Fault-Tolerant Rendezvous of Multirobot Systems

Hyongju Park, *Member, IEEE*, and Seth A. Hutchinson, *Fellow, IEEE*

***Abstract*—In this paper, we propose a distributed control policy to achieve rendezvous by a set of robots even when some robots in the system do not follow the prescribed policy. These nonconforming robots correspond to faults in the multirobot system, and our control policy is thus a fault-tolerant policy. Each robot has a limited sensing range and is able to directly estimate the state of only those robots within that sensing range, which induces a network topology for the multirobot system. We assume that it is not possible for the fault-free robots to identify the faulty robots, and thus our approach is robust even to undetected faults in the system. The main contribution of this paper is a fault-tolerant distributed control algorithm that is guaranteed to converge to consensus under certain reasonable connectivity conditions. We first present a general algorithm that exploits the notion of a Tverberg partition of a point set in $\mathbb{R}^d$, and give a proof of convergence. We then provide three instantiations of this algorithm, based on three different sensing models. For each case, we analyze performance via extensive simulations. The effectiveness and performance of our algorithms on real platforms are demonstrated through experiments on a multirobot testbed.**

***Index Terms*—Decentralized control, distributed control, distributed rendezvous, fault-tolerant consensus, Tverberg partition.**

I. Introduction

In THIS paper, we consider a fault-tolerant version of the multirobot rendezvous problem. In almost all approaches that have been reported for the rendezvous problem, it is implicitly assumed that all of the individual robots will faithfully and accurately execute an agreed upon, decentralized control policy. Here, we consider the case in which some of the robots, termed *faulty* robots, fail to follow the policy. For multirobot systems, this could result from physical failure due to component malfunction (e.g., sensor error), or to depletion of energy (e.g., dead batteries). More generally, faults might even include adversarial scenarios, such as the presence of malicious robots which try to maximally degrade the performance of coordination tasks [1], [2].

We present what we believe to be the first fault tolerant, distributed consensus algorithm for the case of robots operating in a multidimensional workspace. The contributions of this paper are twofold. First, we present a distributed algorithm for convergence of fault-free robots in the presence of malicious robots. The algorithm relies on the concept of a Tverbeg partition of a pointset in $\mathbb{R}^d$. The key property of a Tverberg partition is that there is a nonempty intersection of the convex hulls of the elements in partition. This property allows the construction of fault-tolerant control policies that do not rely on coordinate averaging schemes. Second, we give a rigorous analysis of the algorithm's performance, combining elements of the methods given in the decentralized control community [3], [4], in which faulty robots are not considered, and from the fault-tolerant computing community [5], [6], in which network topology is independent of system state. Our analysis also provides a bound on the number of iterations required to achieve approximate convergence within a specified error bound. We demonstrate our algorithm for three different sensing models, evaluating performance via extensive simulation.

Fault tolerance has been a concern in digital computing since its earliest days [7], and our fault-tolerant rendezvous problem has a close relationship to the classical *Byzantine Generals* problem in distributed computing [8]. The two problems have several key differences, e.g., for the Byzantine Generals problem the communication links are fixed, whereas in the multirobot scenario the communication links depend on the distances between pairs of robots; however, the overall goal for the Byzantine Generals problem (achieving agreement among the fault-free processors, in spite of the presence of faulty processors in the network) is analogous to the goal of achieving rendezvous. The approximate Byzantine consensus problem is an extension of the original Byzantine generals problem for which the goal is to allow the fault-free processors to agree on a value asymptotically [9]–[11]. Recently, there have been studies of approximate Byzantine consensus problems where the consensus value is a d-dimensional vector in Euclidean space. This problem is termed as Byzantine *vector* consensus [12]. Because of the similarities between the Byzantine Generals problem and the rendezvous problem, we are able to exploit a number of results from the area of fault-tolerant distributed computing, particularly the results found in [5] and [6].

Independent from those studies on fault-tolerant consensus in digital computing communities, there is also a body of research on fault-tolerant consensus [13]–[25] and fault-tolerant rendezvous or gathering [26]–[33] in the control and robotics

Manuscript received October 10, 2016; accepted January 11, 2017. Date of publication March 1, 2017; date of current version June 5, 2017. This paper was recommended for publication by Associate Editor C. Secchi and Editor C. Torras upon evaluation of the reviewers' comments. This paper was presented in part at the IEEE International Conference Robotics and Automation, Stockholm, Sweden, May 2016 [40] and in part at the IEEE/RSJ International Conference on Intelligent Robots and Systems, Hamburg, Germany, September–October, 2015 [41].

H. Park is with the Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: hjcpark@umich.edu).

S. A. Hutchinson is with the Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA (e-mail: seth@illinois.edu).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TRO.2017.2658604

communities, respectively. We will review here a few of those studies that are closely related to the problem of interest in this paper.

In the control community, a number of studies have focused on the consensus problem for the special case of a scalar consensus variable. A median-based consensus algorithm is described in [15] and [34], in which each node uses only its own value and the median of its neighbors' values for the state update. In a similar manner, Leblanc *et al.* [10], [35] consider a protocol in which each node removes neighbors' values that are extreme with respect to its own value. Their study is an extension of *resilient consensus protocol* previously investigated in [36]. There are also several studies that use control theoretic approaches to provide provably correct distributed consensus algorithms for cases in which malicious or unreliable links are present, e.g., [17]–[20], [22], and [23]. Again, in all of these approaches, the consensus variables are scalars, taking their values on the real line.

A number of approaches have focused on robustness to changes in network topology, caused, for example, by adversarial nodes that can cause loss of edges or even nodes from the connectivity graph, e.g., [10], [17], and [37]. In [37], Zhang *et al.* prove robustness of a special class of random graphs to such adversarial behaviors, and also show that the problem of determining robustness is coNP-complete. In [16], Khanafer *et al.* proposed a zero-sum game between groups of nodes and an adversary, in which the group of nodes executes robust distributed averaging whereas the adversary strategically disconnects a set of links to prevent the nodes from converging. They formulate two versions of the problem, which are competition between two players whose action is reversed, i.e., min–max and max–min problems. The maximum principle was used to obtain optimal strategies for both problems and also to provide sufficient conditions for existence of saddle points. We note that these approaches consider only a limited class of possible faults—those that result in changes to network connectivity. In contrast, our approach is able to deal a more general class of faults in which nodes may exhibit malicious behavior while remaining anonymous, and without altering network topology.

Finally, it is worth mentioning the work of Zhu and Martinez [13], [14] who consider the special case of adversarial nodes launching replay attacks.[1] They have proposed a novel distributed resilient algorithm for multivehicle systems based upon a receding-horizon control method that converges to a desired formation regardless of any replay attacks by adversarial nodes. Their model requires each node to have a memory. Again, our approach is able to handle this type of fault, as well as much more general classes of faults, and our method does not require memory.

There have been a number of attempts to solve the fault-tolerant gathering/rendezvous problem in the robotics community [26]–[33]. However, all of these rely on the assumption that each fault-free robot can see all other robots in the workspace, i.e., each fault-free robot has unlimited visibility, which implies a fully connected communication graph. Under this condition, Agmon and Peleg [26] presented a correct algorithm to gather all functioning robots when one of the robots crashes permanently. Defago *et al.* [27] extended Agmon and Peleg's previous work and showed feasibility of probabilistic gathering under various assumptions related to synchrony and crash/Byzantine faults. Bouzid *et al.* [28] proposed an algorithm to gather all fault-free robots in the presence of multiple crash faults. In their algorithm, Weber points,[2] which have the key property of remaining unchanged under straight line movements of any of the points toward or away from it, were used. They also proposed a Byzantine tolerant gathering algorithm [29]–[31] for multirobots moving in a line. Their approach is a combination of the algorithms of Leblanc [10] and Zhang [15], in that, in their algorithms, each robot uses a trimming method to remove up to f largest and f smallest values from their neighbors, and takes the median of the values that are left.

In the remainder of this paper, we present our approach to solving the fault-tolerant distributed consensus problem. We begin by introducing necessary notation and background material in Section II. Then, in Section III, we present the concept of a Tverberg partition, which is the key concept underlying our approach. In Section IV, we present our fault-tolerant algorithm, which we refer to as ADRC. The main theoretical result of this paper, a theorem that guarantees convergence under mild connectivity assumptions, is presented in Section V. Section VI presents three different instantiations of the ADRC algorithm based on different sensing models and their performance is analyzed via number of simulation results in Section VII. Experimental results are included in Section VIII to validate the effectiveness of our algorithms on real multirobot platforms. In addition to the conclusions in Section IX, Appendix A gives brief details on ergodic theory and Appendix B provides several proofs for auxiliary lemmas that are used in the main proof of Section V.

II. Preliminaries

A. Notation and Terminology

Throughout this paper, we use the symbol $\mathbb{Z}_{\geq 0}$ for the set of nonnegative integers, $\mathbb{N}$ for the set of natural numbers, and $\mathbb{R}$ for the set of real numbers. We work throughout in Euclidean space $\mathbb{R}^d$, using lower case letters for scalars, points, and vectors, and upper case letters for matrices.

We consider a group of n autonomous mobile robots, each with a bounded state space $\mathcal{X} \subseteq \mathbb{R}^d$. Each robot has an index $i \in \mathcal{I} = \{1, \ldots, n\}$, where $\mathcal{I}$ is called the *index set*. The state of the multirobot system is represented by an $n \times d$ matrix $\mathbf{x} = [x_1, \ldots, x_n]^\top$, where $x_i \in \mathcal{X}$ is a $d \times 1$ column vector representing the location of the ith robot. We use the notation $x = \{x_1, \ldots, x_n\}$ to denote the *set* of robot positions and $u = \{u_1, \ldots, u_n\}$ to denote the set of inputs.

The interconnection topology of the multirobot system is represented by a *directed graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ in which $\mathcal{V} = \mathcal{I}$ is

[1]Adversarial nodes consecutively repeating the control commands for a period of time.

[2]The geometric median of set of points in a Euclidean space is the point minimizing the sum of distances to the points [38].

the set of vertices, each of which corresponds to the identifier of a robot, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of directed edges. Given the set of edges, the index set of in-neighbors for robot $i \in \mathcal{I}$ is defined by $\mathcal{N}_i = \{j \in \mathcal{I} \mid (j, i) \in \mathcal{E}\}$. We use the notation $\mathcal{A}(\mathcal{G})$ or $\mathcal{A}$ for the $n \times n$ adjacency matrix of $\mathcal{G}$. A node $i \in \mathcal{V}$ is *globally reachable* if it can be reached from any other node $j \in \mathcal{V} \setminus \{i\}$ by traversing a directed path.

A matrix is *positive*, if every element in the matrix is positive. Given a square matrix $\mathbf{A}$, we denote by $[\mathbf{A}]_{ij}$ the (i, j)th element of $\mathbf{A}$, and by $[\mathbf{A}]_k$ the kth column of $\mathbf{A}$. We denote by $\mathbf{I}_n$ the $n \times n$ identity matrix, and by $\mathbf{1}_{n\times 1}$ the $n \times 1$ column vector whose elements are all 1s. For arbitrary $n \times n$ square matrices $\mathbf{B}, \mathbf{C}$, we write $\mathbf{B} \le \mathbf{C}$ if $[\mathbf{B}]_{ij} \le [\mathbf{C}]_{ij}$ for all pairs $i, j \in \{1, \ldots, n\}$.

For a given set $\mathcal{C} \subseteq \mathbb{R}^d$, we denote by conv$(\mathcal{C})$ the *convex hull* of the set $\mathcal{C}$, by ri$(\mathcal{C})$ the *relative interior* of $\mathcal{C}$, and by $|\mathcal{C}|$ the *cardinality* of $\mathcal{C}$. For a convex polytope $\mathcal{P} \subseteq \mathbb{R}^d$, we denote by ver$(\mathcal{P})$ the set of *vertices* of $\mathcal{P}$. A set of at least $d + 1$ points in $\mathbb{R}^d$ is said to be in *general position* if no hyperplane of dimension $d - 1$ or less contains more than d points.

B. Faulty Multirobot Systems (F-MRS)

An F-MRS consists of n robots, of which some are faulty. We denote by $\mathcal{F} \subseteq \mathcal{I}$ the index set of faulty robots, and by n_f the number of faulty robots, $n_f = |\mathcal{F}|$.

Because we will frequently refer to the set of fault-free robots, it is convenient to define the following notation. We denote by $\overline{\mathcal{I}}$ the index set of the fault-free robots and by $\overline{x}$ the set of their positions. The number of *fault-free* robots is denoted $\overline{n} = n - n_f$. Without loss of generality, in the sequel, we will assume that the robots are indexed such that the fault-free robots have indices $\overline{\mathcal{I}} = \{1, \ldots, \overline{n}\}$ and the faulty robots have indices $\mathcal{F} = \{\overline{n} + 1, \ldots, n\}$.

The state of the fault-free robots is represented by an $\overline{n} \times d$ matrix $\overline{\mathbf{x}} = [x_1, \ldots, x_{\overline{n}}]^\top$, where $x_i \in \mathcal{X}$ is a $d \times 1$ column vector representing the location of the ith fault-free robot.

We define the interconnection topology of the fault-free robots by a directed graph $\overline{\mathcal{G}} = (\overline{\mathcal{V}}, \overline{\mathcal{E}})$ with $\overline{\mathcal{V}} = \overline{\mathcal{I}}$ and $\overline{\mathcal{E}} \subseteq \overline{\mathcal{V}} \times \overline{\mathcal{V}}$ obtained by removing from $\mathcal{E}$ all edges incident to faulty robot vertices. For the ith robot, we denote by $\overline{\mathcal{N}}_i$ its index set of fault-free in-neighbors in the graph $\overline{\mathcal{G}}$, and by n_{f_i} the number of its faulty neighbors in the graph $\mathcal{G}$ (i.e., $n_{f_i} = |\mathcal{N}_i \cap \mathcal{F}|$).

C. System Properties

As in much previous research on multirobot systems (see, e.g., [41]), we also consider a sequential motion cycle: *Look, Compute*, and *Move*. In the *Look* state, each robot takes a snapshot of the current state of its neighbors. Based upon this information, in the *Compute* state, each robot calculates its next control input, which is applied to the system in the *Move* state. Every robot is *memoryless* such that it generates its control input based upon only the information provided at the current time. We also assume that robots are *dimensionless*, so that multiple robots are allowed to be located at a same position; collision is not an issue (we briefly address this assumption in Section IX). However, we do assume that each robot can distinguish if a point is occupied by multiple robots.[3] Finally, the robots are *anonymous*, i.e., each robot is *indistinguishable* from all other robots. Thus, a fault-free robot cannot identify which of its neighbors are faulty, and which are not.

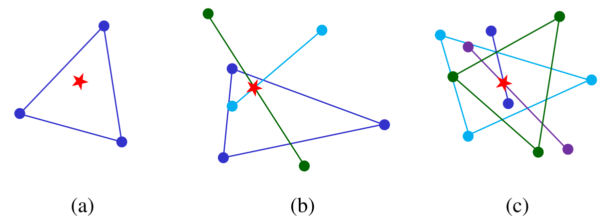


Fig. 1. Examples showing Tverberg points obtained with different number of points and division in $\mathbb{R}^2$ (circle: point, star: a Tverberg point). (a) 4 points, $r = 2$, (b) 7 points, $r = 3$, and (c) 10 points, $r = 4$.

