Q and R be two positive constants. Suppose Assumptions 2 and 3 hold. Then, the leader robots in each cluster achieve cluster formation tracking with the following decentralized and adaptive CFT protocol

$$\begin{cases} u_i = (c_i + \rho_i)K\xi_i + \dot{h}_i - \mu f(\xi_i), \\ \dot{c}_i = \xi_i^T Q \xi_i, \quad c_i(0) \geq 0, \end{cases} \quad (18)$$

where $c_i(t)$ denotes the time-varying coupling weight associated with the i^{th} leader, $K = -\sqrt{\frac{Q}{R}}I_n$, $\rho_i = \sqrt{QR} \xi_i^T \xi_i$ and the nonlinear function $f(\xi_i)$ is designed as

$$f(\xi_i) = \begin{cases} \frac{\xi_i}{\|\xi_i\|} & \text{when } \|\xi_i\| \neq 0, \\ 0 & \text{when } \|\xi_i\| = 0, \end{cases} \quad (19)$$

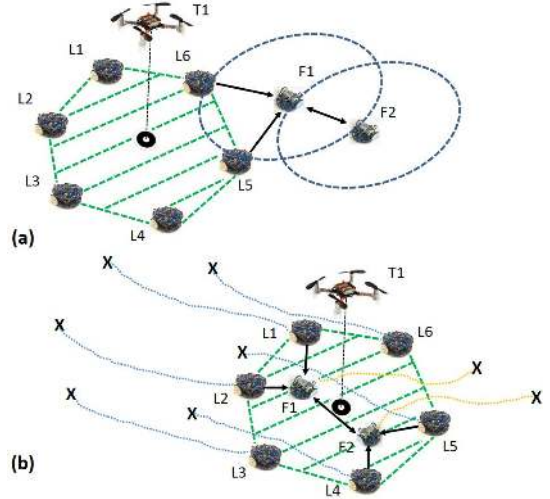


Fig. 5. A visual illustration of the formation containment activity involving six leader robots, two follower robots and a target (the quadrotor): (a) The leader robots have already achieved a hexagonal formation and keep tracking the target while the followers have entered into the sensing range of two right-most leaders; (b) The leaders detect the followers, start communicating with them and finally, make them converged into the convex hull (indicated by Green dotted lines) spanned by the positions of the leaders.

for all $i \in \mathcal{L}$.

Proof: Please see the Appendix. ■

Remark 6: The CFT scheme proposed in Theorem 2 remains applicable in applications involving non-stationary or time-varying targets. This is achieved by including the term $\mu f(\xi_i)$ in the designed control input u_i , given in (18), which counteracts the effect of time-varying target. Furthermore, the term \dot{h}_i in (18) enables the CFT scheme to achieve time-varying formation apart from the static formation since the former is more realistic in real-world applications than the latter. □

Remark 7: During the real-time implementation, some of the leader robots may stop operating due to hardware faults or communication problems. In such a scenario, the proposed CFT scheme still remains applicable and continues to work with the healthy robots assuming that they are connected to the network. Hence, the entire multi-robot system does not collapse. To deal with such issues, the CFT scheme first initiates an autonomous re-clustering process to reconfigure the topology of the MRS on the basis of the healthy robots and to redistribute them into new clusters and subsequently, the new clusters proceed to develop new formations around the given targets. Thus, in case of sudden loss of robots, the proposed framework makes the entire MRS quickly adapt to the altered operating condition and continues the mission with the available resources. The same course of action is pursued if some new leaders join the network during an ongoing mission. Hence, the proposed framework facilitates reconfigurability and scalability which are key features of the distributed control scheme. □

C. Containment control of the follower robots

Followed by the autonomous clustering process and the decentralized CFT protocol introduced in the previous two subsection, we will now present the *containment control*

scheme which is the third part of the proposed cluster formation containment control framework to be utilized in the multi-robot applications. Once the cluster formation is attained, the next task is to steer the followers to converge into the convex hull formed by those leaders. Fig. 5 illustrates this process – six leader robots has attained a hexagonal formation and two follower robots will eventually enter into the hexagonal area (marked by Green dotted lines) surrounded by the leaders. Note that at least two leaders are required to form a convex hull; otherwise, in case of a cluster having single leader, the containment problem specializes to the *leader-following* problem. Below, we declare the two technical assumptions to be satisfied by the follower robots for the containment scheme to work.

Assumption 4: Each follower has limited sensing range such that it will only receive position information from nearby leaders from one cluster or other nearby followers.

Assumption 5: For each follower, there exists at least one leader that has a directed path to it.

It is important to note here that the containment scheme to be developed does not require the followers to be clustered beforehand by the designer. Rather, the followers dynamically decide on their own which cluster to join depending on the relative distance from the leaders of the adjacent clusters. For the sake of the theoretical development of the containment scheme, it is only assumed that the followers are split into p clusters without imposing any specific pattern. However, some of the clusters may be empty, which means that no followers have actually joined those clusters. We also assume that the followers in each cluster are connected by the leaders belonging to the same cluster. Let $\hat{n}_{\bar{i}} \geq 0$ be the number of followers in the \bar{i} th cluster where $\bar{i} \in \{1, 2, \dots, p\}$. Also let $\hat{\phi}_{\bar{i}} = \sum_{k=0}^{\bar{i}-1} \hat{n}_k$ and $\hat{n}_0 = p + M$. $\hat{x}_{\bar{i}} = [x_{\hat{\phi}_{\bar{i}+1}}^T(t), x_{\hat{\phi}_{\bar{i}+2}}^T(t), \dots, x_{\hat{\phi}_{\bar{i}}+\hat{n}_{\bar{i}}}^T(t)]^T$ represents the position vector of the followers in the \bar{i} th cluster. Since the followers in different clusters do not communicate with each other, the sub-Laplacian matrices $L_{f\bar{l}}$ and $L_{f\bar{f}}$ extracted from the graph Laplacian matrix L defined in (2) enjoy the simplified structures

$$L_{f\bar{l}} = \begin{bmatrix} L_{l1} & 0 & \cdots & 0 \\ 0 & L_{l2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & L_{lp} \end{bmatrix} \quad \text{and} \quad (20)$$

$$L_{f\bar{f}} = \begin{bmatrix} L_{f1} & 0 & \cdots & 0 \\ 0 & L_{f2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & L_{fp} \end{bmatrix}. \quad (21)$$

The follower robots in each of the p clusters are said to achieve containment if for any given bounded initial positions

$$\lim_{t \rightarrow \infty} \left(\hat{x}_{\bar{i}}(t) - (-L_{f\bar{i}}^{-1} L_{\bar{l}\bar{i}} \otimes I_n) \bar{x}_{\bar{i}}(t) \right) = 0. \quad (22)$$

Let us now define the containment error for each of the

follower robots by

$$\varsigma_i = \sum_{j=p+1}^{p+N} a_{ij} (x_i - x_j) \quad \forall i \in \mathcal{F}. \quad (23)$$

A decentralized and adaptive containment control scheme for the follower robots has been proposed in Theorem 3 given below.

Theorem 3: Consider a set of followers robots \mathcal{F} that are distributed into p number of disjoint clusters and suppose Assumptions 4 and 5 hold. Let $\eta > \mu$ for a given $\mu > 0$, Q and R be two positive constants. Then, the follower robots in each cluster achieve containment via the decentralized and adaptive control law

$$\begin{cases} u_i = (\hat{c}_i + \hat{\rho}_i) K \varsigma_i - \eta f(\varsigma_i), \\ \dot{\hat{c}}_i = \varsigma_i^T Q \varsigma_i, \quad c_i(0) \geq 0, \end{cases} \quad (24)$$

where $\hat{c}_i(t)$ denotes the time-varying coupling weight associated with the i th follower, $K = -\sqrt{\frac{Q}{R}} I_n$, $\hat{\rho}_i = \sqrt{QR} \varsigma_i^T \varsigma_i$ and the nonlinear function $f(\cdot)$ is defined as:

$$f(\varsigma_i) = \begin{cases} \frac{\varsigma_i}{\|\varsigma_i\|} & \text{when } \|\varsigma_i\| \neq 0, \\ 0 & \text{when } \|\varsigma_i\| = 0, \end{cases} \quad (25)$$

for all $i \in \{p + M + 1, p + M + 2, \dots, p + M + N\}$.

Proof: Please see the Appendix. ■

Note that Theorem 3 provides the containment control technique for the follower robots but does not explicitly address the issue whether the position of a follower inside a convex hull can also be regulated. In some practical applications, for instance, in coverage control of mobile sensor networks, it is often required to maintain precise locations of the followers in a convex hull. We will now show that the proposed containment control scheme (Theorem 3) can also be used to maintain desired locations of the followers after achieving containment. In order to do that, we will invoke the lemma stated below.

Lemma 3: [40] If the directed interaction topology \mathcal{G} satisfies Assumptions 3 and 5, then all the eigenvalues of $L_{f\bar{f}}$ and $L_{l\bar{l}}$ have positive real parts, each entry of $-L_{f\bar{f}}^{-1} L_{f\bar{l}}$ is nonnegative, and each row of $-L_{f\bar{f}}^{-1} L_{f\bar{l}}$ has a sum equal to one.

Now, (64) can be rewritten as

$$\lim_{t \rightarrow \infty} \left(\hat{x}_{\bar{i}}(t) - (-L_{f\bar{i}}^{-1} L_{\bar{l}\bar{i}} \otimes I_n) \bar{h}_{\bar{i}}(t) - \mathbf{1}_{\hat{n}_{\bar{i}}} \otimes x_{\bar{i}}(t) \right) = 0 \quad (26)$$

after substituting the term $\bar{x}_{\bar{i}}(t)$ from (51), since $-L_{f\bar{i}}^{-1} L_{\bar{l}\bar{i}} \mathbf{1}_{\hat{n}_{\bar{i}}} = \mathbf{1}_{\hat{n}_{\bar{i}}}$ holds from Lemma 3. The derived condition in (26) indicates that the positions of the followers are jointly determined by the properties of the communication graph $(L_{f\bar{i}}, L_{\bar{l}\bar{i}})$, the formation of the leaders $(\bar{h}_{\bar{i}}(t))$ and the movement of the target $(x_{\bar{i}}(t))$. Therefore, the locations of the followers can be controlled by modifying any of the above three factors. However, in practice, $x_{\bar{i}}(t)$ is not user defined and $\bar{h}_{\bar{i}}(t)$ depends on the application. Thus, most conveniently, this tuning can be done by modifying the graph properties, i.e., by varying the weights of the links.

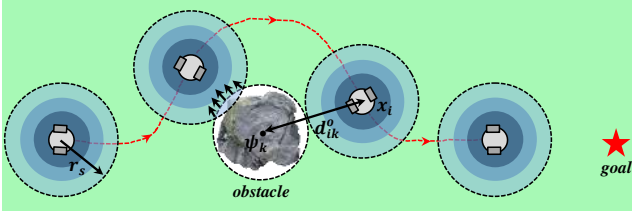


Fig. 6. A scenario shows the obstacle avoidance via the DAPF technique. The black dotted circles in each position around the robot indicate the safe circle having sensing radius r_s . If an obstacle enters even partially into the safe circle, a repulsive force is generated to move away the robot from the obstacle.

