2013 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS)
November 3-7, 2013. Tokyo, Japan

Worst-Case Performance of Rendezvous Networks in the Presence of Adversarial Nodes

Hyongju Park and Seth Hutchinson

Abstract— **In this paper, we consider the performance of distributed control algorithms for networked robotic systems when one or more robots fail to execute the optimal policy. In particular, we investigate the performance of the *circumcenter algorithm with connectivity maintenance* [1]–[3] when one or more adversarial agents act maliciously to maximally disrupt convergence of the remaining, cooperative agents. To this end, we formulate a performance objective for each adversary node in terms of the circumradii of its cooperative neighbors in a communication graph which does not require omniscience of adversaries as is often assumed in the literature (e.g., [4], [5]). We provide an optimization algorithm based on finite-horizon dynamic programming, and obtain solutions through numerical simulation. Our results show that in general adversarial nodes are able not only to impede convergence toward consensus, but can also affect global changes in the topology of the communication graph for the cooperative agents.**

I. INTRODUCTION

There has been much research in the past regarding the *consensus* problem (also known as the *opinion* problem). In a multi-agent system, consensus has various application areas in terms of *information consensus*. A number of linear algorithms for information consensus have been developed in [6], [7], [8], [9]. In these studies, analysis of convergence, performance, and stability with linear protocol were performed in synchronous/asynchronous networks with switching topology, and communication delays.

There were also studies concerning consensus of multi-vehicle networks. In particular, [1], [2], [3] study the case when agents' *physical positions* are considered as consensus variables. Ando et al. [1] first introduced an algorithm with which agents locally converge to the center of the smallest circle containing the set of locally connected agents, and also proposed conditions to preserve the network's connectivity between pairwise agents. In [2], Lin, and Moore first described this type of problem as *multi-agent Rendezvous problem*, and applied their algorithm in both synchronous and asynchronous network. In their work in [3], [10], Cortés, et al. formalized those ideas from [1], [2], and showed that with their circumcenter algorithm with connectivity constraint, rendezvous can be achieved in finite time. In this paper, we extensively use the terminology defined in [3].

Also there have been many studies regarding the consensus problem in which adversarial nodes are present. Especially in computer networking community, this type of problem is better known as the *Byzantine Generals problem*. The original problem and and its variations can be found in [4], [11], [12] to name a few. In recent years, there were also attempts to develop linear algorithms based on [6], [7] which are *resilient* to various types of malicious activity. Gupta et al. in his paper [13], introduced general possible failure modes for the malicious agents depending on how they update their states. Similar to their terms, we define three types of adversary as follows. *Type-A* adversary simply does not communicate to other nodes, which is similar to jamming problem in networks. *Type-B* adversary uses a constant rule in updating its state. *Type-C* adversary updates its state in an arbitrary manner. In [10], Cortés, et al. considered link failures (which can be modeled as *Type-A* adversaries) between pairwise agents, and showed that even with the occurrence of link failures, rendezvous can be achieved under special topological assumptions. Jadbabaie et al. in their work [14], showed that a network that evolves with linear protocol in [6] converges to a consensus whenever there is a *Type-B* adversary. In their work, an adversary which they call the *leader* moves with constant speed, and fixed heading. In his paper [5], [15], Leblanc developed a robust linear consensus protocol named as ARC-P. ARC-P is resilient to *Type-C* adversaries whose numbers are upper bounded. Their main idea is to give zero weight to nodes that have the most deviated state values from others. In [16], Sundaram, and Hadjicostis also considered *Type-C* adversary, and studied the relation between resilience of linear consensus network and its topological property.

In this paper, we consider the worst-case performance of a multi-agent rendezvous network where part of the nodes act maliciously. Especially we put our attention to a special case when all the agents execute the circumcenter algorithm with connectivity maintenance from [1], [2], [3]. The problem is a maximization problem where the reward function is given as the minimum circumradius of the local network. Here, the term *local* implies that we consider only the circumcircles of set of non-adversarial agents which are within the communication (i.e., sensing) range of adversaries. The adversaries in our paper are *Type-C*, which have the potential to actively disrupt the performance of the network. Both our previous work in [17], and this paper utilize finite-horizon dynamic programming as a solution method to obtain the worst-case behavior of the distributed multi-vehicle networks pursuing optimal sensor deployment, and rendezvous respectively.

The outline of this paper is as follows. First, we take a brief review on terms and notable properties of average-

H. Park is in the Department of Mechanical and Science Engineering at the University of Illinois. park334@illinois.edu

S. Hutchinson is a professor of Electrical and Computer Engineering at the University of Illinois. seth@illinois.edu

This material is based in part upon work supported by AFOSR (award AF FA9550-12-1-0193) and the National Science Foundation (award CNS 0931871).

978-1-4673-6357-0/13/$31.00 ©2013 IEEE

consensus. Then, we introduce the circumcenter algorithm with connectivity maintenance from [1], [2], [3] and show that the algorithm converges to consensus using the properties of proper convex-hull averaging maps. Next, we model our system with adversarial agents, and formulate our optimization problem, which aims at maximally disrupting the performance of cooperative-agent-only network. Then we briefly investigate the physical, and topological constraints for adversarial nodes, and present our solution method i.e, a finite-horizon dynamic programming. Lastly, we show numerical simulation results, and conclude our paper with a few remarks.