III. Tverberg Partitions and Safe Points

Our fault-tolerant algorithms rely on the ability to construct a partition of n points into $n_f + 1$ disjoint subsets whose convex hulls have a nonempty intersection. As we describe below, this implies that the nonempty intersection will consist of points that lie in the convex hull of the set of fault-free nodes. In this section, we review results from discrete geometry that establish when and how such a partition can be constructed.

Definition III.1 (An r-divisible point set [42]): A set of n points is r-divisible if it can be partitioned into r pairwise disjoint subsets such that the intersection of the convex hulls of these r subsets is nonempty.

Using the definition, we state the classical Tverberg's theorem, which provides conditions that guarantee a given point set to be r-divisible.

Theorem III.1 (Tverberg's Theorem[42]): Any set of n points in $\mathbb{R}^d$ is r-divisible if $n \ge (d + 1)(r - 1) + 1$.

Corollary III.1 (Maximum Tverberg Partition): The maximum value of r for which a set of n points in $\mathbb{R}^d$ is guaranteed to be r-divisible using Theorem III.1 is $r = \lceil n/(d + 1)\rceil$.

The result follows from straightforward computations using the bound in Theorem III.1.

Fig. 1 shows examples of point sets in $\mathbb{R}^2$ that are r-divisible, for $n = 4, 7, 10$ and $r = 2, 3, 4$, respectively. Note in Fig. 1(a), the four points are partitioned into a set containing the three vertices of the triangle and a set containing only the point lying inside the triangle, the latter also being the only point in the intersection of the convex hulls of the two elements of the partition.

A partition $\Pi = \{P_1 \ldots P_r\}$ such that $\cap \text{conv}(P_i) \neq \emptyset$ is called a *Tverberg partition*, and the size of the partition $|\Pi| = r$ is called the *Tverberg depth* or merely *depth* when the context is clear. Note that for a given point set of size n, the Tverberg partition of depth r is not necessarily unique. A point $p \in \cap_i \text{conv}(P_i)$ is called a *Tverberg point of depth* r. A Tverberg point for a point set in $\mathbb{R}^d$ is analogous to the concept of the median for a point set in $\mathbb{R}$.

[3]This is sometimes called *multiplicity detection* capability.

Suppose $x = \{x_1 \dots x_n\}$ specifies the set of states of n robots, of which n_f are faulty. If x is $(n_f + 1)$ divisible, i.e., if $n \geq (d+1)n_f + 1$, then we can partition x into subsets $P_1, \dots P_{n_f+1}$, and at least one of these sets will contain only fault-free robots (since there are $n_f + 1$ sets but only n_f faulty robots). Furthermore, since $\cap_i \text{conv}(P_i) \subseteq \text{conv}(P_j)$ for all $j \in \{1, \dots, n_f + 1\}$, any Tverberg point of depth $n_f + 1$ is contained in the convex hull of a set of fault-free nodes. This motivates the following definition of safe point.

Definition III.2 (Safe point): For a set of n points in $\mathbb{R}^d$, of which at most n_f correspond to the positions of faulty nodes, a point p is n_f-safe (referred to as an n_f-safe point) if it has a neighborhood that is guaranteed to lie in the convex hull of the $n - n_f$ fault-free nodes.

There are at least two ways to ensure that a point p is n_f-safe: 1) It is a Tverberg point of depth $n_f + 1$ that lies in the relative interior of the nonempty intersection of the convex hulls of $(n_f + 1)$-disjoint subsets. These subsets constitute the associated Tverberg partition. 2) For every subset of size $\overline{n}$, p has a neighborhood that lies in the convex hull of the subset.

Our algorithm for fault-tolerant rendezvous explicitly constructs d-dimensional neighborhoods of safe points at each iteration. The following proposition provides a method to construct such a neighborhood.

Proposition III.1: For a set of n points in $\mathbb{R}^d$, that is, $(n_f + 1)$-divisible, if $z_1, \dots, z_{d+1}$ are Tverberg points of depth $n_f + 1$ in general position, each $q \in \text{ri}(\text{conv}(\{z_1, \dots, z_{d+1}\}))$ is n_f-safe.

Proof: By Lemma B.1, for each choice of the subset Q with size $\overline{n}$

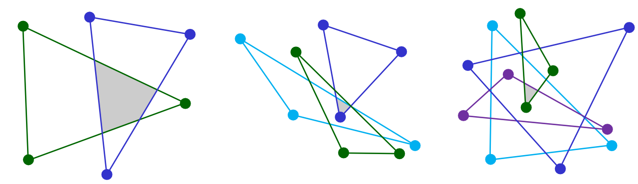
$$z_i \in \text{conv}(Q), \ \ i = 1, \dots, d+1. \quad (1)$$

Under the condition that $z_1, \dots, z_{d+1}$ is in general position $\text{ri}(\text{conv}(z_1, \dots, z_{d+1})) \neq \emptyset$. By the definition of convex sets, (1) implies $\text{conv}(z_1, \dots, z_{d+1}) \subset \text{conv}(Q)$. Hence, for each choice of $q \in \text{ri}(\text{conv}(z_1, \dots, z_{d+1}))$, and Q with size $\overline{n}$ $q \in \text{ri}(\text{conv}(Q))$. By Definition III.2, q is n_f-safe, and the proof is complete. ■

Unfortunately, even if a point set is $(n_f + 1)$-divisible, it is not always the case that there exist $d + 1$ Tverberg points of depth $n_f + 1$ in general position. This can be seen in the example shown in Fig. 1(a), in which the set of Tverberg points is a 0-dimensional (0-D) subset of $\mathbb{R}^2$ (i.e., a single point). In such cases, the relative interior of the Tverberg points is empty, and a d-dimensional neighborhood of safe points does not exist. In such cases, Tverberg points may lie on the boundary, rather than in the interior, of the convex hull of fault-free nodes. For example, in Fig. 1(a), if any vertex of the triangle corresponds to a faulty node, then the Tverberg point will be a vertex of the convex hull of the fault-free nodes, and not an interior point. This motivates the following definition.

Definition III.3 ((r, k)-divisible point set [43]): A set of n points in $\mathbb{R}^d$ is (r, k)-divisible if it can be partitioned into r pairwise disjoint subsets such that the intersection of the convex hulls of these r subsets is at least k-dimensional ($0 \leq k \leq d$).

If a point set is $(n_f + 1, d)$-divisible, then there exists a set of $d + 1$ Tverberg points of depth $n_f + 1$ in general position,



(a) (b) (c)

Fig. 2. Examples of $(r, 2)$-divisible points set in $\mathbb{R}^2$ (circle: position of nodes, shaded area: 2-D intersection). (a) 6 points, $r = 2$, (b) 9 points, $r = 3$, and (c) 12 points, $r = 4$.

which allows the application of Proposition III.1. Reay [43] and Roudneff [44], [45] have given conditions under which a point set is (r, k)-divisible.

Conjecture III.1 (Reay's conjecture [43]): A set of n points in general position in $\mathbb{R}^d$ (with $0 \leq k \leq d$) is (r, k)-divisible if $n \geq (d+1)(r-1) + k + 1$.

For the case of $k = d$, Reay's conjecture has been shown to be true for $2 \leq d \leq 8$ [43]–[46].

Proposition III.2 (Birch [46] and Roudneff [44], [45]): For $d = 2, 3, \dots, 8$, any set of n points in general position in $\mathbb{R}^d$ is (r, d)-divisible if $n \geq r(d+1)$.

This result allows us to apply our algorithm to robots with state spaces $\mathcal{X} \subseteq \mathbb{R}^d$ for $d \leq 8$, and provides the following sufficient condition for constructing a neighborhood of n_f-safe points.

Corollary III.2 (Sufficient condition for n_f-safe neighborhood): For $d = 2, 3 \dots, 8$, any set of $n \geq (n_f + 1)(d+1)$ points in general position will have a Tverberg partition of depth $n_f + 1$, along with a set of $d + 1$ Tverberg points $z_1, \dots z_{d+1}$ such that every $q \in \text{ri}(\text{conv}(\{z_1, \dots, z_{d+1}\}))$ is n_f-safe.

Three examples are shown in Fig. 2.

A. Computational Complexity and Approximations

In general, the problem of computing Tverberg partitions is NP-Hard. Given n points in d dimensional space, the best known algorithm to obtain a Tverberg partition of depth $r = \lceil n/(d+1) \rceil$ requires up to $O(n^d)$ computation time.

A recent study [47] reports a *Lifting Algorithm* that computes an approximate[4] Tverberg partition of size $\lceil n/2^d \rceil$ and a sample Tverberg point in linear time in n and quasi-polynomial time in d, i.e., $d^{O(1)}n$, under the condition that the n points are in general position.[5] The algorithm is a recursive projection-lifting algorithm. First, the point set in $\mathbb{R}^d$ is projected onto hyperplanes, H_m, of successively lower dimension, m, eventually onto the real line, $H_1 = \mathbb{R}^1$, and a partition (not a Tverberg partition) is constructed for this projection onto H_1. Then, for $m = 2, \dots, d$,

[4] *Approximate* in the sense that, given the same number of points, the algorithm obtains a Tverberg partition with decreased depth.

[5] The set of nongeneral configurations is measure zero, and thus any randomly drawn set of n points will be in general position with probability 1. Nevertheless, since our algorithms run on computers with finite precision arithmetic, the general position assumption must be verified in practice. When the general position assumption does not hold, small random perturbations may be applied to produce a point set in general configuration.

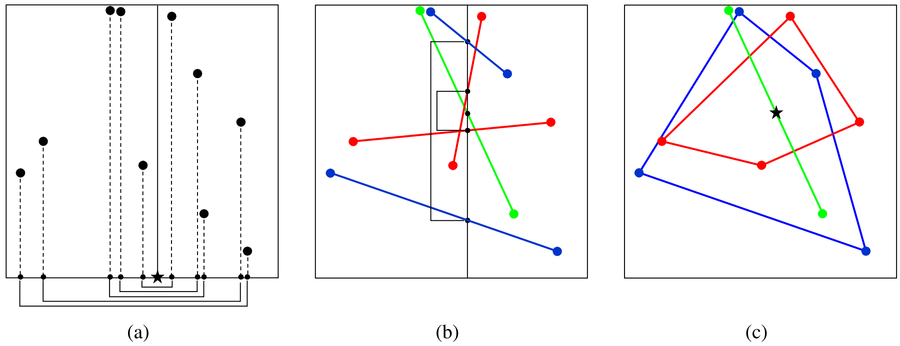


Fig. 3. Procedure to obtain a Tverberg point by the lifting method (the figure is inspired by that contained in [47]).

a partition for H_m is computed using a lifting method applied to the partition of H_{m-1}. The algorithm terminates when $m = d$, and the resulting partition is a Tverberg partition of depth $\lceil n/2^d \rceil$.

While a general description of the algorithm is beyond the scope of this paper, it can easily be illustrated for the case of a point set in $\mathbb{R}^2$. Fig. 3 illustrates the procedure. In this case, there is only one step of projection (onto hyperplane $H_1 = \mathbb{R}^1$) and one lifting step. First, the n points are projected onto $\mathbb{R}^1$, and a partition is formed by creating 2-tuples of points that have successively increasing distance to the left and the right of the median. For the example shown in Fig. 3(a), the partition consists of five 2-tuples of points. Next, the line l through the median and perpendicular to $\mathbb{R}^1$ is constructed. For each 2-tuple in the partition of $\mathbb{R}^1$, the points are lifted back into $\mathbb{R}^2$, and the intersection of l with the convex hull of the lifted points is computed, as shown in Fig. 3(b). The median of these intersection points is computed, and 2-tuples of convex hulls are constructed by establishing symmetric correspondences about this median (analogous to the process for H_1). In this example, since there were an odd number (five) of elements in the partition of H_1, we construct two 2-tuples, and one 1-tuple. These points that are included in corresponding 1- or 2-tuples define the Tverberg partition of the original point set, which is shown in Fig. 3(c). The depth of the resulting Tverberg partition is $\lceil n/2^d \rceil = 3$. Complete details of the lifting method can be found in [47].

IV. Fault-Tolerant Algorithm for Distributed Consensus

In this section, we introduce our *Approximate Distributed Robust Convergence* algorithm, which we call ADRC. The basic idea behind our algorithm is that each fault-free robot constructs $d+1$ maximal depth Tverberg points, uses these to define a safe point, and then executes a motion toward the safe point. The basic procedure is shown in Algorithm 1. We now present the details of the algorithm.

For each $i \in \overline{\mathcal{I}}$, define $\widetilde{n}_{f_i}(t)$ as the maximal value of n_{f_i} for which we can ensure the existence of an n_{f_i}-safepoint with respect to the neighborhood $\mathcal{N}_i(t)$. From Corollary III.2, we have

$$\widetilde{n}_{f_i}(t) \leq \frac{|\mathcal{N}_i(t)|}{d+1} - 1 \qquad (2)$$

Algorithm 1: ADRC

Require: $\delta > 0$, $\{\{\mathcal{N}_i(t), \alpha_i(t)\}_{i\in\overline{\mathcal{I}}}\}_{t\in\mathbb{Z}_{\geq 0}}$
foreach *iteration* t **do**
  for *each fault-free robot* i **do**
    // Look
    Compute $\widetilde{n}_{f_i}(t)$
    // Compute
    Compute an $\widetilde{n}_{f_i}(t)$-safe point, $s_i(t)$
    $u_i(t) \leftarrow \alpha_i(t)(s_i(t) - x_i(t))$
    // Move
    if $\|u_i(t)\| < \delta$ **then**
      Halt
    else
      Execute control $u_i(t)$
    end
  end
end

We use the *Lifting Algorithm* mentioned in Section III-A to construct a Tverberg partition of $\mathcal{N}_i(t)$ of depth $r = \lceil |\mathcal{N}_i(t)|/2^d \rceil$. This imposes the constraint that

$$\widetilde{n}_{f_i}(t) \leq \left\lceil \frac{|\mathcal{N}_i(t)|}{2^d} \right\rceil - 1 \qquad (3)$$

Combining (2) and (3), we obtain

$$\widetilde{n}_{f_i}(t) \leq \min\left\{ \left\lceil \frac{|\mathcal{N}_i(t)|}{2^d} \right\rceil, \frac{|\mathcal{N}_i(t)|}{d+1} \right\} - 1 \qquad (4)$$

Proposition IV.1 (Existence of a safe point): For a node with neighbors $\mathcal{N}_i(t)$, the Lifting Algorithm will construct a Tverberg partition of depth $\widetilde{n}_{f_i}(t) + 1$ and an $\widetilde{n}_{f_i}(t)$-safe point for

$$\widetilde{n}_{f_i}(t) = \min\left\{ \left\lceil \frac{|\mathcal{N}_i(t)|}{2^d} \right\rceil, \left\lfloor \frac{|\mathcal{N}_i(t)|}{d+1} \right\rfloor \right\} - 1. \qquad (5)$$

For the ith robot, our algorithm generates $d+1$ Tverberg points of depth $\widetilde{n}_{f_i}(t) + 1$. This is done by invoking the lifting algorithm $d+1$ times, each time using a randomly chosen direction for the hyperplane projection steps. The center of mass of these $d+1$ points, $s_i(t)$, is an $\widetilde{n}_{f_i}(t)$-safe point. The basic procedure is shown in Algorithm 2. Fig. 4 illustrates the process for $d = 2$ and $|\mathcal{N}_i(t)| = 20$. Fig. 4(a)–(c) shows the results of three applications of the lifting algorithm, each of which produces a Tverberg point of depth 5. Fig. 4(d) shows the resulting safe point s_i.