D. Collision and obstacle avoidance

When operating a multi-robot system in an unpredictable or highly cluttered environment, all robots are required to navigate through a safe path avoiding collisions with other robots and/or obstacles so that the mission can be accomplished successfully. In order to implement obstacle and collision avoidance, we will now apply a Decentralized Artificial Potential Field (DAPF) method for all the robots.

In this approach, each robot is assumed to be surrounded by a virtual circle, called ‘safe circle’, to mark an area or zone of obstruction (see Fig. 6). The radius of the safe circle r_s is equal to the distance between the centre point and the furthest point of the observed obstacle boundary. When the safe zone of a robot is invaded by obstacles or neighboring robots, a repulsive force is generated to push that robot into a safe place away from the colliding robots or the obstacles.

Assume there are q obstacles in the operating environment of MRS, let $\psi_k \forall k \in \{1, \dots, q\}$ be the position of the k^{th} obstacle. The safe distance between the centres of two adjacent robots, say the i^{th} robot and the j^{th} robot, is denoted by d_{ij}^c , while the safe distance between the i^{th} robot and the k^{th} obstacle is given by d_{ik}^o . In this work, an obstacle is assumed to be a circle in \mathbb{R}^2 , or a sphere in \mathbb{R}^3 , with radius smaller than d_{ik}^o .

Motivated by [41], we include the following collision avoidance protocols to design additional input components for each robot to implement collision avoidance:

$$u_{c,i} = \sum_{j=1}^N \theta_c \tau_{ij} a_{ij} \left(\frac{1}{\|x_i - x_j\|} - \frac{1}{d_{ij}^c} \right) \frac{x_i - x_j}{\|x_i - x_j\|^2}, \quad (27)$$

$$u_{o,i} = \sum_{k=1}^q \theta_o \phi_{ik} \left(\frac{1}{\|x_i - \psi_k\|} - \frac{1}{d_{ik}^o} \right) \frac{x_i - \psi_k}{\|x_i - \psi_k\|^2}, \quad (28)$$

where θ_c and θ_o are positive constants which determine the magnitude of the repulsive force generated to repel a robot from another robot or an obstacle in the vicinity, and the factors τ_{ij} and ϕ_{ik} are defined as follow:

$$\tau_{ij} = \begin{cases} 1 & \forall \|x_i - x_j\| \leq d_{ij}^c, \\ 0 & \forall \|x_i - x_j\| > d_{ij}^c, \end{cases} \quad (29)$$

$$\phi_{ik} = \begin{cases} 1 & \forall \|x_i - \psi_k\| \leq d_{ik}^o, \\ 0 & \forall \|x_i - \psi_k\| > d_{ik}^o. \end{cases} \quad (30)$$

In order to produce sufficiently strong repulsive force to repel the i^{th} robot from all the adjacent robots and the obstacles (if any) in the surrounding region, θ_c and θ_o must be set appropriately to make $u_{c,i}$ and $u_{o,i}$ stronger than the formation tracking input (18) and containment control input (24).

Remark 8: The proposed collision and obstacle avoidance policy (27)–(28) (built on the DAPF methodology) can effectively deal with convex obstacles. However, in the case of non-convex or scattered obstacles in a cluttered environment, the above policy is not guaranteed to work well. In such cases, we can still find a remedy by increasing the radius of the safe circle around the robots and by grouping the non-convex obstacles (zone wise) within a virtual circle so that the robots can safely keep away from those obstacles, even though it may result in a conservative result. Note that more powerful and state-of-the-art obstacle avoidance algorithms can also be integrated with the proposed CFC framework to handle non-convex obstacles and to guarantee safe navigation in highly cluttered environments. In that scenario, the low-level obstacle avoidance algorithms implemented in the robots may be different, but the proposed CFC framework used as a high-level coordination strategy will remain the same. \square

Towards this end, we consolidate the aforementioned formation tracking and containment control techniques into an algorithm (Algorithm 2) which also provides a systematic set of guidelines for the robotics-control practitioners to implement the formation-containment methodologies involving the obstacle and collision avoidance scheme.

Algorithm 2 A structured set of guidelines to apply the two-layer formation containment control scheme for multi-robot applications

- 1: **for** each robot $i \in \{1, \dots, N\}$ **do**
 - 2: **if** Nash stable partition is guaranteed by the autonomous clustering algorithm **then**
 - 3: set the desired formation configuration vector h_i for leaders
 - 4: set the communication graph and link-weights a_{ij}
 - 5: $K \leftarrow -\sqrt{Q}/RI_n$
 - 6: $\hat{\rho}_i \leftarrow \sqrt{QR} \zeta_i^T \zeta_i$; $\rho_i \leftarrow \sqrt{QR} \xi_i^T \xi_i$
 - 7: choose positive gains $\eta > \mu > \sigma$ for a given $\sigma \geq 0$
 - 8: **if** $\|\chi_i\| \neq 0$ (χ_i represents either ξ_i or ζ_i) **then**
 - 9: $f(\chi_i) \leftarrow \frac{\chi_i}{\|\chi_i\|}$
 - 10: **end if**
 - 11: construct the decentralized and adaptive formation containment control law u_i given in (18) and (24)
 - 12: **if** $\|x_i - x_j\| \leq d_{ij}^c$ **then**
 - 13: $\tau_{ij} \leftarrow 1$
 - 14: add the term $u_{c,i}$ to the input u_i to each robot
 - 15: **end if**
 - 16: **if** $\|x_i - \psi_k\| \leq d_{ik}^o$ **then**
 - 17: $\phi_{ik} \leftarrow 1$
 - 18: add the term $u_{o,i}$ to the input u_i to each robot
 - 19: **end if**
 - 20: **end if**
 - 21: **end for**
-

Remark 9: In this paper, we have introduced a unified CFC framework for multi-robot applications, which integrates an autonomous clustering process, a cluster formation tracking scheme and a containment control technique. It is interesting to observe that for $p = 1$ (i.e., in case of a single cluster), the proposed cluster, formation containment scheme reduces to the ‘formation tracking’ and ‘containment control’ problems as done in [19], [20], [29], [30], [42]; for $N = M$, the present scheme specializes to the ‘cluster navigation’ problem as discussed in [28], [43]–[45]; the CFC framework also specialises to the ‘multi-robot rendezvous’ problem, as discussed in [46]–[48], when the leader’s formation configuration vector $h_i = 0 \forall i \in \mathcal{L}$. Thus, the CFC framework proposed in this paper generalises all the above works and can be utilized as a decentralized control scheme for solving many well-known cooperative control problems such as rendezvous problems, formation tracking problems, cluster consensus problems, containment problems, etc. In addition to that, the CFC framework proposed in this paper also invokes the re-clustering process autonomously when some robots stop operation due to hardware faults or loose connection with the network due to link failure or when two or more targets come close apart. \square

IV. CASE STUDY: A REAL-TIME SEARCH AND RESCUE MISSION

A. Objectives

In this section, a real-time search and rescue operation is considered to highlight a potential application of the proposed CFC framework and also to demonstrate the feasibility of the framework in achieving the mission in real robot experiments. In the case study, a fleet of networked autonomous mobile robots is used consisting of leader, follower and target (or reference) robots. In order to achieve the search and rescue mission via the proposed CFC framework, we assign particular roles to the leaders, followers and targets: (i) *followers* are to be viewed as *lost* robots; (ii) *targets* are assigned the task of exploring an unknown area in search of the lost robots and hence, targets are recognised as *explorers*; and (iii) *leaders* guide the lost robots into a safe region enclosed by the leaders after a lost robot is detected by an explorer and hence, leaders are recognised as *rescuers*. The mission is to deploy a fleet of rescuers and explorers to search for the lost robots in an unknown environment and to bring them back into a safe region surrounded by the rescuers. The case study may be considered as a heterogeneous MRS since the robots (even though structurally homogeneous) are assigned different roles. The CFC framework can also be applied to multi-dimensional search and rescue operations for which proper synergy between the aerial robots and the ground vehicles needs to be established. In the present scenario, we provide a complete set of simulation results showing clustering, formation tracking and containment phases leading to achieve the search and rescue mission. Simulation results also focus on collision and/or obstacle avoidance capabilities of the proposed CFC framework and its robustness against loss of some rescuer robots during an ongoing mission due to hardware/communication faults.

B. Simulation results

The simulation has been performed using the Webots platform, a GUI-based, open-source, multi-platform mobile robot simulation software. Webots facilitates advanced robot models which are useful for academic and industrial research; provides commonly used sensor and actuator components for robotics applications; and also enables the users to incorporate practical constraints during the simulation. In this case study, the ‘E-puck’ mobile robot [49] is used to demonstrate the effectiveness of the proposed CFC framework. Simulation results are given in Fig. 7(a)–Fig. 7(f). In Fig. 7(a), initial positions of all the robots – five rescuers, one lost robot and one explorer (considered as virtual target in this case) and the obstacles are shown. In this simplified task, the virtual target transmits its own relative location to one of the rescuers directly via wireless communication and all the five rescuers are connected by a communication network. Fig. 7(b) depicts that the rescuers are approaching the virtual target (marked by Red star) and en route, they have encountered the obstacles. Fig. 7(c) reveals that the rescuers bypassed the obstacles and have almost clustered around the virtual target. Fig. 7(d) indicates that the team of rescuers have already attained a pentagonal formation surrounding the virtual target and furthermore, the lost robot is getting drawn towards the formation (i.e. the containment phase has already started). Fig. 7(e) confirms that the lost robot has been enclosed within the pentagonal area (i.e. the convex hull) formed by the team of rescuers and the whole formation is navigating the lost robot back towards the left side of the arena. However, meanwhile, two rescuers (marked by White crosses) suddenly stopped operating due to a hardware fault. Finally, Fig. 7(f) portrays that the remaining three healthy robots have attained a new triangular formation surrounding the virtual target and the lost robot and they are guiding them towards the left side of the arena despite the loss of two rescuers. Hence, the search and rescue mission is accomplished.

Now, in order to test the resilience of the designed formation tracking controller against communication delays, network topology changes and variation in control parameters (Q and R), we increase the number of rescuers to 10 and repeat the same simulation case study. 50 trials are tested for each case corresponding to 50 different sets of initial positions of the rescuers. Some analysis with respect to topology switching and variation in controller parameters are also done based on the simulation data sets which reflects the effectiveness of the proposed controller in a non-ideal environment.

