Resilient Consensus in Robot Swarms With Periodic Motion and Intermittent Communication

Xi Yu, *Member, IEEE*, David Saldaña, *Member, IEEE*, Daigo Shishika, *Member, IEEE*, and M. Ani Hsieh, *Member, IEEE*

***Abstract*—In this article, we propose an approach to construct a time-varying communication topology with a resilient consensus performance for robot swarms with limited communication ranges. The robots are deployed to explore a large task space and achieve consensus despite the existence of a finite number of noncooperative members in the team. Existing methods encouraged robots to stay close to each other to achieve certain robustness requirements on the connectivity of the communication topology. We leverage on the robots' mobility to design a time-varying integrated topology composed of several subgroups of robots deployed on nonoverlapping closed-loop paths. Robots are spread out and move along the paths, forming periodic communication links within or across groups. We analyze the time-varying topology synthesized and provide sufficient conditions for individual subgroups and the interconnection between them. We show designs satisfying the conditions with simulated examples in a lattice space, as well as in a task space with predefined paths.**

***Index Terms*—Consensus, multiagent systems, resilience systems.**

I. Introduction

ROBOT swarms composed of a large number of robots that cooperatively measure, monitor, or patrol large environments have shown great interest in recent years [1]–[4]. Multiple robots can offer an efficient way to perform tasks by combining collective work in difficult environments with limitations in motion and communication. There is a growing interest in designing robot swarms that can work in a *resilient* way, such that the swarms can accomplish tasks in different challenging environments when part of the team becomes "noncooperative."

Manuscript received July 31, 2020; revised February 13, 2021; accepted May 10, 2021. Date of publication July 1, 2021; date of current version February 8, 2022. This article was recommended for publication by Associate Editor A. Prorok and Editor P. Robuffo Giordano upon evaluation of the reviewers' comments. This work was supported in part by the Army Research Laboratory under Grant DCIST CRA W911NF-17-2-0181 and in part by the Office of Naval Research under Award N00014-17-1-2690. *(Corresponding author: Xi Yu.)*

Xi Yu is with the Department of Mechanical and Aerospace Engineering, West Virginia University, Morgantown, WV 26506 USA (e-mail: xi.yu1@mail.wvu.edu).

David Saldaña is with the Autonomous and Intelligent Robotics Laboratory, Lehigh University, Bethlehem, PA 18015 USA (e-mail: saldana@lehigh.edu).

Daigo Shishika is with the Department of Mechanical Engineering, George Mason University, Fairfax, VA 22030 USA (e-mail: daigo.shishika@gmail.com).

M. Ani Hsieh is with the General Robotics, Automation, Sensing and Perception Laboratory, The University of Pennsylvania, Philadelphia, PA 19104 USA (e-mail: m.hsieh@seas.upenn.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TRO.2021.3088765.

Digital Object Identifier 10.1109/TRO.2021.3088765

One of the important joint tasks that a swarm usually faces is the consensus problem [5]–[7], which arises in scenarios where different team members are trying to provide a collective estimation or decision of a global variable. The convergence of the whole team's estimation to a common value is realized by every robot exchanging information with its neighbors and updating its estimation based on the information acquired through communication. A robot may have many different types of "noncooperative" behaviors. One of them that is hard to detect is broadcasting corrupted information to its neighbors, causing its team members to fail to converge or hijack the team's consensus to a value that is way off. A partial failure in a robot swarm can be a disruption of the communication network or the malfunction of a subset of the robots due to internal disturbance or external attacks. In either way, the effect of spreading wrong information through the network can generate a snowball effect leading the swarm to a failure in its task. It is usually hard to identify and reject this type of noncooperative robots. A robot swarm that is resilient against such type of failure must have the capability to achieve consensus with the existence of noncooperative team members.

There have been a myriad of algorithms solving consensus problems [8]–[11]. Existing methods overcome the effect of these noncooperative robots by increasing the connectivity of the team [12], [13]. While the communication ranges of robots are usually limited, deploying any robot in a persistent connection with many of its teammates may cause a lot of robots conglomerate in a small space and severely reduces the coverage of the task space with the same number of robots.

Saldaña *et al.* [12] have shown that any network that is resilient against a finite number of noncooperative robots can maintain its resilience if an arbitrary subset of the communication links becomes *periodic*. Leveraging on this result, we allow robots in our swarm to disconnect periodically from *some* current neighbors, travel around, and seek additional *periodic* connections with those who are not in its immediate proximity. Their motion should be carefully planned, such that all the connections are preserved in a periodic fashion with a common period, and the robots are able to reach consensus while covering a large area that needs to be taken care of persistently.

When robots move around in the task space, they come in and out of communication range with other team members. The communication link formed across them are temporal and vary over time. We say these robots jointly synthesize a communication network that preserves an *intermittent connectivity*. One of

the earliest works considering this concept is [14], where the full connectivity of a team of robots is periodically regained by letting the robots travel far from each other's communication range but regularly gather in certain locations. In [15] and [16], periodically occurred communication links between disconnected subgroups were generated by robots traveling across these subgroups, while in [17], the robots were intermittently connected through moving back and forth on a ring. It was shown in [18] that robots deployed on different predefined routes can form a periodic time-varying communication network, which can be mapped to an equivalent static network, where standard graph analyzing tools can be used. All these works have focused on the synthesis of a network that is *connected* when time accumulates. For reaching resilient consensus, staying in touch with the team is not enough. The underlying topology of the time-varying communication network must meet a more structured feature, namely, the r-robustness, which will be elaborated in Section II. We leverage the techniques of synthesizing periodic intermittent connections to *enhance* the connectivity of the robotic network in a periodic fashion, such that the feature of r-robustness can be met over time.

There are various existing algorithms, such as the weighted mean-subsequence-reduced (W-MSR) algorithm presented in [19]–[21], following which the cooperative robots in a team can converge their estimation to a value within the convex hull of their initial measurements despite the existence of a finite number of noncooperative ones in the team. When the communication topology is given and the highest possible number of noncooperative robots in one's neighborhood is known, a method was presented in [21]–[24] to analyze the structured connectivity of this topology and, therefore, evaluate whether this team is able to deliver a resilient consensus performance. However, the analysis simply requires going over every possible binary grouping of the whole team, which is computationally inefficient [21], [23]. When it comes to our proposed method of establishing time-varying communication links by routing the robots in the task space, it becomes extremely hard to manually define the motion of each robot [12] such that the conditions on the robustness of the communication topology are met. Every time an adjustment is made in the robot team in the environment, new trajectories need to be defined, and the connectivity of the topology and its robustness need to be analyzed and verified all over again.

There are some successful attempts to simplify the verification of the robustness of the topology. The authors of [22] and [24] proposed to use algebraic connectivity of the communication topology as a verification. According to their results, a threshold can be determined, such that the topology is guaranteed to meet the robustness conditions for delivering resilient consensus performances if its algebraic connectivity exceeds the threshold. However, computing algebraic connectivity itself is still expensive, especially when the number of nodes scales up. Besides, increasing algebraic connectivity is simply driving all robots to form a tight cluster, such that every robot can have as many neighbors as possible.

The authors of [13] and [25] proposed the design of well-defined repeating structures that can be composed to create scalable static networks. An elemental formation, in this case, a multilayer hexagon in a lattice space with a robot deployed on each node of each layer, is designed, and its robustness is tested and proved. The authors then showed a synthesis method by sweeping the hexagon along a connected trajectory around the task space. The required robustness in the topology is proved to be preserved by doing so. This method provides a way to compose a desired robust topology for a large number of robots by construction, rather than running expensive verification algorithms. This technique also keeps the robots away from clustering into tight formations, but necessitates a large communication range since the number of noncooperative robots every robot can handle purely depends on this parameter. The authors of [26]–[28] introduced heterogeneous sets of robots to build robotic networks satisfying the robustness conditions. Such methods require a certain number of always-trusted robots that are not universally available in real-world applications.

In our earlier work [29], we proposed an alternative approach to build a robust topology for robot swarms deployed in a lattice task space. We designed modular closed-loop paths with robots circulating on it. This "circulating modules" can repeat itself and cover a task space just like tessellating tiles of the same shape. The shapes of the modular paths are carefully chosen such that the robustness of the topology is preserved during such "tessellation." This method is easy to scale up and does not require a large communication range of each robot. In the swarm it synthesized, the robots can stay barely in its neighbor's communication range, such that the conglomeration of robots in a small area is avoided.

It is worth noting that in [29], such a structured robust topology is guaranteed by requiring every robot module to circulate on a closed-loop path of exactly the same shape. The shape of the path is also carefully designed to consist of special geometries to enable a high intraconnectivity in each subgroup. Such realization, although with great potentials, may not fit for a variety of environments. In reality, robots patrolling a task space may form multiple subgroups, but each subgroup usually follows predefined or semidefined routes of different shapes.

In this article, we propose an "integrated" robot swarming and routing strategy. The robot swarm synthesized is composed of multiple subgroups, each deployed on a predefined closed-loop path. The number of robots in each subgroup and the common circulating period are decided based on a global knowledge of the task space. Each subgroup will realize a high intragroup connectivity by tuning every robot's speed periodically and reserve sufficient opportunities for periodic cross-group communications to take place in designated "interfacing sections." The intragroup and cross-group connections will jointly synthesize a time-varying network that delivers the resilient consensus performance we desire.

The rest of this article is organized as follows. Section II explains how resilient consensus is achieved and elaborates the robustness conditions on the topology to guarantee a resilient consensus performance. Section III presents our approach. Section IV provides conditions on our design to guarantee a resilient consensus performance. Section V and VI shows how this method works in different types of environment with

different sets of constraints. Some results in Section V have been published in [29]. Section VII shows a simulation example supporting the results in Section V and a simulation example supporting the main results in Sections IV and VI. The second one has not been published before. Finally, Section VIII concludes this article.

II. Problem Statement

Consider a scenario that many robots are deployed and circulating along multiple closed-loop paths. Each robot generates a value of a measurement (e.g., temperature, salinity in the water body, or pollution level in the air) or a decision (e.g., desired altitude for air vehicles or task duration). Pairs of robots can communicate and exchange their values while they are within each other's communication range. We assume that robots follow the communication disk model with a range of R. All robots deployed on this task need to agree on a common value within the convex set spanned by everyone's original values.

Typical environments for robot applications, such as urban neighborhoods, underground pits, postdisaster ruins, or ocean surfaces, contain obstacles or nontrespassing areas. Some of them consist of a collection of predefined routes that robots can move along, such as the streets in a neighborhood, the tunnel in a pit, or the current flow in the ocean. While a pair of moving robots enter each other's neighborhood, a temporal connection is activated between them, providing an opportunity to exchange information. All temporal connections activated and deactivated over time together form a time-varying communication network across all robots.

To reach consensus, the robots repeatedly acquire values from their neighbors in a time-varying network. Each robot processes its value according to predesigned consensus algorithms and broadcasts the result to its neighbors. If a robot does not follow the same update rule or fails to broadcast the correct result, we say it is *noncooperative*. While one or more robots may be compromised and become noncooperative, they may still circulating the route, as usual, such that the rest of the team cannot identify them. The team will continue using the fake values broadcast by the noncooperative ones, and the whole team's consensus performance could be interrupted or hijacked.

Our goal is to design routing and swarming strategies for these robots to reach *resilient consensus*. That is to say, despite the existence of a finite number of noncooperative robots, the whole team can still reach consensus by synthesizing a time-varying communication network that processes certain properties.

A. Resilient Consensus

In this section, we introduce definitions and preliminary results on resilient consensus. Consider an undirected communication graph formed by a set of N robots, $\mathcal{V} = \{1, 2, \ldots, N\}$. The time-varying connections (capability of communication) between robots are described with the edge set $\mathcal{E}[t]$, where $\mathcal{E}[t] \subseteq \mathcal{V} \times \mathcal{V}$ for any $t \in \mathbb{Z}_{\geq 0}$. The communication topology is represented by a sequence of graphs $\mathcal{G}[t] = (\mathcal{V}, \mathcal{E}[t])$. We denote the neighbors of node i at time t as $\mathcal{N}_i[t] = \{j | (i, j) \in \mathcal{E}[t]\}$.

Each robot has a scalar value $z_i[t] \in \mathbb{R}$ at time step t. This value represents an estimation variable, heading direction, inter-robot distance, or any local variable that involves global coordination. The initial value is denoted by $z_i[0]$. At each time step, robots can update their values based on their values and the values from their neighbors. We say that the robot network *achieves consensus* when all robots converge to a similar value, i.e., $z_1[t] \approx \ldots \approx z_N[t]$, when t goes to infinity.

Some robots in the team may fail to follow the update rules and send out erroneous information to their neighbors. These robots are referred to as *noncooperative* ones. To avoid the whole team being affected or manipulated by those noncooperative robots, a particular consensus algorithm has been introduced in [20]. Let F denote the maximum possible number of noncooperative robots in any robot's neighborhood. The W-MSR algorithm [20] requires that at every time step t, every cooperative robot i creates a sorted list of all the values it received from its neighbors and compares them with $z_i[t]$. The W-MSR algorithm removes F highest and lowest values in the list before computing the updated value $z_i[t+1]$.