Algorithm 2: Calculate $\widetilde{n}_{f_i}(t)$-Safe point

```
Require: N_i(t), d, maxIter
ñ_{f_i}(t) ← min{ ⌈|N_i(t)|/2^d⌉, ⌊|N_i(t)|/(d+1)⌋ } − 1
j ← 0, k ← 0, z_i[] ← ∅
while j < d+1 and k < maxIter do
    H_{d−1} ← genRandHyperplane
    /* Find a d−1 dimensional hyperplane that are
       normal to a random unit vector in R^d          */
    z_ic ← tverbergPnt(N_i(t), ñ_{f_i}(t)+1, H_{d−1})
    /* Calculate a Tverberg point of depth
       ñ_{f_i}(t)+1 using the lifting algorithm with
       the initial hyperplane generated at random,
       H_{d−1}                                        */
    if isGeneralPosition(z_i[], z_ic) then
        z_i[j] ← z_ic
        j ← j+1
    end
    k ← k+1
end
if j < d then
    s_i(t) ← safePnt(N_i(t), ñ_{f_i}(t)+1)
    /* Obtain ñ_{f_i}(t)-safe point directly if for any
       reason, lifting method fails to generate
       unique d+1 Tverberg points in general
       position                                       */
else
    s_i(t) ← mean(z_i[])
end
return s_i(t)
```

The action taken by the ith robot at time t is simply to move toward the safe point $s_i(t)$. This leads to the following state update equation:

$$x_i(t+1) = x_i(t) + u_i(t) \tag{6}$$

where

$$u_i(t) = \alpha_i(t)(s_i(t) - x_i(t)) \tag{7}$$

and $\alpha_i(t)$ is dynamically chosen parameter in the range, $0 < \alpha_{\min} \leq \alpha_i(t) \leq \alpha_{\max} < 1$, such that $u_i(t)$ does not violate constraints, e.g., maximum allowable displacement per stage. It is possible to consider systems with more complex dynamics than those of (6), but we do not do so in this paper.

Remark IV.1: Every safe point can be represented as a convex combination of the positions of a node's fault-free neighbors. In this sense, a safe point is independent of the locations of the nonconforming robots. Thus, for each fault-free robot, the motion toward a safe point depends only on the positions of fault-free neighbors and the position of the robot itself.

Our algorithm is *aggressive*, in that it computes the maximal number of partitions possible, thus providing maximal fault-tolerance relative to the neighborhood size of each robot. This aggressiveness comes at a computational cost. We address this, and other computational issues, in Section VI.

Without loss of generality, in the remainder of this paper, we assume that robots evolving under ADRC are always in general position. The assumption is often found in the computational geometry literature [48], [49], and is used to avoid computational issues due to the occurrence of degenerate points.

V. Analysis of ADRC

In this section, we analyze the convergence of ADRC. The main result is given in Theorem V.1, which establishes conditions under which ADRC will converge. Our analysis builds on previous work in distributed control of fault-free systems [3], [50] and in fault-tolerant computing [5], [6]. The former is not concerned with cases in which individual nodes may fail, while the latter does not consider the case of time-varying network topology.

We begin by showing that under ADRC the evolution of the fault-free nodes can be described as a time-varying linear system that depends *only on the fault-free nodes*, and deriving certain properties of the corresponding time-varying system matrix. We then establish an appropriate concept of joint connectivity [3], [50], which we call *repeated reachability*. The concept will be used to provide a minimally restrictive condition on the connectivity graph of the fault-free nodes for convergence under ADRC. Finally, we employ properties of stochastic matrices [51]–[54] to demonstrate convergence of ADRC.

A. Evolution of the Fault-Free Nodes as an Linear Time-Varying System

The behavior of the fault-free robots executing ADRC can be described as a discrete-time linear time-varying (LTV) system. In particular, the system evolution can be simply expressed as a backward product of nonhomogeneous system matrices. Our approach closely follows that given in [5].

Proposition V.1: For an F-MRS in which the fault-free nodes execute ADRC, if $n_{f_i}(t) \leq \widetilde{n}_{f_i}(t)$ for each $i \in \overline{\mathcal{I}}$, then the evolution of the fault-free nodes, $\overline{\mathbf{x}}(t)$, can be represented by an LTV system of the form

$$\overline{\mathbf{x}}(t+1) = \mathbf{M}(t)\overline{\mathbf{x}}(t), \;\; t = 0, 1, 2, \ldots, \tag{8}$$

in which $\overline{\mathbf{x}}(t)$ is an $\overline{n} \times d$ matrix, and $\mathbf{M}(t)$ is an $\overline{n} \times \overline{n}$ row-stochastic matrix with $[\mathbf{M}]_{ij}(t) > 0$ for $i = j$, or $j \in \overline{\mathcal{N}}_i(t)$.

Proof: Let $\overline{\mathcal{Y}}_i := \{x_j\}_{j \in \overline{\mathcal{N}}_i}$ denote the set of positions of the fault-free neighbors of node i. By Proposition IV.1, ADRC constructs an $n_{f_i}(t)$-safe point $s_i(t)$ for each $i \in \overline{\mathcal{I}}$. Thus, there exists some set of fault-free neighbors with positions $P \subseteq \overline{\mathcal{Y}}_i$ such that

$$s_i(t) \in \mathrm{ri}(\mathrm{conv}(P)) \subseteq \mathrm{ri}(\mathrm{conv}(\overline{\mathcal{Y}}_i(t)).$$

Thus (see Proposition B.1), for each $i \in \overline{\mathcal{I}}$, there is a set of nonzero weights $\{\lambda_{ij}\}_{j \in \overline{\mathcal{N}}_i(t)}$ such that $s_i(t)$ can be represented by

$$s_i(t) = \sum_{j \in \overline{\mathcal{N}}_i(t)} \lambda_{ij}(t) x_j(t) \tag{9}$$

with $\sum_{j \in \overline{\mathcal{N}}_i(t)} \lambda_{ij}(t) = 1$, and $\lambda_{ij}(t) > 0$ for all $j \in \overline{\mathcal{N}}_i(t)$.

Plugging equation (9) into (6) and (7), for each $i \in \overline{\mathcal{I}}$, we obtain

$$x_i(t+1) = (1 - \alpha_i(t))x_i(t) + \alpha_i(t) \left(\sum_{j \in \overline{\mathcal{N}}_i(t)} \lambda_{ij}(t) x_j(t) \right).$$

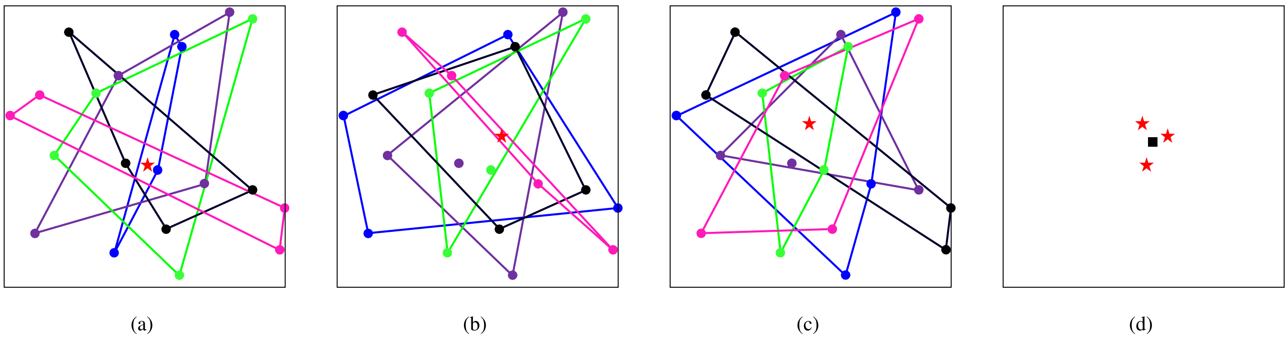


Fig. 4. Safe point calculated by lifting method with 20 points. In (a)–(c), stars are Tverberg points and circles are positions of the nodes, and in (d) square symbol is the safe point, stars are Tverberg points obtained from (a)–(c).

There is such an equation for each $i \in \overline{\mathcal{I}}$, and these may be combined to obtain (8) by defining the $\overline{n} \times \overline{n}$ matrix $\mathbf{M}(t)$ as follows. For $i, j \in \overline{\mathcal{I}}$

$$\mathbf{M}(t) = \begin{cases} 1 - \alpha_i(t) & \text{if } j = i \\ \alpha_i(t)\lambda_{ij}(t) & \text{if } j \neq i \text{ and } j \in \overline{\mathcal{N}}_i(t) \\ 0 & \text{otherwise.} \end{cases}$$

Note that $\mathbf{M}(t)$ is row stochastic, with diagonal elements $[\mathbf{M}]_{ii}(t) = 1 - \alpha_i(t) \geq 1 - \alpha_{\max}$, and $[\mathbf{M}]_{ij}(t) = \alpha_i(t)\lambda_{ij}(t)$ for $i \neq j$ if $j \in \overline{\mathcal{N}}_i(t)$. ∎

Using (8), we may define the state transition matrix $\Phi(t_F, t_I)$, for $t_I, t_F \in \mathbb{Z}_{\geq 0}$, where $t_I \leq t_F$ using a *backward product* of system matrices

$$\begin{aligned} \Phi(t_F, t_I) &= \mathbf{M}(t_F)\mathbf{M}(t_F - 1)\ldots\mathbf{M}(t_I) \\ &= \prod_{t=t_I}^{t_F} \mathbf{M}(t). \end{aligned} \tag{10}$$

B. Jointly Reachable Graphs

In this section, we define the concept of *joint reachability*, which is analogous to the concept of joint connectivity introduced in [3] and [50]. For a jointly reachable sequence of graphs, we give a relationship between the adjacency matrix for the union of the graphs and the adjacency matrices for the individual graphs in the sequence. We then extend the concept of joint reachability to infinite sequences of graphs by defining the concept of *repeated reachability*, which will play a key role to establish a minimally restrictive condition in Theorem V.1.

We denote by $\overline{\mathcal{G}}(t)$ the connectivity graph of the fault-free nodes at time t, and by $\overline{\mathcal{A}}(t)$ the corresponding adjacency matrix.

Definition V.1 (Jointly reachable sequence of graphs): For $j \in \mathbb{N}$, consider a sequence of graphs

$$\overline{\mathcal{G}}(T_j), \ldots, \overline{\mathcal{G}}(T_{j+1} - 1)$$

of length $L_j = T_{j+1} - T_j$, and with common vertex set $\overline{\mathcal{V}}$, such that $\overline{\mathcal{G}}(t) = (\overline{\mathcal{V}}, \overline{\mathcal{E}}(t))$ for $t = T_j, \ldots, T_{j+1} - 1$. The union of these graphs is defined by

$$\widetilde{\mathcal{G}}_{T_j, T_{j+1}-1} = \bigcup_{t=T_j}^{T_{j+1}-1} \overline{\mathcal{G}}(t) = \left(\overline{\mathcal{V}}, \bigcup_{t=T_j}^{T_{j+1}-1} \overline{\mathcal{E}}(t)\right).$$

We say that the sequence is jointly reachable if there exists some $v \in \overline{\mathcal{V}}$ such that for each $v' \neq v$ in $\overline{\mathcal{V}}$ there exists a path from v' to v in $\widetilde{\mathcal{G}}_{T_j, T_{j+1}-1}$.

For a sequence of graphs $\overline{\mathcal{G}}(T_j), \ldots, \overline{\mathcal{G}}(T_{j+1} - 1)$, we denote by $\widetilde{\mathcal{A}}_j$ the adjacency matrix for $\widetilde{\mathcal{G}}_{T_j, T_{j+1}-1}$. The following lemma provides a useful relationship between $\widetilde{\mathcal{A}}_j$ and the individual adjacency matrices $\overline{\mathcal{A}}(t)$, $T_j \leq t \leq T_{j+1} - 1$.

Lemma V.1: For $\overline{\mathcal{A}}(T_j), \ldots, \overline{\mathcal{A}}(T_{j+1} - 1)$, adjacency matrices for a sequence of graphs $\overline{\mathcal{G}}(T_j), \ldots, \overline{\mathcal{G}}(T_{j+1} - 1)$, with the adjacency matrix $\widetilde{\mathcal{A}}_j$ for $\widetilde{\mathcal{G}}_{T_j, T_{j+1}-1}$, the following inequality holds:

$$\mathbf{I}_{\overline{n}} + \widetilde{\mathcal{A}}_j \leq \prod_{t=T_j}^{T_{j+1}-1} (\mathbf{I}_{\overline{n}} + \overline{\mathcal{A}}(t)). \tag{11}$$

Proof: Since $\widetilde{\mathcal{G}}_{T_j, T_{j+1}-1}$ is the *union* graph of the finite graph sequence $\{\overline{\mathcal{G}}(t)\}_{T_j}^{T_{j+1}-1}$

$$\widetilde{\mathcal{E}}_{T_j, T_{j+1}-1} = \bigcup_{t \in [T_j, T_{j+1}-1]} \overline{\mathcal{E}}(t)$$

which implies

$$\widetilde{\mathcal{A}}_j = \widetilde{\mathcal{A}}_{T_j, T_{j+1}-1} \leq \sum_{t=T_j}^{T_{j+1}-1} \overline{\mathcal{A}}(t). \tag{12}$$

Simple calculation yields

$$\mathbf{I}_{\overline{n}} + \sum_{t=T_j}^{T_{j+1}-1} \overline{\mathcal{A}}(t) \leq \prod_{t=T_j}^{T_{j+1}-1} (\mathbf{I}_{\overline{n}} + \overline{\mathcal{A}}(t)). \tag{13}$$

Now, we can combine the two inequalities (12) and (13) to obtain

$$\mathbf{I}_{\overline{n}} + \widetilde{\mathcal{A}}_j \leq \prod_{t=T_j}^{T_{j+1}-1} (\mathbf{I}_{\overline{n}} + \overline{\mathcal{A}}(t)).$$

∎

The following related result holds for any sequence of adjacency matrices. This includes $\overline{\mathcal{A}}(t)$, $\widetilde{\mathcal{A}}_j$, or a general adjacency matrix $\mathcal{A}_j$.

Lemma V.2: Let $\mathcal{A}_0, \mathcal{A}_1, \ldots$ be a sequence of $\overline{n} \times \overline{n}$ adjacency matrices and let $\eta = (\overline{n}-1)2^{\overline{n}(\overline{n}-1)}$. Then, for any $l = 0, 1, \ldots,$ we have

$$(\mathbf{I}_{\overline{n}} + \mathcal{A}_{l^*})^{\overline{n}-1} \leq \prod_{j=l}^{l+\eta-1} (\mathbf{I}_{\overline{n}} + \mathcal{A}_j)$$

for $l^* \in \mathbb{Z}_{\geq 0}$ such that the term $(\mathbf{I}_{\overline{n}} + \mathcal{A}_{l^*})$ appears at least $\overline{n}-1$ times in the right-hand side.