II. Preliminaries on average-consensus

A. Consensus in general

We consider a finite number of n agents, indexed $1, \dots, n$. Let I be a set of n indices such that $I = \{1, \dots, n\}$. Then our dynamic system with each agent $i \in I$ is a tuple (X, X_0, U, f_i), where $X \subseteq \mathbb{R}^d$ is a continuous state space where d is a dimension of the space, $X_0 \subseteq \mathbb{R}^d$ is an initial state space, $U \subseteq \mathbb{R}^d$ is an input space, and f_i is a time independent discrete-time evolution map $f_i : X \times U \to X$. Each state vector is denoted as $x_i \in X$ where $i \in I$, and the set of all state vectors is denoted as $x = (x_1, \dots, x_n) \in X^n$. In similar manner, $f : X^n \times U^n \to X^n$ is the evolution map of all n agents such that $f = (f_1, \dots, f_n)$. Hence, the discrete-time dynamics for all n agents are given by,

$$x(l+1) = f(x(l), u(l)) \tag{1}$$

where l is symbol for discrete-time step such that $l \in \mathbb{N} \cup \{0\}$. In this paper, for each agent $i \in I$, we consider the following discrete version of distributed-integrator dynamics.

$$x_i(l+1) = x_i(l) + u_i(l) \tag{2}$$

Now we are ready to state the definition of consensus.

Definition 1: Given with a dynamical system defined as above, set of state vectors $(x_1(l), \dots, x_n(l)) \in X^n$ is in consensus at time l if,

$$x_1(l) = x_2(l) = \cdots = x_n(l). \tag{3}$$

The state space X was also named as *opinion space*, and denoted by S in [18], [19].

B. Average consensus on one-dimensional state space

There are various types of consensus problem most generally can be put as $\mathcal{X}-$consensus problem. The symbol $\mathcal{X}$ is defined to be a consensus value which is a function of state vectors. (e.g. $\mathcal{X}(x) = \sum_i^n x_i$ is average-consensus, and $\mathcal{X}(x) = \max_i x_i$ is maximum consensus.) The average-consensus is a well-studied area especially in computer science [8], [12], [20]. Typically, graph theory along with matrix theory has been used as a tool to study the convergence properties, and connectivity of the network in [7], [9], [8]. The aforementioned papers are mainly concerned with averaging-consensus algorithms, which are linear mappings each defined by a matrix. Those algorithms are designed to be used in a one-dimensional state space. Therefore for the case $d > 1$, a more suitable mapping uses the concept of *convex hull*, which is reviewed in the next section. In our problem, we only deal with the state space in $\mathbb{R}^2$. Hence, unless otherwise noted $d = 2$.

C. Convex-hull averaging map

Recall that the weighted average of n state vectors is;

$$\sum_{i=1}^{n} w_i x_i \tag{4}$$

where $\sum_i w_i = 1$, and $w_i \geq 0$ for each $i \in I$. Especially when $d > 1$, (4) is more often called a *convex combination* of n vectors $(x_1, \dots, x_n)$. Recall that the convex hull of a set $x = (x_1, \dots, x_n)$, $\text{conv}(x)$, is the smallest convex set which contains x, which coincides with the set of all convex combinations of vectors x. Now we are ready to see a new term convex-hull averaging map which is defined as follows.

Definition 2: [18] Suppose that there are n agents $(x_1, \dots, x_n)$ each in the state space $X \in \mathbb{R}^d$ for $d \geq 2$, then $f : X^n \times U^n \to X^n$ is a *convex hull averaging map*[1], if all of its component functions $\{f_i\}_1^n$ are convex combinations of $(x_1, \dots, x_n)$.

The necessary condition for $f_i(x_i, u_i)$ to be a convex combination of $(x_1, \dots, x_n)$ under (2) is for each $i \in I$,

$$u_i(x) \in \text{conv}(x) - x_i \tag{5}$$

Also, according to the definition 2, the following relation holds.

$$\text{conv}(f(x, u)) \subseteq \text{conv}^2 \tag{6}$$

we say that the convex-hull averaging map f is *proper*, if inclusion in (6) is strict everywhere else but when x is consensus i.e. (3). The following theorem shows the convergence property of the convex-hull averaging map.

Theorem 1: [18] Let $f : X^n \times U^n \to X^n$ be a *continuous*, and *proper* convex-hull averaging map, with X being a *compact* state space. Suppose that agents evolve with discrete-time dynamics as (1). Given that initial state $x(0) \in X_0$ for each $i \in I$,

$$\lim_{l \to \infty} x_i(l) = \gamma\left(x(0)\right) \tag{7}$$

where $\gamma\left(x(0)\right) \in X$ is the *consensus value*.

For the interested reader, more details of this theorem and its variations are contained in [18], [21] along with proofs. This is a general result because it states that given any convex averaging map, consensus is always guaranteed.