For time-varying networks, a sliding-time-window version of the W-MSR algorithm, abbreviated as SW-MSR, was introduced in [12] to update $z_i[t]$ based on the most recent values received from all robots that have communicated with i within the last k steps. After removing F values in the sorted list, the set of the values left is denoted by $\mathcal{R}_i[t]$. The update rule of $z_i[t]$ is given by $z_i[t+1] = w_{ii}[t] z_i[t] + \sum_{j \in \mathcal{R}_i[t]} w_{ij}[t] z_{ij}^k[t]$, where $z_{ij}^k[t]$ denotes the most recent value of $z_j[t]$ that has been received by i during $(t-k, t]$. w_{ii} and w_{ij} are weights that are lower-bounded by some $\alpha \in (0, 1/2)$ and satisfy

$$w_{ii}[t] + \sum_{j \in \mathcal{R}_i[t]} w_{ij}[t] = 1 \quad \forall i \in \mathcal{V}.$$

B. Time-Varying r-Robust Topology

An important sufficient condition for both W-MSR and SW-MSR to converge asymptotically toward consensus is established on the r-*robustness* of the topology. In this subsection, we present definitions of r-robustness for both static and time-varying graphs, together with theorems that are useful in determining r-robustness for both types of graphs.

Definition 2.1 (r-robust graph): A static graph $\mathcal{G}$ is r-*robust* if for any pair of nonempty, disjoint subsets of $\mathcal{V}$, at least one subset of them, denoted as $\mathcal{S}$, contain a node $i \in \mathcal{S}$, such that $|\mathcal{N}_i \backslash S| \geq r$.

For a time-varying topology, we define its robustness as follows.

Definition 2.2 (Time-varying r-robust graph): Given $k \in \mathbb{Z}_{>0}$, a dynamic graph $\mathcal{G}[t]$ is (k, r)-robust if the union $\cup_{\tau=0}^{k} \mathcal{G}[t-\tau]$ is r-robust for all $t \geq k$.

It was shown in [20] that resilient consensus could be achieved through the W-MSR algorithm if the underlying communication topology is r-robust. The existing results for the time-varying network are reviewed in the following.

Theorem 2.1 (see [12]): Given a communication network $\mathcal{G}[t]$, resilient asymptotic consensus is achieved by the SW-MSR update rule in the presence of F noncooperative robots in any robots' neighborhood, if $\exists\, k \in \mathbb{Z}_{>0}$, such that $\mathcal{G}[t]$ is $(k, 2F+1)$-robust.

Determining whether a given graph is r-robust or not is a co-NP-complete problem [21]; however, there are some results that define a systematic way to construct large-scale graphs that are r-robust. Notice that a time-varying graph is r-robust if the union of all its temporal topology is an r-robust static graph. The following theorems are introduced to check the r-robustness for static graphs and can, therefore, be used to check the unions of the temporal topology of time-varying graphs.

Theorem 2.2 (F-elemental graph [30]): A static graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $4F+1$ nodes is $(2F+1)$-robust, if it satisfies the following.

1) $\exists \mathcal{S} \subset \mathcal{V}$, such that $|\mathcal{S}| = 2F$ and $\mathcal{N}_i \cup \{i\} = \mathcal{V}$ for all $i \in \mathcal{S}$.
2) $\mathcal{V}/\mathcal{S}$ is a connected graph.

A graph that satisfies those conditions is called *F-elemental graph*. Robust graphs, including F-elemental graphs, can extend their number of vertices based on the following conditions.

Theorem 2.3 (see [19]): Suppose $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is an r-robust static graph. Consider a new vertex v^*, and $\mathcal{E}^* \subseteq \{(v^*, j) | j \in \mathcal{V}\}$. Then, $\mathcal{G}^* = (\mathcal{V} \cup \{v^*\}, \mathcal{E} \cup \mathcal{E}^*)$ is also r-robust if $|\mathcal{N}_{v^*}| \geq r$.

Theorems 2.1 and 2.2 provide a set of properties that any network possessing such properties in the unions of their temporal topology can reach resilient consensus against a certain number of noncooperative robots. Theorem 2.3 pointed a method to expand a network by adding new robots without losing its capability of reaching resilient consensus. Our approach introduced in the next section is based on the three theorems together.

III. Integrated Topology of Robot Swarms

We approach the problem introduced in Section II by designing an integrated communication topology that consists of a collection of networked subteams. The subteams are indexed from $I = 1, \ldots, M$. Subteam I consists of n_I robots indexed as $i^I = 1^I, \ldots n_I^I$ and deployed on a closed-loop path with no self-intersection. The closed paths of different subteams do not overlap or cross each other. The path of subteam I is described by an arc-length-parameterized curve $\boldsymbol{\gamma}_I : [0, L_I) \mapsto \mathbb{R}^2$, where L_I is the arc length of the path.

Different from the work in [17], the robots considered here always move in the same direction along the path if they are in the same subgroup. We index them along their moving direction, such that robot $i+1$ is the closest robot ahead of robot i. A universal speed control strategy $v_i = f_I(s_i)$, with $s_i \in [0, L_I)$ representing a location on the curve I and $f_I : [0, L_I) \mapsto \mathbb{R}$, is designed for all robots in the same subteam, such that every robot will pass the same position on the path with the same speed. Therefore, all robots in the same subteam complete a cycle of the path with the same periodicity T_I. We refer to the arc length between any two consecutive robots i and $i+1$ as the spacing between i and $i+1$. Each spacing varies periodically and is always within a range $\delta_{i,i+1}(t + T_I) = \delta_{i,i+1}(t) \in (0, \Delta]$, satisfying $\Delta \leq R$. Thus, the robots on the same path always keep the same order.

Let the position of robot i on the path I at time t be denoted by $s_i(t) \in [0, L_I)$ based on the arc-length parameterization, and let the Cartesian coordinate be $\boldsymbol{\gamma}_I(s_i(t))$. Without loss of generality, we define $\boldsymbol{\gamma}_I$ so that the robots' motion is in the positive direction of the arc length. That is to say, after a small amount of time δt, robot i's position on the path will be $s_i(t) + v_i(t)\delta t$, with $v_i(t) = f(s_i(t))$ indicating its velocity at time t. If a pair of robots (i, j) are on the same path I, we say that i is $\tilde{s}_{ij}(t)$ ahead of j at time t, with $\tilde{s}_{ij}(t) = s_i(t) - s_j(t)$. For simplicity in the rest of this article, we use $\tilde{s}_{ij}(t) = (\tilde{s}_{ij}(t) \mod L_I)$ for all s_i and s_j on path I. The arc length between i and j at t is then $\min(\tilde{s}_{ij}(t), \tilde{s}_{ji}(t))$. The Euclidean distance between i and j at t is $||\boldsymbol{\gamma}_I(s_i(t)) - \boldsymbol{\gamma}_I(s_j(t))||$. Each subteam and its path are defined as follows.

Definition 3.1 (Circulating module): A *circulating module*, or a *module*, is a tuple of a closed path with no self-intersection and a team of robots. Every robot moves along the path in the same direction and passes the same position on the path with the same speed, and the spacing between any consecutive robots is bounded by $\delta(t) \in (0, \Delta]$ with $\Delta \leq R$.

A. Inter-Robot Connection

Inter-robot communication occurs when the Euclidean distance between a pair of robots drops below the communication range of R. Note that this condition may be true for robots that are in the same module or different modules. For the robots in the same module I, we always have $||\boldsymbol{\gamma}_I(s_i)(t) - \boldsymbol{\gamma}_I(s_j)(t)|| \leq \min(\tilde{s}_{ij}(t), \tilde{s}_{ji}(t))$. If $\max_{t \in [0, T_I)}(\min(\tilde{s}_{ij}(t), \tilde{s}_{ji}(t))) \leq R$, robots i and j are guaranteed to be connected *at every time step*, which we call a *persistent* connection.

For pairs with a spacing that is not always smaller than R, and pairs in different modules, the communication links may temporarily be created when they enter certain sections of their paths. The proximity can be generated due to either the decrease of arc length between robots or the geometry of the modules. Since communication is only possible at a certain timing, we call this a *temporal* connection. Fig. 1 illustrates both types of connections. If both robots enter this section periodically, and they have a common period, the temporal connection between them is also periodic. Since robots in the same module have the same period of motion, all inter-robot connections within the same module are periodic.

Preserving persistent connections across nonconsecutive robots requires a large communication range and is not always feasible. In this article, we require persistent connections only between the closest neighbors on the path (i.e., between robot i and robot $i \pm 1$), such that every module is guaranteed to be *connected*.

B. Cross-Module Connection

A robot i in module I can communicate with robot j from a different module J if at some time step, both i and j find themselves in communication range with each other. If there

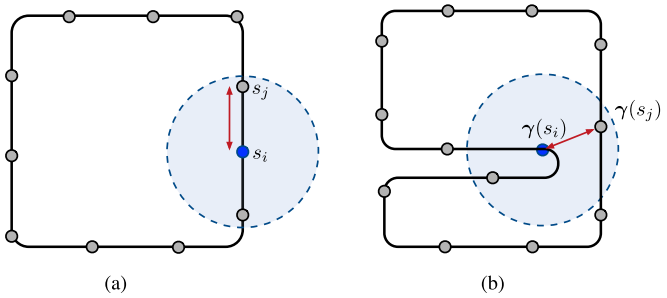


Fig. 1. Examples illustrating the two types of inter-robot connection. (a) Persistent connection. The arc length between i and j is shorter than the communication range. All robots in the light blue circle remain connected at all times when they circulate the path. (b) Temporal connection. The arc length between i and j is longer than the communication range, but they are able to communicate temporally, and this connection occurs every round of their circulation. (a) Persistent connection. (b) Temporal connection.

exists at least one such pair of robots, we say that modules I and J are *periodically connected*. In the rest of this section, we describe how this intermodule connection is formed.

Given a pair of circulating modules I and J, let $\mathbf{l}_1 = [s_1, s_1 + l_1]$ be a segment on path I and $\mathbf{l}_2 = [s_2, s_2 + l_2]$ be a segment on path J, where l_1 and l_2 are the arc lengths of the segments. Similar to the definition of $\boldsymbol{\gamma}_I$, if $s_1 + l_1 \geq L_I$, then $\mathbf{l}_1 = [s_1, s_1 + l_1]$ represents the segment of $[s_1, L_I) \cup [0, (s_1 + l_1 \mod L_I)]$, and the same for $\mathbf{l}_2$. We define *interfacing section* as follows.

Definition 3.2 (Interfacing section): Consider a pair of circulating modules I and J with segments $\mathbf{l}_1$ and $\mathbf{l}_2$, respectively. We say $\mathbf{l}_1$ and $\mathbf{l}_2$ are *interfacing sections* if for all $\sigma_1 \in [0, l_1]$ and $\sigma_2 \in [0, l_2]$, $\exists t_{\sigma_1}, t_{\sigma_2} \geq 0$ such that

$$||\boldsymbol{\gamma}_I(s_1 + \sigma_1 + \int_{t=0}^{t_{\sigma_1}} f_I(s_1(t))\delta t) - \boldsymbol{\gamma}_J(s_2 + \int_{t=0}^{t_{\sigma_1}} f_J(s_2(t))\delta t)|| \leq R \quad (1)$$

$$||\boldsymbol{\gamma}_J(s_2 + \sigma_2 + \int_{t=0}^{t_{\sigma_2}} f_J(s_2(t))\delta t) - \boldsymbol{\gamma}_I(s_1 + \int_{t=0}^{t_{\sigma_2}} f_I(s_1(t))\delta t)|| \leq R \quad (2)$$

where $s_1(t)$ denotes the position at t of any robot starts at s_1 at $t = 0$, and so does $s_2(t)$. R is the communication range. f_I and f_J are the universal control laws designed for all robots in modules I and J, respectively. Under such control laws, all robots in the same module will always pass the same location on the path with the same speed.

Equations (1) and (2) require that, when a robot i enters $\mathbf{l}_1$, every robot in $\mathbf{l}_2$ is expected to enter robot i's communication range soon after. The same for any robot that enters $\mathbf{l}_2$.

Notice that robots on I and J may travel in reverse directions or the same direction when they are on the interfacing sections depending on how the modules are arranged. In the reverse directions case, a robot i in module I traversing through $\mathbf{l}_1$, it will be able to communicate with every robot in module J that *appears on* $\mathbf{l}_2$ during i's dwelling time in $\mathbf{l}_1$. The number of robots from J that can be connected with i could be determined according to the control law and the distribution of robots on both paths. Fig. 2 illustrates the interfacing sections with robots traveling on them in reverse directions.

