2021 60th IEEE Conference on Decision and Control (CDC)
December 13-15, 2021. Austin, Texas

Rolling Horizon Games for Cluster Formation of Resilient Multiagent Systems

Yurid Nugraha, Ahmet Cetinkaya, Tomohisa Hayakawa, Hideaki Ishii, and Quanyan Zhu

2021 60th IEEE Conference on Decision and Control (CDC) | 978-1-6654-3659-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/CDC45484.2021.9683409

***Abstract*— In this paper we formulate a two-player game-theoretic problem on resilient graphs in a multiagent consensus setting. An attacker is capable to disable some of the edges of the network with the objective to divide the agents into clusters by emitting jamming signals while, in response, the defender recovers some of the edges by increasing the transmission power for the communication signals. We consider repeated games between the attacker and the defender where the optimal strategies for the two players are derived in a rolling horizon fashion based on the agents' states and number of agents in each cluster. The players' actions at each discrete-time steps are constrained by their energy for transmissions of signals, with a less strict constraint for the attacker. Simulation results are provided to demonstrate the effects of players' actions on the cluster formation and to illustrate the performance comparison with a non-rolling horizon approach.**

I. Introduction

Applications of large-scale networked systems have rapidly grown in various areas of critical infrastructures including power grids and transportation systems. Such systems can be considered as multiagent systems where a number of agents capable of making local decisions interact over a network and exchange information [1]. While wireless communication plays an important role for the functionality of the network, it is also prone to cyber attacks initiated by malicious adversaries [2].

Jamming attacks on consensus problems of multiagent systems have been studied in [3]. Noncooperative games between the attacker and another player protecting the network are widely used to analyze security problems, including jamming attacks [4] and injection attacks [5].

In this paper, we consider a security problem in a two-player game setting between an attacker, who is motivated to disrupt the communication among agents by attacking communication links, and a defender, who attempts to recover some of the attacked links. This game is played repeatedly over time in the context of multiagent consensus. Their utilities are determined by how agents are divided into clusters as well as how the players' actions affect the states of the agents at each time.

Yurid Nugraha and Tomohisa Hayakawa are with the Department of Systems and Control Engineering, Tokyo Institute of Technology, Tokyo 152-8552, Japan. yurid@dsl.sc.e.titech.ac.jp, hayakawa@sc.e.titech.ac.jp

Ahmet Cetinkaya is with the Information Systems Architecture Science Research Division, National Institute of Informatics, Tokyo 101-8430, Japan. cetinkaya@nii.ac.jp

Hideaki Ishii is with the Department of Computer Science, Tokyo Insitute of Technology, Yokohama 226-8502, Japan. ishii@c.titech.ac.jp

Quanyan Zhu is with the Department of Electrical and Computer Engineering, New York University, Brooklyn NY, 11201, USA. quanyan.zhu@nyu.edu

We formulate the problem based on [6], [7], which use graph connectivity to characterize the game and players' strategies. Specifically, we address how clusters among agents may form in this security game setting. Cluster formation in multiagent systems has been studied in, e.g., [8], where the weights in the agents' state updates may take negative values, representing the possibly hostile relations among certain agents. In this paper, we approach clustering from a different viewpoint based on a game-theoretic formulation. This approach can be related to the concept of network effect/externality [9], where the utility of an agent in a certain cluster depends on how many other agents belong to that particular cluster. Such concepts have been used to analyze grouping of agents on, e.g., social networks and computer networks, as discussed in [10], [11].

Moreover, in comparison to our recent work [6], the contribution of this paper is threefold: (i) we introduce more options for the attacker's jamming signals strengths; (ii) the game consists of multiple attack-recovery actions, resulting in more complicated strategies; and (iii) we consider a rolling horizon approach for the players, so that their strategies may be modified as they obtain new knowledge of the system each time. Rolling horizon approaches in noncooperative security games have been discussed in [12].

More specifically, it is now possible for the attacker to disable the links with stronger intensity of attack signals so that the defender is unable to recover those links as in [13]. On the other hand, we consider games consisting of multiple parts, where the players need to consider their future utilities and energy constraints when deciding their strategies at any point in time. The players recalculate and may change their strategies as time goes on, according to the rolling horizon approach. A related formulation without rolling horizon is discussed in [14], where the players are not able to change their strategies once they are decided.

The paper is organized as follows. In Section II, we introduce the framework for the rolling horizon game, cluster formation among agents, and energy consumption models of the players. In Section III, we analyze some conditions of consensus among agents, and establish relations to parameters of the system and the players. We continue by discussing the cluster formation of agents when consensus is not achieved in Section IV. We then provide numerical examples on consensus and cluster formation in Section V and conclude the paper in Section VI.

The notations used in this paper are fairly standard. We denote $|\cdot|$ as the cardinality of a set. The floor function and ceiling function are denoted by $\lfloor\cdot\rfloor$ and $\lceil\cdot\rceil$, respectively. The set of nonnegative integers $\{0, 1, 2, \ldots\}$ is denoted by

$\mathbb{N}_0$. The proofs of all lemmas in Sections III and IV are omitted due to space limitations.

II. Problem Formulation

We explore a multiagent system of n agents communicating to each other in discrete time. The network topology is described by an undirected and connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. It consists of the set $\mathcal{V}$ of vertices representing the agents and the set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ of edges representing the communication links. Each agent i has the scalar state x_i following the consensus update rule at time $k \in \mathbb{N}_0$

$$x_i[k+1] = x_i[k] + u_i[k], \tag{1}$$

$$u_i[k] = \sum_{j \in \mathcal{N}_i[k]} a_{ij}(x_j[k] - x_i[k]), \tag{2}$$

as in [15], where $x[0] = x_0$, $a_{ij} > 0$, $i, j \in \mathcal{V}$, are such that $\sum_{j=1, j\neq i}^{n} a_{ij} < 1$, and $\mathcal{N}_i[k]$ denotes the set of agents that can communicate with agent i at time k. This set may change due to the attacks.