D. Circumcenter-law with connectivity constraints

In this paper, we are particularly concerned with networks in which communication between nodes is constrained by separation distance. Such constraints define a connectivity graph $\mathcal{G}_{\text{disk}}(r) = (V, E)$ in which elements of V correspond to nodes, and an edge exists between nodes i and j if

[1] In [21], Krause used a term *compromise map* instead.

[2] Given a set S and a vector x, the difference of the set with the vector: $S' = S - x$ is defined by $S' = \{s - x \mid s \in S\}$.

$\text{dist}(x_i, x_j) \leq r$, where $\text{dist}(\cdot,\cdot)$ is a Euclidean distance, and r is the communication range (see, e.g., [3], [22]). The neighborhood index set of node i defined as

$$N_i = \{j \in I \mid j \neq i, \text{dist}(x_i, x_j) \leq r\}$$

and X_i denotes the set of positions of all nodes that are within range of node i, including node i

$$X_i = \{x_j \mid \text{dist}(x_i, x_j) \leq r\}.$$

In the absence of adversarial nodes, [1] presents a control strategy for consenus that relies on geometric properties of the neighborhoods of $\mathcal{G}_{\text{disk}}$. For each neighborhood X_i, we define a *circumcenter*, denoted as CC_i, to be the *center* of the smallest circle that contains X_i, and *circumradius*, denoted as CR_i to be the *radius* of the smallest circle that contains X_i:

$$\text{CR}_i = \min_{q \in X} \max_{x_j \in X_i} \|q - x_j\| \tag{8}$$

and equivalently,

$$\text{CC}_i = \arg\min_{q \in X} \max_{x_j \in X_i} \|q - x_j\|. \tag{9}$$

Loosely speaking, the idea of the circumcenter law with connectivity maintenance in [1]–[3] is that at each time step, each agent $i \in I$ moves toward the circumcenter CC_i, while maintaining connectivity with its neighbors by staying within a connectivity constraint set $\mathcal{X}_i$. The set $\mathcal{X}_i$ is the intersection of the pairwise constraints $\mathcal{X}_{i,j}$ between node i and each of its neighbors:

$$\mathcal{X}_{i,j} = \overline{B}\left(\frac{x_i + x_j}{2}, \frac{r}{2}\right) \tag{10}$$

and

$$\mathcal{X}_i = \bigcap_{j \in N_i} \mathcal{X}_{i,j} \tag{11}$$

in which $\overline{B}(a, b)$ denotes *closed* ball with radius b centered at a. It is easy to show that if agent i and j are neighbors at time l, then as long as $u_i(l) \in \mathcal{X}_{i,j} - x_i(l)$, and $u_j(l) \in \mathcal{X}_{i,j} - x_j(l)$, then i and j remain neighbors at time $l + 1$.

Let CC_i^* be the point in the boundary of $\mathcal{X}_i$ along the line between x_i and CC_i (i.e., movement by node i to CC_i^* is the maximal motion toward the circumcenter CC_i that remains within $\mathcal{X}_i$). A circumcenter control law that maintains connectivity constraints is given by

$$u_i(l) = k_i\left(\text{CC}_i^* - x_i(l)\right) \tag{12}$$

in which k_i is a time-invariant proportional control gain for agent i, and $k_i \in (0\ 1)$.

The following lemma describes the convergence properties of the evolution map $f : X^n \times U^n \to X^n$ when circumcenter law with connectivity constraint is applied to all agents.

Lemma 1: The discrete dynamic system given as (1), and (2) which starts from initial position $x(0) \in X_0^n$, with control defined in (12) converges to a consensus.

A proof of this lemma is given in Appendix I.

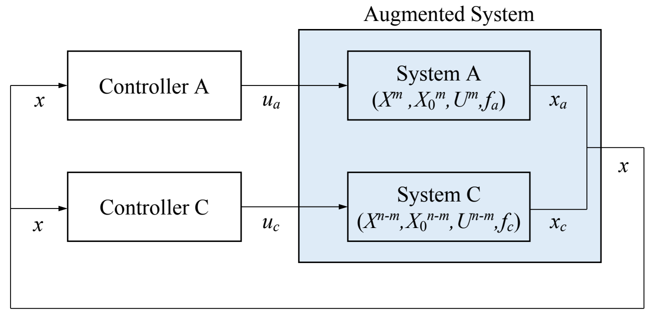


Fig. 1: Block diagram of our system.

III. Problem Statement

In this paper, we consider the problem in which m of the n nodes in the network act as adversaries. The remaining nodes follow the control law in (2), and are called *cooperative nodes*. We denote the index set for the adversarial nodes as $I_a = \{1, \ldots, m\}$, and the index set for the cooperative nodes as $I_c = \{m+1, \ldots, n\}$. Thus, the overall network can be partitioned into adversarial and cooperative subsystems, as shown in Figure 1. For the adversary nodes, we denote by $x_a = \{x_i\}_{i \in I_a}$, $u_a = \{u_i\}_{i \in I_a}$, and $f_a = \{f_i\}_{i \in I_a}$ the set of positions, controls, and evolution maps, respectively, and x_c, u_c, and f_c are defined analogously for the cooperative nodes.

