Achieving Collaborative Resilience in Multi-Layer Heterogeneous Robotic Systems

Jason Hughes[1], Juntao Chen[2] and Christopher Korpela[1]

***Abstract*— As the prevalence of adversarial attacks to multi-robot systems rises, increasing the resiliency of these systems such that the robotic agents can continually remain connected during task execution is highly important. Existing literature on connectivity and control of robotic swarms mostly considers scenarios with homogeneous agents. This work investigates the resiliency of a multi-layered system of robots that aims to equip agents with heterogeneous levels of connectivity depending on their security requirements to withstand attacks. We propose a metric, dubbed (k_1, k_2)-connectivity, that extends the conventional k-connectivity to capture the agents' heterogeneous resiliency needs. Specifically, the robotic agents in distinct layers can consistently stay connected after removing any k_1 or k_2 links. We further develop a computationally efficient algorithm to guide the real-time control of robotic agents that optimizes their mobility under an assigned task while maintaining the required (k_1, k_2)-connectivity holistically. Finally, we show the effectiveness of the designed algorithm through simulations.**

I. Introduction

Communication between agents in a multi-robot system is pertinent to performing real-time collaborative tasks. These tasks can include collaborative mapping, search and rescue missions, and hazardous area detection and mapping. All of these tasks, and many others, require information sharing amongst members of the system. Information exchange can include location data, task data, controller data, or sensor data used to build databases and make decisions. Agents can exchange information via multiple types of wireless links including WiFi, radio, or Bluetooth as long as they are within communication range of each other. Communication range can be guaranteed by limiting how far the agents can move from each other, but it may also limit what the swarm can accomplish. In this regard, many researchers have leveraged connectivity metrics, such as k-connectivity, from graph theory to guide the control design for robotic agents that maintains connections between agents during task executions.

As the tasks become complex, heterogeneous multi-robot systems are increasingly adopted for collaborative missions [1]–[3]. In these scenarios, maintaining the same level of performance for all agents (e.g., connectivity and security) may not be the desired solution. For example, in a collaborative air-ground autonomous system shown in Fig. 1, it is desirable to maintain a higher level of connectivity amongst the aerial agents as they could have more valuable information to communicate to the rest of the swarm. This heterogeneous connectivity consideration is also necessary when some agents in the system are more fragile, have less advanced autonomy, or have different communication capabilities. Let us consider another example of collaborative autonomy with a swarm of wheeled robots and quadrupeds in mapping an urban environment. Since the quadrupeds are able to navigate in and out of buildings and upstairs more easily, they are responsible for mapping the upper levels of buildings. Thus, it is feasible for the quadrupeds to maintain a relatively lower level of connectivity such that they can navigate areas where communication links are more challenging to establish. Alternatively, the wheeled robots can maintain a higher level of connectivity to share data amongst the swarm more quickly.

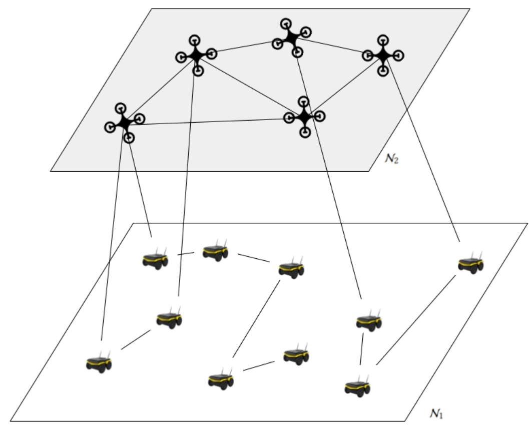


Fig. 1. Example of a two-layered heterogeneous multi-robot system composed of air and ground robots for collaborative missions. Robots in different layers have heterogeneous requirements for network resiliency during their task execution.

Enhancing the resiliency of the heterogeneous swarm is imperative as cyber attacks to autonomous systems become ubiquitous [4]. The autonomous agent could be isolated from the swarm through targeted communication link compromises, e.g., by denial-of-service (DoS) or jamming attacks [5], [6]. One mechanism to improve robotic network resilience is by keeping the agents well connected to withstand malicious attacks. In particular, it is necessary for those critical agents in the swarm to maintain a higher level of

[1]The authors are with the Robotics Research Center, Department of Electrical Engineering and Computer Science, United States Military Academy, West Point NY, 10996 USA. [2]The author is with the Department of Computer and Information Sciences, Fordham University, New York NY, 10023 USA. E-mail: {jason.hughes,christopher.korpela}@westpoint.edu, jchen504@fordham.edu.

connectivity, so that it is more likely for them to stay connected with the rest of the swarm under the strategic attack. In this work, we aim to establish a systematic framework for designing resilient dynamic multi-layer robotic networks with heterogeneous connectivity requirements. Specifically, we extend the conventional k-connectivity metric to (k_1,k_2)-connectivity to capture the distinct needs for resilience by the agents in the multi-layer swarm system. We further propose a computationally efficient algorithm to guide the real-time decision-making of robotic agents during their task completion while preserving (k_1,k_2)-connectivity. The developed mechanism is shown effective in achieving collaborative resilient autonomy by extensive case studies.