Proof: Since there are at most $2^{\overline{n}(\overline{n}-1)}$ possible adjacency matrices in any set of $\eta = (\overline{n}-1)2^{\overline{n}(\overline{n}-1)}$ adjacency matrices, at least one matrix, $\mathcal{A}_{l^*}$ must appear at least $\overline{n}-1$ times. The inequality then follows from the fact that $\mathcal{A}_j$ consists of non-negative entries. ■

Joint reachability is a property of finite-length sequences of graphs. To establish conditions for convergence of ADRC, we extend this notion to infinite graph sequences with the following definition, which is similar to the concept of *repeatedly jointly rooted graph sequence* previously defined in [50].

Definition V.2 (Repeatedly reachable graph sequence): An infinite sequence of graphs $\overline{\mathcal{G}}(0), \overline{\mathcal{G}}(1), \overline{\mathcal{G}}(2)\ldots$ is said to be *repeatedly reachable* if there exists a sequence of times $0 = T_1 < T_2 \ldots$ such that $T_{j+1} - T_j = L_j < \infty$ and the subsequence $\overline{\mathcal{G}}(T_j),\ \overline{\mathcal{G}}(T_j+1),\ \ldots,\ \overline{\mathcal{G}}(T_{j+1}-1)$ is jointly reachable for all j.

We denote by $L_{\max}$ the least uniform upper bound for all L_j, i.e., $L_j \leq L_{\max},$ for all $j \in \mathbb{N}$.

In other words, the sequence $\overline{\mathcal{G}}(0), \overline{\mathcal{G}}(1), \overline{\mathcal{G}}(2)\ldots$ is repeatedly reachable if it can be partitioned into contiguous finite-length subsequences of lengths $L_j \leq L_{\max}$ that are themselves jointly reachable. Note that the condition "the sequence $\overline{\mathcal{G}}(0), \overline{\mathcal{G}}(1), \overline{\mathcal{G}}(2)\ldots$ is repeatedly reachable" is significantly less restrictive than the condition "$\overline{\mathcal{G}}(t)$ strongly connected for all $t \in \mathbb{Z}_{\geq 0}$."

C. Convergence of ADRC

In this section, we provide the main theoretical result of this paper. We begin by using properties of jointly reachable graphs to infer a lower bound on the backward product of system matrices (Proposition V.2 whose proof relies on Lemma V.3). Then, using Proposition V.2, we derive a bound on the coefficient of ergodicity for a finite backward product of system matrices (Lemma V.4). Finally, we present the main result in Theorem V.1.

Lemma V.3: For an F-MRS in which the fault-free nodes execute ADRC, and for a jointly reachable sequence of graphs $\overline{\mathcal{G}}(T_j), \ldots, \overline{\mathcal{G}}(T_{j+1}-1)$ with $\widetilde{\mathcal{A}}_j$ the adjacency graph for $\widetilde{\mathcal{G}}_{T_j, T_{j+1}-1}$, if $n_{f_i}(t) \leq \widetilde{n}_{f_i}(t)$ for all $i \in \overline{\mathcal{I}}$, for each $j \in \mathbb{N}$ there exists $\gamma_j \in (0,\ 1)$ such that

$$\gamma_j^{L_j}(\mathbf{I}_{\overline{n}} + \widetilde{\mathcal{A}}_j) \leq \prod_{t=T_j}^{T_{j+1}-1} \mathbf{M}(t) \tag{14}$$

where $L_j = T_{j+1} - T_j$.

Proof: By Proposition V.1, $[\mathbf{M}(t)]_{ij} > 0,$ for $i = j$ or $[\overline{\mathcal{A}}(t)]_{ij} \neq 0$, and thus, for each t there exists $\gamma(t) \in (0,\ 1)$ such that

$$\gamma(t)(\mathbf{I}_{\overline{n}} + \overline{\mathcal{A}}(t)) \leq \mathbf{M}(t).$$

Taking the product for each side over the sequence, we obtain

$$\prod_{t=T_j}^{T_{j+1}-1} \gamma(t)(\mathbf{I}_{\overline{n}} + \overline{\mathcal{A}}(t)) \leq \prod_{t=T_j}^{T_{j+1}-1} \mathbf{M}(t). \tag{15}$$

Let γ_j be a lower bound on $\gamma(t)$ for the interval T_j to $T_{j+1}-1$. Then

$$\gamma_j^{L_j} \prod_{t=T_j}^{T_{j+1}-1} (\mathbf{I}_{\overline{n}} + \overline{\mathcal{A}}(t)) \leq \prod_{t=T_j}^{T_{j+1}-1} \gamma(t)(\mathbf{I}_{\overline{n}} + \overline{\mathcal{A}}(t)). \tag{16}$$

Finally, Combining (15) and (16) and applying Lemma V.1, we obtain

$$\gamma_j^{L_j}(\mathbf{I}_{\overline{n}} + \widetilde{\mathcal{A}}_j) \leq \prod_{t=T_j}^{T_{j+1}-1} \mathbf{M}(t).$$

■

Proposition V.2: For an F-MRS in which the fault-free nodes execute ADRC, if

1) $n_{f_i}(t) \leq \widetilde{n}_{f_i}(t)$ for each $i \in \overline{\mathcal{I}}$ and $t \in \mathbb{Z}_{\geq 0}$, and
2) the sequence of connectivity graphs for the fault-free nodes, $\overline{\mathcal{G}}(0), \overline{\mathcal{G}}(1), \overline{\mathcal{G}}(2), \ldots,$ is repeatedly reachable,

then for each $l \in \mathbb{Z}_{\geq 1}$, there exists $\gamma \in (0,\ 1)$, and $i \in \overline{\mathcal{I}}$ such that

$$\gamma^{T_{l+\eta}-T_l}\mathbf{1}_{\overline{n}\times 1} \leq \left[\prod_{t=T_l}^{T_{l+\eta}-1} \mathbf{M}(t)\right]_i \tag{17}$$

in which $0 = T_0 < T_1 < T_2 \ldots$ is a sequence of times such that $T_{j+1} - T_j = L_j < \infty$ and $\overline{\mathcal{G}}(T_j), \ldots, \overline{\mathcal{G}}(T_{j+1}-1)$ is jointly reachable for all j; and $\eta = (\overline{n}-1)2^{\overline{n}(\overline{n}-1)}$.

Proof: By Lemma V.3, there exists $\gamma_j \in (0,\ 1)$ such that

$$\gamma_j^{L_j}(\mathbf{I}_{\overline{n}} + \widetilde{\mathcal{A}}_j) \leq \prod_{t=T_j}^{T_{j+1}-1} \mathbf{M}(t) \tag{18}$$

for $L_j = T_{j+1} - T_j$.

We may compute the product of each side over the interval from $j = l$ to $j = l + \eta - 1$

$$\prod_{j=l}^{l+\eta-1} \gamma_j^{L_j}(\mathbf{I}_{\overline{n}} + \widetilde{\mathcal{A}}_j) \leq \prod_{j=l}^{l+\eta-1} \prod_{t=T_j}^{T_{j+1}-1} \mathbf{M}(t). \tag{19}$$

Now, let γ be a uniform lower bound for the γ_j, then

$$\prod_{j=l}^{l+\eta-1} \gamma^{L_j}(\mathbf{I}_{\overline{n}} + \widetilde{\mathcal{A}}_j) \leq \prod_{j=l}^{l+\eta-1} \gamma_j^{L_j}(\mathbf{I}_{\overline{n}} + \widetilde{\mathcal{A}}_j). \tag{20}$$

Since $L_j = T_{j+1} - T_j$, simple calculations yield

$$\sum_{j=l}^{l+\eta-1} L_j = \sum_{j=l}^{l+\eta-1} T_{j+1} - T_j = T_{l+\eta} - T_l.$$

Combining this result with (19) and (20), we obtain

$$\gamma^{T_{l+\eta}-T_l}\prod_{j=l}^{l+\eta-1}(\mathbf{I}_{\overline{n}}+\widetilde{\mathcal{A}}_j)\leq\prod_{t=T_l}^{T_{l+\eta}-1}\mathbf{M}(t). \tag{21}$$

We now apply Lemma V.2. Let l^* be such that the term $(\mathbf{I}_{\overline{n}}+\widetilde{\mathcal{A}}_{l^*})$ appears at least $\overline{n}-1$ times in the product on the left-hand side of (21). Then (by Lemma V.2)

$$\left(\mathbf{I}_{\overline{n}}+\widetilde{\mathcal{A}}_{l^*}\right)^{\overline{n}-1}\leq\prod_{j=l}^{l+\eta-1}\left(\mathbf{I}_{\overline{n}}+\widetilde{\mathcal{A}}_j\right)$$

and thus

$$\gamma^{T_{l+\eta}-T_l}\left(\mathbf{I}_{\overline{n}}+\widetilde{\mathcal{A}}_{l^*}\right)^{\overline{n}-1}\leq\gamma^{T_{l+\eta}-T_l}\prod_{j=l}^{l+\eta-1}(\mathbf{I}_{\overline{n}}+\widetilde{\mathcal{A}}_j). \tag{22}$$

But the matrix $\widetilde{\mathcal{A}}_{l^*}$ is an adjacency matrix for a graph with a globally reachable node, and thus, using Lemma B.2, there is some i such that

$$\gamma^{T_{l+\eta}-T_l}\mathbf{1}_{\overline{n}\times 1}\leq\gamma^{T_{l+\eta}-T_l}\left[\left(\mathbf{I}_{\overline{n}}+\widetilde{\mathcal{A}}_{l^*}\right)^{\overline{n}-1}\right]_i. \tag{23}$$

Finally, combining (21), (22), and (23), we obtain

$$\gamma^{T_{l+\eta}-T_l}\mathbf{1}_{\overline{n}\times 1}\leq\left[\prod_{t=T_l}^{T_{l+\eta}-1}\mathbf{M}(t)\right]_i.$$

■

The following lemma provides a bound on the coefficient of ergodicity, τ_1, for a finite backward product of system matrices evolving under ADRC. A brief review of stochastic matrices and ergodicity is provided in Appendix A. A more comprehensive review can be found in [51] and [52].

Lemma V.4: Let $\tau_1(\mathbf{S})$ denote the coefficient of ergodicity for a stochastic matrix $\mathbf{S}$. For an F-MRS in which the fault-free nodes execute ADRC, if

1) $n_{f_i}(t)\leq\widetilde{n}_{f_i}(t)$ for each $i\in\overline{\mathcal{I}}$ and $t\in\mathbb{Z}_{\geq 0}$, and
2) the sequence of connectivity graphs for the fault-free nodes, $\overline{\mathcal{G}}(0),\overline{\mathcal{G}}(1),\overline{\mathcal{G}}(2),\ldots$, is repeatedly reachable,

then the following inequality holds:

$$\prod_{h=0}^{v}\tau_1\left(\prod_{t=T_{h\eta}}^{T_{(h+1)\eta}-1}\mathbf{M}(t)\right)\leq(1-\gamma^{L_{\max}\eta})^v \tag{24}$$

in which $0=T_0<T_1<T_2\ldots$ is a sequence of times such that $T_{j+1}-T_j=L_j<\infty$ and $\overline{\mathcal{G}}(T_j),\ldots,\overline{\mathcal{G}}(T_{j+1}-1)$ is jointly reachable for all j; and $\eta=(\overline{n}-1)2^{\overline{n}(\overline{n}-1)}$.

Proof: Proposition A.1 together with Proposition V.2 implies

$$\tau_1\left(\prod_{t=T_{h\eta}}^{T_{(h+1)\eta}-1}\mathbf{M}(t)\right)\leq 1-\gamma^{T_{(h+1)\eta}-T_{h\eta}} \tag{25}$$

which holds for all $h\in\mathbb{Z}_{\geq 0}$. Let $L_{\max}\in\mathbb{N}$ be a uniform upper bound for the L_j. Thus, for all $h\in\mathbb{Z}_{\geq 0}$

$$1-\gamma^{T_{(h+1)\eta}-T_{h\eta}}\leq 1-\gamma^{L_{\max}\eta}. \tag{26}$$

Combining (25) and (26), and computing the product of each side for $h=0$ to $h=v$ for some v, we obtain

$$\prod_{h=0}^{v}\tau_1\left(\prod_{t=T_{h\eta}}^{T_{(h+1)\eta}-1}\mathbf{M}(t)\right)\leq(1-\gamma^{L_{\max}\eta})^v.$$

Note that the values taken by t in the left-hand side range from $t=0$ when $h=0$ to $t=T_{(v+1)\eta}-1$ for $h=v$. ■

We now state the main theorem of this paper, which guarantees that under certain connectivity conditions, the fault-free robots executing ADRC will converge to within any desired ϵ bound, regardless of the behavior of the faulty robots. The proof of the theorem provides a bound on the time required to achieve ϵ-convergence.

Theorem V.1 (Convergence of ADRC): For an F-MRS in which the fault-free nodes execute ADRC, if

1) $n_{f_i}(t)\leq\widetilde{n}_{f_i}(t)$ for each $i\in\overline{\mathcal{I}}$ and $t\in\mathbb{Z}_{\geq 0}$, and
2) the sequence of connectivity graphs for the fault-free nodes, $\overline{\mathcal{G}}(0),\overline{\mathcal{G}}(1),\overline{\mathcal{G}}(2),\ldots$, is repeatedly reachable,

then for every $t\geq 0$, fault-free pair $i,j\in\overline{\mathcal{I}}$, and $\epsilon>0$, there is some $t_\epsilon>0$ such that $\|x_i(t)-x_j(t)\|<\epsilon$ for all $t>t_\epsilon$.

Proof: Our proof, which uses the ergodicity of a backward product of stochastic matrices, is directly inspired by the results given in [5] and [4], and relies on classical results given in [55]. Let $0=T_0<T_1<T_2\ldots$ denote a sequence of times such that $T_{j+1}-T_j=L_j<\infty$, and $\overline{\mathcal{G}}(T_j),\ldots,\overline{\mathcal{G}}(T_{j+1}-1)$ is jointly reachable for all j. Such a sequence exists, since $\overline{\mathcal{G}}(0),\overline{\mathcal{G}}(1),\overline{\mathcal{G}}(2),\ldots$ is repeatedly reachable. For an arbitrary time $t\geq 0$, let $v^\star$ be the largest nonnegative integer that satisfies $T_{(v^\star+1)\eta}-1\leq t$, and $t^\star=T_{(v^\star+1)\eta}-1$, with $\eta=(\overline{n}-1)2^{\overline{n}(\overline{n}-1)}$.

We can express the system evolution of fault-free robots using state transition matrix Φ given in (10) as

$$\begin{aligned}\overline{\mathbf{x}}(t)&=\Phi(t,0)\overline{\mathbf{x}}(0) &(27)\\ &=\Phi(t,T_{(v^\star+1)\eta})\Phi(T_{(v^\star+1)\eta}-1,0)\overline{\mathbf{x}}(0) &(28)\\ &=\Phi(t,t^\star)\underbrace{\Phi(t^\star,0)\overline{\mathbf{x}}(0)}_{\overline{\mathbf{x}}(t^\star)}. &(29)\end{aligned}$$

Now, for each time $t\geq 0$, there are the following two possibilities:

1) $t=t^\star=T_{(v^\star+1)\eta}-1$ for some $v^\star$, or
2) $T_{(v^\star+1)\eta}<t<T_{(v^\star+1)\eta}-1$, for some $v^\star$.