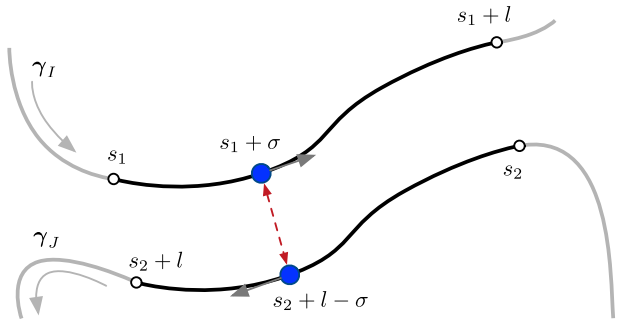


Fig. 2. Illustration of the interfacing sections.

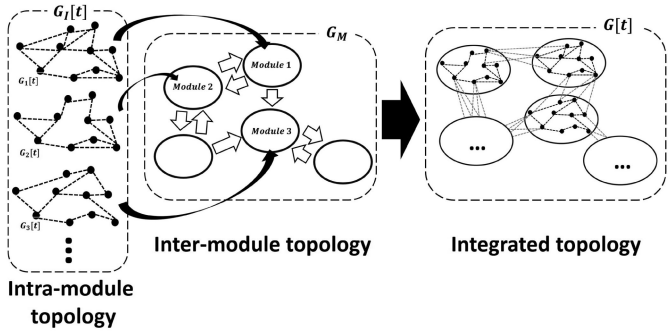


Fig. 3. Three types of graphs describing the time-varying topology. The left column shows the intramodule topology of each circulating module. The connections between robots from different modules are captured by the weighted links in the intermodule topology shown in the middle, with each node representing a module rather than a robot. All intramodule connections and cross-module connections together form the integrated topology, as shown in the right column.

If I and J have a shared common period, the connection between i and j will be periodic, and we consider I and J *periodically connected*. That is to say, the temporal connections that occurred between a pair of robots, one from each module, are reoccurring periodically. We can count the number of periodic cross-module connections a robot established with a given neighboring module.

Definition 3.3 (Interfacing index): The *interfacing index* of robot i in I with module J, denoted by $D_i(J)$, is the number of robots in J that i periodically connects with. Module I's *interfacing index* with module J is $D_I(J) = \min\{D_i(J)\}$ for all robots in module I.

Notice that for a pair of neighboring modules I and J, $D_I(J)$ does not necessarily equal to $D_J(I)$.

C. Three Types of Topology in Our Design

All the intramodule connections and cross-module connections together form a time-varying network across all the robots deployed in the task space. If there is a common period for all the modules, then the modules jointly synthesize a periodically time-varying topology. We use three types of graph to describe the different levels of the topology of a swarm. We illustrate the graphs in Fig. 3.

The *intramodule topology* of module I is a periodically time-varying *undirected* graph defined as $\mathcal{G}_I[t] = (\mathcal{V}_I, \mathcal{E}_I[t])$, where $\mathcal{V}_I = 1^I, \ldots n_I^I$, and $\mathcal{E}_I[t] \subseteq \mathcal{V}_I \times \mathcal{V}_I$ contains the active connections between any pair of robots in module I at time t. $\mathcal{G}_I(t + T_I) = \mathcal{G}_I(t)$, where T_I is the period of I. The static undirected graph $\mathcal{G}_I^T = (\mathcal{V}_I, \mathcal{E}_I^T)$, where $\mathcal{E}_I^T = \cup_{t=0}^{T_I} \mathcal{E}_I(t)$, contains all periodic connections activated over time in $\mathcal{G}_I(t)$. We call $\mathcal{G}_I^T$ the *accumulated intramodule topology*.

The *intermodule topology* is a *directed* weighted graph defined as $\mathcal{G}_M = (\mathcal{V}_M, \mathcal{E}_M)$, where $\mathcal{V}_M = 1, \ldots M$, and $\mathcal{E}_M \subseteq \mathcal{V}_M \times \mathcal{V}_M$. Every node I in the intermodule topology represents a module I, and every directed edge (I, J) has a weight equal to the interfacing index $w(I, J) = D_I(J)$. This graph gives an idea of how tightly a module is connected with its neighboring modules.

The *integrated topology* is a time-varying *undirected* graph defined as $\mathcal{G}[t] = (\mathcal{V}, \mathcal{E}[t])$, where $\mathcal{V}$ contains all robots in the swarm, and $\mathcal{E}[t]$ contains all inter-robot connections activated at t. If $\mathcal{G}_M$ is a connected graph, then there exists an overall common period across all modules, since every connected pair of modules in $\mathcal{G}_M$ have a common period. We write that overall common period as T, and we have $\mathcal{G}(t + T) = \mathcal{G}[t]$. Similar to the intramodule topology, we write the *accumulated integrated topology* $\mathcal{G}^T = (\mathcal{V}, \mathcal{E}^T)$, where $\mathcal{E}^T = \cup_{t=0}^{T} \mathcal{E}[t]$ to describe all periodic connections activated over time in $\mathcal{G}[t]$.

Notice that we only consider periodic cross-module connections; therefore, any pair of adjacent modules must have a common period, and so do the whole team if the intermodule topology is connected. Having a common period does not necessarily mean that all modules must follow the same period. A common period can be the common multiple of all modules' periods. For the easiness of discussion, we require all robots to have the same circulating period in the rest of this article.

In the next section, we analyze the communication topology of this swarm with the help of these graphs and provide conditions to guide the construction of an integrated topology that guarantees the resilient consensus performance of the whole team.

IV. Guaranteed Resilience by Construction

According to Theorem 2.1, a time-varying communication topology $\mathcal{G}[t]$ with a period T achieves resilient consensus in the presence of F noncooperative robots in one's neighborhood if $\mathcal{G}^T = \cup_{\tau=0}^{T} \mathcal{G}[t - \tau]$ is $(2F + 1)$-robust. Therefore, a robot swarm is able to achieve a resilient consensus if its accumulated integrated topology is $(2F + 1)$-robust. We consider that such a robot swarm has a $(2F + 1)$-*robust topology*.

In the rest of this section, we propose a construction of robot swarms with a $(2F + 1)$-robust topology. We start from a core topology fortified by the concept of F-elemental graph (see Theorem 2.2), and we expand it by adding modules in such a way that preserves the $(2F + 1)$-robustness.

Definition 4.1 (Core topology): We say that a composition of connected modules forms a $(2F + 1)$-*core topology* if the accumulated integrated topology of this composition is $(2F + 1)$-robust.

A. Construction of the $(2F + 1)$-Core Topology

An F-elemental graph contains $4F + 1$ nodes. If a module contains at least $4F + 1$ robots, an F-elemental graph may be generated within a single module. A sufficient condition to generate an F-elemental graph within this single module is described as follows.

Lemma 4.1: A circulating module I forms a $(2F + 1)$-core topology if its intramodule topology $\mathcal{G}_I^T = (\mathcal{V}_I, \mathcal{E}_I^T)$ is a complete graph, and $|\mathcal{V}_I| \geq 4F + 1$.

Proof: Choose an arbitrary subset $\mathcal{V}_F \subseteq \mathcal{V}_I$ that contains $4F + 1$ nodes. Denote the accumulated intramodule topology of this subset as $\mathcal{G}_F^T$. Since $\mathcal{G}_I^T$ is a complete graph, $\mathcal{G}_F^T$ is a complete graph as well. According to Theorem 2.2, $\mathcal{G}_F^T$ is F-elemental and, therefore, is $(2F + 1)$-robust. For any $v^* \in \mathcal{V}_I \setminus \mathcal{V}_F$, we have $|\mathcal{N}_{v^*} \cap \mathcal{V}_F| \geq 2F + 1$. Since $\mathcal{G}_F^T$ is a subgraph of $\mathcal{G}_I^T$, according to Theorem 2.3, adding any node from $\mathcal{V}_I / \mathcal{V}_F$ back to the F-elemental graph will yield a new $(2F + 1)$-robust graph. The same logic is repeated until all nodes in $\mathcal{V}_I \setminus \mathcal{V}_F$ are added back, and the final topology $\mathcal{G}_I^T$ is $(2F + 1)$-robust as well.

Designing control and routing strategies for all the robots in a module to reach a complete communication topology could be hard especially when the number of robots is large. We show that it is also possible to create an accumulated intramodule topology that is F-elemental but not a complete graph. Consider the robot indexed as i^I. If i^I can periodically connected with the closest $\rho_i[I]$ robots on its both sides, (i.e., $(i^I, (i + p)^I), (i^I, (i - p)^I) \in \mathcal{E}_I^T, \forall p = 1, \ldots, \rho_i[I]$), we say that $\rho_i[I]$ is its *intramodule communication radius*. For all robots in module I, we compute $\rho_I = \min_{i^I = 1^I}^{n^I} \rho_i[I]$ and denote it as the minimum periodic communication radius of I. Notice that if $\rho_I \geq \frac{|\mathcal{V}_I| - 1}{2}$, module I's accumulated intramodule topology is complete and, therefore, F-elemental.

Lemma 4.2: A circulating module I forms a $(2F + 1)$-core topology if $|\mathcal{V}_I| \geq 6F + 1$, and its minimum periodic communication radius satisfies $\rho_I \geq 3F$.

Proof: Consider a sequence of consecutive robots from i^I to $(i + 4F)^I$ in module I. We, again, denote the set of these robots as $\mathcal{V}_F$. We partition $\mathcal{V}_F$ in three parts: $\mathcal{V}_{N1} = \{i^I, \ldots, (i + F)^I\}$, $\mathcal{V}_{N2} = \{(i + 3F + 1)^I, \ldots, (i + 4F)^I\}$, and $\mathcal{V}_S = \{(i + F + 1)^I, \ldots, (i + 3F)^I\}$. Since $\rho_I \geq 3F$, i^I is able to periodically communicate with every robot labeled from $(i + 1)^I$ to $(i + 3F)^I$. Similarly, $(i + 1)^I$ is able to periodically communicate with everyone from i^I to $(i + 3F + 1)^I$ except for itself, and so on. Therefore, every robot from $\mathcal{V}_S$ is able to communicate with everyone but itself in $\mathcal{V}_F$. Furthermore, the accumulated intramodule topology of $\mathcal{V}_{N1}$ is connected, and so is $\mathcal{V}_{N2}$. Since $(i + F)^I$ is periodically connected with $(i + 3F + 1)^I$, the accumulated intramodule topology of $\mathcal{V}_{N1} \cup \mathcal{V}_{N2}$ is also connected. According to Theorem 2.2, the accumulated intramodule topology of $\mathcal{V}_F$ is F-elemental. Take the robot right next to $\mathcal{V}_F$, i.e., robot $(i - 1)^I$ and $(i + 4F + 1)^I$. Each of them has at least $3F \geq 2F + 1$ periodically connected neighbors inside $\mathcal{V}_F$. The joint topology formed by $\mathcal{V}_F$ and these two robots is, therefore, $(2F + 1)$-robust according to Theorem 2.3. By repeatedly adding robots until everyone in I has been added, the new topology is always $(2F + 1)$-robust. Therefore, the circulating module I forms a $(2F + 1)$-core topology.

When a standard module contains less than $4F+1$ robots, it is possible to construct an F-elemental graph with multiple modules. Lemma 4.3 provides an example of constructing a $(2F+1)$-core topology with two modules.

Lemma 4.3: Consider two circulating modules, I and J, with their accumulated intramodule topologies denoted as $\mathcal{G}_I[t]=(\mathcal{V}_I,\mathcal{E}_I[t])$ and $\mathcal{G}_J[t]=(\mathcal{V}_J,\mathcal{E}_J[t])$, respectively, with $|\mathcal{V}_I|,|\mathcal{V}_J|<4F+1$. I and J have a common period T. The integrated topology, denoted as $\mathcal{G}_{IJ}[t]$, is $(2F+1)$-robust if $\mathcal{G}_I^T$ is a complete graph, and there exists $\mathcal{V}_N\subseteq\mathcal{V}_J$, satisfying the following conditions.

1) $|\mathcal{V}_N|+|\mathcal{V}_I|=4F+1$.
2) $\mathcal{G}_N^T$ is a connected graph.
3) $|(\cup_{j\in\mathcal{V}_N}\mathcal{N}_j)\cap\mathcal{V}_I|>|(\cap_{j\in\mathcal{V}_N}\mathcal{N}_j)\cap\mathcal{V}_I|\geq 2F$.

Proof: We first show that there exists a subgraph $\mathcal{G}_F^T\subseteq\mathcal{G}_{IJ}^T$ that is F-elemental if the conditions in Lemma 4.3 are satisfied.