Impact of the communication delay: Fig. 8 shows a cumulative plot of the time variation of the 2-norm of the formation tracking error $\|\xi_L(t)\|$ of the rescuers for all $t \in [0, 50]$ s in the presence of communication delays (0 s, 1 s, 2 s and 3 s). Each subplot depicts a coloured zone and a solid curve where the coloured zone contains all fifty $\|\xi_L(t)\|$ vs. t graphs obtained from fifty sets of simulation responses corresponding to fifty different sets of initial positions of the rescuers and the solid curve represents the arithmetic mean curve. As observed from Fig. 8, the best result is achieved when there is no delay, as expected; the proposed control scheme works fine

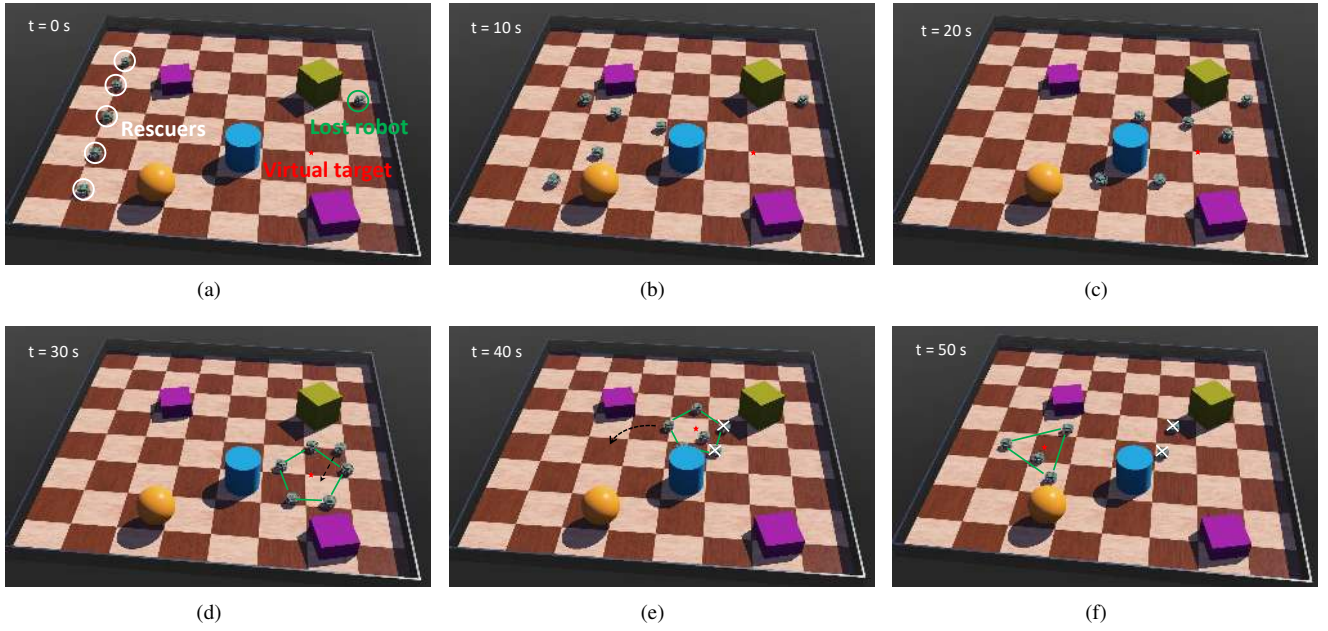


Fig. 7. Webots simulation results of the search and rescue mission involving 5 rescuers, 1 explorer (virtual target, marked by Red star) and 1 lost robot: (a) At $t = 0$ s, initial positions of all the robots including the virtual target in the X-Y plane; (b) At $t = 10$ s, rescuers are heading towards the virtual target; (c) At $t = 20$ s, rescuers are being clustered around the virtual target after successfully avoiding the obstacles; (d) At $t = 30$ s, a pentagon formation is attained by the rescuers surrounding the virtual target and the lost robot has established communication with two rescuers; (e) At $t = 40$ s, the lost robot is already contained within the pentagonal area formed by the team of rescuers and the rescuer team is guiding the lost robot to its home – meanwhile two rescuers (marked by White crosses) stopped working due to sudden hardware fault; (f) At $t = 50$ s, the remaining 3 rescuers have rebuilt a new triangular formation surrounding the virtual target and containing the lost robot within it and finally reached the initial spot – mission accomplished.

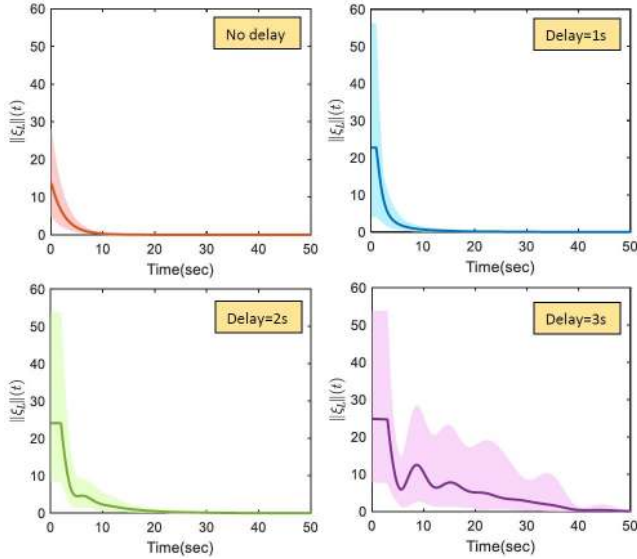


Fig. 8. Variation of 2-norm of the formation tracking error $\|\xi_L(t)\| \forall t \in [0, 50]$ s of all 10 leader robots during the search and rescue mission in presence of different communication delays, subjected to 50 random initial positions. Lines indicate the median of the errors and shaded areas indicate the minimum and maximum of the errors obtained.

(the formation tracking error $\|\xi_L(t)\|$ smoothly decays to zero within 15 s) as long as the delay remains within 2 s; but starts deteriorating significantly when the delay increases to 3 s or beyond.

Effect of the network topologies: The edge connectivity of a graph reflects its structure and complexity [50]. The

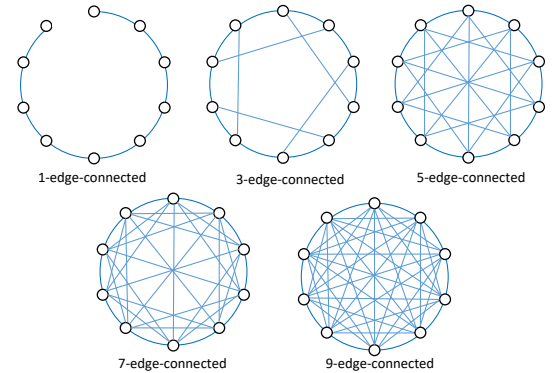


Fig. 9. Five different communication topologies have been chosen in the simulation case study for making analysis of the controller performance.

concept is illustrated through Fig. 9 which shows five different configurations (corresponding to 1-edge-connected, 3-edge-connected, 5-edge-connected, 7-edge-connected and 9-edge-connected) of a graph containing 10 nodes. Note that when the edge connectivity is 1, it represents the simplest communication graph (e.g. line following or star topology) that ensures cooperation among all robots. When the edge connectivity is $k - 1$, where k is the number of nodes of a graph, it signifies a ‘fully connected’ graph.

Here, the controller performance subjected to network topology changes as shown in Fig. 9 is presented. For each case, the results are collected from 50 sets of simulation responses corresponding to 50 different sets of randomly selected initial positions of the rescuers as mentioned before. Fig. 10(a) shows

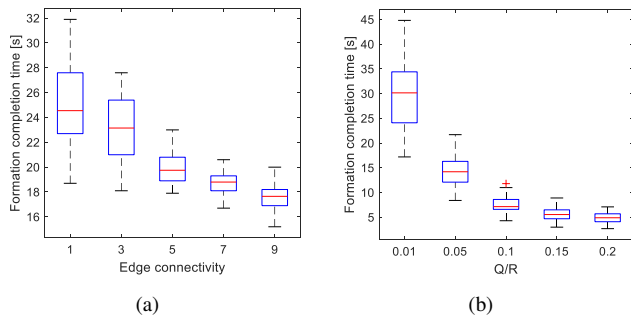


Fig. 10. Formation tracking accomplishment time of the leader robots with respect to (a) five different network topologies (as shown in Fig. 9) and (b) five different Q/R ratios (0.01, 0.05, 0.1, 0.15 and 0.2) subjected to 50 different sets of initial positions of the leaders.

the formation tracking settling time (i.e. the time required for $\|\xi_L(t)\|$ to decay to zero) of the rescuers corresponding to the given five configurations having edge connectivities 1, 3, 5, 7 and 9 respectively. The figure suggests that the edge connectivity has a direct impact on the decay rate of the formation tracking error. From this investigation, we can also conclude that as the edge connectivity increases (that is, the more complex the topology is), the multi-robot system will be more robust to communication failure. However, it will increase the bandwidth requirement and the communication costs. Hence, a trade-off between the controller performance and the complexity of the network should be determined based on the applications and available resources.

Effect of varying Q/R ratio: The control parameters $Q > 0$ and $R > 0$ used in the distributed formation control protocol (see Theorem 2) also have a significant impact on the dynamic response of the robots. In order to get a faster dynamic response, a larger Q and a smaller R should be chosen in the controller design. However, depending on the physical limit of the control input for each robot, the difference between Q and R may not be arbitrarily large and should be properly selected depending on performance specifications.

Fig. 10(b) shows the formation tracking settling time of the rescuers with respect to varying Q/R ratios (both Q and R are positive scalar parameters according to Theorem 2) and subjected to 50 different sets of initial positions as considered in the previous analysis. The figure indicates that the tracking time decreases exponentially as the Q/R ratio increases and the decrement becomes insignificant when the Q/R ratio goes beyond 0.15 due to actuator constraints.

C. Experimental set-up with real robots

1) *Robotic platform:* Mona robots [51] are used in the hardware experiment on a two-dimensional search and rescue problem. Fig. 11 shows a Mona robot and its various components. It is a small-size (with diameter of 8 cm) mobile robot which was developed as an open-source robotic platform.

The robot has two DC motors with direct reduction gears as its actuator. These motors are controlled independently via two pulse-width modulation (PWM) channels of the main microcontroller, which generate control signals for the left and

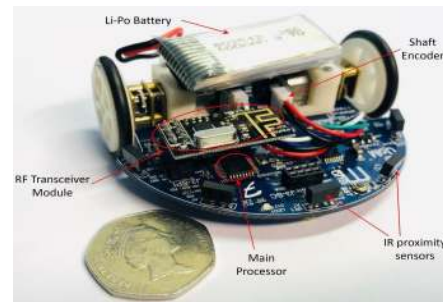


Fig. 11. Mona is an open-source miniature mobile robot suitable for multi-robot applications [51].

right motors simultaneously. The motors actuate Mona with relatively slow speed in the range from 1.5 cm/s to 10 cm/s.

There are three communication channels (I2C, RS232, and SPI) which are used to communicate with the base-station and other robots. Serial Peripheral Interface (SPI) is used to communicate with the RF (radio frequency) module which is attached to each robot as an additional part. Although this module may not guarantee a fast and reliable communication channel, its performance is adequate for this work which requires transferring short-size messages containing only speed of the left and right motors. Fig. 12 illustrates the control loop implemented in the experiment.