A two-player game between the attacker and the defender is considered in terms of the communication among the agents. The attacker is capable to block the communication by jamming some targeted edges and therefore delay (or completely prevent) the consensus among agents. These jamming attacks are represented by the removal of edges in $\mathcal{G}$. In response to the actions of the attacker, the defender tries to recover the inter-agent communications by rebuilding some of the attacked edges. From this one sequence of attacks and recoveries, we may say that the graphs characterizing the networked system are *resilient* if the group of agents is able to recover from the damages caused by the attacker.

In this paper, we consider that the attacker has two types of jamming signals in terms of their strength, *strong* and *normal*. The defender is able to recover only the edges that are attacked with normal strength. Practically, the defender may be able to differentiate between normal and strong attacks by measuring the signal-to-interference ratio on some edges; the strong attack will result in an even lower signal-to-interference ratio, which may make the recovery not possible.

A. Attack-recovery sequence

In our setting, the players make their attack/recovery actions at every discrete time $k \in \mathbb{N}_0$. At the beginning of each time k, the communication topology of the system is represented by $\mathcal{G}$. Then, the players decide to attack/recover certain edges in the two stages, with the attacker acting first and then the defender.

We assume that at time k the attacker attacks $\mathcal{G}$ by deleting $\mathcal{E}_k^{\mathrm{A}} \subseteq \mathcal{E}$ with normal jamming signals and $\overline{\mathcal{E}}_k^{\mathrm{A}} \subseteq \mathcal{E}$ with strong jamming signals with $\mathcal{E}_k^{\mathrm{A}} \cap \overline{\mathcal{E}}_k^{\mathrm{A}} = \emptyset$, whereas the defender recovers $\mathcal{E}_k^{\mathrm{D}} \subseteq \mathcal{E}_k^{\mathrm{A}}$. Due to the attacks and then the recoveries, the network changes from $\mathcal{G}$ to $\mathcal{G}_k^{\mathrm{A}} := (\mathcal{V}, \mathcal{E} \setminus (\mathcal{E}_k^{\mathrm{A}} \cup \overline{\mathcal{E}}_k^{\mathrm{A}}))$ and further to $\mathcal{G}_k^{\mathrm{D}} := (\mathcal{V}, (\mathcal{E} \setminus (\mathcal{E}_k^{\mathrm{A}} \cup \overline{\mathcal{E}}_k^{\mathrm{A}})) \cup \mathcal{E}_k^{\mathrm{D}})$ at the kth time. The agents then communicate to their neighbors based on this resulting graph $\mathcal{G}_k^{\mathrm{D}}$.

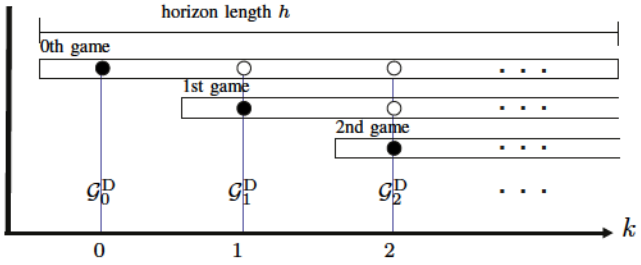

Fig. 1. Illustration of the games played over discrete time k with rolling horizon approach for the players.

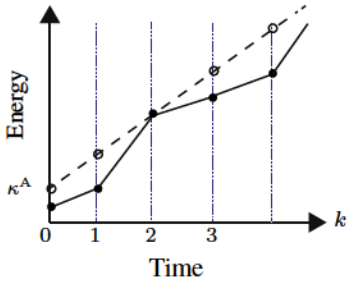

Fig. 2. Energy constraint of the attacker considered in the formulation. The dashed line represents the allowable energy to spend. The solid line representing the actual energy consumed by the player should be below the dashed line.

In this game, the players attempt to choose the best strategies in terms of edges attacked/recovered $(\overline{\mathcal{E}}_k^{\mathrm{A}}, \mathcal{E}_k^{\mathrm{A}})$ and $\mathcal{E}_k^{\mathrm{D}}$ to maximize their own utility functions. Here the lth game is defined over the horizon of h steps from time l to $l+h-1$. The players make decisions in a rolling horizon fashion as explained more in Section II-D; the optimal strategies obtained at the past time l for the future time may change when the players recalculate their strategies at a future time. Fig. 1 illustrates the discussed sequence over time; in the figure, the filled black circles indicate the implemented strategies and the empty circles indicate the strategies of the game that are not implemented.