Related Works: Connectivity control has been vastly explored for robotic applications [7]–[10]. Among various connectivity measures, k-connectivity has received significant attention with various applications, including static IoT networks and mobile multi-robot systems [11]–[16]. In terms of static networks, the authors of [17]–[19] have studied network construction algorithms to achieve k-connectivity and m-coverage. Resilient multi-layer network design with heterogeneous connectivity requirements have been investigated in [20], but it is not directly applicable to dynamic multi-robot networks. The authors of [11] specifically investigated the scenario of maintaining k-connectivity of a multi-agent robotic system under a minimally invasive controller. In comparison, our work considers a multi-layer robotic network with the goal of achieving heterogeneous resilience for agents depending on the requirements.

Organization of the Paper: The rest of the paper is organized as follows. In Section II, we present the preliminaries and formulate the problem. Section III analyzes the problem and develops an algorithm to guide the dynamic decision-making of resilient multi-layer robotic systems. In Section IV, we corroborate our results using simulation case studies. Finally, Section V concludes the paper.

II. Problem Formulation

In this section, we present the preliminaries of control barrier certificates, discuss the problem setting, and formulate the problem of achieving a resilient multi-agent system.

A. Problem Setting

We consider a heterogeneous multi-agent system in which the agents execute mission-critical tasks collaboratively. Connectivity plays an essential role in such scenarios to equip agents with high-level situational awareness. Depending on the roles of agents, their requirements on the level of connectivity can be distinct. In this work, we focus on a collaborative autonomy consisting of two types of mobile agents that hold heterogeneous connectivity demands.

The basic setup of the interdependent two-layer swarms is as follows. Let the graph $\mathscr{G} = \{\mathscr{N},\mathscr{E}\}$ denote the communication topology among the mobile agents, where $\mathscr{N} := \{1,...,|\mathscr{N}|\}$ denotes all the agents in the network and $\mathscr{E}$ represents all possible communication links between agents. The agent set is divided into two subsets, $\mathscr{N}_1 := \{0,...,n_1\} \subset \mathscr{N}$ and $\mathscr{N}_2 := \{n_1+1,...,n_1+n_2\} \subset \mathscr{N}$ such that $\mathscr{N}_1 \cup \mathscr{N}_2 = \mathscr{N}$, capturing the heterogeneity of two groups of swarms. Denote by $n := n_1+n_2$ the total number of agents in the system. We further denote $\{i,j\} \in \mathscr{E}$ as an undirected connection between nodes i and j which signifies communication between those two agents.

Each agent i's dynamics are described by $\dot{x}_i = u_i$, where $x_i \in \mathbb{R}^3$ denotes its position and $u_i \in \mathbb{R}^3$ is its corresponding control input, $\forall i \in \mathscr{N}$. In addition, two agents stay connected if they are within a distance of r_c, where r_c is the communication radius of the agents.

B. Control Barrier Certificates

In order to enforce the connections and ensure safety, we need to define two types of barrier certificates [21]. Let $\mathbf{x} := [x_1^T,...,x_n^T]^T \in \mathbb{R}^{3n}$ denote the position state of the system. To enable collision avoidance through boundary certificates we define a radius of safety r_s, meaning all other agents in the system must maintain a distance of at least r_s from one another. We can encode this with the pairwise set $\mathscr{S}_{ij}$, defined as the following:

$$s_{ij}(\mathbf{x}) = ||x_i - x_j||^2 - r_s^2, \quad \forall i > j, \tag{1}$$

$$\mathscr{S}_{ij} := \{\mathbf{x} \in \mathbb{R}^{3n} | s_{ij}(\mathbf{x}) \geq 0\}. \tag{2}$$

The set $\mathscr{S}_{ij}$ is the safety set such that the paths of robots i and j will intersect each other's radius of safety. The safety set for the swarm as a whole can be composed of the following:

$$\mathscr{S} = \bigcap_{\{v_i,v_j \in \mathscr{N} | i>j\}} \mathscr{S}_{ij}. \tag{3}$$

The barrier certificate for the safety set constrains the control $\mathbf{u} := [u_1^T,...,u_n^T]^T \in \mathbb{R}^{3n}$ of the entire swarm using (3). The admissible control space is defined as

$$\mathscr{B}_s := \left\{ \mathbf{u} \in \mathbb{R}^{3n} \middle| \frac{\partial s_{ij}(\mathbf{x})}{\partial x} u + \gamma s_{ij}(\mathbf{x}) \geq 0, \ \forall i > j \right\}, \tag{4}$$

where γ is a defined parameter controlling the available sets. It has been shown that as long as the initial state of the system is collision-free, every control input $\mathbf{u}$ in the set $\mathscr{B}_s$ also preserves such property [22].

We define an additional boundary certificate that will ensure pairwise connectivity over the constructed network, meaning that if the constructed network $\mathscr{G}$ connects agents i and j, the distance between these two agents must within the radius of connectivity r_c. To this end, we construct the following set:

$$c_{ij}(\mathbf{x}) = r_c^2 - ||x_i - x_j||^2, \tag{5}$$

$$C_{ij} = \{\mathbf{x} \in \mathbb{R}^{3n} | c_{ij}(\mathbf{x}) \geq 0\}. \tag{6}$$