We will first consider the case when $t=t^\star$, and evaluate the maximum difference of the rows of $\Phi(t^\star,0)$ to provide a uniform upper bound for the Euclidean distance between the positions of any fault-free pair at $t^\star$. Then, using the contracting property of the map Φ, we will show that the maximum Euclidean distance between fault-free pairs is nonincreasing for $T_{(v^\star+1)\eta}<t<T_{(v^\star+1)\eta}-1$, for any $v^\star$.

1) Case 1 $t=t^\star$: For notational convenience, we denote by x_i^l the lth coordinate of x_i, and we define $q_{ij}(t):=[\Phi(t,0)]_{ij}$ as the (i,j)th entry of the matrix product $\Phi(t,0)$. We denote by α_l the absolute value of the largest lth component of the initial position of all robots

$$\alpha_l=\max_{m\in\overline{\mathcal{I}}}|x_m^l(0)|.$$

Consider the difference in lth coordinate of the positions of any two fault-free robots $i,\, j \in \overline{\mathcal{I}}$ at time $t^\star > 0$

$$
\begin{aligned}
|x_i^l(t^\star) - x_j^l(t^\star)| &= \left| \sum_{g=1}^{\overline{n}} (q_{ig}(t^\star) - q_{jg}(t^\star)) x_g^l(0) \right| \\
&\le \sum_{g=1}^{\overline{n}} \left| (q_{ig}(t^\star) - q_{jg}(t^\star)) x_g^l(0) \right| \quad (30)\\
&\le \sum_{g=1}^{\overline{n}} \left| (q_{ig}(t^\star) - q_{jg}(t^\star)) \right| \left| x_g^l(0) \right| \quad (31)\\
&\le \alpha_l \sum_{g=1}^{\overline{n}} |q_{ig}(t^\star) - q_{jg}(t^\star)| \quad (32)\\
&\le \alpha_l \overline{n}\, \delta \left(\prod_{t=0}^{t^\star} \mathbf{M}(t) \right) \quad (33)\\
&= \alpha_l \overline{n}\, \delta \left(\prod_{t=0}^{T_{(v^\star+1)\eta}-1} \mathbf{M}(t) \right) \quad (34)\\
&= \alpha_l \overline{n}\, \delta \left(\prod_{h=0}^{v^\star} \prod_{t=T_{h\eta}}^{T_{(h+1)\eta}-1} \mathbf{M}(t) \right) \quad (35)\\
&\le \alpha_l \overline{n} \prod_{h=0}^{v^\star} \tau_1 \left(\prod_{t=T_{h\eta}}^{T_{(h+1)\eta}-1} \mathbf{M}(t) \right) \quad (36)\\
&\le \alpha_l \overline{n} (1 - \gamma^{L_{\max}\eta})^{v^\star}. \quad (37)
\end{aligned}
$$

In the steps above, (30) follows from the triangle inequality, (31) follows from the Cauchy–Schwartz inequality, (32) uses the definition of α_l, (33) uses the definition of maximum range given by (43), (34) is obtained using $t^\star = T_{(v^\star+1)\eta} - 1$, (36) follows from Lemma A.1, (37) follows from (24) of lemma V.4.

Using (37), we can compute a bound on the distance between the positions of any two robots at time $t^\star$

$$
\begin{aligned}
\|x_i(t^\star) - x_j(t^\star)\|^2 &= \sum_{l=1}^{d} \left| x_i^l(t^\star) - x_j^l(t^\star) \right|^2 \quad (38)\\
&\le \sum_{l=1}^{d} \alpha_l^2 \overline{n}^2 (1 - \gamma^{L_{\max}\eta})^{2v^\star}. \quad (39)
\end{aligned}
$$

Setting the upper bound to ϵ, i.e., setting

$$\sum_{l=1}^{d} \alpha_l^2 \overline{n}^2 (1 - \gamma^{L_{\max}\eta})^{2v^\star} \le \epsilon^2$$

and solving for $v^\star$ yields

$$v^* \le \frac{\log \epsilon - \log \overline{n} - \frac{1}{2} \log \sum \alpha_l^2}{\log\left(1 - \gamma^{L_{\max}\eta}\right)}. \quad (40)$$

And we can obtain the actual time $t^\star$ at the switching step $v^\star$ using

$$t^\star = T_{(v^\star+1)\eta} - 1.$$

We note in (39) that as the switching time $v^\star \to \infty$, the left-hand side will tend to 0. We now consider the case when t is *not* a switching time.

2) Case 2: $T_{(v^\star+1)\eta} < t < T_{(v^\star+1)\eta} - 1$: For a given configuration $\overline{x}(t)$ of the fault-free nodes, we define the *diameter* of $\overline{x}(t)$ as

$$\mathrm{diam}(\overline{x}(t)) := \max_{i,j \in \overline{\mathcal{I}}} \|x_i(t) - x_j(t)\|. \quad (41)$$

For a switching time $t = t^\star$, (39) provides a uniform upper bound on the diameter of the positions of the fault-free nodes

$$\mathrm{diam}(\overline{x}(t^\star)) \le \overline{n}(1 - \gamma^{L_{\max}\eta})^{v^\star} \left(\sum_{l=1}^{d} \alpha_l^2 \right)^{\frac{1}{2}}.$$

For $t^\star = T_{(v^\star+1)\eta} < t < T_{(v^\star+1)\eta} - 1$, if the fault-free robots execute ADRC, we can apply (29) to obtain

$$\mathrm{conv}(\overline{x}(t)) = \mathrm{conv}(\Phi(t,\, t^\star)\overline{x}(t^\star)) \subset \mathrm{conv}(\overline{x}(t^\star))$$

and this implies

$$\mathrm{diam}(\overline{x}(t)) \le \mathrm{diam}(\overline{x}(t^\star)).$$

Thus, the uniform upper bound obtained for Euclidean distance between all fault-free pairs at switching time $t^\star$ is also a valid for all $t \ge t^\star$. However, it is not known whether for some arbitrary pair $i,\, j \in \overline{\mathcal{I}}$, $\|x_i(t) - x_j(t)\| \le \|x_i(t^\star) - x_j(t^\star)\|$ will hold for $t > t^\star$. Combining the results for Case 1 and Case 2 above, we have shown that for every $\epsilon > 0$, and for all pairs $i,\, j \in \overline{\mathcal{I}}$, there is $t^\star > 0$ such that $t > t^\star$ implies $\|x_i(t) - x_j(t)\| < \epsilon$. ■

Remark V.1: To this point, our proof only demonstrates ergodicity in the *weak* sense, which, on its own, does not imply that the positions of the fault-free robots will converge to a point that is stationary. However, it has been described in [53] that backward products of row-stochastic matrices have a nice property that is summarized in the following theorem.

Theorem V.2 (Chatterjee and Seneta [54]): For backward product of stochastic matrices, weak and strong ergodicity are equivalent.

Using Theorem V.2, we can deduce the following corollary, which implies convergence to a fixed point.

Corollary V.1 (Convergence of ADRC to a fixed point): Consider the assumptions and settings given by Theorem V.1. Then, there is $p \in \mathcal{X} \subseteq \mathbb{R}^d$ such that for all $i \in \overline{\mathcal{I}}$, $x_i(t) \to p$ as $t \to \infty$.

In other words, the strong ergodicity of the infinite product of row-stochastic system matrices for fault-free robots implies convergence to a fixed point.

D. Comments on the Weak Ergodicity of ADRC

In [51], Hajnal discusses various classes of matrices that show ergodic properties. Stochastic, Indecomposable, Aperiodic (SIA) matrices comprise the largest of these classes.

Roughly speaking, SIA matrices are *regular* matrices,[6] and an infinite product of SIA matrices tends to a matrix with identical rows. The set of *scrambling matrices*[7] is a subset of the SIA matrices, with the nice feature that multiplying any two scrambling matrices produces a matrix that is also scrambling. This is in contrast to SIA matrices: multiplying two SIA matrices does not necessarily produce an SIA matrix. The following relationships hold between matrix classes that are used in this paper:

$$\{\text{Positive matrices}\} \subset \{\text{Matrices with a positive column}\} \subset \{\text{Scrambling matrices}\} \subset \{\text{SIA}\}.$$

Recall that Proposition V.2 states that the infinite backward product of system matrices generated by ADRC can be expressed as an infinite backward product of matrices with positive columns. By the inclusion above, these are in fact scrambling matrices. Thus, the weak ergodicity of ADRC follows by the classical results contained in, e.g., [51] and [52] which asserts that *the infinite product of scrambling matrices is weakly ergodic*.

VI. Family of ADRC Algorithms

Like all distributed control algorithms, convergence of ADRC relies on maintaining appropriate connectivity conditions; in the case of ADRC, repeated reachability of the connectivity graphs of the fault-free nodes is a condition of Theorem V.1. In this section, we propose three versions of ADRC, each with its own strategy for maintaining adequate connectivity.

In the case of fault-free networks in which connectivity is determined by the sensing capabilities of each robot, it is often possible to enforce connectivity constraints by limiting the range of motion of each robot, based on the locations of its neighbors (e.g., [33] and [56]). The *circumcenter algorithm* [56], [57] is one such approach. Unfortunately, these approaches cannot be applied in cases when some of the robots are faulty; if a fault-free robot constrains its motion in order to enforce connectivity with a faulty neighbor, then the faulty neighbor has the possibility to impede, or even prevent, convergence of the fault-free nodes by its actions. For example, if connectivity is enforced, and if there is a faulty robot that moves far away, fault-free robots that are initially connected to the faulty robot may have no choice but to follow the faulty robot (in order to maintain connectivity), which may result in a *partitioned connectivity graph* or *divergence* of robot positions.

Rather than constraining the motions of the robots, we have opted to design a class of algorithms that employ variable-range sensing to maintain connectivity. This approach is motivated by work in wireless sensor networks [58], [59], where sensor nodes are capable of adjusting their sensing ranges to conserve energy. In particular, we assume that each fault-free robot can dynamically adjust its sensing range, resulting in a tradeoff between cost of energy for sensing and connectivity maintenance.

[6] A row-stochastic matrix is *regular* if it has a unit eigenvalue, i.e., the eigenvalue $\lambda = 1$ is simple. The powers of every regular matrix converge to a rank one row-stochastic matrix.

[7] A row-stochastic matrix is *scrambling* if and only if any two rows have at least one positive element in a coincident position.

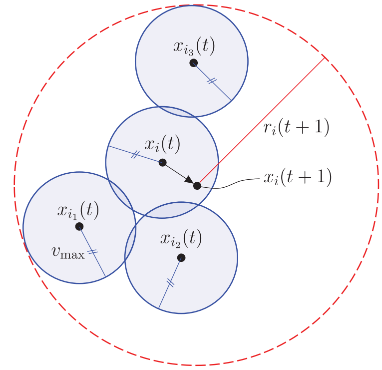


Fig. 5. Illustration of the procedure used by ADRC-II to adjust the sensing radius of ith robot at time t, where $\mathcal{N}_i(t) = \{i_1, i_2, i_3\}$. The areas enclosed by solid circles represent each robot's reachable set and the area enclosed by the dashed circle represents the sensing range of the robot.

Our three algorithms, ADRC-I, ADRC-II, and ADRC-III, are defined as follows.

1) [ADRC-I]—Each robot has a fixed sensing range, r_i. This is the typical case that is considered in distributed control algorithms for fault-free networks.
2) [ADRC-II]—At time $t+1$, the ith robot chooses its sensing range $r_i(t+1)$ so that $\mathcal{N}_i(t) \subseteq \mathcal{N}_i(t+1)$, i.e., the sensing range is chosen so that its set of neighbors is monotonically nondecreasing.
3) [ADRC-III]—At time $t+1$, the ith robot chooses its sensing range $r_i(t+1)$ so that it has $n_{\mathcal{N}}$ neighbors, where
$$n_{\mathcal{N}} \triangleq \max\{(n_{f_{\text{local, max}}} + 1)(d+1),\ n_{f_{\text{local, max}}} 2^d + 1\}$$
and $n_{f_{\text{local, max}}}$ is the maximum number of faults every robot is desired to tolerate.

Fig. 5 illustrates the variable sensing range in ADRC-II. For each time step t, the ith robot adjusts its sensing range based on the $v_{\max}$, the maximum displacement per stage which is uniform for all neighbors, and the current location of its neighbors. By following this procedure, the ith robot will never lose connectivity with its neighbors at the next time step $t+1$, even if some of its neighbors are malicious. Note that ADRC-III imposes a fixed value for the neighborhood size as a function of the maximum number of faulty neighbors, whereas the ADRC-II allows the number of neighbors to grow to $\overline{n}$ as the robots converge. As a result, the computational cost for ADRC-III may be significantly less than that for ADRC-II, particularly during the final stages of convergence (when the presence of faulty neighbors will have less impact on performance). Furthermore, since ADRC-II enforces continued connectivity with all of its initial neighbors, sensing cost can be made arbitrarily high by a malicious neighbor. For these reasons, if a reasonable estimate is available for the number of faulty neighbors, ADRC-III is a more attractive algorithm.

VII. Numerical Simulations

This section presents a suite of numerical simulation results to demonstrate the performance of our proposed algorithms. We will work with an F-MRS where the intercommunication

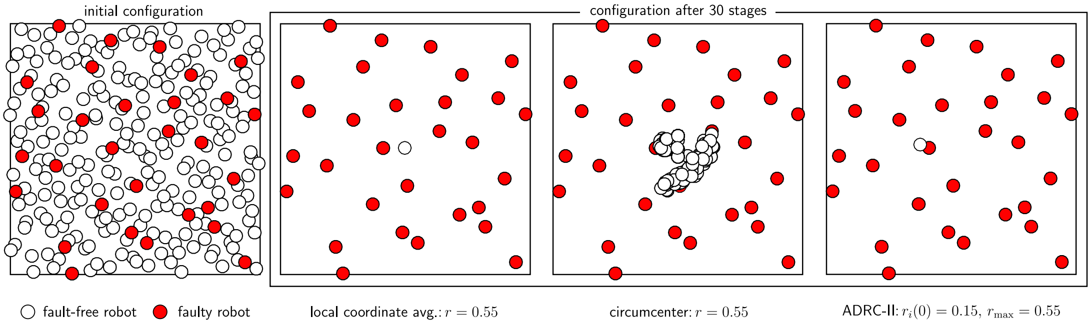
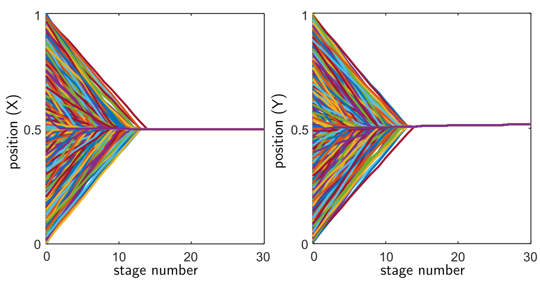
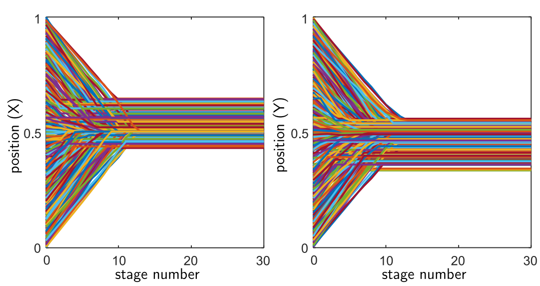


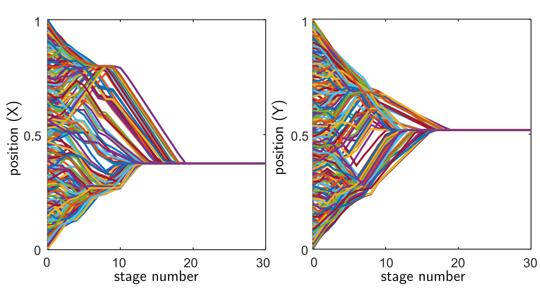
Fig. 6. Initial configuration and the configuration after 30 stages with stationary faults.