B. Energy constraints

The actions of the attacker and the defender are affected by the constraints on the energy resources, which increase linearly in time; however, the energy consumed by the players is proportional to the number of attacked/recovered edges as well. Note that the attacker has two types of jamming signals. Here, the strong attacks on $\overline{\mathcal{E}}_k^{\mathrm{A}}$ take $s > 1$, $s \in \mathbb{R}$, times more energy per edge per unit time compared to the normal attacks on $\mathcal{E}_k^{\mathrm{A}}$, which take β^{A} cost per edge. The total energy used by the attacker is constrained as

$$\sum_{m=0}^{k} \beta^{\mathrm{A}}(s|\overline{\mathcal{E}}_m^{\mathrm{A}}| + |\mathcal{E}_m^{\mathrm{A}}|) \leq \kappa^{\mathrm{A}} + \rho^{\mathrm{A}} k \tag{3}$$

for any time k, where $\kappa^{\mathrm{A}} \geq \rho^{\mathrm{A}} > 0$, $\beta^{\mathrm{A}} > 0$. The condition $\kappa^{\mathrm{A}} \geq \rho^{\mathrm{A}}$ allows the attacker to have enough energy to attack at time 0 with consistent recharge rate. This inequality implies that the total energy spent by the attacker cannot exceed the available energy characterized by the initial energy κ^{A} and the supplied energy $\rho^{\mathrm{A}} k$ by time k.

Fig. 2 shows the energy constraint of the attacker, where the dashed line with slope ρ^{A} represents the total supplied energy and the solid line indicates the total energy spent. A critical case is when $\beta^{\mathrm{A}} < \rho^{\mathrm{A}}$ since it is possible for the

attacker to attack some edges for infinite time, as long as $s|\overline{\mathcal{E}}_k^{\mathrm{A}}|+|\mathcal{E}_k^{\mathrm{A}}|\leq \overline{m}^{\mathrm{A}}$ is satisfied with $\overline{m}^{\mathrm{A}} := \frac{\rho^{\mathrm{A}}}{\beta^{\mathrm{A}}}$.

The energy constraint of the defender is similar to (3) and is given by $\sum_{m=0}^{k} \beta^{\mathrm{D}}|\mathcal{E}_m^{\mathrm{D}}|\leq \kappa^{\mathrm{D}}+\rho^{\mathrm{D}}k$ with $\kappa^{\mathrm{D}} \geq \rho^{\mathrm{D}} > 0$, $\beta^{\mathrm{D}} > 0$. Note that the defender can recover only the edges in $\mathcal{E}_k^{\mathrm{A}}$ under normal jamming attacks.

C. Agent clustering and state difference

By attacking, the attacker tries to make the graph disconnected to separate the agents into several groups. We introduce a few notions related to grouping/clustering of agents. In a given subgraph $\mathcal{G}' = (\mathcal{V}, \mathcal{E}')$ of $\mathcal{G}$, the agents are to be grouped into $\overline{c}(\mathcal{G}')$ number of *groups*, with the groups $\mathcal{V}_1', \mathcal{V}_2', \ldots, \mathcal{V}_{\overline{c}(\mathcal{G}')}'$ being a partition of $\mathcal{V}$ with $\cup_{l=1}^{\overline{c}(\mathcal{G}')}\mathcal{V}_l' = \mathcal{V}$ and $\mathcal{V}_l' \cap \mathcal{V}_m' = \emptyset$ with $l \neq m$. There is no edge connecting different groups, i.e., $e_{ij} \notin \mathcal{E}'$ for $i \in \mathcal{V}_l', j \in \mathcal{V}_m'$. We also call each subset of agents taking the same state at infinite time as a *cluster*, i.e., $\lim_{k\to\infty} x_{i'}[k] = \lim_{k\to\infty} x_{j'}[k]$.

Here, we are interested in the case where the attacker is also concerned about the number of agents in each group, as an extension of [6], [7]. Specifically, we follow the notion of *network effect/externality* [9], where the utility of an agent in a certain cluster depends on how many other agents belong to that particular cluster. In the context of this game, the attacker wants to isolate agents so that fewer agents are in each group, while the defender wants as many agents as possible in the same group. We then represent the level of clustering in the graph $\mathcal{G}'$ by the function $c(\cdot)$ called *cluster distribution*, which is given by $c(\mathcal{G}') := \sum_{l=1}^{\overline{c}(\mathcal{G}')}|\mathcal{V}_l'|^2 - |\mathcal{V}|^2$.

In our problem setting, the players also consider the effects of their actions on the agent states when attacking/recovering, similar to the formulation in [16]. For example, the attacker may want to separate agents having state values with more difference in different groups. We specify the agents' state difference z_k of time k as

$$z_k(\overline{\mathcal{E}}_k^{\mathrm{A}}, \mathcal{E}_k^{\mathrm{A}}, \mathcal{E}_k^{\mathrm{D}}) := x^{\mathrm{T}}[k+1]L_{\mathrm{c}}x[k+1], \tag{4}$$

with L_{c} being the Laplacian matrix of a complete graph with n agents. The attacked and recovered edges $(\overline{\mathcal{E}}_k^{\mathrm{A}}, \mathcal{E}_k^{\mathrm{A}}, \mathcal{E}_k^{\mathrm{D}})$ will affect $x[k+1]$, and in turn influence the value of z_k. Note that the value of z_k does not increase over time [1] because of the protocol given in (1) and (2).

D. Multiple-attack rolling horizon game structure

The utility functions of both attacker and defender of the lth game, $l \in \mathbb{N}_0$, starting at time $k = l$ take account of the cluster distribution $c(\cdot)$ and the difference z_k of agents' states over $h \geq 1$ horizon length from time l to $l+h-1$. With weights $a, b \geq 0$, the utilities are defined by

$$U^{\mathrm{A}} := \sum_{k=l}^{l+h-1} (az_k - bc(\mathcal{G}_k^{\mathrm{D}})), \tag{5}$$

$$U^{\mathrm{D}} := -U^{\mathrm{A}}. \tag{6}$$

We are interested in finding the subgame perfect equilibrium of this game. To find the equilibrium, the game is divided into some subgames/decision-making points. The subgame perfect equilibrium must be an equilibrium in every subgame. The optimal strategy of each player is obtained by using a backward induction approach, i.e., by finding the equilibrium from the smallest subgames.