The connection set C_{ij} is the possible set on $\mathbf{x}$ that the pairwise connection between agents i and j is ensured. In section II-C, we describe our algorithm to form such connections. The constructed network $\mathscr{G} = \{\mathscr{N},\mathscr{E}_c\}$, where $\mathscr{E}_c \subset \mathscr{E}$ is the defined set of connections to be maintained. We can impose the subgraph $\mathscr{G}_c$ onto the system with the following set:

$$\mathscr{C}(\mathscr{G}_c) = \bigcap_{\{i,j \in \mathscr{N} | (i,j) \in \mathscr{E}_c\}} C_{ij}. \tag{7}$$

We construct a similar barrier certificate to the one in (4) that imposes another class of linear constraints on the control input $\mathbf{u}$ to the system, detailed as:

$$\mathscr{B}_c(\mathbf{x}, \mathscr{G}_c) = \left\{ \mathbf{u} \in \mathbb{R}^{3n} \middle| \frac{\partial c_{ij}(\mathbf{x})}{\partial x} u + \gamma c_{ij}(\mathbf{x}) \geq 0, \forall (i,j) \in \mathscr{E}_c \right\}. \tag{8}$$

Similar to the safety certificate in (4), if the joint control input $\mathbf{u}$ is in the set $\mathscr{B}_c$, the pair-wise connection between i and j will be maintained.

C. Heterogeneous Resiliency

The primary objective of this work is to investigate how a heterogeneous two-layer multi-agent system can maintain different connectivity levels to satisfy the distinct needs for the resiliency of the agents. In the established framework, the agents in the two subsets of $\mathscr{N}_1$ and $\mathscr{N}_2$ have heterogeneous security requirements. In the following, we present the definition of k-connectivity and then extend it to (k_1, k_2)-connectivity which will be used to quantify the performance of multi-layer heterogeneous swarms.

Definition 1 (k-Connectivity [23])**.** *In a network with a set of nodes $\mathscr{N}$ and links $\mathscr{E}$, the network is said to be k-connected if the network is still connected after the removal of any k links.*

Note that the network is said to be connected if there exists a path between any two nodes in the network. To account for the heterogeneous levels of resilience, we leverage static (k_1, k_2)-connectivity from [20] and adapt it to a dynamic multi-agent system. Recall that the set of nodes $\mathscr{N}$ are divided into two layers, described by $\mathscr{N}_1$ and $\mathscr{N}_2$. We assume that the nodes in $\mathscr{N}_1$ and $\mathscr{N}_2$ aim to maintain k_1 and k_2 level of connectivity, respectively. It means that subnetworks 1 and 2 (i.e., two layers of networks) should remain connected after the compromise of any k_1 and k_2 links in the network.

We further have the following assumption about the system parameters.

Assumption 1. *The system parameters satisfy: i) $k_1 \leq k_2$, ii) $n_1 > k_1$, iii) $n_2 > k_2 - k_1$, and iv) $n_1 \geq \frac{n_2(k_1+1)}{2}$.*

Assumption i) indicates that those agents in subnetwork 2 are relatively more important than those in subnetwork 1, and thus are required to be more resistant to link compromise attacks. Assumptions ii) and iii) indicate that there are relatively more nodes than the number of link compromises. Assumption iv) implies that the size of subnetwork 1 is larger than that of subnetwork 2. Assumptions ii)-iv) together ensure the existence of a network configuration that achieves (k_1, k_2)-connectivity.

A formal definition of (k_1, k_2)-connectivity is presented below.

Definition 2 ((k_1, k_2)-Connectivity)**.** *Consider a network $\mathscr{G} = \{\mathscr{N} = \mathscr{N}_1 \cup \mathscr{N}_2, \mathscr{E}\}$. The network is (k_1, k_2)-connected if the following conditions are satisfied: i) with the removal of any k_1 links, the nodes in $\mathscr{N}_1$ are still connected, and ii) with the removal of any k_2 links, the nodes in $\mathscr{N}_2$ are still connected.*

Note that since $k_1 \leq k_2$, any k_1 link removal cannot disconnect subnetwork 2. Thus, scenario i) in Definition 2 is equivalent to the following: with the removal of any k_1 links, all the nodes in $\mathscr{N}$ are still connected. In this case, the two sets of agents are connected, ensuring the global swarm's resilience.

D. Problem Formulation

In this section, we formulate the minimally invasive control problem for the agents that conforms to the connectivity and safety constraints during mission executions. Considering given the optimal goal-oriented control vector $\hat{u}_i \in \mathbb{R}^3$, the system operator needs to modify $\hat{u}_i$ for each agent $i \in \mathscr{N}$ such that the connected agents remain within their connectivity range, and all agents do not collide to others during the operation. We propose the following optimization problem:

$$\mathbf{u}^* = \arg\min_{\mathscr{G}_c, \mathbf{u}} \sum_{i=1}^{n} ||u_i - \hat{u}_i||^2 \tag{9}$$

$$\text{s.t.} \quad \mathscr{G}_c = \{\mathscr{N}, \mathscr{E}_c\} \subset \mathscr{G} \text{ is } (k_1, k_2)-\text{connected}, \tag{10}$$

$$\mathbf{u} \in \mathscr{B}_x(\mathbf{x}) \bigcap \mathscr{B}_c(\mathbf{x}, \mathscr{G}_c), \tag{11}$$

$$||u_i|| \leq \alpha_i, \ \forall i = 1, ..., n, \tag{12}$$

where α_i is the maximum velocity of each agent i. The output $\mathbf{u}^* \in \mathbb{R}^{3n}$ is the control inputs of the entire swarm that adheres to the subnetwork $\mathscr{G}_c$ and the safety and velocity parameters. The objective of the optimal task-based control is preserved in $\hat{u}_i$. The desired control $\hat{u}_i, \forall i \in \mathscr{N}$, may not be preserved due to the connectivity and safety constraints. Thus, the problem (9) ensures that the nodes are (k_1, k_2)-connected and outside of each other's radius of safety while minimizing the deviation from $\hat{u}_i$. Another remark is that the problem (9) is solved repeatedly over the considered time horizon, and hence it yields a dynamic heterogeneous resilient swarm.