In this paper, we assume that the cooperative nodes are unaware of the presence (and identities) of the adversarial nodes. Thus, the cooperative nodes execute control strategy u_i, defined by (12), for each $i \in I_c$. Note that when computing the circumcenter CC_i, all neighboring nodes, including adversary nodes, are considered (since the cooperative nodes are not aware of which nodes might be adversarial).

The adversary nodes have as their goal to maximally interfere with the achievement of consensus (or rendezvous) by the cooperative agents. There are various ways that one could formalize this objective; here we opt to define the objective of an adversary node as maximizing the size of the smallest circumcirle for those cooperative nodes in its neighborhood. To formalize this objective, let $\overline{X}_i = \{x_j \mid j \in N_i \setminus I_a\}$ denote the set of positions of cooperative nodes that are neighbors of node i. Then the circumcenter and circumradius of this set are given by

$$\overline{\text{CR}}_i = \min_{q \in X} \max_{x_j \in \overline{X}_i} \|q - x_j\| \tag{13}$$

and

$$\overline{\text{CC}}_i = \arg\min_{q \in X} \max_{x_j \in \overline{X}_i} \|q - x_j\| \tag{14}$$

The objective for adversary node $k \in I_a$ at position x_k is then given by

$$\mathcal{L}(x_k) = \min_{i \in N_k} \overline{\text{CR}}_i \tag{15}$$

and the goal of the adversary is to maximize $\mathcal{L}$ by its choice of control. Note that this is a local objective function, since each adversary node considers only those cooperative nodes

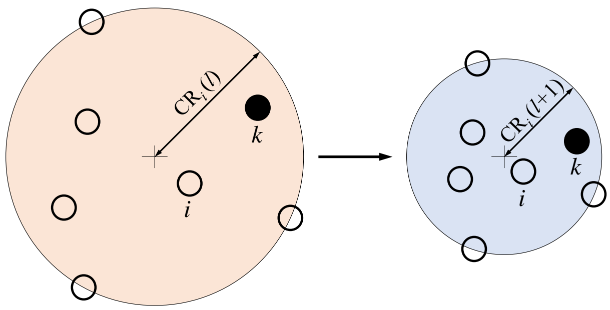


Fig. 2: An example showing ineffective adversarial node's control. The solid circle is the adversary, and blank circles are cooperative agents (left: time l, right: time $l+1$).

that lie within its communication radius.

In the following, we make the following assumptions regarding the network.

- All agents use a same clock (i.e., synchronous network).
- Initially, the communication network defined with $\mathcal{G}_{\text{disk}}(r)$ is *strongly connected*.
- The system is memory-less, and time-invariant.
- Adversaries have twice the sensing rage of cooperative agents (this assumption can easily be relaxed or modified).

IV. Optimal Adversarial Strategies

As the first step in this section, we define a constraint set to confine adversarial node's motion. First, for each time step l, each adversarial agent's input is bounded as follows,

$$u_i(l) \in \overline{B}\left(0, u_{\max}\right), \;\; i \in I_a, \, u_i(l) \in U \tag{16}$$

where $u_{\max}$ is a positive real number which is related to the node's mobile specifications.

A. Connectivity of adversarial nodes with other nodes

In this section, we investigate a more general version of connectivity constraint which can be used for adversarial nodes in special cases. For instance, suppose that there is a cooperative node which needs to be monitored by adversarial node all the time, then it is desirable that the cooperative node be connected all the time. We state our result as follows.

Lemma 2: Suppose that $j \in I_a$, $i \in I_c$ and $i \in N_j$ at time l. Then, $\text{dist}(x_j(l), x_i(l)) \leq r$ implies $\text{dist}(x_j(l+1), x_i(l+1)) \leq r$ if and only if,

$$u_j(l) \in u_i(l) + \overline{B}\Big(x_i(l) - x_j(l), \text{dist}(x_i(l), x_j(l))\Big) \tag{17}$$

Since the proof of the lemma is simple and algebraic, it will be omitted from this paper. As you can see from (17), not only the state information is required, but also $u_i(l)$ with $i \in N_j$ (i.e., neighbor agents' control inputs) is necessary to obtain the largest constraint set. In other words, only with the redundancy of information we can obtain more *relaxed* version of connectivity constraint set than (10).

B. Effectiveness of adversarial node's control

In this section, we specify the condition under which adversarial node's control is *effective* or *ineffective*. Given an adversarial node indexed with $k \in I_a$ with some cooperative agent $i \in N_k$, if adversary agent's control at time l satisfies,

$$u_k(l) \in \overline{B}\Big(\overline{\text{CC}}_i(l+1), \overline{\text{CR}}_i(l+1)\Big) - x_k(l) \tag{18}$$

then the position of adversary $x_k(l)$ does not have any affect in the motion of other agents because at time $l+1$ for the same i,

$$\text{CR}_i = \overline{\text{CR}}_i \tag{19}$$

Fig. 2 shows an example of this graphically.