Due to the nature of the rolling horizon approach, the strategies obtained from the lth game, i.e., attacked and recovered edges, are not applied, except those for time l. Specifically, in the lth game for time l to $l+h-1$, the strategies of both players are denoted by $((\overline{\mathcal{E}}_{l,0}^{\mathrm{A}}, \mathcal{E}_{l,0}^{\mathrm{A}}, \mathcal{E}_{l,0}^{\mathrm{D}}), \ldots, (\overline{\mathcal{E}}_{l,h-1}^{\mathrm{A}}, \mathcal{E}_{l,h-1}^{\mathrm{A}}, \mathcal{E}_{l,h-1}^{\mathrm{D}}))$, where only $(\overline{\mathcal{E}}_{l,0}^{\mathrm{A}}, \mathcal{E}_{l,0}^{\mathrm{A}}, \mathcal{E}_{l,0}^{\mathrm{D}})$ is applied. Therefore, for the lth game at time l, the strategy applied can be written as $(\overline{\mathcal{E}}_l^{\mathrm{A}}, \mathcal{E}_l^{\mathrm{A}}, \mathcal{E}_l^{\mathrm{D}}) = (\overline{\mathcal{E}}_{l,0}^{\mathrm{A}}, \mathcal{E}_{l,0}^{\mathrm{A}}, \mathcal{E}_{l,0}^{\mathrm{D}})$. The same notations apply for the functions z_k.

We look at how the optimal edges can be found by an example with $h = 2$. In this case, for the lth game over time l and $l+1$, the optimal strategies of the players are given by

$$\mathcal{E}_{l,1}^{\mathrm{D}*}(\overline{\mathcal{E}}_{l,1}^{\mathrm{A}}, \mathcal{E}_{l,1}^{\mathrm{A}}) \in \arg\max_{\mathcal{E}_{l,1}^{\mathrm{D}}} U_1^{\mathrm{D}}, \tag{7}$$

$$(\overline{\mathcal{E}}_{l,1}^{\mathrm{A}*}(\mathcal{E}_{l,0}^{\mathrm{D}}), \mathcal{E}_{l,1}^{\mathrm{A}*}(\mathcal{E}_{l,0}^{\mathrm{D}})) \in \arg\max_{(\overline{\mathcal{E}}_{l,1}^{\mathrm{A}}, \mathcal{E}_{l,1}^{\mathrm{A}})} U_1^{\mathrm{A}}, \tag{8}$$

$$\mathcal{E}_{l,0}^{\mathrm{D}*}(\overline{\mathcal{E}}_{l,0}^{\mathrm{A}}, \mathcal{E}_{l,0}^{\mathrm{A}}) \in \arg\max_{\mathcal{E}_{l,0}^{\mathrm{D}}} U^{\mathrm{D}}, \tag{9}$$

$$(\overline{\mathcal{E}}_{l,0}^{\mathrm{A}*}, \mathcal{E}_{l,0}^{\mathrm{A}*}) \in \arg\max_{(\overline{\mathcal{E}}_{l,0}^{\mathrm{A}}, \mathcal{E}_{l,0}^{\mathrm{A}})} U^{\mathrm{A}}, \tag{10}$$

with U_1^{A} and U_1^{D} being parts of U^{A} and U^{D} associated with the strategies of time $(l+1)$ of the lth game, respectively.

Note that to find $(\overline{\mathcal{E}}_{l,0}^{\mathrm{A}*}, \mathcal{E}_{l,0}^{\mathrm{A}*})$, one needs to obtain $\mathcal{E}_{l,0}^{\mathrm{D}*}(\overline{\mathcal{E}}_{l,0}^{\mathrm{A}}, \mathcal{E}_{l,0}^{\mathrm{A}})$ beforehand. Likewise, to find $\mathcal{E}_{l,0}^{\mathrm{D}*}$, one needs to obtain $(\overline{\mathcal{E}}_{l,1}^{\mathrm{A}*}(\mathcal{E}_{l,0}^{\mathrm{D}}), \mathcal{E}_{l,1}^{\mathrm{A}*}(\mathcal{E}_{l,0}^{\mathrm{D}}))$. For $h > 2$, the optimization problems are similar to those in (7)–(10), solved in $2h$ steps from the $(h-1)$th step of the lth game. They are solved by the players at every time $k = l$.

In this paper, we focus on cluster formation over time k rather than characterizing the equilibrium at each individual game. We will find the optimal strategies of the players by computing all possible combinations, since the choices of edges are finite.

Our previous works [6], [7], [14] considered related games in continuous time, where the timings for launching attack/defense actions are also part of the decision variables. This aspect complicated the formulation, making it difficult to study games over a time horizon. In the current paper, we simplify the timing issue and instead introduced the rolling horizon feature. This enables the players to consider the cluster forming in a longer time range, which is especially useful when consensus among agents is considered.

III. Consensus Analysis

In this section, we examine the effect of the game structure and players' energy constraints on consensus.

We first discuss the defender's optimal strategy on some games with specific conditions. Lemma 3.1 provides the defender's optimal edges for the last step of the lth game.