III. Problem Analysis

In this section, we aim to solve the formulated minimally invasive optimal control problem in (9) to obtain a (k_1, k_2)-connected heterogeneous swarm. The main challenge in addressing (9) is due to the presence of connectivity and safety constraints. First, it is nontrivial to construct a network satisfying the heterogeneous connectivity requirements, even for a static network. Second, the decision for the control input $\mathbf{u}$ and the network configuration $\mathscr{G}_c$ are naturally coupled. The control input to the agent determines the trajectory of the movement, impacting the agent's safety and network connectivity. On the other hand, the safety and connectivity requirements constrain the decision on the possible control

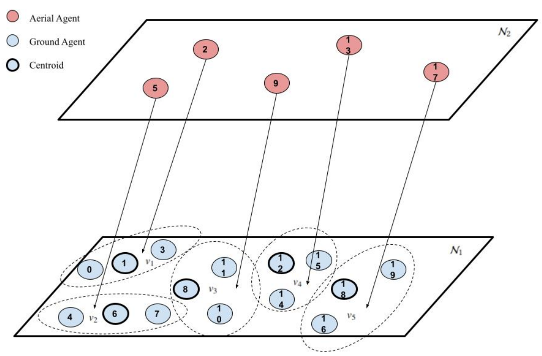


Fig. 2. An air-ground teaming example that illustrates how Algorithm 1 is used to index the agents to facilitate configuring a (k_1, k_2)-connected network. The ground agents in blue (all nodes in $\mathcal{N}_1$) constitute subnetwork 1, while the aerial agents in red (all nodes in $\mathcal{N}_2$) constitute subnetwork 2.

actions. Indeed, directly optimizing the control decision variables with the network design as a constraint is NP-hard. We take an alternative approach by developing a two-step heuristic method to address the established problem.

A. (k_1, k_2)-Connected Network Construction

To design a (k_1, k_2)-connected network $\mathcal{G}_c = \{\mathcal{N}, \mathcal{E}_c\}$, the most critical task is to specify the subset of edges $\mathcal{E}_c \subseteq \mathcal{E}$. Before presenting the algorithm to construct such networks, we present the definition of Harary networks as follows.

Definition 3 (Harary Network [24]). *Harary network is the optimal network configuration that uses the minimum number of links, $\lceil \frac{(k+1)n}{2} \rceil$, such that the network is still connected after removing any k links.*

The construction of a Harary network is as follows. First, it creates links between nodes i and j such that $(|i-j| \mod n) = 1$, and then between nodes i and j with $(|i-j| \mod n) = 2$, and so on until $(|i-j| \mod n) = k$ when the constructive cycle stops. When n is odd, the last cycle of connection is slightly changed as $\frac{(k+1)n}{2}$ is not an integer. However, the lower bound $\lceil \frac{(k+1)n}{2} \rceil$ on the number of links can still be achieved. The above algorithm to construct a Harary network has a linear time computational complexity in terms of both the number of nodes n and the connectivity requirement k.

Harary network plays an essential role in constructing a (k_1, k_2)-connected robotic network. The first step is to number the nodes in the network strategically. To this end, we start by clustering the nodes in $\mathcal{N}_1$ into $|\mathcal{N}_2|$ clusters (e.g., k-means clustering with $\mathcal{O}(|\mathcal{N}_1|^2)$ computational complexity) such that each agent in $\mathcal{N}_2$ can be assigned its own cluster. We define the set of clusters as $\mathcal{V}_i \subset \mathcal{N}_1$, for $i = 1, ..., |\mathcal{N}_2|$, such that $\cup_{i=1}^{|\mathcal{N}_2|} \mathcal{V}_i = \mathcal{N}_1$, where $\mathcal{V}_i$ is a cluster. Denote by v_i^c the corresponding centroid of the cluster containing nodes $\mathcal{V}_i$. Note that the assignment of nodes in $\mathcal{N}_2$ to a cluster in $\mathcal{N}_1$ is achievable due to $n_1 \geq n_2$. We next use the centroids of the clusters to enable the matching between the clusters and the nodes in $\mathcal{N}_2$. Specifically, we assign each node $i \in \mathcal{N}_2$ to a cluster j^* based on the distance metric:

$$j^* = \arg\min_j ||v_j^c - x_i||^2, \text{ for } i \in \mathcal{N}_2. \tag{13}$$

The assignment is determined in a greedy fashion for simplicity, but more complex matching algorithms can be leveraged.