C. A finite-horizon forward Dynamic Programming

Similar to our work in [17], we are interested in obtaining the solution to $\max_{x_k} \mathcal{L}(x_k)$ over a finite horizon using *forward* dynamic programming [23]. With a slight abuse of notation, let $\max_{x_k} \mathcal{L}(x_k) = \mathcal{L}^*(x_k)$. The strongest point of dynamic programming approach is that it provides both necessary and sufficient conditions for optimality. To make our approach tractable, we discretize the input space U into a grid, and confine the motion of adversary to move over it. Suppose that we have total N stages. Then given a time horizon $l = 0, \ldots, N-1$, and an adversarial node with index $k \in I_a$, we expect to solve the following recursive equations.

$$\tilde{J}_N(x(0)) = 0, \tag{20}$$

$$\begin{aligned}\tilde{J}_l(x(N-l)) = \max_{u_k(N-l-1) \in U} \Big[& g_{N-l-1}\big(x(N-l-1)\big) \\ & + \tilde{J}_{l+1}\Big(\tilde{f}\big(x(N-l), u(N-l-1)\big)\Big)\Big]\end{aligned} \tag{21}$$

where $\tilde{f}$ is a backward state transition[3] given as,

$$x(l-1) = \tilde{f}\big(x(l), u(l-1)\big) \tag{22}$$

and the current reward at time l is given by

$$g_l\big(x(l)\big) = \mathcal{L}\big(x_k(l)\big) \tag{23}$$

The total reward over N stage is,

$$\tilde{J}_0\big(x(N)\big) = \max_{u_k(0), \ldots, u_k(N-1)} \left(\sum_{l=0}^{N-1} g_l\big(x(l)\big) \right) \tag{24}$$

We denote by $\pi^* = \{u_k^*(0), \ldots, u_k^*(N-1)\}$ which solves (24) to be the optimal policy of adversarial agent k over N stages.

D. Stopping condition

Our DP algorithm in (20)-(21) terminates at time N when either one of the following conditions is satisfied.

- $u_a^*(N) = 0$, i.e., $u_i = 0$ for all $i \in I_a$.
- $x_{m+1}(N) = \cdots = x_n(N)$.

[3]The term 'backward' was used simply because with $\tilde{f}$, state proceeds backward in time.

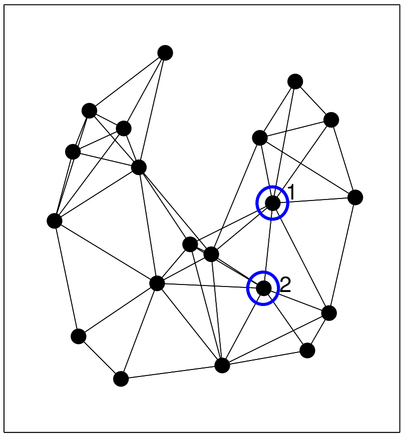
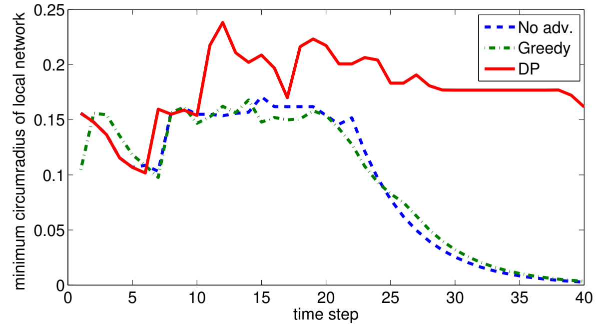


Fig. 3: A disk graph $\mathcal{G}_{\text{disk}}(r)$ showing initial positions of agents and their connectivity. (circled nodes are adversaries)

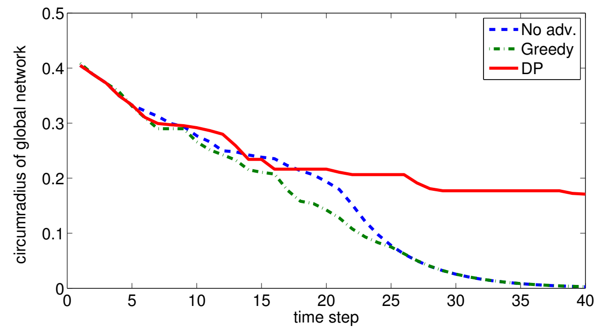
The 1st condition relies on a heuristic that if $u_a^*(N) = 0$ is the best control for adversarial agents at time N, it will remain the best in subsequent time, too. The second condition is to simply avoid trivial cases in this problem in which all cooperative agents are in consensus except for the adversaries.