We construct $\mathcal{G}_F^T$ by defining $\mathcal{V}_F=\mathcal{V}_I+\mathcal{V}_N$. Let $\mathcal{S}_F=(\cap_{j\in\mathcal{V}_N}\mathcal{N}_j)\cap\mathcal{V}_I$; it is easy to show that every robot in $\mathcal{S}_F$ is connected with the whole team but itself of $\mathcal{V}_F$. Nodes in $\mathcal{V}_I/\mathcal{S}_F$ are periodically connected since $\mathcal{G}_I$ is complete. Nodes in $\mathcal{V}_N$ are periodically connected according to the second condition in Lemma 4.3. Since $|(\cup_{j\in\mathcal{V}_N}\mathcal{N}_j)\cap\mathcal{V}_I|$ is strictly greater than $|(\cap_{j\in\mathcal{V}_N}\mathcal{N}_j)\cap\mathcal{V}_I|$, there exists at least one node in $\mathcal{V}_I/\mathcal{S}_F$ that is periodically connected with at least one node in $\mathcal{V}_N$. Therefore, nodes in $\mathcal{V}_F/\mathcal{S}_F$ are periodically connected. According to Theorem 2.2, the topology $\mathcal{G}_F^T=(\mathcal{V}_F,\mathcal{E}_F^T)$ is F-elemental.

Since every circulating module forms a connected accumulated intramodule topology, there exists at least one robot j^* in $\mathcal{V}_J/\mathcal{V}_N$ that is periodically connected with at least one robot in $\mathcal{V}_N$. The joint topology formed by $\mathcal{V}_F$ and j^* is, therefore, $(2F+1)$-robust according to Theorem 2.3. By repeatedly adding robots until everyone in J has been added, the new topology is always $(2F+1)$-robust. Therefore, $\mathcal{G}_{IJ}^T$ is $(2F+1)$-robust.

The $(2F+1)$-core topology composed of more than two modules can be verified with similar methods.

B. Expanding a $(2F+1)$-Robust Topology

We discuss how to expand a swarm with a $(2F+1)$-robust topology by adding new modules to the current swarm while preserving its $(2F+1)$-robustness.

Lemma 4.4: Consider a set of connected modules $\mathcal{V}_M=\{1,2\ldots,M\}$. The accumulated integrated topology $\mathcal{G}^T$ is $(2F+1)$-robust. Deploy a new module M^* next to $\mathcal{V}_M$, and define $\mathcal{V}_M^*=\mathcal{V}_M\cup\{M^*\}$. The integrated topology formed by $\mathcal{V}_M^*$, denoted as $(\mathcal{G}^*)^T$, is also $(2F+1)$-robust, if there exists a subset of modules $\mathcal{N}_{M^*}\subseteq\{1,2,\ldots,M\}$, such that $\sum_{I\in\mathcal{N}_{M^*}}D_{M^*}(I)\geq 2F+1$.

Proof: Since $\sum_{I\in\mathcal{N}_{M^*}}D_{M^*}(I)\geq 2F+1$, we have that $|\mathcal{N}_{i^*}\cap\mathcal{V}|\geq 2F+1$ holds for all nodes $i^*\in\mathcal{V}_{M^*}$. $|\mathcal{N}_{i^*}$ is the set of all robots periodically connected with i^*. Therefore adding any node from $\mathcal{V}_{M^*}$ to $\mathcal{G}^T$ will create a new swarm with a $(2F+1)$-robust topology. By adding nodes in $\mathcal{V}_{M^*}$ to the swarm one after another, we will create a new swarm with a $(2F+1)$-robust topology every time until every node in $\mathcal{V}_{M^*}$ has been added. The topology of the swarm finally synthesized is $(\mathcal{G}^*)^T$, and it is $(2F+1)$-robust as well.

Lemma 4.4 provides us with a method to examine the $(2F+1)$-robustness of a constructed robot swarm by analyzing its intermodule topology.

Theorem 4.1: Consider a set of connected circulating modules, $\mathcal{V}_M$, with $|\mathcal{V}_M|>2$. The intermodule topology $\mathcal{G}_M$ is a connected weighted directed graph, with the weights $w(I,J),I,J\in\mathcal{V}_M$ representing interfacing indices $D_I(J)$. The accumulated intramodule topology of each module is $\mathcal{G}_I^T$, with $I=1,\ldots,M$, and the accumulated integrated topology is denoted as $\mathcal{G}^T$. Let $\mathcal{A}_M=(\mathcal{V}_M,\mathcal{E}_M^A)$ be a spanning *directed acyclic graph* of the intermodule topology $\mathcal{G}_M$. The integrated topology of the connected modules is $(2F+1)$-robust, if there exists some $\mathcal{A}_M$, such that:

1) there exists a set of one or multiple modules $\mathcal{V}_C\subseteq\mathcal{V}_M$ that can form a $(2F+1)$-core topology;
2) for every module that is not in $\mathcal{V}_C$, it is satisfied that $\sum_{(I,J)\in\mathcal{E}_M^A}w(I,J)\geq 2F+1$.

Proof: According to Lemma 4.4, adding modules satisfying the conditions stated in the theorem will preserve the $(2F+1)$-robustness of the topology. Since every module but the one that forms the core topology satisfies the conditions, the integrated topology is $(2F+1)$-robust.

According to Theorem 2.1, a robot swarm with a $(2F+1)$-robust topology can achieve resilient consensus through SW-MSR if there are no more than F noncooperative robots in any robot's neighborhood. Therefore, we can evaluate whether a robot swarm can achieve resilient consensus by counting the number of noncooperative robots in any module and its neighboring modules.

Corollary 4.1.1: Consider a set of connected circulating modules, $\mathcal{V}_M$, with $|\mathcal{V}_M|>2$. The intermodule topology $\mathcal{G}_M$ is a connected weighted directed graph, with the weights $w(I,J),I,J\in\mathcal{V}_M$ representing interfacing indices $D_I(J)$. The accumulated intramodule topology of each module is $\mathcal{G}_I^T$, with $I=1,\ldots,M$, and the accumulated integrated topology is denoted as $\mathcal{G}^T$. Let $\mathcal{N}_I=\{J|(I,J)\in\mathcal{G}_M\}$ be the set of neighboring modules of I. Let $\mathcal{A}_M=(\mathcal{V}_M,\mathcal{E}_M^A)$ be a spanning *directed acyclic graph* of the intermodule topology $\mathcal{G}_M$. Let $\mathcal{U}$ be the set of all noncooperative robots, and $\mathcal{U}_I=\mathcal{U}\cap\mathcal{V}_I$ be the set of noncooperative robots in module I. This swarm can achieve resilient consensus by SW-MSR if the conditions in Theorem 4.1 are met, and $|\mathcal{U}_I|+\sum_{J\in\mathcal{N}_I}|\mathcal{U}_J|\leq F$.

Proof: Corollary 4.1.1 holds directly upon Theorems 2.1 and 4.1.

Theorem 4.1 and Corollary 4.1.1 provide us with guidelines in designing routing and swarming strategies for robot teams to achieve resilient consensus in the existence of F noncooperative robots. The goal of our strategies is to synthesize a topology composed of multiple circulating modules, such that its integrated topology is $(2F+1)$-robust. The integrated topology can be regulated through the accumulated intramodule topology of each of the modules, as well as the intermodule topology that specifies how the modules are assembled. The accumulated intramodule topology defines a robots' neighbors within the same circulating module. We can change it by either altering the intramodule

communication radius permanently or temporally or creating temporal links to connect robots with a long arc length in between. The intermodule topology specifies the neighboring modules of a circulating module and the interfacing indices between them. We can alter it by either changing the layout of the modules or increasing/decreasing the interfacing indices.

In real-world applications, we do not always enjoy the flexibility to alter all the factors listed above. In the next two sections, we will introduce two scenarios. The first, as shown in Section V, is to design modular paths for robots deployed in a large and open lattice space, where the robots' speeds are fixed and the intramodule communication radius is fixed as 1. We show how to adjust the accumulated intramodule topology and the universal interfacing index by the geometric design of the circulating path. The latter scenario, as shown in Section VI, is to control the robots deployed on a set of predefined circulating paths. In this case, the intermodule topology is fixed with only the interfacing indices flexible. We show how to realize sufficient intramodule communication radii and interfacing indices by controlling the speeds of the robots.

V. Modular Design

In the previous section, sufficient conditions are provided to ensure resilient consensus on swarms composed of circulating modules. In this section and the next one, we illustrate the strategies to design resilient robot swarms under different sets of constraints. This section focuses on the situation where all robots are set to maintain the same speed of $\bar{v}$ throughout the task execution. In this case, the spacing between robots is fixed, and the intramodule communication radius and the intermodule interfacing indices are controlled by the geometric designs of the circulating routes. We approach this problem by introducing a modular design for all the modules in the whole team such that any swarm constructed by these identical modules will deliver a desired resilient consensus performance. This approach fits the deployment in large and empty environments with adequate flexibility to place the modules.

A. Identical Modules to Guarantee Resilience

We consider the resilient swarm that is composed of multiple modules with their paths described by rotating, translating, shifting phase, or reversing the direction of the same shape $\bar{\boldsymbol{\gamma}} : [0, \bar{L}) \mapsto \mathbb{R}^2$. The robots in the same module are distributed on the route *uniformly*, such that the spacing between any pair of consecutive robots remains constant. We denote it as $\bar{\Delta}$. To let the robots spread out as much as possible, we set $\bar{\Delta}$ slightly smaller than R. Therefore, each of the modules is always connected, but every robot's intramodule communication radius is only $\rho = 1$.

We design the interfacing section of a module in a way such that when a pair of modules are aligned next to each other, their interfacing sections are paralleled. Since all modules are identical, all interfacing sections have the same length of $\bar{l}$. The interfacing index can, therefore, be determined.

In this section, we discuss the actual design of the path's shape in a latticed space [31], [32] to satisfy those conditions. We use a triangular lattice, as introduced in [13]. The lattice is defined as $\mathbb{L} = \{d(x\vec{v}_1 + y\vec{v}_2) | x, y \in \mathbb{Z}\}$, where d is the scale factor of the lattice, which denotes the length of the triangular sides. $\vec{v}_1 = [1, 0]^\top$ and $\vec{v}_2 = [\frac{1}{2}, \frac{\sqrt{3}}{2}]^\top$ are the base vectors. Let $d = \bar{\Delta}$, where $\bar{\Delta}$ is the spacing in a circulating module, which is slightly smaller than the robots' communication range R. When all robots arrive at the top of some lattice points, communication is only available between those on the lattice points next to each other. Connecting lattice nodes with straight lines of a length of d creates a lattice grid.

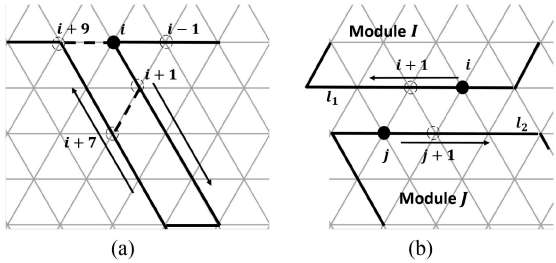


Fig. 4. Examples of periodic connections. (a) shows δi-bridges formed by a *slit* shaped bent on the path. The dash line connection i and $i + 9$ is a 9-bridge, and the one connection $i + 1$ and $i + 7$ is a 6-bridge. (b) shows a pair of interfacing sections with a path length of $4d$. The interfacing indexes will be $D_I(J) = D_J(I) = 9$.

We design the path of a module as a closed curve composed of a limited number of line segments along the grid lines. At time $t = 0$, there is a robot on every lattice node on the path. Let the number of robots in one module to be n, and the robots indexed as $1, 2, \ldots, n$ along the moving direction of robots. The location of a robot i at t is denoted as $s_i[t]$. Let one time step to be the time that a robot takes to move from one lattice node to the next, $s_i[t + 1] = s_{i+1}[t]$, for all $i = 1, 2, \ldots, n - 1$, and $s_n[t + 1] = s_1[t]$.

Robot i is always connected with $i + 1$ and $i - 1$. At some lattice node along the path, robot i is in communication range with some robots $i + \delta i, \delta i \in \mathbb{Z}_{\geq 2}$, and we say that there exists a δi-communication bridge. As shown in Fig. 4, the dashed lines in part (a) are a 6-bridge and a 9-bridge. The connections across δi-bridges are periodic.

When two identical modules are periodically connected through a pair of interfacing sections, the path length of an interfacing section is always an integer multiple of d, as shown in Fig. 4(b). Let the arc length of the interfacing section be $pd, p \in \mathbb{Z}_{\geq 1}$. Assuming that robots from a pair of neighboring modules always travel on the interfacing sections in the reversed direction, as shown in the figure, it will maximize their interfacing index. It is easy to find out that the interfacing index of one module with another is always $2p + 1$.

B. Path Features

Since we are designing identical circulating modules with exactly the same number of robots and the same time-varying intramodule topology, a set of sufficient conditions that guarantee a $(2F + 1)$-robust integrated topology of multiple modules can be written as follows.

Lemma 5.1: Consider two modules, I and J, with their accumulated intramodule topology $\mathcal{G}_I[t] = (\mathcal{V}_I, \mathcal{E}_I[t])$ and $\mathcal{G}_J[t] =$

$(\mathcal{V}_J, \mathcal{E}_J[t])$ identical to each other with $|\mathcal{V}_I| = |\mathcal{V}_J| < 4F + 1$. The period of both modules is T. Module J's interfacing index with module I is $D_J(I)$. The integrated topology, denoted as $\mathcal{G}_{IJ}[t]$, is $(2F + 1)$-robust if $\mathcal{G}_I^T$ is a complete graph, and $D_J(I) + |\mathcal{V}_I| \geq 6F$.