Lemma 3.1: In the $(h-1)$th step of the lth game, the set of edges $\mathcal{E}_{l,h-1}^{\mathrm{D}*}$ satisfies $|\mathcal{E}_{l,h-1}^{\mathrm{D}*}|=$

$\min(\lfloor \frac{\kappa^{\mathrm{D}}+\rho^{\mathrm{D}}(l+h-1)-\beta^{\mathrm{D}}(\sum_{m=0}^{l-1}|\mathcal{E}_m^{\mathrm{D}}|+\sum_{m=0}^{h-2}|\mathcal{E}_{l,m}^{\mathrm{D}}|)}{\beta^{\mathrm{D}}} \rfloor,$
$|\mathcal{E}_{l,h-1}^{\mathrm{A}*}|)$.

Lemma 3.2 states that if the defender has enough energy, it recovers all possible edges at the 0th step of the game.

Lemma 3.2: If the defender's energy satisfies

$$\left\lfloor \frac{\kappa^{\mathrm{D}}+\rho^{\mathrm{D}} l-\sum_{m=0}^{l-1}\beta^{\mathrm{D}}|\mathcal{E}_m^{\mathrm{D}}|}{\beta^{\mathrm{D}}} \right\rfloor \geq h|\mathcal{E}| \tag{11}$$

at time l, then $\mathcal{E}_l^{\mathrm{D}*}=\mathcal{E}_l^{\mathrm{A}*}$.

The next lemma states that the condition (11) in Lemma 3.2 always occurs in some interval of discrete time.

Lemma 3.3: There is at least one occurrence of either $\mathcal{E}_l^{\mathrm{D}}\neq\emptyset$ or $\mathcal{E}_l^{\mathrm{A}}=\emptyset$ in every $\lceil \frac{h|\mathcal{E}|\beta^{\mathrm{D}}}{\rho^{\mathrm{D}}} \rceil$ time steps.

The following two results provide necessary conditions for the agents to be separated into different states for infinitely long duration without achieving consensus. We consider a more general condition in Theorem 3.4, whereas in Theorem 3.5 we consider a more specific situation for the utility functions that leads to a tighter condition.

Theorem 3.4: A necessary condition for consensus not to happen is $\lambda \leq \overline{m}^{\mathrm{A}}$, with λ being the connectivity of $\mathcal{G}$.

Proof: We note that, without any recovery from the defender ($\mathcal{E}_k^{\mathrm{D}}=\emptyset$), the attacker must attack at least λ number of edges with normal signals at any time k in order to make $\mathcal{G}_k^{\mathrm{D}}$ disconnected. If the attacker attacks λ edges with normal jamming signals at all time, the energy constraint (3) becomes $(\beta^{\mathrm{A}}\lambda-\rho^{\mathrm{A}})k \leq \kappa^{\mathrm{A}}$. From this inequality, it is clear that the attacker needs to have high enough recharge rate ρ^{A} to attack λ edges at all time. Specifically, the condition $\bar{m}^{\mathrm{A}} \geq \lambda$ has to be satisfied. ■

We now limit the utilities in (5) and (6) to the case of $b=0$ in the weights. This means that the players do not take account of the clustering in the graph, but only the status in consensus.

Theorem 3.5: Suppose that $b=0$. A necessary condition for consensus not to happen is $\lambda \leq \overline{m}^{\mathrm{A}}/s$.

Proof: We prove by contrapositive; namely, we prove that consensus always happens if $\overline{m}^{\mathrm{A}} < s\lambda$.

We first suppose that the attacker attempts to attack λ edges strongly at all time to disconnect the graph $\mathcal{G}_k^{\mathrm{D}}$. From (3), the energy constraint of the attacker at time k becomes $(\beta^{\mathrm{A}}s\lambda-\rho^{\mathrm{A}})k \leq \kappa^{\mathrm{A}}$. This inequality is not satisfied for higher k if $\overline{m}^{\mathrm{A}} < s\lambda$, since the left-hand side becomes positive and κ^{A} is finite. Therefore, the attacker cannot attack λ edges strongly at all time if $\overline{m}^{\mathrm{A}} < s\lambda$, and is forced to disconnect the graph by attacking with normal jamming signals instead.

By Lemma 3.3, there exists an interval where the defender always recovers if there are edges attacked normally, i.e., $\mathcal{E}_{l'}^{\mathrm{D}}\neq\emptyset$ are optimal given that $\mathcal{E}_{l'}^{\mathrm{A}}\neq\emptyset$. From the definitions in (6), given that $b=0$, we can see that the defender obtains a higher utility if the agents are closer, which means that given a nonzero number of edges to recover (at time jl' described above), the defender recovers the edges connecting further agents. Specifically, for interval $[jl',(j+i)l']$, there is a time step where $U^{\mathrm{D}}(\mathcal{E}_k^{\mathrm{D}}=\mathcal{E}_1) \geq U^{\mathrm{D}}(\mathcal{E}_2)$, with edges $\mathcal{E}_1$ connecting agents with further states than agents connected by $\mathcal{E}_2$. This implies that when recovering, the defender always chooses the further disconnected agents, and since by communicating with the consensus protocol as in (1) the agents' states are getting closer, the defender will choose different edges to recover if the states of agents connected by recovered edges $\mathcal{E}_k^{\mathrm{D}}$ become close enough. Consequently, if $\overline{m}^{A} < s\lambda$, then there exists $i \in \mathbb{N}$ where the union of graphs, i.e., the graph having the union of the edges of each graph, $(\mathcal{V}, \bigcup((\mathcal{E}\setminus(\overline{\mathcal{E}}_k^{\mathrm{A}}\cup\mathcal{E}_k^{\mathrm{A}}))\cup\mathcal{E}_k^{\mathrm{D}}))$ over the time interval $[jl',(j+i)l']$ becomes a connected graph, where $l'=\lceil\frac{h|\mathcal{E}|\beta^{\mathrm{D}}}{\rho^{\mathrm{D}}}\rceil$ as in Lemma 3.3 above. These intervals $[jl',(j+i)l']$ occur infinitely many times, since the defender's energy bound keeps increasing over time.