(a)

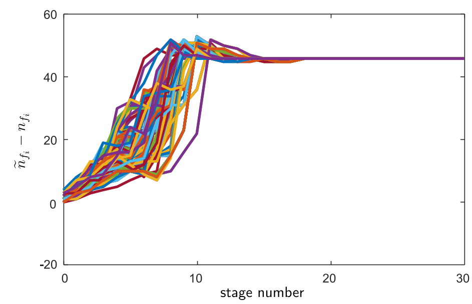


(b)

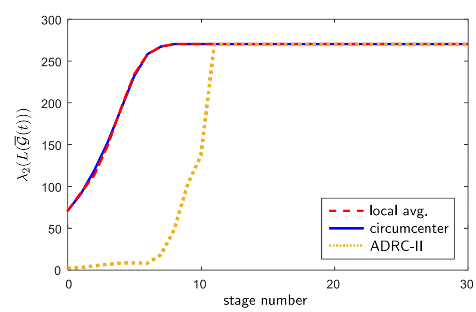


(c)

Fig. 7. Positions change of fault-free robots during 30 stages. (a) Local averaging, (b) circumcenter, and (c) ADRC-II.



(a)

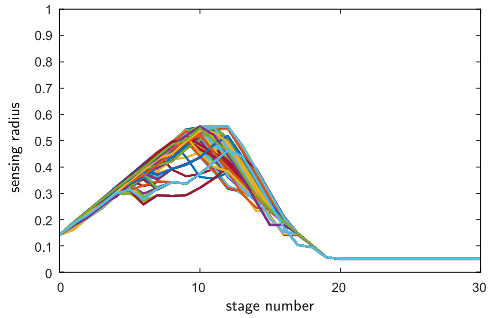


(b)

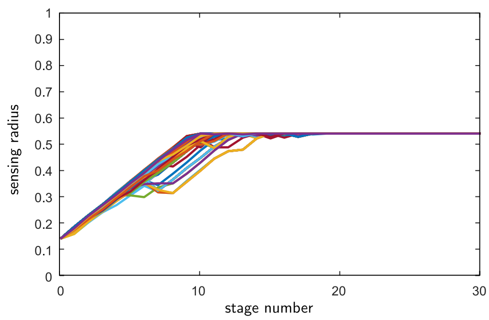
Fig. 8. Connectivity changes over the evolution. (a) $\widetilde{n}_{f_i} - n_{f_i}$ and (b) algebraic connectivity.

topology is characterized by the proximity of the robot positions. In order to provide a more fair comparison to other consensus methods (where the interconnection topologies are captured by disk graphs with radius $r_{\max}$), we have also imposed an upper bound, $r_{\max}$, on the maximum sensing range in our numerical simulations of ADRC-II and ADRC-III. Thus, the actual behavior of our algorithms (allowing for larger sensing radii) will be at least as good as the behavior reported below. Under this constraint, the interconnection topology of the robots at time t is defined by a disk graph $\mathcal{G}_{\text{disk}}(t) = (\mathcal{V}, \mathcal{E}(t))$, where $\mathcal{E}(t) \in \mathcal{V} \times \mathcal{V}$ and $(j, i) \in \mathcal{E}(t)$ if and only if $i, j \in \mathcal{V}$, $i \neq j$, and $\|x_j(t) - x_i(t)\| \leq \min\{r_i(t), r_{\max}\}$.

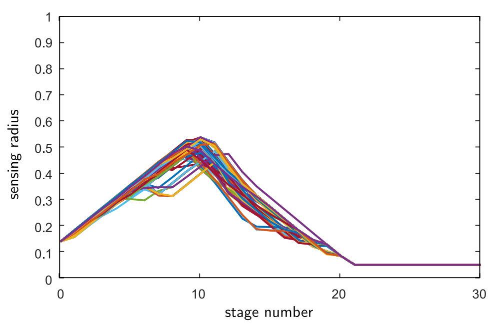
The workspace for our simulations is $\mathcal{X} = [0, 1] \times [0, 1] \in \mathbb{R}^2$, in which 300 robots are initially deployed in general position. Comparisons of rendezvous performance in the presence of faulty robots are made between three algorithms: a local coordinate averaging algorithm [60], the circumcenter algorithm [57], and ADRC (in its three versions with upper bounded sensing range). For local coordinate averaging and the circumcenter algorithm, the sensing radius is fixed and uniform with $r_i(t) = 0.55$. For fair comparison, we apply controllable sensing radius for our algorithm where the upper bound is $r_{\max} = 0.55$. The maximum displacement per stage is $v_{\max} = 0.05$, the convergence error bound is $\epsilon = 1 \times 10^{-7}$, and the number of faults is $n_f = 30$. For ADRC-III, we uniformly set the number of tolerable faulty neighbors for every



(a)



(b)



(c)

Fig. 9. Radii change during the evolution with ADRC-II, III with 0 and 30 stationary faults. (a) $n_f = 0$ (ADRC-II), (b) $n_f = 30$ (ADRC-II), and (c) $n_f = 30$ (ADRC-III).

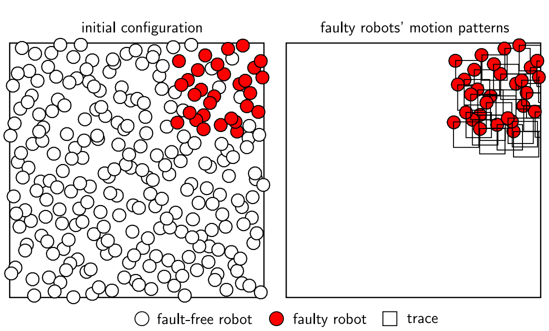
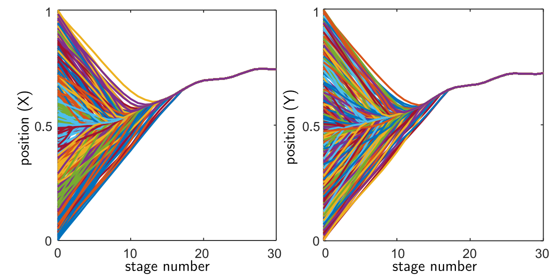


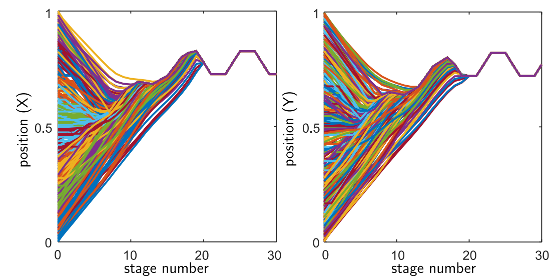
Fig. 10. Initial configuration and faulty robots' motion pattern.

fault-free robot as $n_{f_{\mathrm{local,max}}} = 40$ such that $n_{\mathcal{N}} = 161$, and $\widetilde{n}_{f_i} = n_{f_{\mathrm{local,max}}} = 40$ for all $i \in \overline{\mathcal{I}}$.

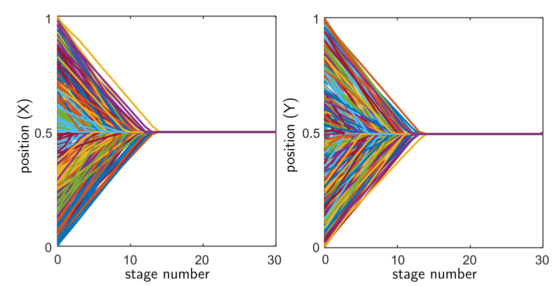
The first simulation considers the case of *stationary faults*. Fig. 6 shows the initial configuration, and configuration at stage 30 for the three algorithms. Fig. 7 shows position change over



(a)



(b)



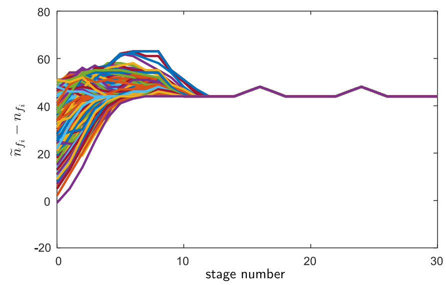
(c)

Fig. 11. Positions change of fault-free robots during 30 stages. (a) local averaging. (b) circumcenter. (c) ADRC-II.

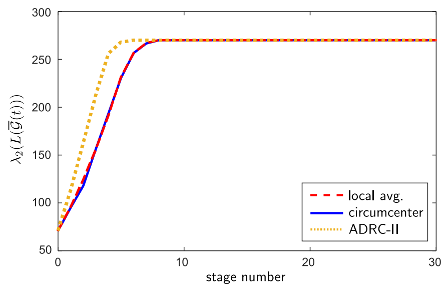
the evolutions of each of the three algorithms. As can be seen from the figures, the circumcenter law does not converge in the presence of stationary faults, while both local-averaging and ADRC-II do converge. The value $\widetilde{n}_{f_i} - n_{f_i}$ for ADRC-II is shown in Fig. 8(a), and algebraic connectivity[8] over 30 stages is shown in Fig. 8(b). The two plots in Fig. 8 show that the current example satisfies two connectivity conditions found in Theorem V.1. Fig. 9 shows $r_i(t)$ for the fault-free robots (a) for ADRC-II when there are no faults, (b) for ADRC-II when there are 30 faults, (c) for ADRC-III when there are 30 faults. Compared to Fig. 9(a) where there are no faults, Fig. 9(b) shows that overall $r_i(t)$ for fault-free robots do not decrease as fault-free robots positions converge to a point. On the other hand, Fig. 9(c) shows that if ADRC-III is used under the identical settings, the sensing radii converge. The example shows one advantage of ADRC-III over ADRC-II.

The second simulation results correspond to the case of *dynamic faults*. For this simulation, each faulty robot merely traces out a square pattern (each side of length $2 \times v_{\max}$). The initial configuration, and the faulty robots motion pattern is shown in

[8]It was described in [61] that if the algebraic connectivity of a digraph $\mathcal{G}$ is positive, i.e., $\lambda_2(L(\mathcal{G})) > 0$, then $\mathcal{G}$ has a globally reachable node.

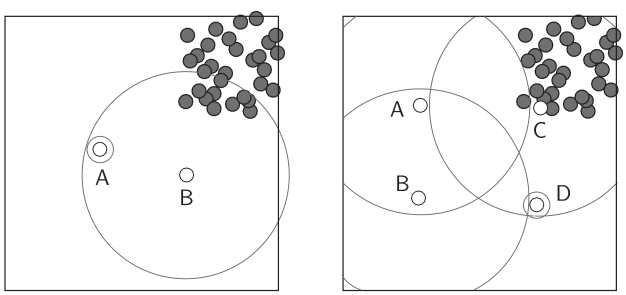


(a)



(b)

Fig. 12. Connectivity changes over the evolution. (a) $\widetilde{n}_{f_i} - n_{f_i}$ and (b) algebraic connectivity.



(a) (b)

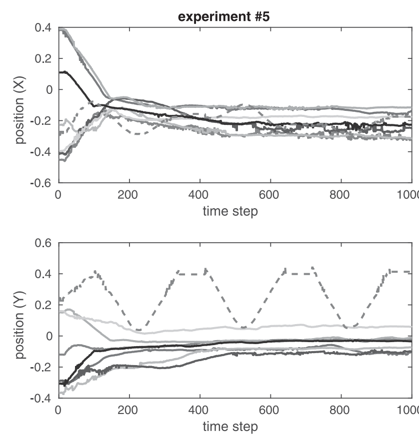
Fig. 13. Convergences to multiple points due to the neighborhood size limit. (a) $n_{f_{\mathrm{local,max}}} = 30$ and (b) $n_{f_{\mathrm{local,max}}} = 20$.

TABLE I
DETAILS OF THE POC IN FIG. 13(B)

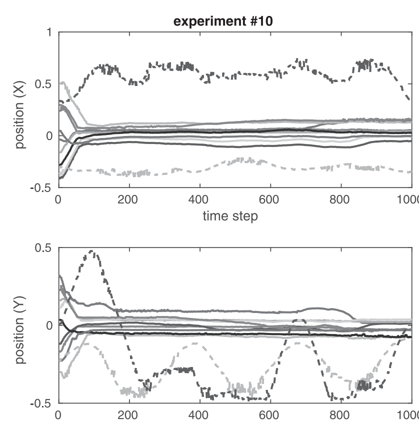
POC	Number of Nodes at the POC	For Each Node at the POC, the Number of Neighbors From				
		A	B	C	D	$\mathcal{F}$
A	81	80	1	0	0	0
B	70	12	69	0	0	0
C	29	0	0	28	25	28
D	90	0	0	0	81	0

(a) (b)

Fig. 14. Multirobot testbed. (a) Robotarium test-bed (b) GRITSBot. Photo courtesy of [62].



(a)

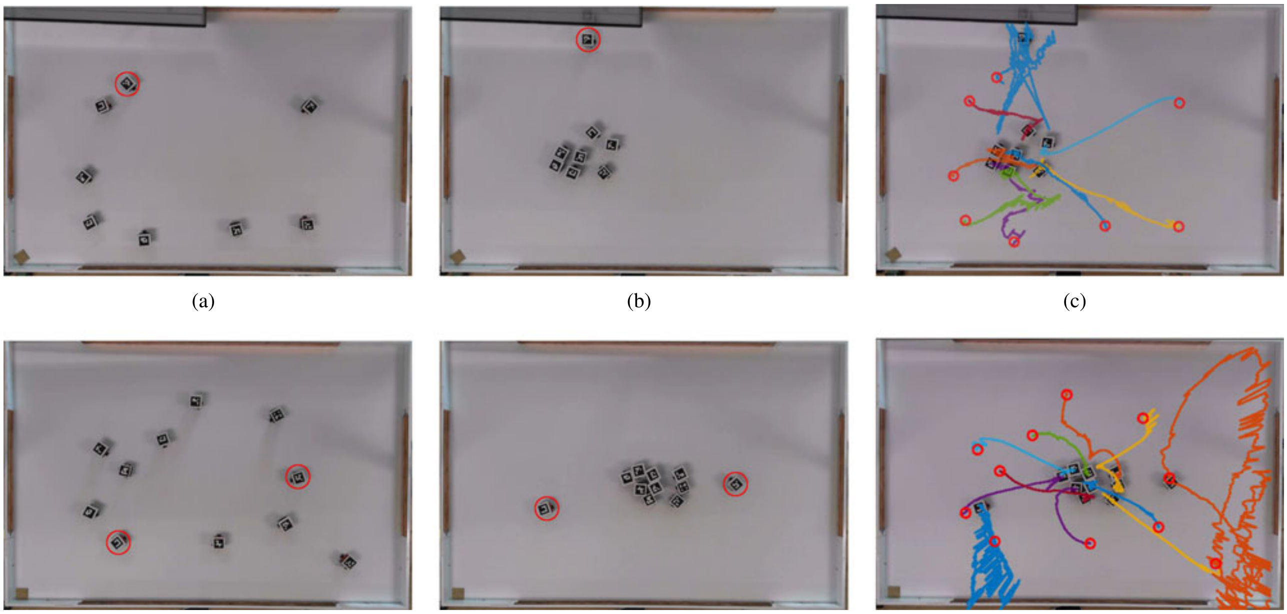


(b)

Fig. 15. Positions change of fault-free robots during 1000 iterations: (a) experiment #5, and (b) experiment #10 (dashed lines: positions of faulty robots, solid lines: positions of fault-free robots).