2) *Arena set-up:* To investigate the feasibility and efficacy of the proposed CFC framework in the hardware experiments with real robots, an experimental set-up was designed which includes the following three parts:

- *Localization system:* This is used to capture the position information of the robots and to generate relative positions with respect to the neighboring robots. An open-source localization system [5] has been used to detect the positions and orientations of the robots using coloured circular tags attached to the robots. The localization system uses a low-cost overhead camera to detect the circular tags via identifying the colours. The estimated positions and orientations of all robots are sent to the formation controller via the ROS (Robot Operating System) communication framework.
- *Communication system:* This uses a modem (transceiver) to modulate control commands generated by the controller and sends those messages to each individual robot using its RF transceiver. Therefore, it is a communication link between the base-station and the robots.
- *Base-station:* This is a laptop which manages localization software and runs the controller in MATLAB. The internal link between these two software is managed by ROS. As it was mentioned before, this laptop only routes the inter-robot communication and hence it can be removed from the system if each robot can detect the relative positions of its neighbors and orientations using on-board range & bearing sensors, which will be implemented and tested in future work.

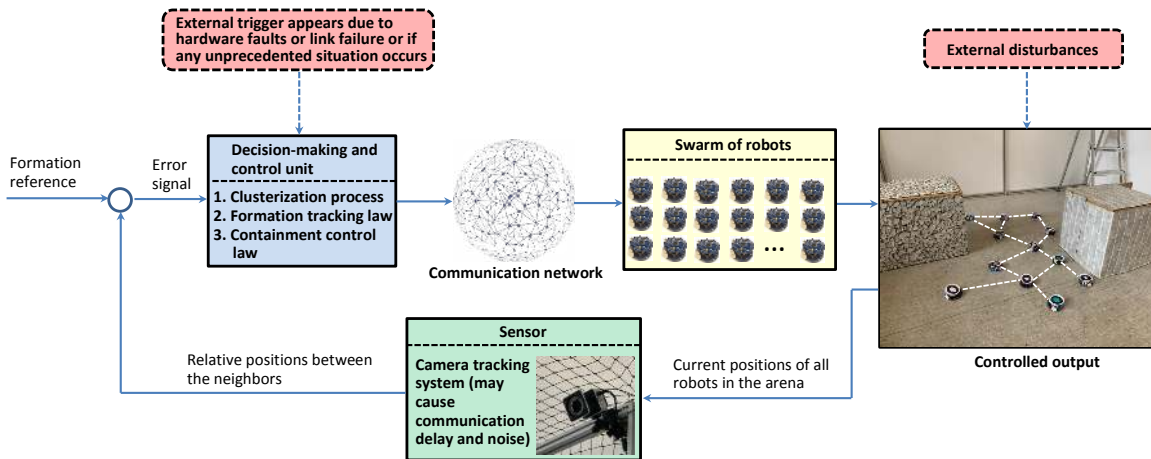


Fig. 12. Hardware control loop associated with the experiment.

D. Results of experiments with real robots

In the experiment, a group of twelve Mona robots which contains two explorers (Red centred), three lost robots (Green centred) and seven rescuers (White centred) is placed in a two dimensional arena. An obstacle (Gray coloured box) is also placed in the arena (see Figures 13(a)–13(h)). The designed sequence of actions can be described as: first, the rescuers will be separated into two clusters depending on the positions and priorities of the explorers; then, the rescuers in each clusters will form specific formations around the explorers and will track the positions of the explorers; subsequently, the explorers will start searching for the lost robots in the arena. Once an explorer locates a lost robot, it informs the rescuers clustered around that explorer. The rescuers then try to communicate with the lost robots to navigate them within a safe region enclosed by the rescuers.

Fig. 13(a) shows the initial orientation of the entire multi-robot system. The lost robots are placed close to the left and right corners of the arena. Fig. 13(b) depicts the first step, that is, all seven rescuers are distributed into two non-overlapping and distinct clusters with respect to the positions and priorities of the explorers. Once the clusters are formed, the rescuers in each cluster align them in a specific geometric pattern called *formation*. Fig. 13(c) reveals that both the clusters have achieved the respective formations - a triangular formation containing three rescuers and a square formation with four rescuers. In Figures 13(c), 13(d), 13(e), 13(f) and 13(h), the green planar regions mark the safe region (convex hull) spanned by the positions of the rescuers of the individual clusters. After the cluster-formation is attained in Fig. 13(c), the explorers start searching in Fig. 13(d) for the lost robots in the arena using a standard cooperative search algorithm [52]. During this phase, as an explorer keeps roaming in the arena, the associated rescuers (i.e., the whole formation) also keep moving to track the explorer. Fig. 13(d) reflects this phenomena that the clusters have moved away from each other compared to their spatial positions in Fig. 13(c). If an explorer comes close to a lost robot (usually measured in terms of ‘sensing radius’) during the course of searching such that the

explorer can detect the presence of a lost robot, the rescuers associated with that explorer build communication with the lost robot. Fig. 13(d) portrays this situation where both the explorers are approaching towards the lost robots (indicated by the Red dashed arrow) and enter into the respective zone of detection. As soon as the rescuers contact the lost robots, the latter start moving towards the Green safe region spanned by the rescuers. This phase is known as *containment*. Fig. 13(e) illustrates that all three lost robots are made to converge into the safe zones - two of them are encompassed by the square formation while the third one is surrounded by the triangular formation. Note that obstacle and collision avoidance have been implemented throughout the entire experiment. Even though the lost robots are all rescued, they still need to be protected and navigated further towards a base, which is assumed to be the middle of the arena. While doing so, the two formations come close to each other as the search space is common to both robot teams. Fig. 13(f) indicates such a situation when one cluster enters into the surrounding region of the other and the distance between the explorers corresponding to those two clusters is less than a threshold value. In this particular case, according to the proposed clustering algorithm, the existing formations are first dismantled (see Fig. 13(g)) and then, merged into a single larger cluster. Subsequently, the new larger cluster will achieve a complete formation comprised of all seven rescuers and encompass the lost robots while keeps tracking the centroid of the positions of two explorers. Fig. 13(h) shows that finally a single heptagonal formation is attained by the new larger cluster with two explorers viewed as a combined target and the lost robots contained within the safe region spanned by the seven rescuers. We have also plotted the 2-norms of the formation tracking error of rescuers and the containment error of lost robots evolved during the real robot experiment to show that the errors decay quickly to zero due to the proposed CFC framework.

E. Discussion

In this paper, we conducted a lab-based experiment involving affordable small-scale mobile robots (Mona), which relies

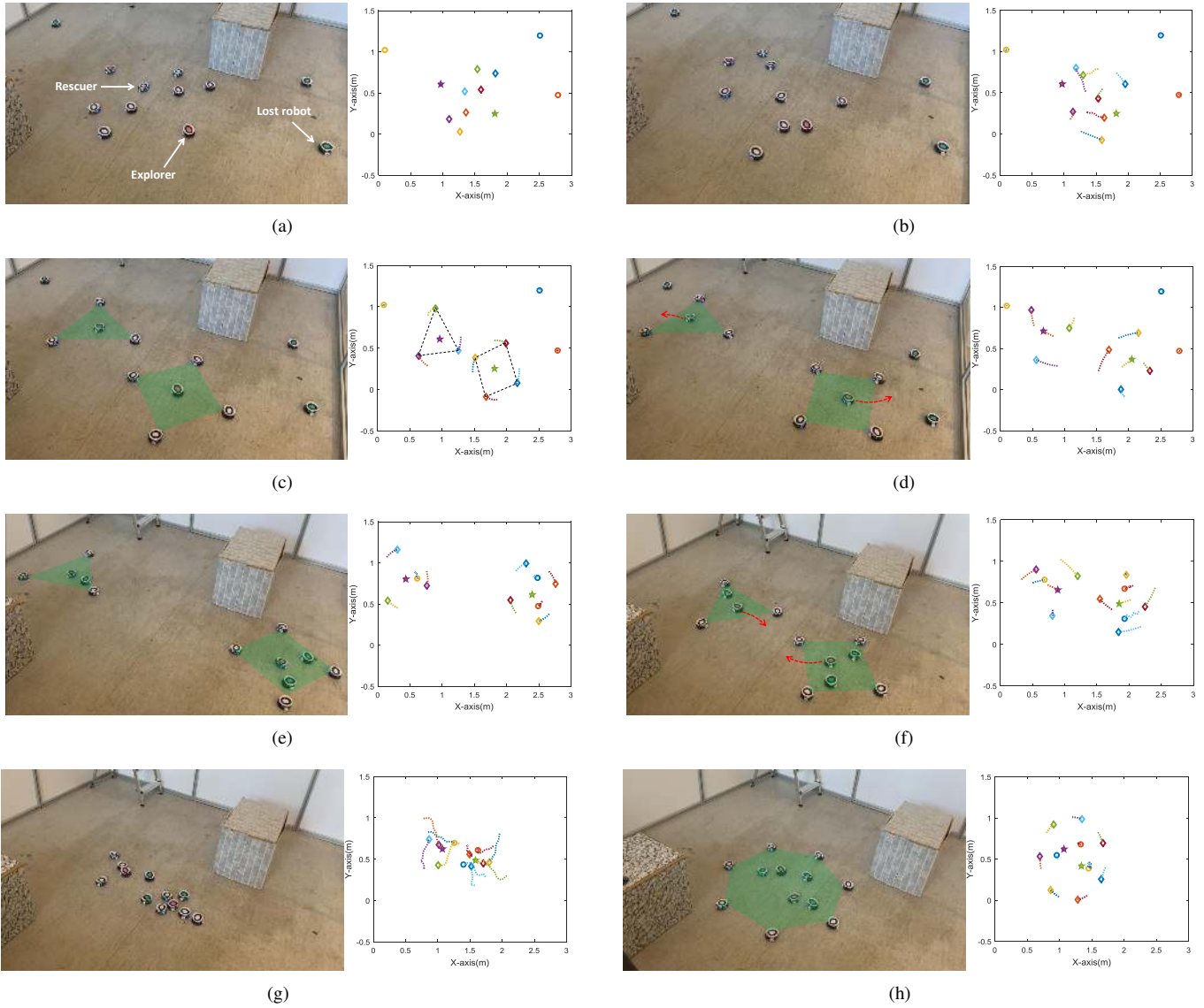


Fig. 13. Experimental observation of the search and rescue mission involving twelve robots (two Red robots act as explorers, three Green robots represent the lost robots and seven White robots being the rescuers); In all the simulated pictures complementing the experimental snapshots, the ‘stars’ denote the explorers, the ‘diamonds’ represent the rescuers and the ‘circles’ denote the lost robots: (a) haphazard position of the robots at the beginning of the mission, (b) rescuers (White robots) are distributed into clusters with respect to the positions of the explorers (Red robots), (c) two cluster-formations attained, (d) searching for the lost robots (Green robots) in the arena, (e) lost robots found and rescued - i.e., containment achieved, (f) explorers approach each other, (g) existing two clusters are dismantled to form a single large cluster, and (h) the new large cluster is formed with a heptagonal formation surrounding all three lost robots and it also keeps tracking the centroid of all explorers.

on a camera tracking system and a base-station for navigation and control of the robots. Although in this set-up, the robots are not equipped with onboard position and orientation sensors for estimating the relative range and bearing of the neighboring robots, the experiment still validates the feasibility of the proposed CFC scheme because the host computer is used only as an intermediary for processing the feedback signal from the camera and doing the mathematical computation to construct the control protocols based on only relative position information. The experimental results suggest that the proposed scheme fulfils the desired objectives in the presence of communication delays, actuator noises, unaccounted factors (e.g. static and rolling frictions, nonlinearities of the robots) and other limitations (e.g. inaccuracies of the sensors and the

camera tracking system). We believe that these results will motivate future development of more realistic multi-robot test rigs for improving the current experimental performance.