Proof: We show that there exists $\mathcal{V}_N \subseteq \mathcal{V}_J$ that satisfies all conditions listed in Lemma 4.3, and the same conclusion as in Lemma 4.3 holds.

The conditions in Lemma 5.1 require $\mathcal{V}_I \geq 3F$, through $D_J(I) \leq |\mathcal{V}_I|$. Therefore, $|\mathcal{V}_{IJ}| = 2|\mathcal{V}_I| \geq 6F > 4F + 1$.

We construct $\mathcal{V}_N$ by selecting b consecutive robots along the path of J, with $b = 4F + 1 - |\mathcal{V}_I|$. Since every robot is persistently connected with its closest neighbors in the same module, these b robots form a connected graph. We index the b robots as $1_J, 2_J, \ldots, b_J$ and index the robots in module I such that the first robot in I to connect with i_J on the interfacing section will be indexed as 1_I and the second as 2_I, and so on. In other words, robot 1_J will periodically connect with robots $1_I, 2_I, \ldots, D_J(I)_I$. Robot 2_J will temporally connect with $2_I, 3_I, \ldots, (D_J(I) + 1)_I$, and so on.

Similar to the proof of Lemma 4.3, we construct $\mathcal{G}_F^T$ by selecting a subset $\mathcal{S}_J$ of b consecutive robots along the path of J, with $b = 4F + 1 - |\mathcal{V}_I|$. We index the b robots as $1_J, 2_J, \ldots, b_J$. Without loss of generality, we index the first $D_J(I)$ robots from module I that robot 1_J is able to connect with as $1_I, 2_I, \ldots, D_J(I)_I$. Since I and J are identical, robot 2_J will connect to $2_I, 3_I, \ldots, (D_J(I) + 1)_I$, and so on.

Therefore, robots indexed as $b_I, (b + 1)_I, \ldots, D_J(I)_I$ are those who periodically connect with all the b robots in $\mathcal{V}_N$. Since $b = 4F + 1 - |\mathcal{V}_I|$ and $D_J(I) + |\mathcal{V}_I| \geq 6F$, the set $|(\cap_{j \in \mathcal{V}_N} \mathcal{N}_j) \cap \mathcal{V}_I|$ contains at least $2F$ nodes, and $|(\cup_{j \in \mathcal{V}_N} \mathcal{N}_j) \cap \mathcal{V}_I|$ is strictly greater than $|(\cap_{j \in \mathcal{V}_N} \mathcal{N}_j) \cap \mathcal{V}_I|$.

According to Lemma 4.3, the integrated topology $\mathcal{G}_{IJ}[t]$ is $(2F + 1)$-robust.

Since we design all circulating modules that are exactly the same, we require all the interfacing sections to have exactly the same length, yielding the same interfacing index both ways. Theorem 4.1 can, therefore, be rewritten as follows.

Theorem 5.1: Consider a set of connected identical circulating modules, $\mathcal{V}_M$, with $|\mathcal{V}_M| > 2$. The intermodule topology $\mathcal{G}_M$ is a connected weighted *symmetric* directed graph with a universal weight $2p + 1$ for all edges. The identical accumulated intramodule topology of each module is $\mathcal{G}_I^T$, with $I = 1, \ldots, M$, and the accumulated integrated topology is denoted as $\mathcal{G}^T$. $\mathcal{G}^T$ is $(2F + 1)$-robust, if the following conditions hold.

1) $\mathcal{G}_I^T$ is a complete graph.
2) $p \geq F$.
3) $|\mathcal{V}_I| + 2p + 1 \geq 6F$.
4) There exists a connected spanning tree $\mathcal{T}_M = (\mathcal{V}_M, \mathcal{E}_M^T) \subseteq \mathcal{G}_M$.

Proof: With $\mathcal{G}_I^T$ a complete graph and $|\mathcal{V}_I| + 2p + 1 \geq 6F$, we are always able to find one of two modules to form a core topology. With $p \geq F$, we have all the interfacing indices no smaller than $2F + 1$. Therefore, any robot swarm constructed by adding modules one after another after a core topology has been formed will preserve the $(2F + 1)$-robustness.

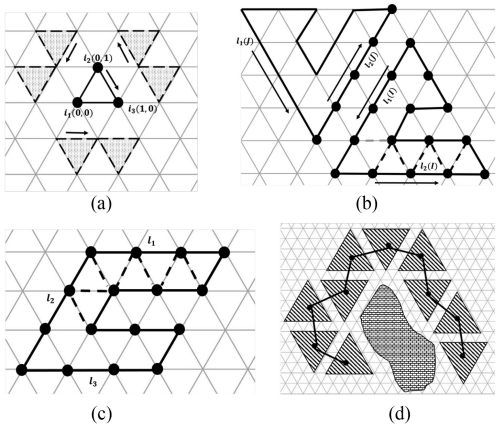


Fig. 5. (a) Unit triangle module I_3. (b) Triangle module I_{15}. (c) U-shape module I_{16}. (d) Formation composed of multiple I_{15} modules is resilient against $F = 4$ noncooperative robots in one's neighborhood.

Theorem 5.1 points out two sets of conditions that we should focus on while designing modular paths that can be assembled while maintaining resilience in consensus performance.

1) Fully Connected Graph: One of the important conditions to construct a $(2F + 1)$-core topology is to ensure that in each module, every robot is periodically connected with all other robots. For the n robots in the same module, there should be δi-bridges in the path for all $\delta i = 2, 3, \ldots, \lceil \frac{n}{2} \rceil$, with $\lceil \frac{n}{2} \rceil = \frac{2n+1}{2}$ if n is an odd number. One method to ensure the δi-bridges is to create a "slit" shape, as shown in Fig. 4. A section of the path with a length of $\lceil \frac{n}{2} \rceil$ and being bent as a slit shape can provide a series of δi-bridges for $\delta i = 2, 3, \ldots, \lceil \frac{n}{2} \rceil$.

2) Interfacing Section: The other important condition to construct a $(2F + 1)$-robust topology is to provide a sufficient interface number while connecting two modules. For example, as shown in Fig. 4(b), a straight line segment of length pd on the path can provide a sufficient interfacing index while serving as an interfacing section if $p \geq F$ and $2p + 1 \geq 6F - |\mathcal{V}_I|$. If we are connecting multiple identical modules, each module should contain at least two nonoverlapping sections that can provide sufficient interfacing indexes.

C. Examples

Following the constraint of path features analyzed, we provide some example designs.

1) Unit Triangles: A module I_3 with three robots on a triangle path with each edge of a unit length d, as shown in Fig. 5(a), is always fully connected. It has three sections, each with a length of d, that can be interfacing sections. Hence, placing another triangle module at one of the six locations (shown in the shaded triangles) can establish an interfacing index of three between them. Therefore, this module with $n_{I_3} = 3$ and $p_{I_3} = 1$ can form large-scale formations that is resilient in consensus performance against $F_{I_3} = 1$ noncooperative robots in one's neighborhood.

2) Triangles With Slits: Fig. 5(b) shows an example module I_{15} with 15 robots. Its communication topology is $\mathcal{G}_{I_{15}}[t]$. The robots form a triangle with two sections of length $4d$ that can be interfacing sections. Black dash lines show δi-bridges of

$\delta i = 2, 3, \ldots, 6$. The dashed line highlighted in blue shows a communication bridge that is both a 7-bridge and an 8-bridge. Therefore, $\mathcal{G}^T_{I_{15}}$ is fully connected. It can be shown that two I_{15} modules can jointly form a $(2F+1)$-core topology with $F_{I_{15}} = 4$, and connecting multiple identical I_{15} modules can form a topology that is resilient against $F_{I_{15}} = 4$ noncooperative robots in any robot's neighborhood. Fig. 5(d) shows an example. Notice that each module can only connect with two other modules.

3) U-Shape: Fig. 5(c) shows a U-shape path with a slit bent on it. This module I_{16} has 16 robots and the path contains three sections, each of the length $3d$, that can be interfacing sections. The communication topology of this module is $\mathcal{G}_{I_{16}}[t]$. Dash lines show δi-bridges for $\delta i = 2, 3, \ldots, 8$. Therefore, $\mathcal{G}^T_{I_{16}}$ is fully connected. It can be shown that a single I_{16} forms a $(2F+1)$-core topology with $F_{I_{16}} = 3$, and connecting multiple I_{16} modules can form a topology that is resilient against three noncooperative robots. Every module can be assembled with three other modules.

VI. Predefined Circulating Routes

In this section, we consider the situations where the circulating routes are predefined and fixed, such as when the robots are patrolling the perimeters of building blocks, cruising along the orbits in the space, or drifting in the gyre-like ocean flows. Robots on the same route are moving in the same direction and in the same order but can control their speeds to adjust the spacing between each other. All robots deployed in the same module form a persistently connected communication network. The paths of different modules are not overlapping with each other but may share some proximity, where robots from different modules entering the same proximity are able to establish cross-module connections. In this article, the parts of their paths sharing proximity are referred to as interfacing sections.

Unlike Section V, here, the circulating paths are given with no guaranteed "bridge" induced by the geometric shape of the paths. We can, still, realize a sufficient level of connectivity within a circulating module by regulating the speeds of the robots and, therefore, the intramodule communication radius. On the other hand, the layout of the paths of all modules is given, so the interfacing sections are predefined as well. It means that the underlying graph of the intermodule topology is a predefined connected graph, leaving only the interfacing indices flexible by altering the spacing between robots while they traversing the interfacing sections.

In the rest of this section, we present our strategy to control the robots' speeds circulating on a set of closed-loop paths with a connected intermodule topology, such that the robots are guaranteed to deliver resilient consensus performance in the existence of no more than F noncooperative robots in any robot's neighborhood.

A. Sufficient Intramodule Communication Radius

As stated in Lemmas 4.1 and 4.2, an important feature that a circulating module containing n_I robots forms a $(2F+1)$-core topology is to have a sufficient intramodule communication radius, i.e.,

$$\rho_I \geq \begin{cases} \lfloor \frac{n_I}{2} \rfloor, & \text{if } n_I \leq 6F \\ 3F, & \text{otherwise} \end{cases}. \tag{3}$$

As introduced in Section III, we design a universal controller for all robots in the same module as $v_i = f_I(s_i), \forall i = 1^I, 2^I, \ldots, n_I^I$. In other words, all robots in the same module are always passing the same location on the path at the same speed. With this universal speed allocation, although the spacing between two robots may vary along the path, the time lag between these two robots passing the same location remains constant. That is to say, for any robots $i, j \in \{1^I, 2^I, \ldots, n_I^I\}$, there exists $\delta t_{i,j}$ such that $s_i(t) = s_j(t + \delta t_{i,j})$ always hold. Such a speed allocation strategy also ensures that the order of robots on the same path remains unchanged. In the part of the path where the robots move faster, the distances between robots are larger, while in the part of the path associated with a slower speed, distances between robots decrease.

In this section, we require all robots deployed on the same module to have exactly the same time lag δt_I between any consecutive pair of robots i and $i+1$. Since there are n_I robots, the universal time lag in module I will be computed by $\delta t_I = \frac{T}{n_I}$, $\forall i \in \{1^I, 2^I, \ldots, n_I^I\}$, where T is the given circulating period. The spacing between a pair of consecutive robots can, therefore, be calculated.

Lemma 6.1: Consider a circulating module with a closed path $\boldsymbol{\gamma}_I : [0, L_I) \mapsto \mathbb{R}^2$, and its universal control strategy $f_I : [0, L_I) \mapsto \mathbb{R}$ for all the n_I robots deployed on $\boldsymbol{\gamma}_I$ is given. The dynamics of any robot i on the path is given by $\dot{s}_i(t) = f_I(s_i(t)), s_i(t) \in [0, L_I)$ with $s_i(t) \equiv s_i(t) \mod L_I$. The robots' common circulating period on their path is T. Assume that the time lag between consecutive robots in module I is given by $\delta t_{i,i+1} = \frac{T}{n_I}$. For any robot i^I in module I locating at $s_i(t_0)$ at time t_0, the location of robot $(i-1)^I$ at time t_0 is

$$s_{i-1}(t_0) = s_i(t_0) + \int_{t=t_0}^{t_0 + \frac{T}{n_I}} f_I(s_i(t))dt.$$

Proof: Since the time lag between consecutive robots i and $i+1$ is universal in module I, robot i will pass the current location of robot $i-1$ after $\frac{T}{n_I}$. Therefore, $s_{i-1}(t_0)$ can be calculated through $s_{i-1}(t_0) = s_i(t_0 + \frac{T}{n_I})$, and the results hold.