Figs. 10 and 11 shows positions change over the evolutions of the three algorithms. As can be seen, ADRC-II converges to a consensus, while both local averaging and the circumcenter algorithm fail to converge. We note that the point of the convergence will always lie in the convex hull of initial positions of fault-free robots; however, the exact location of the convergence point cannot be identified *a priori*. The value $\widetilde{n}_{f_i} - n_{f_i}$ for ADRC-II is shown in Fig. 12(a), and algebraic connectivity over 30 stages is shown in Fig. 12(b).

In our third and final simulation, we provide a few examples that depict shortcomings of ADRC-III due to limiting the neighborhood size by the value $n_{f_{\mathrm{local,max}}}$. Fig. 13 shows



(a) (b) (c)

(d) (e) (f)

Fig. 16. Initial configurations (left column), final configurations (middle column), trajectories of robots (right column) for experiment #5 (top row) and experiment #10 (bottom row). The circled robots shown in (a), (b), (d), and (e) are faulty.

the configuration after 30 stages when applying the value for $n_{f_{\text{local,max}}} = 30$ and 20, respectively (refer to the supplementary video attachment to see a complete set of simulations). The symbols A, B in Fig. 13(a) and A–D in Fig. 13(b) are the points of convergence (POC), and disks indicate the common sensing ranges of the points. In Fig. 13(a), the neighborhood size for every fault-free robot is set to 121. The number of robots at A[9] is 160. Since each robot at A is connected to 121 robots from A, it will not move because its safe point should be found near A. The number of robots at B is 110, and every robot at B is connected to 9 robots from A, 106 robots from B, and 6 faulty robots. Since each safe point for a robot at B is necessarily contained in the convex hull of fault-free neighbors' positions and the value $n_{f_{\text{local,max}}} = 30$ is greater than the number of robots connected to A plus the number of faulty robots (9 + 6 = 15), the safe point must be at B. Thus, all the robots at B will not move. Fig. 13(b) shows convergence to 4 groups of points after 30 stages, and the result can be analyzed in a similar manner. Refer to Table I for the details.

VIII. Robotarium Experiments

This section presents a series of Robotarium experiments to verify our fault-tolerant rendezvous algorithm. The Robotarium [62], [63] is a multirobot testbed developed at the Georgia Institute of Technology. The testbed consists of custom-designed robots that are called the GRITSBots [63]. An image of the Robotarium testbed is shown in Fig. 14(a), and that of the GRITSBot is shown in Fig. 14(b).

After our algorithm (provided as a script) is uploaded to each GRITSBot, robots are initially deployed at a randomly chosen configuration. Then, each robot is commanded to execute the script for a specified number of discrete time steps, which in our case is set to 1000. Total ten experiments are carried out to test our theoretical results on the real multirobot platform. The experiments from #1 to #5 are performed with eight robots from which one of them are faulty and the experiments from #6 to #10 are performed with 11 robots from which two of them are faulty. Those faulty robots are randomly sampled and simply wander around the workspace periodically based on some random nonlinear sinusoidal function. Due to the small workspace size and the limited availability of the number of robots, we assumed complete graph for the interconnection topology throughout the experiments.

For the sake of space, only a part of the experimental results are presented. Fig. 15 shows positions change of all robots during the 1000 iterations from experiments #5 and #10. The solid lines show the change in the positions of fault-free robots, and the dashed lines show the change in the positions of faulty robots. As can be seen from the figures, after 1000 iterations, all the fault-free robots successfully gathered together at a fixed location, regardless of the actions of faulty robots. In particular, Fig. 16 shows a few webcam snapshot images of both initial and final configurations and the motions of all robots obtained from the two experiments: #5 and #10. Refer to the video attachment to see the recordings of actual experiments.

IX. Conclusion

This paper proposed a computationally efficient, decentralized, fault-tolerant algorithm for rendezvous of a group of robots with limited sensing. We provided the convergence analysis of the proposed algorithm by borrowing several tools form ergodic theory, matrix theory, and graph theory. A suite of simulation

[9] In this context, a robot is at A, if it is sufficiently close to A.

and experimental results is provided to illustrate the theoretical results.

Our study in this paper is a first step toward the design of practical yet fully capable fault-tolerant rendezvous algorithms. Throughout this paper, we have only focused on higher level path planning of fault-free robots. It will be the topic of future research to extend our results to include issues such as dealing with uncertainties and partial observability, long-term planning and exploration, navigating and mapping in complex environments, etc. Other future work will involve relaxing the dimensionless robots assumption, e.g., considering fat robot models [64] with collision avoidance, and proposing tighter connectivity constraints for each robot by taking into account the cooperative actions between fault-free robots.

Appendix A
Few Results From Matrix theory

The coefficient of ergodicity [51], [52] provides a measure of the degree of *ergodicity* of row-stochastic matrices. Given a row-stochastic matrix $\mathbf{P} \in \mathbb{R}^{n\times n}$, the coefficient of ergodicity $\tau_1(\mathbf{P})$ is defined by

$$\tau_1(\mathbf{P}) = 1 - \min_{i,j} \sum_h \min\left([\mathbf{P}]_{ih}, [\mathbf{P}]_{jh}\right). \tag{42}$$

The following proposition is immediate from this definition.

Proposition A.1: Consider a row-stochastic matrix $\mathbf{P}$. If $\mathbf{P}$ has at least one column, all of whose elements are nonzero and lower bounded by $\alpha > 0$, then $\tau_1(\mathbf{P}) \le 1 - \alpha$.

Given any square matrix $\mathbf{P} \in \mathbb{R}^{n\times n}$, the *maximum range of* $\mathbf{P}$, denoted $\delta(\mathbf{P})$, [52] is defined as

$$\delta(\mathbf{P}) = \max_j \max_{i_1,i_2} \left|[\mathbf{P}]_{i_1 j} - [\mathbf{P}]_{i_2 j}\right|. \tag{43}$$

The value $\delta(\mathbf{P})$ is the maximum difference between any pair of elements in the same column, and it provides an upper bound for the difference in the rows among all columns.[10] The relationship between the coefficient of ergodicity $\tau_1(\mathbf{P})$ and the maximum range $\delta(\mathbf{P})$ of a matrix $\mathbf{P}$ is given in the following lemma.

Lemma A.1 (Hajnal [51] and Wolfowitz [52]): For row-stochastic matrices $\mathbf{S}(0), \ldots, \mathbf{S}(t)$, with $\mathbf{S}(l) \in \mathbb{R}^{n\times n}$

$$\delta\left(\prod_{l=0}^{t} \mathbf{S}(l)\right) \le \prod_{l=0}^{t} \tau_1\left(\mathbf{S}(l)\right)$$

holds for $t = 0, 1, 2, \ldots$.

Appendix B
Lemma B.1, Proposition B.1, and Lemma B.2

Lemma B.1: Consider X, a set of n points in $\mathbb{R}^d$ that is $r+1$ divisible with $r \ge 0$. For each choice of subset $Q \subset X$ with size $n - r$, every Tverberg point of depth $r+1$ lies in the convex hull of Q.

Proof: Consider $q \in \mathbb{R}^d$ to be a Tverberg point of depth $r+1$ of X. By the definition of the Tverberg point, there is a partition Π such that q lies in the intersection of convex hulls of $r+1$ disjoint subsets of X that partitions X. Consider $\Pi := \{P_1, \ldots, P_{r+1}.\}$ to be an $r+1$ disjoint subsets of X that partitions X whose intersection of the convex hulls is nonempty. Then, $P_i \subset X$, $|P_i| \ge 1$ for $i = 1, \ldots, r+1$, $\bigcup_{i=1}^{r+1} P_i = X$, and $P_i \cap P_j = \emptyset$ for all pairs $i, j \in \{1, \ldots, r+1\}$ with $i \ne j$. If q is a Tverberg point of depth $r+1$ of X

$$q \in \bigcap_{i=1}^{r+1} \mathrm{conv}(P_i). \tag{44}$$

Consider choosing a subset S of X with size r. We denote the complement set of S to be Q such that Q is given by $Q = X \setminus S$, and the size of Q is exactly $n - r$. Regardless of the choice of the set S, there is at least one set among $P_1, \ldots, P_{r+1}$ which does not contain any elements from S. For every choice of $S \subset X$, there is a $P_j \in \Pi$ such that $P_j \subset Q$ and $q \in \mathbf{conv}(P_j)$ [by (44)]. Thus, for each choice of $Q \subset X$, $q \in \mathrm{conv}(Q)$. In other words, for each choice of subset $Q \subset X$ with size $n - r$, every Tverberg point of depth $r+1$ of X lies in the convex hull of Q. This completes the proof. ∎

Proposition B.1: Given a point set x in $\mathbb{R}^d$, any point $p \in \mathrm{ri}(\mathrm{conv}(x))$ can be written as nonzero convex combination of all points in $\mathrm{ver}(\mathrm{conv}(x))$ whenever $\mathrm{ri}(\mathrm{conv}(x)) \ne \emptyset$.

Proof: Let $m \in \mathrm{conv}(x)$ denote the coordinate average of the vertices of $\mathrm{conv}(x)$. For each $p \in \mathrm{ri}(\mathrm{conv}(x))$ there exists some $\epsilon > 0$ such that $q = (1+\epsilon)p - \epsilon m \in \mathrm{ri}(\mathrm{conv}(x))$. Writing p in terms of q and m

$$p = \frac{1}{1+\epsilon} q + \frac{\epsilon}{1+\epsilon} m.$$

Since $q \in \mathrm{ri}(\mathrm{conv}(x))$, it can also be written as a convex combination of some subset of the vertices of $\mathrm{ver}(\mathrm{conv}(x))$, and since m is a nonzero convex combination of all vertices of $\mathrm{ri}(\mathrm{conv}(x))$, p can be represented by nonzero convex combination of all vertices, i.e., $\mathrm{ver}(\mathrm{conv}(x))$, as claimed. ∎

The proof of Lemma B.2 depends on the following proposition.

Proposition B.2: Let $\mathbf{A}_i, \mathbf{B}_i \in \mathbb{R}^{n\times n}$ be nonnegative matrices for $i = 1, \ldots, m$. If for each i, there is $\tau_i > 0$ such that $\mathbf{A}_i \ge \mathbf{B}_i \ge \tau_i \mathbf{I}_n$ for $i = 1, \ldots, m$

$$\mathbf{A}_m \mathbf{A}_{m-1} \cdots \mathbf{A}_1 \ge \mathbf{B}_m \mathbf{B}_{m-1} \cdots \mathbf{B}_1 \ge \prod_{i=1}^{m} \tau_i \mathbf{I}_n.$$

Proof: For the sake of brevity, we give only a brief sketch of the proof. The proof is by induction on m. The key observation required for the proof is the following: $\mathbf{A}_i \ge \mathbf{B}_i$ implies $\mathbf{A}_i = \mathbf{B}_i + \mathbf{B}_i'$ where $\mathbf{B}_i' \ge 0$, and since each $\mathbf{B}_i$ has positive diagonal, there is $\mathbf{B}_i'' \ge 0$ such that

$$\mathbf{B}_i = \mathbf{B}_i'' + \tau_i \mathbf{I}_n$$

with $\tau_i > 0$. The remainder of the proof proceeds by merely applying this observation and carrying out matrix algebra to arrive to the desired result. ∎

Lemma B.2: Consider a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with n vertices where $\mathcal{A}$ is the adjacency matrix for $\mathcal{G}$. If the ith node

[10] Because of its close relationship $\tau_1(\mathbf{P})$, the value $\delta(\mathbf{P})$ is also sometimes called a *coefficient of ergoditicy*, e.g., [52].

is globally reachable, then

$$\mathbf{1}_{n\times 1} \leq [(\mathbf{I}_n + \mathcal{A})^l]_i \tag{45}$$

whenever $l \geq n-1$.

Proof: We will prove (45) by showing that

$$1 \leq [(\mathbf{I}_n + \mathcal{A})^l]_{ji}$$

holds for all $j \in \mathcal{V}$ whenever $l \geq n-1$.

Case 1: Consider $j = i$. By Proposition B.2

$$\mathbf{I}_n = \mathbf{I}_n^m \leq (\mathbf{I}_n + \mathcal{A})^m, \ \ m = 1, 2, \ldots,$$

and this implies that $1 \leq [(\mathbf{I}_n + \mathcal{A})^m]_{ii}$ for all $i \in \mathcal{V}$, and $m \in \mathbb{N}$.

Case 2: Consider $j \in \mathcal{V} \setminus \{i\}$. Since the graph $\mathcal{G}$ has globally reachable node i, for every $j \in \mathcal{V} \setminus \{i\}$ there is a directed acyclic path from node j to i which consists of vertices $j, l_1, l_2, \ldots, l_{k-1}, i$, where $l_0 = j$ and $k < n-1$. Thus, $(j, l_1), (l_1, l_2), \ldots (l_{k-2}, l_{k-1}), (l_{k-1}, i) \in \mathcal{E}$, and this implies that $[\mathcal{A}]_{jl_1} = 1$, $[\mathcal{A}]_{l_{k-1}i} = 1$, and $[\mathcal{A}]_{l_m, l_{m+1}} = 1$ for all $m = 1, \ldots, k-2$. Hence, for each j there exists $k \leq n-1$ such that the product $[\mathcal{A}]_{jl_1}[\mathcal{A}]_{l_1 l_2} \cdots [\mathcal{A}]_{l_{k-2}l_{k-1}}[\mathcal{A}]_{l_{k-1}i} = 1$, and this implies

$$1 \leq [\mathcal{A}^k]_{ji}. \tag{46}$$

By Proposition B.2

$$\mathcal{A}^m \leq (\mathbf{I}_n + \mathcal{A})^m$$

for all $m \in \mathbb{N}$ such that for any pair $i, j \in \mathcal{V}$

$$[\mathcal{A}^m]_{ji} \leq [(\mathbf{I}_n + \mathcal{A})^m]_{ji}. \tag{47}$$

Combining (46) and (47) leads to

$$1 \leq [(\mathbf{I}_n + \mathcal{A})^k]_{ji}$$

and again by Proposition B.2

$$(\mathbf{I}_n + \mathcal{A})^k \leq (\mathbf{I}_n + \mathcal{A})^m$$

holds for all $m \geq k$. Thus

$$1 \leq [(\mathbf{I}_n + \mathcal{A})^k]_{ji} \leq [(\mathbf{I}_n + \mathcal{A})^m]_{ji}$$

for all $m \geq k$. Note that the uniform upper bound for k is $n-1$. Hence, as long as $m \geq n-1$

$$1 \leq [(\mathbf{I}_n + \mathcal{A})^m]_{ji}$$

holds for all $j \in \mathcal{V} \setminus \{i\}$.