In the next phase of our work, we plan to develop more realistic and fully distributed experimental set-up including advanced mobile robots equipped with onboard position and proximity sensors. We will also test controller resilience to more challenging scenarios (e.g. non-convex obstacle avoidance, dynamic priority changes of the targets) and robustness to sudden loss of robots and network failure. Future research will also be pursued to develop an effective swarm intelligence algorithm to fit the proposed CFC framework in applications related to agri-robotics, safety-critical systems and disaster-recovery.

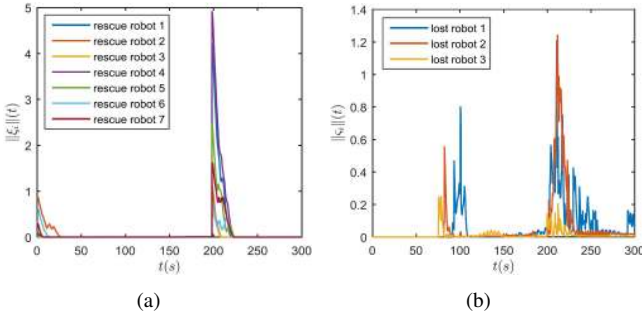


Fig. 14. Time evolution of 2-norms of (a) formation error of the rescuers (i.e., the leader robots) during the formation tracking phase and (b) containment error of the lost robots (i.e., the followers) during the containment phase.

V. CONCLUSIONS

Cooperative control of MRS is an active area of robotics research because it promises to solve many real-world engineering problems especially in precision agriculture (e.g. automated weed monitoring and control), in military applications (e.g. surveillance, target localization and tracking, search and rescue in extreme environments), etc. Coordination of MRS presents challenges on all aspects including platform sensing and actuation, control algorithmic challenges and mission formulation. This paper proposes a hedonic game-based decentralized cluster, formation containment (CFC) methodology, which can be applied as an appropriate control framework to deal with multi-robot applications. An autonomous clustering algorithm has been developed exploiting the idea of Nash-stable partitioning to distribute a group of leaders into several clusters depending on the positions and priorities of the individual targets. The paper considers an exhaustive case study on a real-time search and rescue operation in which the problem is first modelled as a two-layer multi-robot coordination problem. Then, the proposed CFC framework has been applied to achieve the desired goal. The simulation study addresses non-trivial issues like collision and obstacle avoidance, the impact of communication delays and network topology changes on the performance of the controller, and robustness of the controller to faulty conditions (e.g. loss of robots). A hardware experiment involving a group of networked miniature mobile robots has been performed to validate the feasibility of the proposed CFC framework.

APPENDIX

A. Proof of Theorem 2

Proof: Since the communication topology of each cluster is connected, it is shown in [16] that all the eigenvalues of $L_{\bar{i}\bar{i}}$ have positive real part for all $\bar{i} \in \{1, \dots, p\}$. Applying Lemma 1, there exists a set of real diagonal matrices $\Xi_{\bar{i}} > 0$ such that

$$\Xi_{\bar{i}} L_{\bar{i}\bar{i}} + L_{\bar{i}\bar{i}}^T \Xi_{\bar{i}} > 0 \quad \forall \bar{i} \in \{1, 2, \dots, p\}. \quad (31)$$

Define a block diagonal matrix $\Xi = \text{diag}\{\Xi_1, \dots, \Xi_p\}$. It is evident that $\Xi L_{ll} + L_{ll}^T \Xi > 0$ via (31) since Ξ and L_{ll} are block diagonal matrices.

Define the following vectors which will be used in this proof $x_L = [x_{p+1}^T, \dots, x_{p+M}^T]^T$, $x_T = [x_1^T, \dots, x_p^T]^T$, $u_T = [u_1^T, \dots, u_p^T]^T$, $\dot{h}_L = [\dot{h}_{p+1}^T, \dot{h}_{p+2}^T, \dots, \dot{h}_{p+M}^T]^T$ and $F(\xi) = [f^T(\xi_{p+1}), f^T(\xi_{p+2}), \dots, f^T(\xi_{p+M})]^T$. The closed-loop system dynamics of the leaders embedded with the CFT protocol (18) can be expressed in a structured form as

$$\begin{cases} \dot{x}_L = ((C + \rho)L_{ll} \otimes K)x_L + ((C + \rho)L_{lt} \otimes K)x_T \\ \quad + (I_M \otimes I_n)\dot{h}_L - ((C + \rho)L_{ll} \otimes K)h_F \\ \quad - \mu(I_M \otimes I_n)F(\xi), \\ \dot{x}_T = (I_p \otimes I_n)u_T, \end{cases} \quad (32)$$

where $C = \text{diag}\{c_{p+1}, c_{p+2}, \dots, c_{p+M}\}$ and $\rho = \text{diag}\{\rho_{p+1}, \rho_{p+2}, \dots, \rho_{p+M}\}$.

Let the global formation tracking error be $\xi_L = [\xi_{p+1}^T, \dots, \xi_{p+M}^T]^T$. Define $z_i = x_i - h_i \forall i \in \mathcal{L}$ and $z_L = [z_{p+1}^T, \dots, z_{p+M}^T]^T$. Now ξ_L can be written in a compact form as

$$\xi_L = (L_{ll} \otimes I_n)z_L + (L_{lt} \otimes I_n)x_T. \quad (33)$$

Substituting (32) into the derivative of (33), the expressions for $\dot{\xi}_L$ and \dot{c}_i can be found as

$$\begin{cases} \dot{\xi}_L = (L_{ll}(C + \rho) \otimes K)\xi_L + (L_{lt} \otimes I_n)u_E \\ \quad - \mu(L_{ll} \otimes I_n)F(\xi), \\ \dot{c}_i = \xi_i^T Q \xi_i. \end{cases} \quad (34)$$

Motivated by [47], we consider the following Lyapunov function candidate

$$V_1 = \sum_{i=p+1}^{p+M} \frac{1}{2} \varphi_i (2c_i + \rho_i) \rho_i + \frac{1}{2} \sum_{i=p+1}^{p+M} \varphi_i (c_i - \alpha)^2 \quad (35)$$

where $\Xi = \text{diag}\{\varphi_{p+1}, \dots, \varphi_{p+M}\}$ is a positive definite matrix with $\varphi_i \in \mathbb{R}_{>0} \forall i \in \mathcal{L}$ such that $\Xi L_{ll} + L_{ll}^T \Xi > 0$ and α is a positive constant to be determined later. As $c_i(0) > 0$ for all $i \in \mathcal{L}$, it follows from $\dot{c}_i(t) \geq 0$ that $c_i(t) > 0$ for all $t > 0$. Therefore, V_1 is positive definite.

Now, the time derivative of V_1 along any trajectory of (34) is given by

$$\begin{aligned} \dot{V}_1 &= \sum_{i=p+1}^{p+M} (\varphi_i (c_i + \rho_i) \dot{\rho}_i + \varphi_i \rho_i \dot{c}_i) + \sum_{i=p+1}^{p+M} \varphi_i (c_i - \alpha) \dot{c}_i \\ &= \sqrt{QR} \sum_{i=p+1}^{p+M} 2\varphi_i (c_i + \rho_i) \xi_i^T \dot{\xi}_i + \sum_{i=p+1}^{p+M} \varphi_i (\rho_i + c_i - \alpha) \dot{c}_i. \end{aligned} \quad (36)$$

Note that,

$$\sum_{i=p+1}^{p+M} \varphi_i (\rho_i + c_i - \alpha) \dot{c}_i = Q \xi_L^T ((C + \rho - \alpha I) \Xi \otimes I_n) \xi_L \quad (37)$$

and

$$\begin{aligned}
& \sum_{i=p+1}^{p+M} 2\varphi_i(c_i + \rho_i)\xi_i^T \sqrt{QR}\dot{\xi}_i \\
&= 2\sqrt{QR}\xi_L^T((C + \rho)\Xi \otimes I_n)\dot{\xi}_L \\
&= -Q\xi_L^T((C + \rho)(\Xi L_{lu} + L_{lu}^T\Xi)(C + \rho) \otimes I_n)\xi_L \\
&\quad + 2\sqrt{QR}\xi_L^T((C + \rho)\Xi L_{lt} \otimes I_n)u_T \\
&\quad - 2\mu\sqrt{QR}\xi_L^T((C + \rho)\Xi L_{lu} \otimes I_n)F(\xi) \\
&\leq -\xi_L^T(\lambda_1^{\min}(C + \rho)^2 \otimes QI_n)\xi_L \\
&\quad + 2\sqrt{QR}\xi_F^T((C + \rho)\Xi L_{lt} \otimes I_n)u_T \\
&\quad - 2\mu\sqrt{QR}\xi_L^T((C + \rho)\Xi L_{lu} \otimes I_n)F(\xi), \quad (38)
\end{aligned}$$

where λ_1^{\min} represents the smallest positive eigenvalue of $[\Xi L_{lu} + L_{lu}^T\Xi]$.