V. Simulation Results

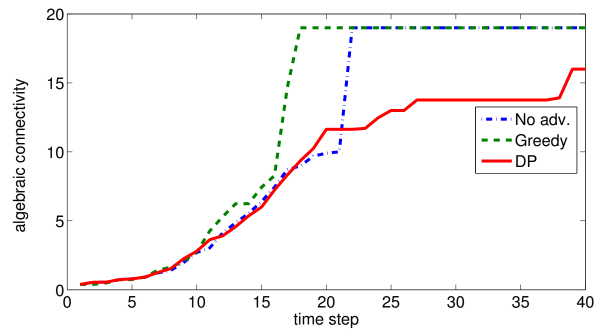
In our simulation, we chose our workspace X to be a unit square $[0\ 1]^2$. Fig 3. shows the initial configuration. There are total 20 nodes where one of them is the adversary ($n = 20$, and $m = 1$). The unfilled circle in Fig. 3 is where the adversarial node (index:1) is placed. The maximum communication range was set to $r = 0.3$, and control gain was uniformly applied to all cooperative agents: $k_2 = \cdots = k_{20} = 0.2$. Fig. 4 shows the evolution of all agents over 40 stages. Solid circles are the final positions, unfilled circles are initial positions, solid lines are traces of cooperative nodes, and thicker lines are traces of adversarial node. The adversarial node moves with 8-neighbor-rule over $80{\times}80$ equally spaced grid on X. Fig. 5 shows comparison of cooperative agents' network properties between 3 distinctive cases: when there is *no adversary* in the network, when there is a single adversary, if *greedy method* is used, and if *DP method* is used to obtain optimal policy π^* is used over 40 stages. In Fig. 5-(a), the vertical axis shows the minimum circumradius of local network with optimal policy π^* denoted by $\mathcal{L}^*(x_k)$. In Fig. 5-(b), the vertical axis shows the circumradii of *all* cooperative agents i.e., $\text{CR}(x_c)$. In both Fig. 5-(a), and Fig. 5-(b), solution from DP method yield the most favorable results in both local, and global performance metric i.e., circumradius value. Furthermore in Fig. 5-(c), the *algebraic connectivity* was compared between 3 cases. Algebraic connectivity is defined as the 2nd smallest eigenvalue of the Laplacian matrix L which is related to the worst-case speed of convergence of the network [7]. When using DP algorithm, the algebraic connectivity was overall the lowest. In sum, results from Fig. 5 implies that



(a) $\mathcal{L}^*(x_k)$ vs time



(b) $\text{CR}(x_c)$ vs time



(c) algebraic connectivity vs time

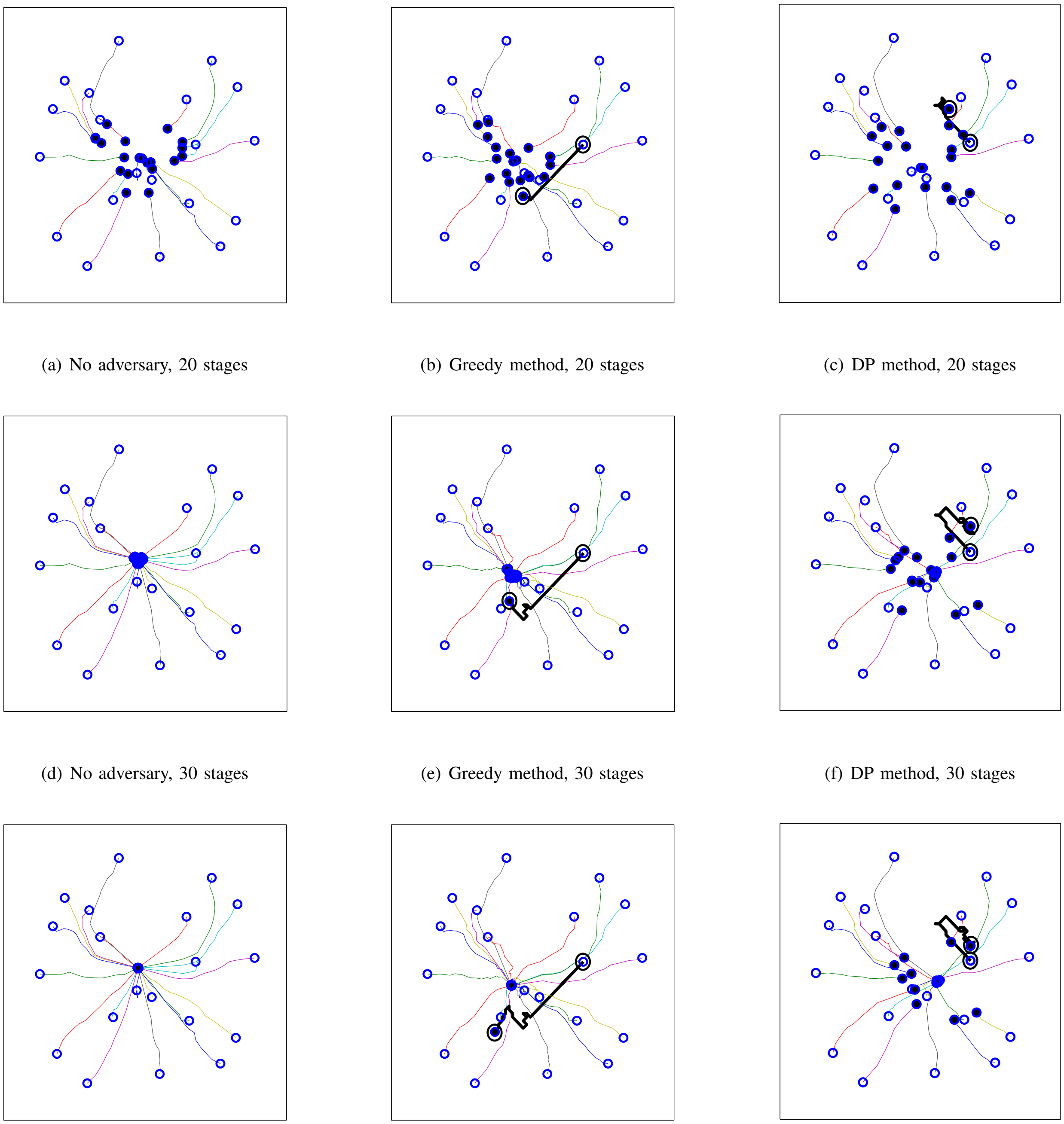
Fig. 5: Comparison of network properties: circumcenter algorithm with no adversary, adversary with greedy algorithm, and DP algorithm as a solution method.