It is clear that we want the robots to spread as much as possible on their routes, such that they can provide better coverage of the task space with the same amount of robots. However, the robots also need to slow down somewhere on the path to decrease the distances between them and establish more intramodule communication links. Consider a circulating module with a closed path $\boldsymbol{\gamma}_I : [0, L_I) \mapsto \mathbb{R}^2$; we designate a *slowdown section* on $\boldsymbol{\gamma}_I$ as $\zeta_I = \boldsymbol{\gamma}_I([z_I, z_I + R])$, for some z_I satisfying $z_I, z_I + R \in [0, L_I)$ or $\zeta_I = \boldsymbol{\gamma}_I([z_I, L_I) \cup [0, z_I + R - L_I])$, for some z_I satisfying $z_I, z_I + R - L_I \in [0, L_I)$. The arc length of the designated slowdown section is R; thus, all robots in this section can communicate with each other.

Our idea is to let all robots slow down to a constant speed when they traverse this section ζ_I, such that any robot i is able

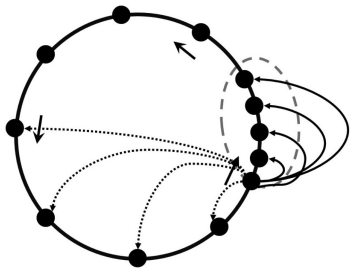


Fig. 6. Robots deployed on a predefined path slow down when traversing a section to build connections with more teammates. The slowdown section ζ_I is circled out by a dashed ellipse. Curves with arrows show the teammates one robot can connect with while entering ζ_I. Dashed curves with arrows show the teammates that will enter ζ_I and, therefore, will connect with this robot before it exists ζ_I. ρ_I here is 4.

to communicate with every other robot in ζ_I between the time that i is on ζ_I, as shown in Fig. 6. If there could be at most $\rho_I + 1$ robots in ζ_I at the same time, any robot in module I is guaranteed to have an intramodule communication radius of ρ_I.

Lemma 6.2: Consider a circulating module with a closed path $\boldsymbol{\gamma}_I : [0, L_I) \mapsto \mathbb{R}^2$ and a designated slowdown section ζ_I of the arc length R. The universal control strategy $f_I : [0, L_I) \mapsto \mathbb{R}$ for the $n_I \geq 4F + 1$ robots deployed on $\boldsymbol{\gamma}_I$ is set. The circulating period for any robot in this module is T. The time for any robot traversing through ζ_I is T_{ζ_I}. Assume that the time lag between any pair of consecutive robots passing the same location on the path is $\delta t_I = \frac{T}{n_I}$. Module I forms a $(2F + 1)$-core topology if

$$\max_{s\in[0,L_I)} f_I(s) \leq \frac{n_I R}{T}$$

$$\text{and } T_{\zeta_I} \geq \begin{cases} T\frac{\lfloor \frac{n_I}{2} \rfloor}{n_I}, & \text{if } n_I \leq 6F \\ T\frac{3F}{n_I}, & \text{otherwise} \end{cases}.$$

Proof: According to Lemma 6.1, the distance between two consecutive robots in module I is $\int_{t=t_0}^{t_0+\frac{T}{n_I}} f_I(s_i(t))dt$. Since $f_I(s) \leq \frac{n_I R}{T}, \forall s \in [0, L_I)$, the distance between two consecutive robots is no greater than $\frac{n_I R}{T} \cdot \delta t_I$, where $\delta t_I = \frac{T}{n_I}$. Therefore, the distance between any pair of consecutive robots in module I is no greater than R. The module remains persistently connected.

Since the time traversing through ζ_I is T_{ζ_I}, there will always be ρ_I or $\rho_I + 1$ robots on ζ_I, with $\frac{\rho_I}{n_I} = \frac{T_{\zeta_I}}{T}$. Since the arc length of ζ_I is R, a robot can connect to its ρ_Ith neighbor before or after it while traversing ζ_I. Therefore, the intramodule communication radius of I is $\rho_I = \lfloor \frac{n_I T_{\zeta_I}}{T} \rfloor$. Equation (3) is satisfied and the module I is $(2F + 1)$-robust, according to Lemmas 4.1 and 4.2.

B. Sufficient Interfacing Indices

As stated in Section III-B, if modules I and J are connected in the intermodule topology with a pair of interfacing indices $D_I(J)$ and $D_J(I)$, then there are a pair of interfacing sections, $\mathbf{l}_1$ on path I and $\mathbf{l}_2$ on path J, such that a robot i in module I will temporally connect with $D_I(J)$ robots during the time that i is traversing $\mathbf{l}_1$, and a robot j in module J will temporally connect with $D_J(I)$ robots from I.

Assuming that robots from modules I and J are traversing the interfacing sections in the reverse directions, then $D_I(J)$ for a robot is the sum of the number of robots that are on $\mathbf{l}_2$ at the time point when a robot in I enters $\mathbf{l}_1$ and the number of robots that are going to enter $\mathbf{l}_2$ before the same robot in I exits $\mathbf{l}_1$. To reach a sufficient interfacing index, say $D_I(J)$, we would like to slow down the robots on either $\mathbf{l}_1$ or $\mathbf{l}_2$.

Lemma 6.3: Consider two circulating modules I and J with closed paths $\boldsymbol{\gamma}_I$ and $\boldsymbol{\gamma}_J$ interfacing each other. Their interfacing sections are $\mathbf{l}_1$ for module I and $\mathbf{l}_2$ for module J. Each of them has a universal control strategy such that they have a common circulating period T. The numbers of robots deployed in module I and J are n_I and n_J, respectively. The time for any robot traversing through $\mathbf{l}_1$ or $\mathbf{l}_2$ is $T_{\mathbf{l}_1}$ or $T_{\mathbf{l}_2}$. The robots circulate the path in such a way that the time lag between any pair of consecutive robots in the same module is always the same. The interfacing indices between these two modules are

$$D_I(J) \geq n_J \frac{T_{\mathbf{l}_2}}{T} + n_J \frac{T_{\mathbf{l}_1}}{T}$$

$$D_J(I) \geq n_I \frac{T_{\mathbf{l}_1}}{T} + n_J \frac{T_{\mathbf{l}_2}}{T}.$$

Proof: To calculate $D_I(J)$, we count $n_J \frac{T_{\mathbf{l}_2}}{T}$, which is the minimum number of robots that are on $\mathbf{l}_2$ at the same time, and $n_J \frac{T_{\mathbf{l}_1}}{T}$, which is the minimum number of robots that will enter $\mathbf{l}_2$ during the time window $T_{\mathbf{l}_1}$ that a robot in module I traverses through $\mathbf{l}_1$. The calculation of $D_J(I)$ is following the same way.

It is easy to see that $D_J(I)$ is guaranteed to be no smaller than $n_I \frac{T_{\mathbf{l}_1}}{T}$, since it takes a nontrivial time for any robot in module J to traverse through $\mathbf{l}_2$. That is to say, we are able to control robots on path I to guarantee a minimum number of $D_J(I)$ without knowing the dynamics of module J. For example, $D_I(J)$ is guaranteed to exceed a required number of $\bar{D}$ if $n_J \frac{T_{\mathbf{l}_2}}{T} \geq \bar{D}$.

C. Assign Robots to a Set of Predefined Paths

As stated in Theorem 4.1, a swarm synthesizes a $(2F + 1)$-robust topology, if its intermodule topology contains at least one connected spanning directed acyclic graph with only one node of a zero indegree, which represents a circulating module that itself, or jointly with its neighboring module, forms a core topology. Every node else has a summed in-flow weight no smaller than $2F + 1$. Based on this theorem, we propose our strategy to assign robots to a given set of closed-loop paths and to circulate the robots on the paths to guarantee the synthesis of a $(2F + 1)$-robust integrated topology.

We first abstract the layout of paths as a symmetric directed graph and consider it as the intermodule topology $\mathcal{G}_M$. In this graph, every node represents a closed-loop path that is ready for a team of robots to circulate on. Every pair of links represent the possible existence of a pair of interfacing sections. Notice that, to retain the $(2F + 1)$-robustness of the integrated topology, only a subset of the edges in $\mathcal{G}_M$ must have their interfacing indices no

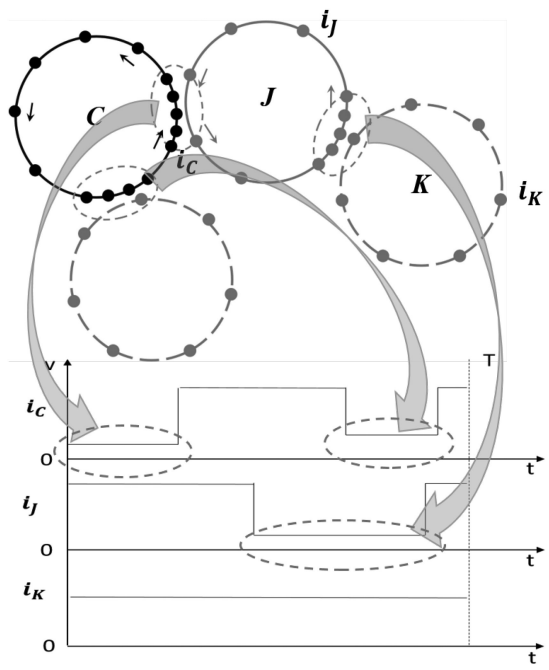


Fig. 7. Three types of circulating modules that form an integrated topology. The black circle is the one that forms a core topology, with an outdegree $\deg(C)^+ = 2$ in the spanning directed acyclic graph of the intermodule topology. The gray circle is a transit module, and the two dashed circles are end modules. Interfacing sections are circled out by dashed ellipses. The bottom part is a sketch showing how the robots' speeds change in a period in three different types of modules. The robots in the core and the transit modules slow down in some regions on the circular path and speed up between these regions. Robots in end modules do not slow down.

smaller than $2F+1$. We then find a spanning directed acyclic graph $\mathcal{A}_M \subseteq \mathcal{G}_M$ with one node C of zero indegree and denote the outdegree of C as $\deg(C)^+ \geq 1$. For the rest of the nodes, we call those of a zero outdegree as *end modules*, and the others with nonzero indegree and outdegree as *transit modules*, as shown in Fig. 7. Let all modules have a common circulating period T, and the weights of the links in $\mathcal{A}_M$ satisfy the conditions stated in Theorem 4.1.

In the rest of this section, we calculate a minimum sufficient number for each module depending on its path length, the common circulating period T, and its role in the spanning directed acyclic graph. A module's role in such a graph can be the core module C, one of the transition modules, or one of the end modules.

We start with constructing a $(2F+1)$-core topology out of module C. That requires module C to contain $n_C \geq 4F+1$ robots always connected. The robots should slow down in a designated section to create a sufficient intramodule communication radius and also slow down in every interfacing sections to create a sufficient interfacing index. Notice that the designated slowdown section can overlap with the interfacing sections. Therefore, we pick the longest interfacing section, and let it serve as the designated slowdown section as well.

Consider the path of C defined by $\boldsymbol{\gamma}_C : [0, L_C) \mapsto \mathbb{R}^2$. The universal control strategy of all of the $n_C \geq 4F+1$ robots in C is $f_C : [0, L_C) \mapsto \mathbb{R}$. All robots have the same communication range R, and the time lag between any two consecutive robots passing the same location is the same $\delta t_C = \frac{T}{n_C}$. The $\deg^+(C)$ interfacing sections, denoted as $\mathbf{l}_1, \mathbf{l}_2, \ldots \mathbf{l}_{\deg^+(C)}$ have lengths as $l_1, l_2, \ldots, l_{\deg^+(C)}$, and the interfacing indices are $D_1(C), D_2(C), \ldots, D_{\deg^+(C)}(C)$, respectively. We denote the starting point of each of the interfacing sections as $s_{l_1}, s_{l_2}, \ldots, s_{l_{deg^+(C)}}$. We require $s_{l_{k+1}} - s_{l_k} > l_k + R, \forall k = 1, 2, \ldots, \deg^+(C) - 1$ and $s_{l_1} + L_C - s_{l_{\deg^+(C)}} > l_{\deg^+(C)} + R$. That is to say, the interfacing sections are not overlapping with each other. There is an arc of a length no shorter than R between any two interfacing sections.

Without loss of generality, let $\mathbf{l}_1$ be the longest interfacing section. We designate a subsegment $\bar{\mathbf{l}}_1 \subseteq \mathbf{l}_1$ with a length no longer than R. If $l_1 > R$, $\bar{\mathbf{l}}_1$ is a subsegment of $\mathbf{l}_1$ with a length of R. If $l_1 \leq R$, we simply let $\bar{\mathbf{l}}_1 = \mathbf{l}_1$. We set the speed of robots traversing this new segment $\bar{\mathbf{l}}_1$ as $\frac{n_C R}{\rho_C T}$, with ρ_C satisfying

$$\rho_C \geq \begin{cases} \lceil \frac{n_C}{2} \rceil, & \text{if } n_C \leq 6F \\ 3F, & \text{otherwise} \end{cases}.$$

Robots on this segment will then reach an intramodule communication radius ρ_C. ρ_C is guaranteed to be greater than $D_1(C)$ because $D_1(C)$ is no greater than $2F+1$, while ρ_C is at least $3F$ or $\lfloor \frac{n_C}{2} \rfloor$ with $n_C \geq 4F+1$. The new $\bar{\mathbf{l}}_1$ is no longer than either $\mathbf{l}_1$ or R, and there are always no less than either $D_1(C)$ or ρ_C robots on this section simultaneously. Therefore, $\bar{\mathbf{l}}_1$ can serve as both an interfacing section and a designated slowdown section ζ_C. Similarly, for the rest interfacing sections on C, the traversing speeds will be set to $\frac{n_C l_2}{D_2(C)T}, \ldots, \frac{n_C l_2}{D_2(C)T}$.