Now that the agents in $\mathcal{N}_2$ are assigned to clusters, we can designate an index to each agent in the network which facilitates the construction of (k_1, k_2)-connected network. We start from the leftmost cluster, i.e., the cluster with the least y value (i.e., the second dimension in the position vector) in its centroid. We then work clockwise around the cluster. One central challenge in this process is to index the node in $\mathcal{N}_2$. We develop the following procedure. If the cluster has an even number of nodes and only one node is being added from $\mathcal{N}_2$, then it gets placed in the middle, i.e., it is inserted with index $|\mathcal{V}_i|/2$ within the cluster. If the cluster has an odd number of nodes, we place it with index $\frac{|\mathcal{V}_i|-1}{2}$. If there is more than one node to be matched with a cluster from $\mathcal{N}_2$, then we place one at the second index of the cluster and the other at the $|\mathcal{V}_i| - 1$ index of the cluster so that they are as far apart as possible, reducing the risk of being indexed next each other in the multi-layer network. For illustrative purposes, an example of the numbering with $|\mathcal{N}_1| = 15$ and $|\mathcal{N}_2| = 5$ is shown in Fig. 2.

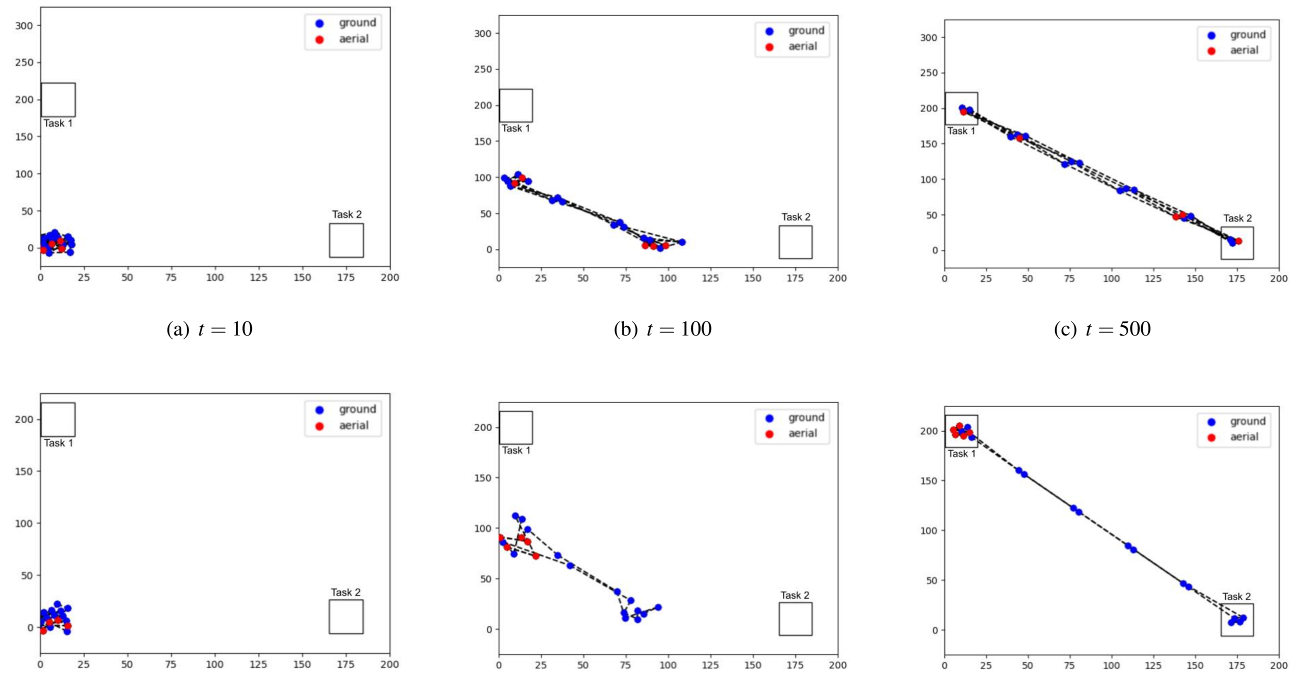
Once all the agents are indexed, we first build a k_1 Harary network. This step ensures that every node in $\mathcal{N}_1$ is at least k_1-connected. The indexing strategy described above results in no links being established between nodes in $\mathcal{N}_2$ in the k_1-connected network. In order to ensure the agents in $\mathcal{N}_2$ are at least k_2-connected, we construct an additional $(k_1 - k_2)$-Harary network solely with the nodes in $\mathcal{N}_2$. These two steps define all the edges we aim to maintain in $\mathcal{E}_c$. For clarity, we summarize the developed algorithm for constructing a (k_1, k_2)-connected network in Algorithm 1.

B. Optimal Control for Heterogeneous Resilience

Once the desired network configuration is obtained, then we can solve the minimally invasive control problem in (9)