adversary's motion impedes the rendezvous of cooperative agents. The fact becomes obvious in Fig. 4 by comparing 3rd column with 1st, and 2nd column. Fig. 6 shows results when there are 2 adversaries. We considered the worst-case maximization strategy by choosing current reward at each time step as $\min_{k\in I_a} \mathcal{L}^*(x_k)$. Fig. 6-(a) shows evolution of agents using approximate DP algorithm (4-step lookahead)[4]. In Fig. 6-(b), current costs at each stage are compared between different algorithms

VI. Conclusion and Future Works

In this study, we formulate our performance objective in terms of circumradii of 1-hop neighborhood of an adversary

[4]More than 4-step lookahead methods were not considered here due to the heavy computational load.



(a) No adversary, 20 stages

(b) Greedy method, 20 stages

(c) DP method, 20 stages

(d) No adversary, 30 stages

(e) Greedy method, 30 stages

(f) DP method, 30 stages

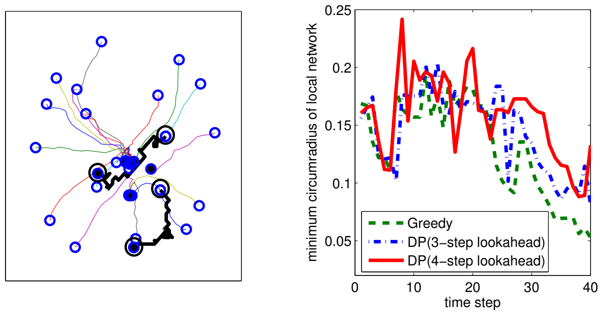
(g) No adversary, 40 stages

(h) Greedy method, 40 stages

(i) DP method, 40 stages

Fig. 4: Evolution of agents over 20, 30, and 40 time steps. Each column shows results when there is no adversary (a),(d),(g), when there is a single adversary, if greedy method was applied (b),(e),(h), and if DP method was applied (c),(f),(i) respectively. Solid circles are the final positions, and unfilled circles are initial positions. In 2nd and 3rd columns of the figure, double circles are the adversarial nodes position, and the thinker lines are their traces.

node, and obtain the local strategy of each adversary node by using DP method which is optimal. It is shown in our simulation result that even a single node failure can alter the convergence property of multi-agent rendezvous networks. We propose a few possible future directions. First, when we solve $\max_{x_k} \mathcal{L}(x_k)$, we consider only the 2-hop neighborhood of adversary which implies that our strategy is local. As you can see from Fig. 4 and 5, when a greedy method is used, the solution falls into local maxima which are trivial solutions, no better than the case when there is no adversary at all. The result suggests that we consider larger number of agents e.g., 3-hop neighborhood. Next, we



(a) Evolution of agents (approx. DP) (b) $\min_{k \in I_a} \mathcal{L}^*(x_k)$ vs time

Fig. 6: Simulation results with two adversaries.

assumed in this paper that adversaries fully share information between each other, and they cooperate to find the worst-case optimal policy. In the future, it will be interesting to relax this assumption such that communications between adversaries are local, and each adversary makes its own decision in independent manner. Furthermore, studying the convergence property of adversary-present networks, and time to reach consensus are other interesting subjects.

Appendix I
Proof of Lemma 1

Lemma 1: The discrete dynamic system given as (1) and (2) which starts from initial position $x(0) \in X_0^n$, with control defined in (12) converges to a consensus.

Proof: In the proof, we show that f is a continuous, and proper convex-hull averaging map, and use Theorem 1 to show that it converges to a consensus. To show the continuity of f, it is sufficient to show that each component function f_i is continuous on X. In [3] it was shown that 'fti' is a continuous map which implies that CC_i^* is also continuous on X. From (2) and (12), it can be shown that each f_i is continuous. According to [3], $\mathrm{CC}_i \in \mathrm{conv}(X_i) \setminus \mathrm{Ver}(X_i)$ where X_i is set of positions for agent i, and its neighbor, and $\mathrm{Ver}(X_i)$ is a set of vertices of $\mathrm{conv}(X_i)$. This implies that $\mathrm{CC}_i \in \mathrm{conv}(X_i)$. By the property of convex set, an open line segment connecting two points in a convex set, is strictly inside the convex set. Using this property, given set of points $(x_1, \dots, x_n)$ each in X, each component function f_i map x_i to $x_i + u_i = x_i + k_i(\mathrm{CC}_i^* - x_i)$ from (2), and (12). Obviously $[x_i\ \mathrm{CC}_i^*] \subseteq [x_i\ \mathrm{CC}_i]$, and since $k_i \in (0\ 1)$ for each i, the following relation holds.