Substituting (37) and (38) into (36), we obtain

$$\begin{aligned}
\dot{V}_1 &\leq Q\xi_L^T((C + \rho)\Xi \otimes I_n \\
&\quad - (\lambda_1^{\min}(C + \rho)^2 + \alpha\Xi) \otimes I_n)\xi_L \\
&\quad + 2\sqrt{QR}\xi_L^T((C + \rho)\Xi L_{lt} \otimes I_n)u_T \\
&\quad - 2\mu\sqrt{QR}\xi_L^T((C + \rho)\Xi L_{lu} \otimes I_n)F(\xi). \quad (39)
\end{aligned}$$

From (19), it is straightforward to show that

$$\sqrt{QR}\xi_i^T f(\xi_i) = \sqrt{QR}\|\xi_i\| \quad \forall i \in \mathcal{L} \quad (40)$$

and by applying the Cauchy-Schwarz inequality [33], we get

$$\begin{aligned}
\sqrt{QR}\xi_i^T f(\xi_j) &\leq \sqrt{QR}\|\xi_i\| \|f(\xi_j)\| \\
&\leq \sqrt{QR}\|\xi_i\| \quad \forall i \neq j. \quad (41)
\end{aligned}$$

Subsequently, we get the following relation

$$\begin{aligned}
& -2\mu\sqrt{QR}\xi_L((C + \rho)\Xi L_{lu} \otimes I_n)F(\xi) \\
&= -2\mu\sqrt{QR}\sum_{i=p+1}^{p+M}(c_i + \rho_i)\varphi_i\xi_i^T \sum_{j=p+1}^{p+M} a_{ij}(f(\xi_i) - f(\xi_j)) \\
&\quad - 2\mu\sqrt{QR}\sum_{i=p+1}^{p+M}(c_i + \rho_i)\varphi_i \sum_{k=1}^p a_{ik}\xi_i^T f(\xi_i) \\
&\leq -2\mu\sqrt{QR}\sum_{i=p+1}^{p+M}(c_i + \rho_i)\varphi_i \sum_{k=1}^p a_{ik}\|\xi_i\|. \quad (42)
\end{aligned}$$

Since $\|u_i\| \leq \sigma \forall i \in \mathcal{T}$, it follows that

$$\begin{aligned}
& 2\xi_L((C + \rho)\Xi L_2 \otimes \sqrt{QR}I_n)u_T \\
&= 2\sqrt{QR}\sum_{i=p+1}^{p+M}(c_i + \rho_i)\varphi_i \sum_{k=1}^p a_{ik}\xi_i^T u_k \\
&\leq 2\sigma\sqrt{QR}\sum_{i=p+1}^{p+M}(c_i + \rho_i)\varphi_i \sum_{k=1}^p a_{ik}\|\xi_i\|. \quad (43)
\end{aligned}$$

Now, using a common matrix property $X^2 + Y^2 \geq 2XY$ where $X > 0$ and $Y > 0$ [53], we find

$$\begin{aligned}
& -Q\xi_L^T((\lambda_1^{\min}(C + \rho)^2 + \alpha\Xi) \otimes I_n)\xi_L \\
&\leq -2Q\xi_L^T(\sqrt{\lambda_1^{\min}\alpha\Xi}(C + \rho) \otimes I_n)\xi_L \quad (44)
\end{aligned}$$

assuming $X = \sqrt{\lambda_1^{\min}(C + \rho)} > 0$ and $Y = \sqrt{\alpha\Xi} > 0$. After that, selecting $\alpha \geq \frac{\max_{i \in \mathcal{L}} \varphi_i}{\lambda_1^{\min}}$ and $\mu > \sigma$, the inequality given in (39) implies

$$\dot{V}_1 \leq -Q\xi_L^T((C + \rho)\Xi \otimes I_n)\xi_L \quad (45)$$

invoking (42), (43) and (44). Introducing a change of variable $\zeta_L = (\sqrt{(C + \rho)\Xi} \otimes I_n)\xi_L$, inequality (45) can be rearranged as

$$\dot{V}_1 \leq -Q\zeta_L^T(I_M \otimes I_n)\zeta_L \quad (46)$$

which implies $\dot{V}_1 \leq 0$ since $Q > 0$. Furthermore, $\dot{V}_1 = 0$ only when $\zeta_L = 0$. Now, invoking LaSalle's invariance principle [33], asymptotic stability of (34) can be asserted. Therefore, $\lim_{t \rightarrow \infty} \zeta_L(t) = 0$ and hence

$$\lim_{t \rightarrow \infty} \xi_L(t) = 0. \quad (47)$$

From (33) and (47), we can write

$$\lim_{t \rightarrow \infty} (x_L(t) - h_L(t) - (-L_{lu}^{-1}L_{lt} \otimes I_n)x_T(t)) = 0. \quad (48)$$

Now, (48) can be expanded using Lemma 2 as

$$\begin{aligned}
& \lim_{t \rightarrow \infty} (x_L(t) - h_L(t) - \\
& \quad \left(\begin{bmatrix} \mathbf{1}_{n_1} & \cdots & \mathbf{0}_{n_1} \\ \vdots & \ddots & \vdots \\ \mathbf{0}_{n_p} & \cdots & \mathbf{1}_{n_p} \end{bmatrix} \otimes I_n \right) x_T(t)) = 0. \quad (49)
\end{aligned}$$

Rearranging (49), we have

$$\lim_{t \rightarrow \infty} \left(\begin{bmatrix} \bar{x}_1(t) - \bar{h}_1(t) - (\mathbf{1}_{n_1} \otimes I_n)\hat{x}_1 \\ \vdots \\ \bar{x}_p(t) - \bar{h}_p(t) - (\mathbf{1}_{n_p} \otimes I_n)\hat{x}_p \end{bmatrix} \right) = 0. \quad (50)$$

That is, for any $\bar{i} \in \{1, 2, \dots, p\}$,

$$\lim_{t \rightarrow \infty} (\bar{x}_{\bar{i}}(t) - \bar{h}_{\bar{i}}(t) - (\mathbf{1}_{n_{\bar{i}}} \otimes I_n)x_{\bar{i}}(t)) = 0. \quad (51)$$

Thus, it is established that the leaders in each cluster will achieve the desired formation (specified by the cluster-formation configuration vector $\bar{h}_{\bar{i}}(t)$) asymptotically under the application of the designed CFT protocol. ■

B. Proof of Theorem 3

Proof: Let $x_F = [x_{\hat{n}_0+1}^T, \dots, x_{p+N}^T]^T$ and $F(\varsigma) = [f^T(\varsigma_{\hat{n}_0+1}), f^T(\varsigma_{\hat{n}_0+2}), \dots, f^T(\varsigma_{p+N})]^T$. The closed-loop system dynamics of the followers embedded with the containment control protocol can be expressed in a structured form as

$$\begin{cases} \dot{x}_F = ((\hat{C} + \hat{\rho})L_{ff} \otimes K)x_F + ((\hat{C} + \hat{\rho})L_{fl} \otimes K)x_L \\ \quad - \eta(I_{N-M} \otimes I_n)F(\varsigma) \text{ and} \\ \dot{\varsigma}_i = \varsigma_i^T Q \varsigma_i \end{cases} \quad (52)$$

where $\hat{C} = \text{diag}\{\hat{c}_{\hat{n}_0+1}, \hat{c}_{\hat{n}_0+2}, \dots, \hat{c}_{p+N}\}$ and $\hat{\rho} = \text{diag}\{\hat{\rho}_{\hat{n}_0+1}, \hat{\rho}_{\hat{n}_0+2}, \dots, \hat{\rho}_{p+N}\}$.

Let the global containment error of the followers be $\varsigma_F = [\varsigma_{\hat{n}_0+1}^T, \varsigma_{\hat{n}_0+2}^T, \dots, \varsigma_{p+N}^T]^T$, then we have

$$\varsigma_F = (L_{fl} \otimes I_n)x_L + (L_{ff} \otimes I_n)x_F. \quad (53)$$

From (24), it can be derived that

$$\begin{cases} \dot{\zeta}_F = ((\hat{C} + \hat{\rho})L_{ff} \otimes K)_{\zeta_F} - \eta(L_{ff} \otimes I_n)F(\zeta) \\ \quad + ((C + \rho)L_{fl}L_{ll} \otimes K)(x_L - h_L \\ \quad - (-L_{ll}^{-1}L_{lt} \otimes I_n)x_T) - \mu(L_{fl} \otimes I_n)F(\xi) \text{ and} \\ \dot{\hat{c}}_i = \zeta_i^T Q \zeta_i. \end{cases} \quad (54)$$

Consider the following Lyapunov function candidate

$$V_2 = \frac{1}{2} \sum_{i=\hat{n}_0+1}^{p+N} \kappa_i (2\hat{c}_i + \hat{\rho}_i) \hat{\rho}_i + \frac{1}{2} \sum_{i=\hat{n}_0+1}^{p+N} \kappa_i (\hat{c}_i - \beta)^2 \quad (55)$$

where $\Psi = \text{diag}\{\kappa_{\hat{n}_0+1}, \kappa_{\hat{n}_0+2}, \dots, \kappa_{p+N}\}$ is a positive definite matrix such that $\Psi L_{ff} + L_{ff}^T \Psi > 0$ and let β be a positive constant to be determined later. According to Lemma 4 and the fact that L_{ff} is a nonsingular M -matrix, the existence of such a positive definite matrix Ψ is guaranteed from Lemma 1. Because $\hat{c}_i(0) > 0$, it follows from $\dot{\hat{c}}_i(t) \geq 0$ that $\hat{c}_i(t) > 0$ for any $t > 0$. Then, it is easy to conclude that V_2 is positive definite.

Thus, the time derivative of V_2 along the trajectory of (54) is obtained as

$$\begin{aligned} \dot{V}_2 &= \sum_{i=\hat{n}_0+1}^{p+N} (\kappa_i (\hat{c}_i + \hat{\rho}_i) \dot{\hat{\rho}}_i + \kappa_i \hat{\rho}_i \dot{\hat{c}}_i) + \sum_{i=\hat{n}_0+1}^{p+N} \kappa_i (\hat{c}_i - \alpha) \dot{\hat{c}}_i \\ &= \sqrt{QR} \sum_{i=\hat{n}_0+1}^{p+N} 2\kappa_i (\hat{c}_i + \hat{\rho}_i) \zeta_i^T \dot{\zeta}_i + \sum_{i=\hat{n}_0+1}^{p+N} \kappa_i (\hat{\rho}_i + \hat{c}_i - \beta) \dot{\hat{c}}_i. \end{aligned}$$

Note that

$$\sum_{i=\hat{n}_0+1}^{p+N} \kappa_i (\hat{\rho}_i + \hat{c}_i - \beta) \dot{\hat{c}}_i = Q_{\zeta_F^T} ((\hat{C} + \hat{\rho} - \beta I) \Psi \otimes I_n)_{\zeta_F},$$

and

$$\begin{aligned} &\sqrt{QR} \sum_{i=\hat{n}_0+1}^{p+N} 2\kappa_i (\hat{c}_i + \hat{\rho}_i) \zeta_i^T \dot{\zeta}_i \\ &= 2\sqrt{QR} \zeta_F^T ((\hat{C} + \hat{\rho}) \Psi \otimes I_n) \dot{\zeta}_F \\ &= -Q_{\zeta_F^T} ((\hat{C} + \hat{\rho}) (\Psi L_{ff} + L_{ff}^T \Psi) (\hat{C} + \hat{\rho}) \otimes I_n)_{\zeta_F} \\ &\quad + 2\zeta_F^T ((C + \rho) \Psi L_{fl} L_{ll} \otimes \sqrt{QR} K) (x_L - h_L - (-L_{ll}^{-1} L_{lt} \\ &\quad \otimes I_n) x_T) - 2\eta \zeta_F^T ((\hat{C} + \hat{\rho}) \Psi L_{ff} \otimes \sqrt{QR} I_n) F(\zeta) \\ &\quad - 2\mu \zeta_F^T ((\hat{C} + \hat{\rho}) \Psi L_{fl} \otimes \sqrt{QR} I_n) F(\xi) \\ &\leq -Q_{\zeta_F^T} (\lambda_2^{\min} (\hat{C} + \hat{\rho})^2 \otimes I_n)_{\zeta_F} \\ &\quad + 2\zeta_F^T ((C + \rho) \Psi L_{fl} L_{ll} \otimes \sqrt{QR} K) (x_L - h_L - (-L_{ll}^{-1} L_{lt} \\ &\quad \otimes I_n) x_T) - 2\eta \sqrt{QR} \zeta_F^T ((\hat{C} + \hat{\rho}) \Psi L_{ff} \otimes I_n) F(\zeta) \\ &\quad - 2\mu \sqrt{QR} \zeta_F^T ((\hat{C} + \hat{\rho}) \Psi L_{fl} \otimes I_n) F(\xi) \end{aligned} \quad (56)$$

where λ_2^{\min} represents the smallest positive eigenvalue of $[\Psi L_{ff} + L_{ff}^T \Psi]$. Similar to (42) and (43), upper bounds can be calculated for the terms involving the nonlinear function $F(\cdot)$ as mentioned below:

$$\begin{aligned} &-2\eta \sqrt{QR} \zeta_F^T ((\hat{C} + \hat{\rho}) \Xi L_{ff} \otimes I_n) F(\zeta) \\ &\leq -2\eta \sqrt{QR} \sum_{i=\hat{n}_0+1}^{p+N} (\hat{c}_i + \hat{\rho}_i) \kappa_i \|\zeta_i\| \sum_{k=p+1}^{p+M} a_{ik} \end{aligned}$$

and

$$\begin{aligned} &-2\mu \sqrt{QR} \zeta_F^T ((\hat{C} + \hat{\rho}) \Xi L_{fl} \otimes I_n) F(\xi) \\ &\leq 2\mu \sqrt{QR} \sum_{i=\hat{n}_0+1}^{p+N} (\hat{c}_i + \hat{\rho}_i) \kappa_i \|\zeta_i\| \sum_{k=p+1}^{p+M} a_{ik}. \end{aligned}$$