We now compute the minimum robots needed in C. Let $L_o = L_C - R - l_2 - \cdots - l_{\deg^+(C)})$ denote the summed length of the sections out of the slowdown sections. Notice that when robots are moving between interfacing sections, they can speed up to $\frac{n_I R}{T}$ but still maintain a connected topology (according to Lemma 6.2) The summed traversing time out of the slowdown sections is, therefore, no shorter than $\frac{L_o T}{n_C R}$. The summed traversing time through the slowdown sections is no less than $T(\frac{\rho_C}{n_C} + \frac{D_2(C)}{n_C} + \cdots + \frac{D_{\deg^+(C)}(C)}{n_C})$ (according to Lemmas 6.2 and 6.3). Therefore, we have

$$\frac{L_o T}{n_C R} + T\left(\frac{\rho_C}{n_C} + \frac{D_2(C)}{n_C} + \cdots + \frac{D_{\deg^+(C)}(C)}{n_C}\right) \leq T.$$

We denote $n'_C = \lceil \frac{L_o}{R} \rceil + D_2(C) + \cdots + D_{\deg^+(C)}(C)$, and the minimum sufficient number of robots required in C is

$$n_C \geq \begin{cases} \max(4F+1, 2n'_C - 1), & \text{if } n'_C < 3F \\ n'_C + 3F, & \text{otherwise} \end{cases}.$$

The speed controller is set as

$$f_C(s) = \begin{cases} \frac{n_C R}{\rho_C T}, & \text{if } s \in \bar{\mathbf{l}}_1 \\ \frac{n_C l_2}{D_2(C)T}, & \text{if } s \in \mathbf{l}_2 \\ \vdots & \\ \frac{n_C l_{\deg^+(C)}}{D_{\deg^+(C)}(C)T}, & \text{if } s \in \mathbf{l}_{\deg^+(C)} \\ \frac{n_C L_o}{T(n_C - \rho_C - D_2(C) \ldots - D_{\deg^+(C)}(C))}, & \text{otherwise} \end{cases}.$$

For the transition modules, we only need to slow down the robots in the interfacing sections that are represented by outflow links to guarantee a minimum number of interfacing index with some other modules. Consider the path of a transition module J defined by $\boldsymbol{\gamma}_J : [0, L_J) \mapsto \mathbb{R}^2$. The universal control strategy of all of the n_J robots in J is $f_J : [0, L_J) \mapsto \mathbb{R}$. All robots have the same communication range R, and the time lag between any two consecutive robots passing the same location is the same $\delta t_J = \frac{T}{n_J}$. The $\deg^+(J)$ interfacing sections, denoted as $\mathbf{l}_1, \mathbf{l}_2, \ldots \mathbf{l}_{\deg^+(J)}$, and the interfacing indices are $D_1(J), D_2(J), \ldots, D_{\deg^+(J)}(J)$, respectively. Following a similar discussion as we did for module C, we have $n_J \geq \lceil \frac{L_o}{R} \rceil + D_1(J) + D_2(J) + \cdots + D_{\deg^+(J)}(J)$, and $f_J(s)$ allocated similarly as $f_C(s)$.

Notice that the robots are not changing speed on the interfacing sections that are represented by inflow links, because the speed allocation calculated on the tail end of these links guarantees that the interfacing indices can be reached as long as the robots in module J traverse through the interfacing section with a nontrivial time.

For the end modules, every robot will receive at least $2F + 1$ periodic connections by traversing through the interfacing sections with a nontrivial time. There is no need of slowing down the robots. The only requirement is to keep the module persistently connected. Consider the path of an end module K defined by $\boldsymbol{\gamma}_K : [0, L_K) \mapsto \mathbb{R}^2$. The universal control strategy of all of the n_K robots in K is $f_K : [0, L_K) \mapsto \mathbb{R}$. All robots have the same communication range R, and the time lag between any two consecutive robots passing the same location is the same $\delta t_K = \frac{T}{n_K}$. Therefore, we have $n_K \geq \lceil \frac{L_K}{R} \rceil$, and the speed of all robots around the module is $\frac{L_K}{T}$. The speeds of robots in all three types of modules are illustrated as in Fig. 7.

With the method proposed, a solution to synthesize a robot swarm that can achieve resilient consensus is only guaranteed when there are a sufficient number of robots assigned to every module.

VII. Simulation

In this section, we present two sets of simulations. The first set[1] includes four different robot swarms composed of subgroups of robots deployed on modular closed-loop paths in a lattice space. The designs were introduced in Section V-C. The black dots show cooperative robots that are achieving resilient consensus in the presence of multiple noncooperative robots. The noncooperative robots are shown in red circles. The initials of each team were randomly chosen, including the noncooperative robots. The cooperative robots follow the SW-MSR algorithm, and the noncooperative robots generate oscillatory and random values. The signal from the noncooperative robots tries to avoid consensus from the cooperative robots. Every design has a topology that is periodic and is proved to be $(2F + 1)$ robust in Section V.

[1][Online]. Available: https://youtu.be/dj2afGyhBB4

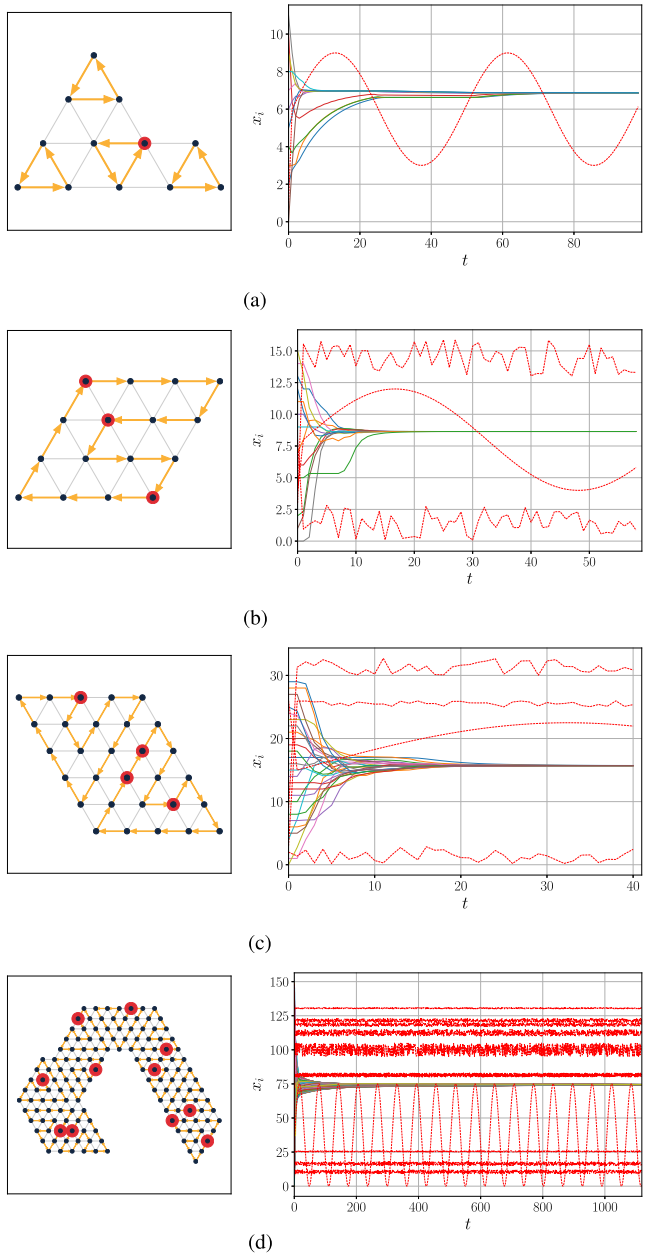


Fig. 8. Simulations for four different modular dynamical networks. On the left side of each panel, we present the network topology. The dark disks represent the robots and the arrows illustrate their direction of motion. On the right side, the evolution of the scalar variable in time. The solid lines and the dashed lines represent the scalar values of the cooperative and noncooperative robots, respectively. (a) Triangular module with $F = 1$ (four modules with three robots each). (b) U-shape module with $F = 3$ (one module with 16 robots each). (c) Large triangular module with $F = 4$ (two modules with 15 robots each). (d) Large network with $F = 4$ (ten modules with 15 robots each).

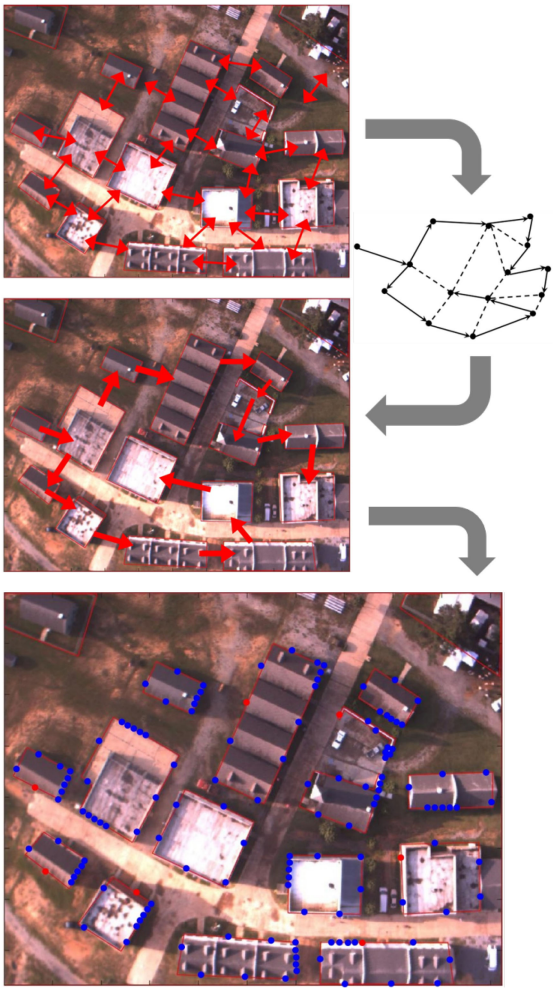


Fig. 9. Top left figure shows the layout of the closed-loop paths with double arrows showing the existence of interfacing sections. The top right sketch is the abstract intermodule topology. The spanning directed acyclic graph is shown with solid arrows. The middle left figure is the projection of the spanning directed acyclic graph to the task space, where every red arrow in this figure indicates a slowdown section. The bottom figure shows the robots on the paths as blue and red dots. The distances between robots decrease in the sections marked by red arrows in the middle left figure. The red dots represent the noncooperative robots broadcasting false information, while the blue dots are the cooperative ones.

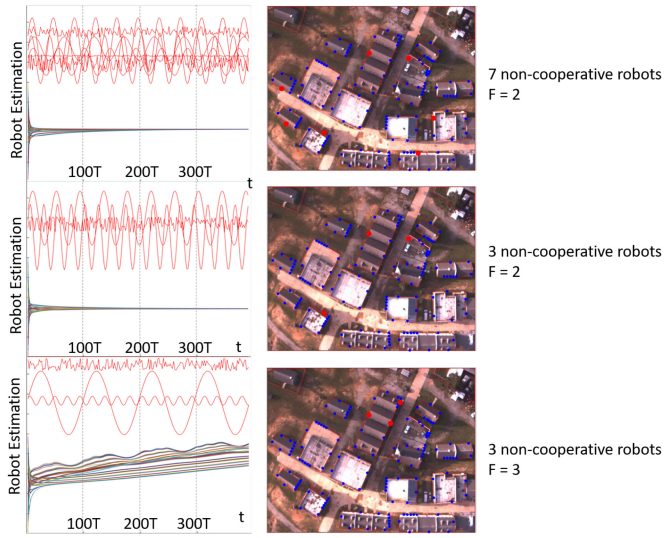


Fig. 10. Consensus performance of the robot swarm shown in Fig. 9 with different sets of noncooperative robots. The first line shows the consensus performance of seven noncooperative robots with a distribution that fits $F = 2$. The second line shows the performance of three noncooperative robots with $F = 2$. The last line shows the performance of three noncooperative robots with $F = 3$. The red dots indicates where the noncooperative robots are. The red curves are the corrupt or manipulative information sent by the noncooperative robots.

The results of the simulations are shown in Fig. 8. In every scenario, the cooperative robots achieved a resilient consensus. The swarms shown in Fig. 8(c) and (d) are composed of the same modular design of the circulating paths. Notice that robots in Fig. 8(c) achieve consensus much faster than robots in Fig. 8(d). The factors that affect the time to achieve consensus varies. In general, a shorter common period, a smaller number of robots in each subgroup, and a smaller diameter of the intermodule topology will result in faster convergence. This result is expected since the network connectivity is reduced by adding new modules to the network. However, in comparison to methods in the literature where high connectivity is required to satisfy the robustness conditions, we can achieve resilient consensus and still maintaining low connectivity. In Fig. 8(d), we show that the modular configuration limits the neighborhood of each robot, allowing up to $F = 4$ noncooperative robots in every robot's neighborhood.