Combining two results for the cases $j = i$ and $j \in \mathcal{V} \setminus \{i\}$, whenever $l \geq n-1$,

$$\mathbf{1}_{n\times 1} \leq \left[(\mathbf{I}_n + \mathcal{A})^l\right]_i$$

and the proof is complete. ■

ACKNOWLEDGMENT

The authors would like to thank the Editors and Anonymous Reviewers for their feedbacks, which has greatly improved the quality of this paper. In addition, the authors would also like to thank Dr. M. Egerstedt and Dr. D. Pickem for giving them an opportunity to test their algorithm on the Robotarium testbed.

REFERENCES

[1] H. Park and S. Hutchinson, "Worst-case performance of rendezvous networks in the presence of adversarial nodes," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 5579–5585.

[2] H. Park and S. Hutchinson, "Worst-case performance of a mobile sensor network under individual sensor failure," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2013, pp. 895–900.

[3] A. Jadbabaie, J. Lin, and A. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. Autom. Control*, vol. 48, no. 6, pp. 988–1001, Nov. 2003.

[4] J. N. Tsitsiklis *et al.*,"Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *IEEE Trans. Autom. Control*, vol. AC-31, no. 9, pp. 803–812, Sep. 1986.

[5] N. H. Vaidya, "Iterative Byzantine vector consensus in incomplete graphs," in *Distributed Computing and Networking*. New York, NY, USA: Springer, 2014, pp. 14–28.

[6] H. Mendes and M. Herlihy, "Multidimensional approximate agreement in Byzantine asynchronous systems," in *Proc. 45th Annu. ACM Symp. Theory Comput.*, 2013, pp. 391–400.

[7] A. Aviziens, "Fault-tolerant systems," *IEEE Trans. Comput.*, vol. C-25, no. 12, pp. 1304–1312, Dec. 1976.

[8] L. Lamport, R. Shostak, and M. Pease, "The Byzantine generals problem," *ACM Trans. Program. Lang. Syst.*, vol. 4, no. 3, pp. 382–401, 1982.

[9] N. H. Vaidya, L. Tseng, and G. Liang, "Iterative approximate Byzantine consensus in arbitrary directed graphs," in *Proc. ACM Symp. Princ. Distrib. Comput.*, 2012, pp. 365–374.

[10] 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.

[11] D. Dolev, N. A. Lynch, S. S. Pinter, E. W. Stark, and W. E. Weihl, "Reaching approximate agreement in the presence of faults," *J. ACM*, vol. 33, no. 3, p. 499–516, 1986.

[12] A. Doudou and A. Schiper, "Muteness detectors for consensus with Byzantine processes," in *Proc. 17th Annu. ACM Symp. Princ. Distrib. Comput.*, PODC 98. ACM, 1998, p. 315.

[13] M. Zhu and S. Martínez, "On distributed constrained formation control in operator-vehicle adversarial networks," *Automatica*, vol. 49, no. 12, pp. 3571–3582, 2013.

[14] M. Zhu and S. Martinez, "On resilient consensus against replay attacks in operator-vehicle networks," in *Proc. IEEE Am. Control Conf.*, 2012, pp. 3553–3558.

[15] H. Zhang and S. Sundaram, "A simple median-based resilient consensus algorithm," in *Proc. IEEE 50th Annu. Allerton Conf. Commun., Control, Comput.*, 2012, pp. 1734–1741.

[16] A. Khanafer, B. Touri, and T. Basar, "Robust distributed averaging on networks with adversarial intervention," in *Proc. IEEE 52nd Annu. Conf. Decis. Control*, 2013, pp. 7131–7136.

[17] S. Sundaram, S. Revzen, and G. Pappas, "A control-theoretic approach to disseminating values and overcoming malicious links in wireless networks," *Automatica*, vol. 48, no. 11, pp. 2894–2901, 2012.

[18] A. Gusrialdi, Z. Qu, and M. A. Simaan, "Robust design of cooperative systems against attacks," in *Proc. IEEE Am. Control Conf.*, 2014, pp. 1456–1462.

[19] G. Parlangeli, "A fault compensation strategy for consensus networks subject to transient and intermittent faults," in *Proc. IEEE 21st Mediterranean Conf. Control Autom.*, 2013, pp. 46–53.

[20] F. Pasqualetti, A. Bicchi, and F. Bullo, "Consensus computation in unreliable networks: A system theoretic approach," *IEEE Trans. Autom. Control*, vol. 57, no. 1, pp. 90–104, Jan. 2012.

[21] S. Sundaram and C. N. Hadjicostis, "Distributed function calculation via linear iterative strategies in the presence of malicious agents," *IEEE Trans. Autom. Control*, vol. 56, no. 7, pp. 1495–1508, Jul. 2011.

[22] S. Sundaram and C. N. Hadjicostis, "Distributed function calculation via linear iterations in the presence of malicious agents—Part II: Overcoming malicious behavior," in *Proc. IEEE Am. Control Conf.*, 2008, pp. 1356–1361.

[23] S. Sundaram and C. N. Hadjicostis, "Distributed function calculation via linear iterations in the presence of malicious agents—Part I: Attacking the network," in *Proc. IEEE Am. Control Conf.*, 2008, pp. 1350–1355.

[24] W. Zeng, M.-Y. Chow, and P. Ning, "Secure distributed control in unreliable D-NCS," in *Proc. IEEE Int. Symp. Ind. Electron.*, 2012, pp. 1858–1863.

[25] D. Silvestre, P. Rosa, J. P. Hespanha, and C. Silvestre, "Finite-time average consensus in a Byzantine environment using set-valued observers," in *Proc. Am. Control Conf.*, 2014, pp. 3023–3028.

[26] N. Agmon and D. Peleg, "Fault-tolerant gathering algorithms for autonomous mobile robots," *SIAM J. Comput.*, vol. 36, no. 1, pp. 56–82, 2006.

[27] X. Défago, M. Gradinariu, S. Messika, and P. Raipin-Parvédy, *Fault-tolerant and Self-stabilizing Mobile Robots Gathering*. New York, NY, USA: Springer, 2006.

[28] Z. Bouzid, S. Das, and S. Tixeuil, "Gathering of mobile robots tolerating multiple crash faults," in *Proc. IEEE 33rd Int. Conf. Distrib. Comput. Syst.*, 2013, pp. 337–346.

[29] Z. Bouzid, M. G. Potop-Butucaru, and S. Tixeuil, "Optimal Byzantine-resilient convergence in uni-dimensional robot networks," *Theor. Comput. Sci.*, vol. 411, no. 34, pp. 3154–3168, 2010.

[30] Z. Bouzid, M. G. Potop-Butucaru, and S. Tixeuil, "Byzantine-resilient convergence in oblivious robot networks," in *Distributed Computing and Networking*. New York, NY, USA: Springer, pp. 275–280, 2009.

[31] Z. Bouzid, M. G. Potop-Butucaru, and S. Tixeuil, "Optimal Byzantine resilient convergence in asynchronous robots networks," in *Stabilization, Safety, and Security of Distributed Systems*. New York, NY, USA: Springer, pp. 165–179, 2009.

[32] C. Auger, Z. Bouzid, P. Courtieu, S. Tixeuil, and X. Urbain, "Certified impossibility results for Byzantine-tolerant mobile robots, " in, *Stabilization, Safety, and Security of Distributed Systems*. New York, NY, USA: Springer, pp. 178–190, 2013.

[33] J. Cortés, S. Martínez, and F. Bullo, "Robust rendezvous for mobile autonomous agents via proximity graphs in arbitrary dimensions," *IEEE Trans. Autom. Control*, vol. 51, no. 8, pp. 1289–1298, Aug. 2006.

[34] M. Franceschelli, A. Giua, and A. Pisano, "Finite-time consensus on the median value by discontinuous control," in *Proc. IEEE Am. Control Conf.*, 2014, pp. 946–951.

[35] H. J. LeBlanc, H. Zhang, S. Sundaram, and X. Koutsoukos, "Consensus of multi-agent networks in the presence of adversaries using only local information," in *Proc. 1st Int. Conf. High Confidence Networked Syst.*, 2012, pp. 1–10.

[36] G. Bracha, "An asynchronous [(n−1)/3]-resilient consensus protocol," in *Proc. 3rd Annu. ACM Symp. Princ. Distrib. Comput.*, 1984, pp. 154–162.

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

[38] A. Weber and C. J. Friedrich, *Alfred Weber's Theory of the Location of Industries*. Chicago, IL, USA: Univ. Chicago Press, 1929.

[39] H. Park and S. Hutchinson, "An efficient algorithm for fault-tolerant rendezvous of multi-robot systems with controllable sensing range," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2016, pp. 358–365.

[40] H. Park and S. Hutchinson, "A distributed robust convergence algorithm for multi-robot systems in the presence of faulty robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2015, pp. 2980–2985.

[41] P. Flocchini, G. Prencipe, N. Santoro, and P. Widmayer, "Gathering of asynchronous robots with limited visibility," *Theor. Comput. Sci.*, vol. 337, no. 1, pp. 147–168, 2005.

[42] H. Tverberg, "A generalization of Radon's theorem," *J. London Math. Soc*, vol. 41, no. 1, pp. 123–128, 1966.

[43] J. R. Reay *et al.*, "An extension of Radon's theorem," *Illinois J. Math.*, vol. 12, no. 2, pp. 184–189, 1968.

[44] J.-P. Roudneff, "Partitions of points into intersecting tetrahedra," *Discrete Math.*, vol. 81, no. 1, pp. 81–86, 1990.

[45] J.-P. Roudneff, "New cases of reays conjecture on partitions of points into simplices with k-dimensional intersection," *Eur. J. Combinatorics*, vol. 30, no. 8, pp. 1919–1943, 2009.

[46] B. J. Birch, "On 3 n points in a plane," in *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 55, no. 04. Cambridge, U.K.: Cambridge Univ. Press, pp. 289–293, 1959.

[47] W. Mulzer and D. Werner, "Approximating Tverberg points in linear time for any fixed dimension," *Discrete Comput. Geom.*, vol. 50, no. 2, pp. 520–535, 2013.

[48] J. O'Rourke and J. E. Goodman, *Handbook of Discrete and Computational Geometry*. Boca Raton, FL, USA: CRC Press, 2004.

[49] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu, *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*. Hoboken, NJ, USA: Wiley, 2009, vol. 501.

[50] M. Cao, A. S. Morse, and B. D. Anderson, "Agreeing asynchronously," *IEEE Trans. Autom. Control*, vol. 53, no. 8, pp. 1826–1838, Sep. 2008.

[51] J. Hajnal, "Weak ergodicity in non-homogeneous Markov chains," in *Proc. Cambridge Philos. Soc.*, vol. 54, no. 2, pp. 233–246, 1958.

[52] J. Wolfowitz, "Products of indecomposable, aperiodic, stochastic matrices," *Proc. Am. Math. Soc.*, vol. 14, no. 5, pp. 733–737, 1963.

[53] E. Seneta, "Coefficients of ergodicity: Structure and applications," *Adv. Appl. Probab.*, vol. 11, no. 3, pp. 576–590, 1979.

[54] S. Chatterjee and E. Seneta, "Towards consensus: Some convergence theorems on repeated averaging," *J. Appl. Probab.*, vol. 14, pp. 89–97, 1977.

[55] J. Hajnal and M. Bartlett, "The ergodic properties of non-homogeneous finite Markov chains," in *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 52, no. 01. Cambridge, U.K.: Cambridge Univ. Press, 1956, pp. 67–77.

[56] J. Lin, A. Morse, and B. Anderson, "The multi-agent rendezvous problem," in *Proc. 42nd IEEE Conf. Decis. Control*, vol. 2, 2003, pp. 1508–1513.

[57] H. Ando, Y. Oasa, I. Suzuki, and M. Yamashita, "Distributed memoryless point convergence algorithm for mobile robots with limited visibility," *IEEE Trans. Robot. Autom.*, vol. 15, no. 5, pp. 818–828, Oct. 1999.

[58] D. Tian and N. D. Georganas, "A coverage-preserving node scheduling scheme for large wireless sensor networks," in *Proc. 1st ACM Int. Workshop Wireless Sensor Netw. Appl.*, 2002, pp. 32–41.

[59] M. Cardei, J. Wu, M. Lu, and M. O. Pervaiz, "Maximum network lifetime in wireless sensor networks with adjustable sensing ranges," in *Proc. IEEE Int. Conf. Wireless. Mobile Comput.*, 2005, vol. 3, pp. 438–445.

[60] R. Olfati-Saber and R. Murray, "Consensus protocols for networks of dynamic agents," in *Proc. 2003 Am. Control Conf.*, 2003, pp. 951–956.

[61] C. W. Wu, "Algebraic connectivity of directed graphs," *Linear Multilinear Algebr.*, vol. 53, no. 3, pp. 203–223, 2005.

[62] D. Pickem *et al.*, "Safe, remote-access swarm robotics research on the robotarium," arXiv:1604.00640, 2016.

[63] D. Pickem, M. Lee, and M. Egerstedt, "The gritsbot in its natural habitat-a multi-robot testbed," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2015, pp. 4062–4067.

[64] J. Czyzowicz, L. Gasieniec, and A. Pelc, "Gathering few fat mobile robots in the plane," *Theor. Comput. Sci.*, vol. 410, no. 6, pp. 481–499, 2009.

Hyongju Park (M'16) received the B.S. and M.S. degrees in mechanical engineering from Korea University, Seoul, South Korea, in 2007 and 2009, respectively, and the Ph.D. degree in mechanical engineering from University of Illinois at Urbana-Champaign, Champaign, IL, USA, in 2016.

He is a Postdoctoral Researcher in Robotics and Optimization for Analysis of Human Motion Laboratory, University of Michigan, Ann Arbor, MI, USA. His research interests include path planning of multi-robot systems, fault-tolerant control, and coordinated control of multivehicles.

Seth A. Hutchinson (F'07) received the Ph.D. degree in electrical engineering from Purdue University, Lafayette, IN, USA, in 1988.

In 1990, he joined the faculty of University of Illinois in Urbana-Champaign, Champaign, IL, USA, where he is currently a Professor in the Department of Electrical and Computer Engineering, the Coordinated Science Laboratory, and the Beckman Institute for Advanced Science and Technology. He served as the Associate Department Head of ECE from 2001 to 2007. He has published more than 200 papers on the topics of robotics and computer vision, and is the coauthor of the books *Principles of Robot Motion: Theory, Algorithms, and Implementations* (MIT Press, Cambridge, MA, USA, 2005) and *Robot Modeling and Control* (Wiley, Hoboken, NJ, USA).

Dr. Hutchinson currently serves on the editorial board of *International Journal of Robotics Research* and chairs the steering committee of IEEE ROBOTICS AND AUTOMATION LETTERS. He was the Founding Editor-in-Chief of the IEEE Robotics and Automation Society's Conference Editorial Board from 2006 to 2008 and the Editor-in-Chief of IEEE TRANSACTION ON ROBOTICS from 2008 to 2013.