When the leaders achieve the desired formation $h_E(t)$, we have

$$\lim_{t \rightarrow \infty} x_L(t) - h_L(t) - (-L_{ll}^{-1} L_{lt} \otimes I_n) x_T(t) = 0, \quad (57)$$

which means that

$$\lim_{t \rightarrow \infty} (I_M \otimes I_n) (x_L(t) - h_L(t) - (-L_{ll}^{-1} L_{lt} \otimes I_n) x_T(t)) = 0. \quad (58)$$

Pre-multiplying both sides of (58) by $(C + \rho) \Psi L_{fl} L_{ll} \otimes \sqrt{QR} K$, we get

$$\begin{aligned} &\lim_{t \rightarrow \infty} \left(((C + \rho) \Psi L_{fl} L_{ll} \otimes \sqrt{QR} K) \right. \\ &\quad \left. (x_L(t) - h_L(t) - (-L_{ll}^{-1} L_{lt} \otimes I_n) x_T(t)) \right) = 0. \end{aligned}$$

Therefore we obtain

$$\begin{aligned} \dot{V}_2 &\leq Q_{\zeta_F^T} ((\hat{C} + \hat{\rho}) \Psi \otimes I_n - (\lambda_2^{\min} (\hat{C} + \hat{\rho})^2 + \beta \Psi) \otimes I_n)_{\zeta_F} \\ &\quad - 2\sqrt{QR} (\eta - \mu) \sum_{i=\hat{n}_0+1}^{p+N} (\hat{c}_i + \hat{\rho}_i) \kappa_i \|\zeta_i\| \sum_{k=p+1}^{p+M} a_{ik}. \end{aligned}$$

Invoking the same matrix property as used in the proof of Theorem 2, we find

$$\begin{aligned} &-Q_{\zeta_F^T} ((\lambda_2^{\min} (\hat{C} + \hat{\rho})^2 + \beta \Psi) \otimes I_n)_{\zeta_F} \\ &\leq -2Q_{\zeta_F^T} (\sqrt{\lambda_2^{\min} \beta} (\hat{C} + \hat{\rho}) \otimes I_n)_{\zeta_F}. \end{aligned} \quad (59)$$

Selecting $\beta \geq \frac{\max_{i \in \mathcal{F}} \kappa_i}{\lambda_2^{\min}}$, $\eta \geq \mu$ and substituting (59) into (59) yields

$$\dot{V}_2 \leq -Q_{\zeta_F^T} ((\hat{C} + \hat{\rho}) \Psi \otimes I_n)_{\zeta_F}. \quad (60)$$

Now, define $\Upsilon = \left(\sqrt{(\hat{C} + \hat{\rho}) \Psi \otimes I_n} \right)_{\zeta_F}$. Therefore, from (60) we get

$$\dot{V}_2 \leq -Q \Upsilon^T (I_{N-M} \otimes I_n) \Upsilon \quad (61)$$

which implies $\dot{V}_2 \leq 0$ since $Q > 0$ and $\dot{V}_2 = 0$ only when $\Upsilon = 0$. Now, invoking LaSalle's invariance principle [33], asymptotic stability of the closed-loop dynamics (24) of the follower robots can be ensured. Therefore,

$$\lim_{t \rightarrow \infty} \zeta_F(t) = 0. \quad (62)$$

From (53) and (62), we have

$$\lim_{t \rightarrow \infty} \left(x_F(t) - (-L_{ff}^{-1} L_{fl} \otimes I_n) x_L(t) \right) = 0, \quad (63)$$

which is equivalent to

$$\lim_{t \rightarrow \infty} \left(\hat{x}_{\bar{i}}(t) - (-L_{f\bar{i}}^{-1} L_{l\bar{i}} \otimes I_n) \bar{x}_{\bar{i}}(t) \right) = 0 \quad (64)$$

for all $\bar{i} \in \{1, 2, \dots, p\}$. It is hence finally concluded that the positions of followers converge into the convex hull formed by the leaders. This completes the proof. \blacksquare

REFERENCES

- [1] W. Ren and R. W. Beard, *Distributed consensus in multi-vehicle cooperative control*. Springer, 2008, vol. 27, no. 2.
- [2] F. L. Lewis, H. Zhang, K. Hengster-Movric, and A. Das, *Cooperative control of multi-agent systems: optimal and adaptive design approaches*. Springer Science & Business Media, 2013.
- [3] G. Antonelli, F. Arrichiello, and S. Chiaverini, "The entrapment/escorting mission," *IEEE Robotics and Automation Magazine*, vol. 15, no. 1, pp. 22–29, 2008.
- [4] I. Jang, H. S. Shin, A. Tsourdos, J. Jeong, S. Kim, and J. Suk, "An integrated decision-making framework of a heterogeneous aerial robotic swarm for cooperative tasks with minimum requirements," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 233, no. 6, pp. 2101–2118, 2019.
- [5] T. Krajnik, M. Nitsche, J. Faigl, P. Vanek, M. Saska, L. Preucil, T. Duckett, and M. Mejail, "A practical multirobot localization system," *Journal of Intelligent and Robotic Systems*, vol. 76, no. 3–4, pp. 539–562, 2014.
- [6] E. Ferrante, A. E. Turgut, M. Dorigo, and C. Huepe, "Elasticity-based mechanism for the collective motion of self-propelled particles with springlike interactions: A model system for natural and artificial swarms," *Physical Review Letters*, vol. 111, no. 26, pp. 268302–1 – 268302–5, 2013.
- [7] E. Ferrante, A. E. Turgut, N. Mathews, M. Birattari, and M. Dorigo, "Flocking in stationary and non-stationary environments: a novel communication strategy for heading alignment," in *International Conference on Parallel Problem Solving from Nature*. Springer, 2010, pp. 331–340.
- [8] J. Hu and A. Lanzon, "An innovative tri-rotor drone and associated distributed aerial drone swarm control," *Robotics and Autonomous Systems*, vol. 103, pp. 162–174, 2018.
- [9] J. Faigl, T. Krajnik, V. Vonasek, and L. Preucil, "On localization uncertainty in an autonomous inspection," in *In: Proceedings of the IEEE International Conference on Robotics and Automation*, 2012, pp. 1119–1124.
- [10] V. Spurny, T. Baca, M. Saska, R. Penicka, T. Krajnik, J. Thomas, D. Thakur, G. Loianno, and K. Vijoy, "Cooperative autonomous search, grasping, and delivering in a treasure hunt scenario by a team of unmanned aerial vehicles," *Journal of Field Robotics*, vol. 36, pp. 125–148, 2018.
- [11] J. Hu, A. E. Turgut, T. Krajnik, B. Lennox, and F. Arvin, "Occlusion-based coordination protocol design for autonomous robotic shepherding tasks," *IEEE Transactions on Cognitive and Developmental Systems*, (in press), 2020. DOI:10.1109/TCDS.2020.3018549.
- [12] L. Yliniemi, A. K. Agogino, and K. Tumer, "Multirobot coordination for space exploration," *AI Magazine*, vol. 35, no. 4, pp. 61–74, 2014.
- [13] J. Hu, H. Niu, J. Carrasco, B. Lennox, and F. Arvin, "Voronoi-based multi-robot autonomous exploration in unknown environments via deep reinforcement learning," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 14413–14423, 2020.
- [14] J. Chen, M. Gauci, W. Li, A. Kolling, and R. Grob, "Occlusion-based cooperative transport with a swarm of miniature mobile robots," *IEEE Transactions on Robotics*, vol. 31, no. 2, pp. 307–321, 2015.
- [15] J. Hu, P. Bhowmick, F. Arvin, A. Lanzon, and B. Lennox, "Cooperative control of heterogeneous connected vehicle platoons: An adaptive leader-following approach," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 977–984, 2020.
- [16] Z. Li and Z. Duan, *Cooperative control of multi-agent systems: a consensus region approach*. CRC Press, 2014.
- [17] J. Alonso-Mora, S. Baker, and D. Rus, "Multi-robot formation control and object transport in dynamic environments via constrained optimization," *The International Journal of Robotics Research*, vol. 36, no. 9, pp. 1000–1021, 2017.
- [18] S. Bandyopadhyay, S.-J. Chung, and F. Y. Hadaegh, "Probabilistic and distributed control of a large-scale swarm of autonomous agents," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1103–1123, 2017.
- [19] G. Antonelli, F. Arrichiello, F. Caccavale, and A. Marino, "Decentralized time-varying formation control for multi-robot systems," *The International Journal of Robotics Research*, vol. 33, no. 7, pp. 1029–1043, 2014.
- [20] J. Alonso-Mora, E. Montijano, T. Nageli, O. Hilliges, M. Schwager, and D. Rus, "Distributed multi-robot formation control in dynamic environments," *Autonomous Robots*, pp. 1–22, 2018.
- [21] J. Qin and C. Yu, "Cluster consensus control of generic linear multi-agent systems under directed topology with acyclic partition," *Automatica*, vol. 49, no. 9, pp. 2898–2905, 2013.
- [22] X. Dong, Q. Li, Q. Zhao, and Z. Ren, "Time-varying group formation analysis and design for general linear multi-agent systems with directed topologies," *International Journal of Robust and Nonlinear Control*, vol. 27, no. 9, pp. 1640–1652, 2017.
- [23] I. Jang, H.-S. Shin, and A. Tsourdos, "Local information-based control for probabilistic swarm distribution guidance," *Swarm Intelligence*, vol. 12, no. 4, pp. 327–359, 2018.
- [24] R. O'Grady, C. Pinciroli, A. L. Christensen, and M. Dorigo, "Supervised group size regulation in a heterogeneous robotic swarm," *Proceedings of ROBOTICA*, pp. 113–119, 2009.
- [25] N. Cambier, V. Fremont, and E. Ferrante, "Group-size regulation in self-organised aggregation through the naming game," in *Proceedings of the Second International Symposium on Swarm Behavior and Bio-Inspired Robotics*, 2017.
- [26] P. Lin, Y. Wang, H. Qi, and Y. Hong, "Distributed consensus-based k-means algorithm in switching multi-agent networks," *Journal of Systems Science and Complexity*, vol. 31, no. 5, pp. 1128–1145, 2018.
- [27] M. Elango, S. Nachiappan, and M. K. Tiwari, "Balancing task allocation in multi-robot systems using k-means clustering and auction based mechanisms," *Expert Systems with Applications*, vol. 38, no. 6, pp. 6486–6491, 2011.
- [28] I. Jang, H.-S. Shin, and A. Tsourdos, "Anonymous hedonic game for task allocation in a large-scale multiple agent system," *IEEE Transactions on Robotics*, vol. 34, no. 6, pp. 1534–1548, 2018.
- [29] Y. Cao, D. Stuart, W. Ren, and Z. Meng, "Distributed containment control for multiple autonomous vehicles with double-integrator dynamics: algorithms and experiments," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 4, pp. 929–938, 2011.
- [30] Z. Li, Z. Duan, W. Ren, and G. Feng, "Containment control of linear multi-agent systems with multiple leaders of bounded inputs using distributed continuous controllers," *International Journal of Robust and Nonlinear Control*, vol. 25, no. 13, pp. 2101–2121, 2015.
- [31] H. Zhang, F. L. Lewis, and Z. Qu, "Lyapunov, adaptive, and optimal design techniques for cooperative systems on directed communication graphs," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 7, pp. 3026–3041, 2012.
- [32] B. Crowther, A. Lanzon, M. Maya-Gonzalez, and D. Langkamp, "Kinematic analysis and control design for a nonplanar multirotor vehicle," *AIAA Journal of Guidance, Control, and Dynamics*, vol. 34, no. 4, pp. 1157–1171, 2011.
- [33] H. K. Khalil, *Nonlinear Systems*, 3rd ed. New Jersey, USA: Prentice-Hall, Upper Saddle River, 2002.
- [34] A. Dutta and A. Asaithambi, "One-to-many bipartite matching based coalition formation for multi-robot task allocation," *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 2019-May, pp. 2181–2187, 2019.
- [35] A. Rauniyar and P. K. Muhuri, "Multi-robot coalition formation problem: Task allocation with adaptive immigrants based genetic algorithms," *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, pp. 137–142, 2017.
- [36] L. B. Johnson, H.-L. Choi, and J. P. How, "The Role of Information Assumptions in Decentralized Task Allocation: A Tutorial," *IEEE Control Systems*, vol. 36, no. 4, pp. 45–58, 2016.
- [37] H. L. Choi, L. Brunet, and J. P. How, "Consensus-based Decentralized Auctions for Robust Task Allocation," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 912–926, 2009.
- [38] P. Segui-Gasco, H.-S. Shin, A. Tsourdos, and V. J. Segui, "Decentralised Submodular Multi-Robot Task Allocation," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Hamburg, Germany, 2015, pp. 2829–2834.
- [39] J. H. Cho, Y. Wang, I. R. Chen, K. S. Chan, and A. Swami, "A Survey on Modeling and Optimizing Multi-Objective Systems," *IEEE Communications Surveys and Tutorials*, vol. 19, no. 3, pp. 1867–1901, 2017.
- [40] Z. Meng, W. Ren, and Z. You, "Distributed finite-time attitude containment control for multiple rigid bodies," *Automatica*, vol. 46, no. 12, pp. 2092–2099, 2010.
- [41] H. Tnunay, Z. Li, C. Wang, and Z. Ding, "Distributed collision-free coverage control of mobile robots with consensus-based approach," *Proceedings of the IEEE International Conference on Control & Automation*, pp. 678–683, 2017.
- [42] J. Hu, P. Bhowmick, and A. Lanzon, "Two-layer distributed formation-containment control strategy for linear swarm systems: Algorithm and experiments," *International Journal of Robust and Nonlinear Control*, vol. 30, no. 16, pp. 6433–6453, 2020.