Algorithm 1 (k_1, k_2)-Connected Network Construction

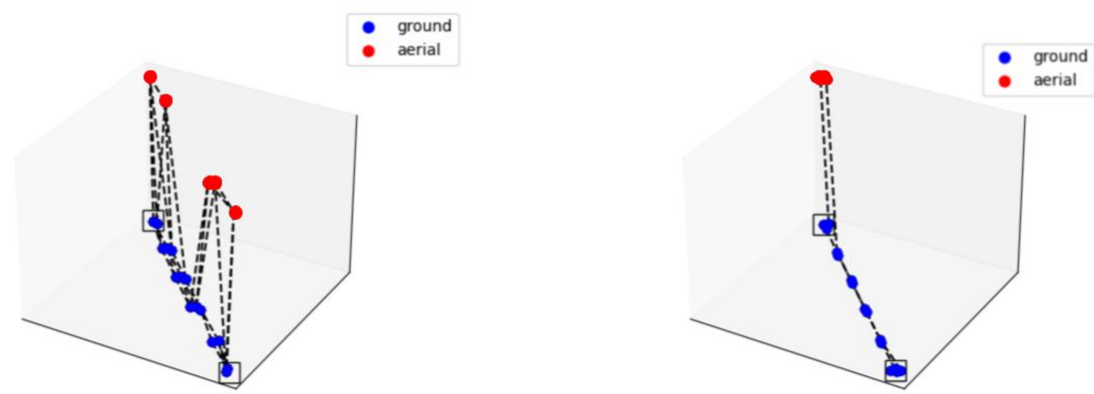
Input: $\mathbf{x}$
Output: $\mathcal{G}_c = \{\mathcal{N}, \mathcal{E}_c\}$
Clustering nodes $\forall i \in \mathcal{N}_1$ into $|\mathcal{N}_2| = n_2$ clusters: cluster $\mathcal{V}_j$ with centroid v_j^c, $j = 1, 2, ..., n_2$
Match every node $i \in \mathcal{N}_2$ to a cluster j using (13)
Determine index of each node $i \in \mathcal{N}$
Construct a k_1-connected network using Harary networks for all nodes $i \in \mathcal{N}$
Construct an additional $(k_2 - k_1)$-connected network using Harary networks solely for all nodes $i \in \mathcal{N}_2$
return $\mathcal{G}_c$



(a) $t = 10$ (b) $t = 100$ (c) $t = 500$

(d) $t = 10$ (e) $t = 100$ (f) $t = 500$

Fig. 3. (a) – (c) show how the robots disperse to tasks while maintaining 2-connectivity. (d) – (f) show the same process while maintaining $(1,2)$-connectivity. The blue dots represent ground agents, which maintain 1-connectivity, the red dots represent aerial agents, which maintain 2-connectivity, and the boxes are the targeted task areas.



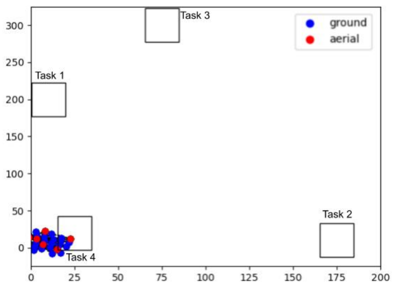
(a) $t = 500$ (b) $t = 500$

Fig. 4. (a) and (b) show the three-dimensional view of the 2-connected robotic network and $(1,2)$-connected robotic network, respectively.

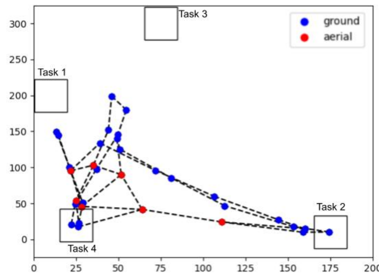
which is a quadratic program. Note that the network configuration achieving (k_1, k_2)-connectivity evolves over time due to the agents' dynamics, and thus Algorithm 1 is repeatedly leveraged to identify the real-time optimal network configuration. The configuration will be enforced when solving the optimal control $\mathbf{u}^*$ through a set of linear constraints based on the control barrier certificates. The proposed two-step heuristic approach provides a computationally feasible framework to enable the control design of the multi-agent system as the exact solution to the original problem is challenging to obtain.

IV. Case Studies

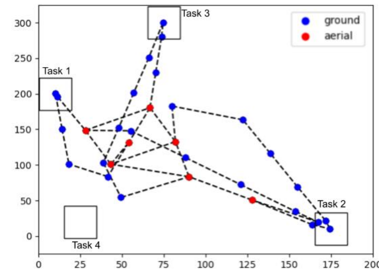
In this section, we use case studies to illustrate the collaborative resilient multi-agent control using the developed strategy.