It is shown in [17] that with protocol (1), the agents achieve consensus in the time-varying graph as long as the union of the graphs over bounded time intervals is a connected graph. This implies that consensus is achieved if $(\mathcal{V}, \bigcup((\mathcal{E}\setminus(\overline{\mathcal{E}}_k^{\mathrm{A}}\cup\mathcal{E}_k^{\mathrm{A}}))\cup\mathcal{E}_k^{\mathrm{D}}))$ is connected over $[l_i', l_{i+j}']$. Thus, if $\overline{m}^{\mathrm{A}} < s\lambda$ then consensus is achieved. ■

The result in Theorem 3.5 only holds for $b=0$, since with $b>0$ the defender may choose to recover the edges connecting agents that already have similar states to maximize $c(\mathcal{G}_k^{\mathrm{D}})$ (instead of those connecting farther agents). The effect of the values of a and b on consensus is illustrated in Section V.

The next result provides a condition for consensus to be prevented. It shows that an attacker who is capable to continuously make strong attacks on the edges, can prevent consensus.

Theorem 3.6: A sufficient condition for consensus not to happen is $|\mathcal{E}|\leq \overline{m}^{\mathrm{A}}/s$.

Proof: With $\overline{m}^{\mathrm{A}} \geq s|\mathcal{E}|$, the attacker can attack all edges of the graph $\mathcal{G}$ with strong jamming signals at any time k (including time 0, since $\kappa^{\mathrm{A}} \geq \rho^{\mathrm{A}}$ by assumption). Note that at the $(h-1)$th step of the lth game, the attacker always attacks as many edges as possible to maximize U^{A}, if there is no recovery by the defender. Since $z_{l,h-1}(\overline{\mathcal{E}}_{l,h-1}^{\mathrm{A}}=\mathcal{E}', \mathcal{E}_{l,h-1}^{\mathrm{A}}, \emptyset) > z_{l,h-1}(\overline{\mathcal{E}}_{l,h-1}^{\mathrm{A}}, \mathcal{E}_{l,h-1}^{\mathrm{A}}, \emptyset)$ and $c(\mathcal{V},(\mathcal{E}\setminus(\mathcal{E}'\cup\mathcal{E}_{l,h-1}^{\mathrm{A}}))\cup\emptyset) \geq c(\mathcal{V},(\mathcal{E}\setminus(\overline{\mathcal{E}}_{l,h-1}^{\mathrm{A}}\cup\mathcal{E}_{l,h-1}^{\mathrm{A}}))\cup\emptyset))$ for any $|\overline{\mathcal{E}}_{l,h-1}^{\mathrm{A}}|<|\mathcal{E}'|$, the function U_{h-1}^{A} always has a higher value if more edges are attacked. Thus, the attacker will attack all edges strongly $\overline{\mathcal{E}}_{l,h-1}^{\mathrm{A}*}=\mathcal{E}$, which also prevents the defender from recovering any edges. It then follows that the function U_{h-1}^{A} does not vary for different choices of edges in the previous attack $((\overline{\mathcal{E}}_{l,0}^{\mathrm{A}},\mathcal{E}_{l,0}^{\mathrm{A}}),\ldots,(\overline{\mathcal{E}}_{l,h-2}^{\mathrm{A}},\mathcal{E}_{l,h-2}^{\mathrm{A}}))$. This implies that the attacker does not need to attack fewer edges at the previous steps to save energy, since it already has enough energy to attack all possible edges $\mathcal{E}$ at the next steps. Thus, $\overline{\mathcal{E}}_l^{\mathrm{A}*}=\mathcal{E}$ at the 0th step of the lth game.

This implies that if $\overline{m}^{\mathrm{A}} \geq s|\mathcal{E}|$, the attacker will attack $\mathcal{E}$ strongly at all time, separating every agent. As a result, consensus is not reached. ■

IV. Clustering Analysis

In this section, we derive some results on the number of formed clusters of agents at infinite time. These results are

related to those in Section III, since agents are separated into different clusters when consensus is not reached.

The first result is a corollary of Theorem 3.6.

Corollary 4.1: There are n clusters formed with each cluster consisting of one agent, if $\overline{m}^{\mathrm{A}} \geq s|\mathcal{E}|$.

The next results discuss relations between the attacker's cost and energy recharge rate with the maximum number of clusters that it may create through jamming. In these results we assume that $b = 0$.

Lemma 4.2: With $b = 0$, the attacker cannot divide the agents into more than $\lfloor \overline{m}^{\mathrm{A}}/s \rfloor + 1$ number of clusters.

We then extend this result to the case where all the agents are connected with each other.

Lemma 4.3: With $b = 0$, in a complete graph $\mathcal{G}$, the attacker cannot divide the agents into more than m clusters, with m satisfying $\sum_{i=1}^{m-1}(n-i) \leq \lfloor \overline{m}^{\mathrm{A}}/s \rfloor < \sum_{i=1}^{m}(n-i)$.

V. Simulation Results

A. Consensus and Clustering across Weights a and b

Here we show how the consensus varies across different weights a and b of the utility functions.