$$f_i(x_i, u_i) \in (x_i, x_i\ \mathrm{CC}_i^*) \subset [x_i\ \mathrm{CC}_i] \subset \mathrm{conv}(X_i) \subseteq \mathrm{conv}(x)$$

For each $i \in I$, $f_i(x_i, u_i)$ is strictly in $\mathrm{conv}(X_i)$ if $x_i \neq \mathrm{CC}_i$, and there is always at least one node $j \in I \setminus \{i\}$ such that $x_j \in \mathrm{Ver}(x)$. Hence, $\mathrm{conv}(f_1(x_1, u_1), \dots, f_n(x_n, u_n)) \subset \mathrm{conv}(x)$, and the inclusion is strict if x is not consensus, which proves that the map f is *proper*. In [3], it is shown that agents evolving with circumcenter-law and connectivity constraint, with bounded input (similar to our gain k_i); $\|u_i\| \leq u_{\max}$ for some $u_{\max} > 0$, converges to a consensus with a different approach using the invariance principle. ■

References

[1] H. Ando, Y. Oasa, I. Suzuki, and M. Yamashita, "Distributed memoryless point convergence algorithm for mobile robots with limited visibility," *Robotics and Automation, IEEE Transactions on*, vol. 15, no. 5, pp. 818–828, 1999.

[2] J. Lin, A. Morse, and B. Anderson, "The multi-agent rendezvous problem," in *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, vol. 2. IEEE, 2003, pp. 1508–1513.

[3] F. Bullo, C. J., and M. S., *Distributed control of robotic networks: a mathematical approach to motion coordination algorithms*. Princeton University Press, 2009.

[4] S. Jaggi, M. Langberg, S. Katti, T. Ho, D. Katabi, and M. Médard, "Resilient network coding in the presence of byzantine adversaries," in *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*. IEEE, 2007, pp. 616–624.

[5] 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 *Proceedings of the 1st international conference on High Confidence Networked Systems*, 2012, p. 110.

[6] R. Olfati-Saber and R. Murray, "Consensus protocols for networks of dynamic agents," 2003.

[7] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *Automatic Control, IEEE Transactions on*, vol. 49, no. 9, p. 15201533, 2004.

[8] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, p. 215233, 2007.

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

[10] J. Cortés, S. Martínez, and F. Bullo, "Robust rendezvous for mobile autonomous agents via proximity graphs in arbitrary dimensions," *Automatic Control, IEEE Transactions on*, vol. 51, no. 8, p. 12891298, 2006.

[11] L. Lamport, R. Shostak, and M. Pease, "The byzantine generals problem," *ACM Transactions on Programming Languages and Systems (TOPLAS)*, vol. 4, no. 3, pp. 382–401, 1982.

[12] M. J. Fischer, N. A. Lynch, and M. S. Paterson, "Impossibility of distributed consensus with one faulty process," *Journal of the ACM (JACM)*, vol. 32, no. 2, pp. 374–382, 1985.

[13] V. Gupta, C. Langbort, and R. M. Murray, "On the robustness of distributed algorithms," in *Decision and Control, 2006 45th IEEE Conference on*, 2006, p. 34733478.

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

[15] H. J. LeBlanc and X. D. Koutsoukos, "Consensus in networked multi-agent systems with adversaries," in *Proceedings of the 14th international conference on Hybrid systems: computation and control*, 2011, p. 281290.

[16] S. Sundaram and C. N. Hadjicostis, "Distributed function calculation via linear iterative strategies in the presence of malicious agents," *Automatic Control, IEEE Transactions on*, vol. 56, no. 7, p. 14951508, 2011.

[17] H. Park and S. Hutchinson, "Wosrt-case performance of a mobile sensor network under individual sensor failure," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, 2013.

[18] J. Lorenz, "Repeated averaging and bounded confidence-modeling, analysis and simulation of continuous opinion dynamics," Ph.D. dissertation, Ph. D. thesis, University of Bremen, Germany, available at http://nbn-resolving. de/urn: nbn: de: gbv: 46-diss000106688, 2007.

[19] R. Hegselmann and U. Krause, "Opinion dynamics and bounded confidence models, analysis, and simulation," *Journal of Artificial Societies and Social Simulation*, vol. 5, no. 3, 2002.

[20] M. Ben-Or, "Another advantage of free choice (extended abstract): Completely asynchronous agreement protocols," in *Proceedings of the second annual ACM symposium on Principles of distributed computing*. ACM, 1983, pp. 27–30.

[21] U. Krause, "Compromise, consensus, and the iteration of means," *Elemente der Mathematik*, vol. 63, pp. 1–8, 2008.

[22] A. Brandstädt, J. P. Spinrad *et al.*, *Graph classes: a survey*. Society for Industrial Mathematics, 1987, vol. 3.

[23] D. P. Bertsekas, *Dynamic programming and optimal control*. Athena Scientific Belmont, MA, 1995, vol. 1, no. 2.