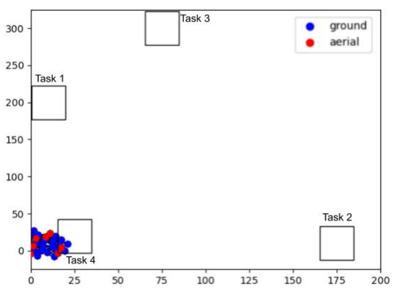
(a) $t = 10$



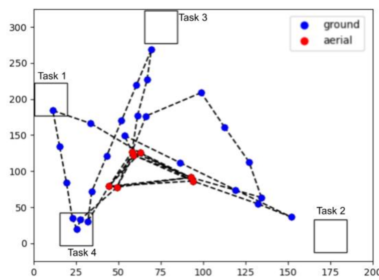
(b) $t = 200$



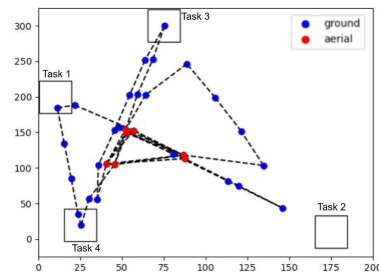
(c) $t = 1000$



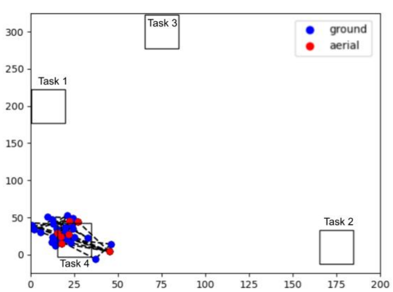
(d) $t = 10$



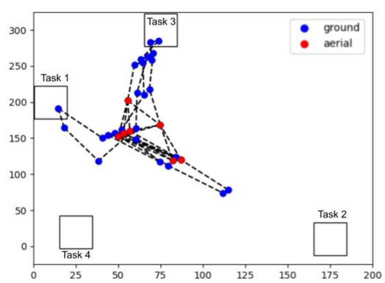
(e) $t = 500$



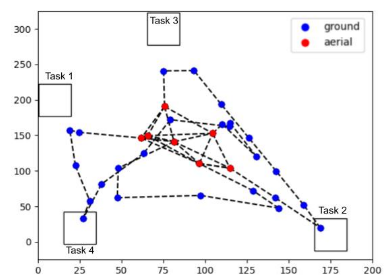
(f) $t = 2000$



(g) $t = 40$



(h) $t = 280$



(i) $t = 580$

Fig. 5. (a)-(c) show how the agents are distributed to tasks over time while maintaining $(1,2)$-connectivity. (d) - (f) show the distribution of agents over time with $(1,4)$-connectivity without dynamic task allocation. (g) - (i) show the distribution of agents while maintaining $(1,4)$-connectivity with dynamic task allocation. The blue dots represent ground agents which maintain 1-connectivity, the red dots represent aerial agents, which maintain 4-connectivity, and the boxes are the targeted task areas.

A. *Simulation Environment*

We start by considering a system with 20 agents consisting of air and ground robots, where $|\mathcal{N}_1| = 15$ and $|\mathcal{N}_2| = 5$. We set $r_c = 50$ and $r_s = 5$ as the radius of connectivity and radius of safety respectively. In this heterogeneous robotic system, the ground and aerial robots have connectivity requirements $k_1 = 1$ and $k_2 = 2$, respectively, i.e., we aim to maintain a (1,2)-connected network during the task execution of the agents. Note that each agent has a targeted location to reach according to the assigned task. We assume that there are two tasks in total. Thus, the prescribed $\hat{u}_i$ of robot i can be computed according to the agent's current location and its task location. Additionally, $\alpha_i = 1, \forall i \in \mathcal{N}$, i.e., the speed of each agent is limited to 1 m/s.

For comparison, we further investigate a scenario with $|\mathcal{N}_1| = 25$ ground agents and $|\mathcal{N}_2| = 7$ aerial agents. Specifically, we consider two cases: $k_1 = 1$ and $k_2 = 2$, and $k_1 = 1$ and $k_2 = 4$, in an environment with four tasks that each agent is randomly assigned to initially. In addition, we study the resilient autonomy under the dynamic task allocation, i.e., once an agent completes its currently assigned task, it will be assigned to a new one subsequently.

B. *k-Connectivity vs (k_1,k_2)-Connectivity*

Fig. 3 shows how the agents disperse through a goal-oriented environment. In Fig. 3(a) – Fig. 3(c), all the agents maintain the same level of connectivity with $k = 2$ ($k_1 = k_2 = 2$). We can see that among the 25 total agents, 6 of them are able to reach their assigned task location, while the other 19 agents act as a bridge between the two tasks such that the desired level of connectivity can be maintained. There are a total of 38 connections among the agents in this case study. In Fig. 3(d) – Fig. 3(f), the agents are required to maintain $(1,2)$-connectivity, i.e., the ground agents are 1-connected and the aerial agents are 2-connected. In this scenario, 17 agents reach their task locations while 8 of them act as a communication bridge between the tasks. It can be observed that by lowering the level of connectivity for the ground agents, more agents are able to complete their assigned tasks. In addition, under the developed heterogeneous resiliency framework, the less resilient aerial agents can maintain the same connectivity level as in the first example. This case study demonstrates that our proposed resiliency design is capable of satisfying the heterogeneous resilience requirements in the collaborative multi-agent system.

C. Resiliency vs Mobility Performance

We next examine the impacts of connectivity/resiliency requirements on the agent's ability to complete their task in larger swarms. In Fig. 5(a) – Fig. 5(c), the agents maintain $(1,2)$-connectivity. In this example, all tasks are reached by multiple agents over time, though there is no instant at which all tasks are fulfilled simultaneously. Under the requirement of $(1,4)$-connectivity, shown in Fig. 5(d) – Fig. 5(f), agents are only able to visit three out of the four tasks successfully. In this example, 8 out of the total 32 agents reach their goal locations after $t = 2000$. To address the issue that not every goal location is reached, we investigate a scenario with dynamic task allocation. In this case study, every task location can be visited by more than one agent multiple times as described in Section IV-A. The results are shown in Fig. 5(g) – Fig. 5(i). In total, 25 out of the 32 agents can reach the assigned goal location. We also note that none of the aerial agents from $\mathcal{N}_2$ are able to reach the desired goal location, which is due to the high level of connectivity they are required to preserve. Another remark is that an agent's ability to get to their assigned task locations is dependent on the network schema and the environment layout. We show that even in the most challenging circumstances, i.e., a high level of connectivity and wide-spread task locations, the designed two-layer multi-robot system is able to maintain (k_1,k_2)-connectivity while still having a set of agents reach their goals.