The second set of simulations shows the performance of robots in a neighborhood with 15 building blocks to be patrolled. A subgroup of robots will be deployed on a closed-loop path and circulate on it, maintaining a persistently connected intramodule topology. The pair of interfacing sections, where robots from different circulating modules may connect with each other, is shown in red double arrows. We identify a connected spanning directed acyclic graph (shown with red single arrows) of the intermodule topology and let the robots slow down in the interfacing sections related to the tails of the single arrows. The speed allocation and the number of robots assigned to each path were calculated following the methods in Section VI-C.

The integrated topology these robots form is $(2F + 1)$ robust, with $F = 2$. Notice that it requires no more than two noncooperative robots in any robot's circulating module *and* neighboring modules in the intermodule topology. The blue dots in the left bottom figure in Fig. 9 show the robots on the path. The red dots are the noncooperative ones. There are 141 robots deployed in total.

Fig. 10 shows the consensus performance of the robots swarm constructed, as shown in Fig. 9, with the different distributions of noncooperative robots. In each simulation, the initial estimation carried out by each robot was randomly chosen. The red curves show the manipulative information broadcasted by the noncooperative robots to their neighbors. Cooperative robots update their estimations based on the information acquired from their neighbors within a period of time. The first and second simulations show that the estimations of the cooperative robots converge despite the existence of different numbers of the

noncooperative robots. Notice that in both cases, there are more than two noncooperative robots in the whole team. However, the noncooperative robots were evenly distributed so that $F = 2$ still holds. On the contrary, the last simulation shows a distribution with only three noncooperative robots, but all three occur in the same group. In this case, $F = 3$ for some of the robots, and therefore, the whole team did not converge.

VIII. Conclusion

This work proposed an approach to construct resilient dynamic communication networks leveraging the mobility of robots deployed in large environments. The robots formed multiple subgroups, with each group periodically circulating on a closed-loop path. The paths are not overlapping with each other but may share some proximity allowing robots from different groups to establish temporal cross-group communication links. We showed that the integrated topology can guarantee the team to reach resilient consensus over time if all robots share a common period in their motion, and certain sufficient conditions on both the individual groups and their interconnections are met.

We provided designs in two different scenarios. In the first scenario, all robots are assumed to move at the same constant speed in a lattice space. We designed a modular shape for all closed-loop paths that robots are circulating on. This formation can expand in the task space by connecting multiple circulating modules and preserve resilience in the consensus performance at the same time. In the other scenario where the closed-loop paths are given and the underlying graph of the intermodule topology is predefined, we developed a controller for each module to slow down the robots in certain sections interfacing other modules. The robots can, therefore, establish a sufficient number of intramodule and cross-module connections. We supported our theoretical results for both scenarios with simulation examples.

Our proposed solution of constructing integrated robot network topology that is r-robust is easy to scale up by integrating more circulating modules. The designs we have provided offer flexibility to alter different sets of factors (path shapes, circulating speeds, etc.) of the robot network according to the demands of the tasks. However, the design was done offline, and each design requires a full knowledge of the task space. If the task space changes (i.e., circulating paths are added, removed, reshaped, etc.), the design must be updated accordingly. Future study could be carried out in developing online methods to synthesize resilient robot swarms that can adapt to dynamic environments.

References

[1] S. L. Smith, M. Schwager, and D. Rus, "Persistent monitoring of changing environments using a robot with limited range sensing," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 5448–5455.

[2] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo, "Swarm robotics: A review from the swarm engineering perspective," *Swarm Intell.*, vol. 7, no. 1, pp. 1–41, 2013.

[3] M. Duarte *et al.*, "Application of swarm robotics systems to marine environmental monitoring," in *Proc. IEEE OCEANS*, Shanghai, China, 2016, pp. 1–8.

[4] S.-J. Chung, A. A. Paranjape, P. Dames, S. Shen, and V. Kumar, "A survey on aerial swarm robotics," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 837–855, Aug. 2018.

[5] W. Ren and R. W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Trans. Autom. Control*, vol. 50, no. 5, pp. 655–661, May 2005.

[6] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.

[7] W. Ren, R. W. Beard, and E. M. Atkins, "Information consensus in multivehicle cooperative control," *IEEE Control Syst. Mag.*, vol. 27, no. 2, pp. 71–82, Apr. 2007.

[8] W. Ren, R. W. Beard, and E. M. Atkins, "A survey of consensus problems in multi-agent coordination," in *Proc. Amer. Control Conf.*, 2005, pp. 1859–1864.

[9] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," Departmental Papers, Dept. Elect. Syst. Eng., Univ. Pennsylvania, Philadelphia, PA, USA, 2003, p. 29.

[10] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1520–1533, Sep. 2004.

[11] L. Moreau, "Stability of multiagent systems with time-dependent communication links," *IEEE Trans. Autom. Control*, vol. 50, no. 2, pp. 169–182, Feb. 2005.

[12] D. Saldaña, A. Prorok, S. Sundaram, M. F. Campos, and V. Kumar, "Resilient consensus for time-varying networks of dynamic agents," in *Proc. Amer. Control Conf.*, 2017, pp. 252–258.

[13] D. Saldaña, L. Guerrero-Bonilla, and V. Kumar, "Resilient backbones in hexagonal robot formations," in *Distributed Autonomous Robotic Systems*. Cham, Switzerland: Springer, 2019, pp. 427–440.

[14] G. A. Hollinger and S. Singh, "Multirobot coordination with periodic connectivity: Theory and experiments," *IEEE Trans. Robot.*, vol. 28, no. 4, pp. 967–973, Aug. 2012.

[15] Y. Kantaros and M. M. Zavlanos, "Distributed intermittent communication control of mobile robot networks under time-critical dynamic tasks," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2018, pp. 1–9.

[16] M. Guo and M. M. Zavlanos, "Multirobot data gathering under buffer constraints and intermittent communication," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 1082–1097, Aug. 2018.

[17] R. Aragues, D. V. Dimarogonas, P. Guallar, and C. Sagues, "Intermittent connectivity maintenance with heterogeneous robots," *IEEE Trans. Robot.*, vol. 37, no. 1, pp. 225–245, Feb. 2021.

[18] X. Yu and M. A. Hsieh, "Synthesis of a time-varying communication network by robot teams with information propagation guarantees," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 1413–1420, Apr. 2020.

[19] H. Zhang and S. Sundaram, "Robustness of information diffusion algorithms to locally bounded adversaries," in *Proc. Amer. Control Conf.*, 2012, pp. 5855–5861.

[20] H. J. LeBlanc, H. Zhang, X. Koutsoukos, and S. Sundaram, "Resilient asymptotic consensus in robust networks," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 4, pp. 766–781, Apr. 2013.

[21] H. Zhang, E. Fata, and S. Sundaram, "A notion of robustness in complex networks," *IEEE Control Netw. Syst.*, vol. 2, no. 3, pp. 310–320, Sep. 2015.

[22] K. Saulnier, D. Saldana, A. Prorok, G. J. Pappas, and V. Kumar, "Resilient flocking for mobile robot teams," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 1039–1046, Apr. 2017.

[23] H. J. LeBlanc and X. D. Koutsoukos, "Algorithms for determining network robustness," in *Proc. 2nd ACM Int. Conf. High Confidence Netw. Syst.*, 2013, pp. 57–64.

[24] E. M. Shahrivar, M. Pirani, and S. Sundaram, "Robustness and algebraic connectivity of random interdependent networks," *IFAC-PapersOnLine*, vol. 48, no. 22, pp. 252–257, 2015.

[25] L. Guerrero-Bonilla, D. Saldana, and V. Kumar, "Design guarantees for resilient robot formations on lattices," *IEEE Robot. Autom. Lett.*, vol. 4, no. 1, pp. 89–96, Jan. 2019.

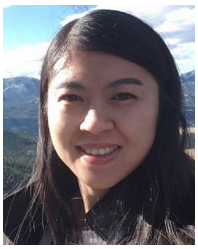
[26] W. Abbas, A. Laszka, and X. Koutsoukos, "Improving network connectivity and robustness using trusted nodes with application to resilient consensus," *IEEE Control Netw. Syst.*, vol. 5, no. 4, pp. 2036–2048, Dec. 2018.

[27] A. Mitra, W. Abbas, and S. Sundaram, "On the impact of trusted nodes in resilient distributed state estimation of LTI systems," in *Proc. IEEE Conf. Decis. Control*, 2018, pp. 4547–4552.

[28] F. Ghawash and W. Abbas, "Leveraging diversity for achieving resilient consensus in sparse networks," *IFAC-PapersOnLine*, vol. 52, no. 20, pp. 339–344, 2019.

[29] X. Yu, D. Shishika, D. Saldana, and M. A. Hsieh, "Modular robot formation and routing for resilient consensus," in *Proc. Amer. Control Conf.*, 2020, pp. 2464–2471.

[30] L. Guerrero-Bonilla, A. Prorok, and V. Kumar, "Formations for resilient robot teams," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 841–848, Apr. 2017.
[31] K. Fujibayashi, S. Murata, K. Sugawara, and M. Yamamura, "Self-organizing formation algorithm for active elements," in *Proc. 21st IEEE Symp. Rel. Distrib. Syst.*, 2002, pp. 416–421.
[32] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Trans. Autom. Control*, vol. 51, no. 3, pp. 401–420, Mar. 2006.



Xi Yu (Member, IEEE) received the B.S. and Dipl.-Ing. degrees in mechanical engineering from the Karlsruhe Institute of Technology, Karlsruhe, Germany, in 2010 and 2011, respectively, and the Ph.D. degree in mechanical engineering from Boston University, Boston, MA, USA, in 2018.

She is currently an Assistant Professor with Department of Mechanical and Aerospace Engineering, West Virginia University, Morgantown, WV, USA. From 2018 to 2020, she was a Postdoctoral Researcher with the General Robotics, Automation, Sensing and Perception Laboratory, University of Pennsylvania, Philadelphia, PA, USA. Her research interests include control and coordination of multirobot systems, especially for long-term and large-scale monitoring tasks.

Dr. Yu is a member of the IEEE Control Systems Society.

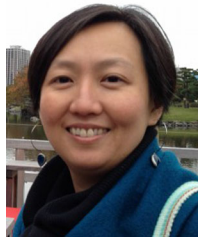
David Saldaña (Member, IEEE) received the B.Sc. and M.Sc. degrees in informatics engineering from the National University of Colombia, Bogotá, Colombia, in 2010 and 2012, respectively, and the Ph.D. degree in computer science, artificial intelligence and robotics from the Federal University of Minas Gerais, Belo Horizonte, Brazil, in 2017.

He is currently an Assistant Professor in Computer Science and Engineering with Lehigh University, Bethlehem, PA, USA. From 2015 to 2019, he was a Visiting Research Scholar and a Postdoctoral Researcher with the General Robotics, Automation, Sensing and Perception Laboratory, University of Pennsylvania, Philadelphia, PA. His current projects include design and control of modular aerial robots, and enhancing resiliency in large-scale robot networks. His main research interests include modular aerial robots, multirobot systems, and robot swarms.

Daigo Shishika (Member, IEEE) received the B.S. degree in aerospace engineering from the University of Tokyo, Tokyo, Japan, in 2012, and the Ph.D. degree in aerospace engineering from the University of Maryland, College Park, MD, USA, in 2017.

He is currently an Assistant Professor with the Department of Mechanical Engineering, George Mason University, Fairfax, VA, USA. From 2017 to 2020, he was a Postdoctoral Researcher with the General Robotics, Automation, Sensing and Perception Laboratory, University of Pennsylvania, Philadelphia, PA, USA. His research interests include cooperative control of multirobot systems in adversarial scenarios.

Dr. Shishika is a member of the IEEE Control Systems Society.



M. Ani Hsieh (Member, IEEE) received the B.S. degree in engineering and the B.A. degree in economics from Swarthmore College, Swarthmore, PA, USA, in 1999, and the Ph.D. degree in mechanical engineering from the University of Pennsylvania, Philadelphia, PA, in 2007.

She is currently a Research Associate Professor with the Department of Mechanical Engineering and Applied Mechanics, University of Pennsylvania. From 2007 to 2008, she was a Visiting Assistant Professor with the Department of Engineering, Swarthmore College. From 2008 to 2017, she was an Associate Professor with the Department of Mechanical Engineering and Mechanics, Drexel University, Philadelphia. Her research interests include robotics, specifically many robot systems and marine robotics, geophysical fluid dynamics, and dynamical systems.

Dr. Hsieh is a Recipient of a 2012 Office of Naval Research Young Investigator Award and a 2013 National Science Foundation CAREER Award.