- [43] M. Aranda, G. López-Nicolás, C. Sagüés, and Y. Mezouar, "Formation control of mobile robots using multiple aerial cameras," *IEEE Transactions on Robotics*, vol. 31, no. 4, pp. 1064–1071, 2015.
- [44] J. Hu, P. Bhowmick, and A. Lanzon, "Distributed adaptive time-varying group formation tracking for multiagent systems with multiple leaders on directed graphs," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 1, pp. 140–150, 2020.
- [45] O. Arslan, D. P. Guralnik, and D. E. Koditschek, "Coordinated robot navigation via hierarchical clustering," *IEEE Transactions on Robotics*, vol. 32, no. 2, pp. 352–371, 2016.
- [46] H. Park and S. A. Hutchinson, "Fault-tolerant rendezvous of multirobot systems," *IEEE Transactions on Robotics*, vol. 33, no. 3, pp. 565–582, 2017.
- [47] Y. Lv, Z. Li, Z. Duan, and G. Feng, "Novel distributed robust adaptive consensus protocols for linear multi-agent systems with directed graphs and external disturbances," *International Journal of Control*, vol. 90, no. 2, pp. 137–147, 2017.
- [48] N. Mathew, S. L. Smith, and S. L. Waslander, "Multirobot rendezvous planning for recharging in persistent tasks," *IEEE Transactions on Robotics*, vol. 31, no. 1, pp. 128–142, 2015.
- [49] F. Mondada, M. Bonani, X. Raemy, J. Pugh *et al.*, "The e-puck, a robot designed for education in engineering," *Proceedings of the Conference on Autonomous Robot Systems and Competitions*, pp. 59–65, 2009.
- [50] N. Biggs, N. L. Biggs, and B. Norman, *Algebraic graph theory*. Cambridge university press, 1993, vol. 67.
- [51] F. Arvin, J. Espinosa, B. Bird, A. West, S. Watson, and B. Lennox, "Mona: an affordable open-source mobile robot for education and research," *Journal of Intelligent and Robotic Systems*, pp. 1–15, 2018.
- [52] M. Polycarpou, Y. Yang, and K. Passino, "A cooperative search framework for distributed agents," in *Proceedings of the IEEE International Symposium on Intelligent Control*, vol. 16. IEEE, 2001.
- [53] D. S. Bernstein, *Matrix mathematics: Theory, facts, and formulas with application to linear systems theory*, 2nd ed. New Jersey, USA: Princeton University Press, 2005.



Junyan Hu received B.Eng degree in Automation from Hefei University of Technology in 2015 and Ph.D degree in Electrical and Electronic Engineering from the University of Manchester in 2020.

Dr. Hu is currently a Postdoctoral Research Associate in Robotics at the University of Manchester. His research interests include multi-robot coordination, networked control systems, autonomous vehicle navigation, and swarm intelligence.



Parijat Bhowmick received his B. Tech degree in Electrical Engineering from West Bengal University of Technology, Kolkata, India in 2008 and received his M.E. degree in Control Systems Engineering from Jadavpur University, Kolkata, India in 2012. He has completed his Ph.D. degree in Control Engineering from Indian Institute of Technology Kharagpur, India, in 2018. He is now working as a Post-doctoral Research Associate in the School of Electrical and Electronic Engineering, University of Manchester, UK. His research interests include robust control of

negative-imaginary systems, passivity-based control and decentralized integral control.



Inmo Jang received the B.Sc. and M.Sc. degrees in mechanical and aerospace engineering from Seoul National University, South Korea, and Ph.D. degree in autonomous decision making of multi-robot systems from Cranfield University, U.K., in 2008, 2010, and in 2018, respectively. From 2010 to 2014, he worked in Korea Aerospace Industries Ltd., Sacheon, South Korea, and from 2014 to 2015, worked in Korea Institute of Aviation Safety Technology, Incheon, South Korea. He is currently a research associate in Robotics for Extreme Environ-

ment Group at the University of Manchester, U.K., where particularly being involved in RAIN (Robotics and AI in Nuclear) project, one of the four big robotics and AI projects funded by EPSRC. His research interests range over the areas of multi-robot/agent systems, decentralized autonomous decision making, control and navigation for robots, human multi-robot interaction, and their real-world applications.



Farshad Arvin joined The University of Manchester as an Assistant Professor in Robotics in July 2018. He received his BSc degree in Computer Hardware Engineering in 2004, MSc degree in Computer Systems Engineering in 2010, and a PhD in Computer Science in 2015. His research interests include swarm robotics and autonomous systems. Farshad joined the University of Manchester in July 2015 as a Post-Doctoral Researcher at Department of Electrical and Electronic Engineering. He was a Research Assistant at the Computational Intelligence Laboratory (CIL) in the University of Lincoln, UK (2012 to 2015). He was awarded a Marie Curie-Sklodowska Fellowship to be involved in the FP7-EYE2E and LIVCODE EU projects during his PhD study. He visited several leading institutes including Artificial Life Laboratory in University of Graz, Austria in 2018, the Italian Institute of Technology (iit) in Genoa, Italy in 2017, Institute of Rehabilitation and Medical Robotics in Huazhong University of Science and Technology (HUST), Wuhan, China in 2014, and the Institute of Microelectronics at Tsinghua University in Beijing, in 2012 and 2013.

Farshad is the founding director of Swarm & Computation Intelligence Laboratory (SwaCIL) formed in 2018. He is coordinating several research projects including a large EU project H2020-FET RoboRoyale and he is a PI in H2020-FET Robocoenosis.



Alexander Lanzon received his Ph.D. degree in Control Engineering and his M.Phil. degree in Robot Control from the University of Cambridge, Cambridge, UK, in 2000 and 1997 respectively and received his B.Eng.(Hons). degree in Electrical and Electronic Engineering from the University of Malta, Msida, Malta, in 1995.

He has held research and academic positions at Georgia Institute of Technology, Atlanta GA, USA, and the Australian National University, Canberra ACT, Australia, and industrial positions at ST-Microelectronics Ltd., Kirkop, Malta; Yaskawa Denki (Tokyo) Ltd., Saitama-Ken, Japan; and National ICT Australia Ltd., Canberra ACT, Australia. In 2006, he joined the University of Manchester, Manchester, UK, where he now holds the Chair in Control Engineering. His research interests include the fundamentals of feedback control theory, negative imaginary systems theory, \mathcal{H}_∞ control theory, robust linear and nonlinear feedback control systems, and applying robust control theory to innovative mechatronic, robotic and drone applications.

Prof. Lanzon is a Fellow of the Institute of Mathematics and its Applications, the Institute of Measurement and Control, and the Institution of Engineering and Technology. He has served as an Associate Editor of the *IEEE Transactions on Automatic Control* from 2012 to 2018, and as a Subject Editor of the *International Journal of Robust and Nonlinear Control* from 2012 to 2015.