We consider the four agents line/path graph 1–2–3–4 with initial states $x_0 = [1, 0.75, 0.75, -1]^{\mathrm{T}}$. The parameters are $\beta^{\mathrm{A}} = \beta^{\mathrm{D}} = 1$, $h = s = 2$, $\kappa^{\mathrm{A}} = \rho^{\mathrm{A}} = 2.6$, $\rho^{\mathrm{D}} = 0.3$, and $\kappa^{\mathrm{D}} = 0.8$, which satisfy the necessary condition in Theorem 3.4. Figs. 3 and 5 show the agent states with small a and large a, respectively. Figs. 4 and 6 illustrate the status of the edges in $\mathcal{G}_k^{\mathrm{D}}$ over time k. There, no circle in the corresponding edges implies that the edges are strongly attacked; likewise, red circles: normally attacked, black circles: not attacked, and filled circles: recovered.

With the small a, the attacker more often divides the agents into more groups, indicated by fewer black circles in Fig. 4. As a result, the attacker fails to prevent consensus (Fig. 3), despite the condition in Theorem 3.4 being satisfied. On the other hand, with the large a, the attacker is more focused to make the agents' state difference larger while separating agents into fewer groups compared to the case with small a, as shown in Figs. 5 and 6 with more black circles and no consensus among the agents.

We next present a comparison in the optimal state difference $z_k(\overline{\mathcal{E}}_k^{\mathrm{A}*}, \mathcal{E}_k^{\mathrm{A}*}, \mathcal{E}_k^{\mathrm{D}*})$ and cluster distribution $c(\mathcal{G}_k^{\mathrm{D}})$ across different a and b. These are shown in Fig. 7, with weight $b = 1-a$. We observe that with larger a, the attacker successfully prevents the consensus among agents (shown with larger values of z_k) up to time $k = 20$. On the other hand, with smaller a, i.e., larger b, the attacker obtains higher $c(\mathcal{G}_k^{\mathrm{D}})$ over time at the cost of low z_k, implying that the attacker fails to group the agents into different states.

B. Comparison with Non-Rolling Horizon Approach

We continue by comparing the proposed approach using the rolling horizon-based strategies with a simpler one (without rolling horizon) using the same horizon length h. Specifically, we consider that the attacker does not behave strategically, and attacks random edges with uniform distribution instead. We then observe the response of the defender under this non-optimal attack. Without the rolling horizon approach, the defender does not recalculate their strategies every time, and applies the strategy that has been determined before.

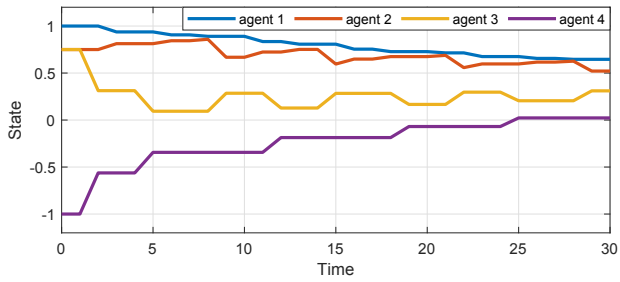

Fig. 3. Agent states with $a = 0.1$ and $b = 0.9$

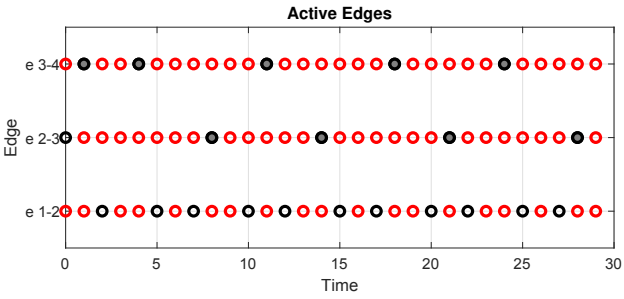

Fig. 4. Attacked and recovered edges with $a = 0.1$ and $b = 0.9$

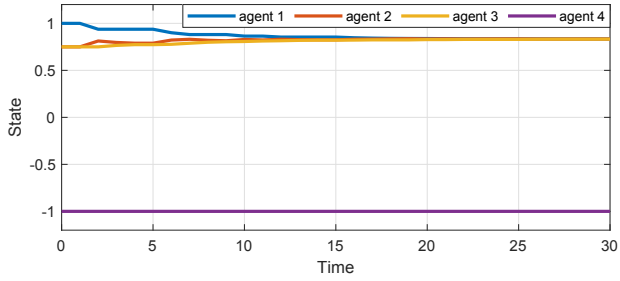

Fig. 5. Agent states with $a = 0.9$ and $b = 0.1$

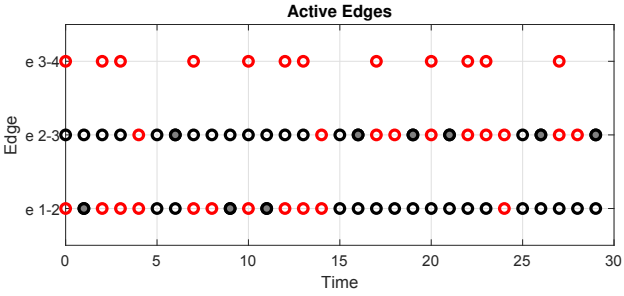

Fig. 6. Attacked and recovered edges with $a = 0.9$ and $b = 0.1$

Note that with the sequential nature of the actions, the optimal strategy of the defender depends on the attacker's strategy. Therefore, if the attacker changes its strategy, the defender may not be able to apply its own strategy in response to that. For example, suppose that the players' optimal strategies are (e_{12}, e_{34}) for the attacker, and e_{12} for the defender. When the attacker deviates by attacking e_{23}, then the defender's strategy is not applicable anymore. Fig. 8 illustrates the decision making process of the players at every k with random edges in the first step for the case of $h = 2$; the defender then has to formulate its optimal