V. Conclusion

In this paper, we have developed a framework for achieving heterogeneous resiliency in multi-layer swarm systems. This has been enabled by maintaining a (k_1,k_2)-connected robotic network in which agents in subnetwork 1 are required to be k_1-connected while those in subnetwork 2 are k_2-connected over the entire task execution time interval. The adopted connectivity and safety barrier certificates have been shown effective and convenient in transforming the (k_1,k_2)-connected network configuration and collision avoidance requirements into a set of linear constraints. Case studies have demonstrated that the proposed methodology is successful in yielding a collaborative multi-layer resilient autonomous system. Future work includes the development of decentralized control for multi-layer resilient autonomy under the adversarial environment.

VI. Acknowledgements

This research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-21-2-0281. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

References

[1] Y. Zhao, B. Huang, J. Yu, and Q. Zhu, "Stackelberg strategic guidance for heterogeneous robots collaboration," in *International Conference on Robotics and Automation (ICRA)*, 2022, pp. 4922–4928.

[2] J. Zhang, R. Liu, K. Yin, Z. Wang, M. Gui, and S. Chen, "Intelligent collaborative localization among air-ground robots for industrial environment perception," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 12, pp. 9673–9681, 2018.

[3] J. Chen and Q. Zhu, "Control of multilayer mobile autonomous systems in adversarial environments: A games-in-games approach," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 3, pp. 1056–1068, 2019.

[4] Q. Zhu, S. Rass, B. Dieber, and V. M. Vilches, "Cybersecurity in robotics: Challenges, quantitative modeling, and practice," *Foundations and Trends® in Robotics*, vol. 9, no. 1, pp. 1–129, 2021.

[5] M. Abomhara and G. M. Køien, "Cyber security and the internet of things: vulnerabilities, threats, intruders and attacks," *Journal of Cyber Security and Mobility*, pp. 65–88, 2015.

[6] J. Chen, C. Touati, and Q. Zhu, "A dynamic game approach to strategic design of secure and resilient infrastructure network," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 462–474, 2019.

[7] M. Zavlanos and G. Pappas, "Controlling connectivity of dynamic graphs," in *Proceedings of the 44th IEEE Conference on Decision and Control*, 2005, pp. 6388–6393.

[8] N. Chakraborty and K. Sycara, "Reconfiguration algorithms for mobile robotic networks," in *IEEE International Conference on Robotics and Automation*, 2010, pp. 5484–5489.

[9] M. M. Zavlanos, M. B. Egerstedt, and G. J. Pappas, "Graph-theoretic connectivity control of mobile robot networks," *Proceedings of the IEEE*, vol. 99, no. 9, pp. 1525–1540, 2011.

[10] W. Burgard, M. Moors, C. Stachniss, and F. Schneider, "Coordinated multi-robot exploration," *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 376–386, 2005.

[11] W. Luo and K. Sycara, "Minimum k-connectivity maintenance for robust multi-robot systems," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 7370–7377.

[12] W. Luo, N. Chakraborty, and K. Sycara, "Minimally disruptive connectivity enhancement for resilient multi-robot teams," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020, pp. 11 809 – 11 816.

[13] A. Cornejo and N. Lynch, "Fault-tolerance through k-connectivity."

[14] P. Szczytowski, A. Khelil, and N. Suri, "Dkm: Distributed k-connectivity maintenance in wireless sensor networks," in *2012 9th Annual Conference on Wireless On-Demand Network Systems and Services (WONS)*, 2012, pp. 83–90.

[15] M. Ji and M. Egerstedt, "Distributed coordination control of multiagent systems while preserving connectedness," *IEEE Transactions on Robotics*, vol. 23, no. 4, pp. 693–703, 2007.

[16] Z. Nutov, "The k-connected subgraph problem," in *Handbook of Approximation Algorithms and Metaheuristics*, 2018.

[17] D. Angluin, J. Aspnes, and L. Reyzin, "Network construction with subgraph connectivity constraints," *J. Comb. Optim.*, vol. 29, no. 2, p. 418–432, feb 2015.

[18] S. K. Gupta, P. Kuila, and P. K. Jana, "Genetic algorithm approach for k-coverage and m-connected node placement in target based wireless sensor networks," *Computers & Electrical Engineering*, vol. 56, pp. 544–556, 2016.
[19] M. S., S. Udgata, and S. Sabat, "M-connected coverage problem in wireless sensor networks," *Sensor Networks*, vol. 2012, 03 2012.
[20] J. Chen, C. Touati, and Q. Zhu, "Optimal secure two-layer IoT network design," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 1, pp. 398–409, 2020.
[21] D. Pickem, P. Glotfelter, L. Wang, M. Mote, A. Ames, E. Feron, and M. Egerstedt, "The robotarium: A remotely accessible swarm robotics research testbed," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 1699–1706.
[22] U. Borrmann, L. Wang, A. D. Ames, and M. Egerstedt, "Control barrier certificates for safe swarm behavior," *IFAC-PapersOnLine*, vol. 48, no. 27, pp. 68–73, 2015.
[23] S. Skiena, *Implementing Discrete Mathematics: Combinatorics and Graph Theory with Mathematica*. Addison-Wesley Longman Publishing, 1991.
[24] F. Harary, "The maximum connectivity of a graph," *Proceedings of the National Academy of Sciences*, vol. 48, no. 7, pp. 1142–1146, 1962.