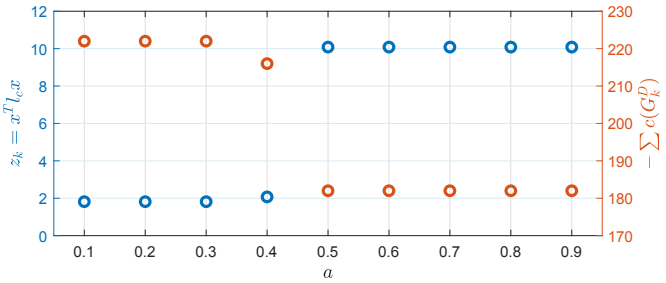

Fig. 7. Comparison of z_k and $-\sum c(\mathcal{G}_k^{\mathrm{D}})$ $(k = 20)$ versus a

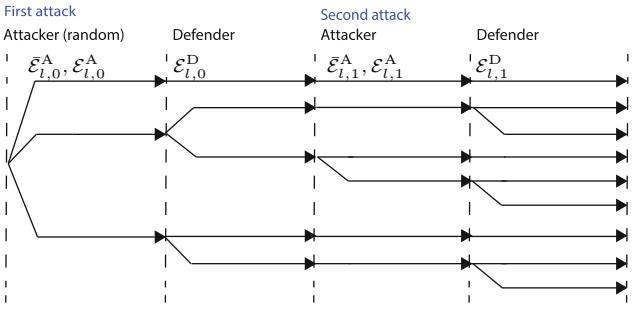

Fig. 8. Decision making process of the players with non-optimal attack as explained in Section V-B

recovery from this random attack, assuming that the second attack will be optimal.

The result with the rolling horizon approach is shown in Fig. 9. We can see here that with non-optimal random attack, the defender is able to adapt and make z_k smaller and thus the agents achieve consensus in all realizations of the random attack. On the other hand, Fig. 10 shows z_k of the non-rolling horizon approach. Here we observe that z_k is relatively large compared to that in Fig. 9, implying that consensus is achieved at a slower pace in this setting. This indicates that in the non-optimal attack situation, the rolling horizon approach can be more effective, since the defender can adapt better to uncertainty.

VI. Conclusion

We have formulated a two-player game in a cluster formation of resilient multiagent systems played over time. The players consider the impact of their actions on future communication topology and agent states, and adjust their strategies according to a rolling horizon approach. Necessary conditions and sufficient conditions for forming clusters among agents have been derived. We have discussed the effect of the weights of the utility functions and different initial conditions on cluster formation, and compared the behaviors of the players in rolling horizon and non-rolling horizon settings.

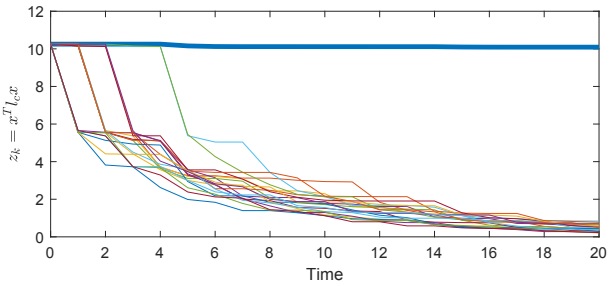

Fig. 9. State difference z_k with the random attack strategy (20 Monte Carlo simulations, each for 20 time steps), where the thick blue line indicates z_k from the optimal strategies of both players

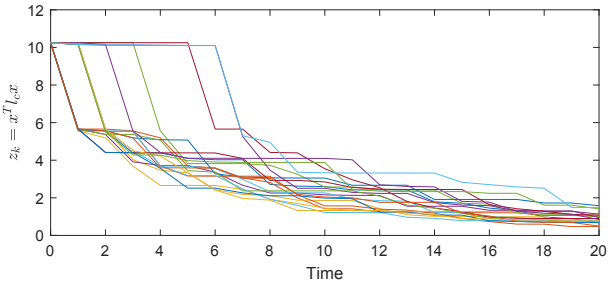

Fig. 10. State difference z_k with random attack strategy without rolling horizon (20 Monte Carlo simulations, each for 20 time steps)

References

[1] F. Bullo, *Lectures on Network Systems*. Kindle Direct Publishing, 2019.

[2] H. Sandberg, S. Amin, and K. H. Johansson, "Special issue on cyberphysical security in networked control systems," *IEEE Control Syst. Mag.*, vol. 35, pp. 20–23, 2015.

[3] A. Cetinkaya, K. Kikuchi, T. Hayakawa, and H. Ishii, "Randomized transmission protocols for protection against jamming attacks in multi-agent consensus," *Automatica*, vol. 117, 2020.

[4] Y. Li, D. E. Quevedo, S. Dey, and L. Shi, "SINR-based DoS attack on remote state estimation: A game-theoretic approach," *IEEE Trans. Control Netw. Syst.*, vol. 4, pp. 632–642, 2017.

[5] M. Pirani, E. Nekouei, H. Sandberg, and K. Johansson, "A graph-theoretic equilibrium analysis of attacker-defender game on consensus dynamics under $\mathcal{H}_2$ performance metric," *IEEE Trans. Control Netw. Syst.*, vol. 8, pp. 1991–2000, 2021.

[6] Y. Nugraha, A. Cetinkaya, T. Hayakawa, H. Ishii, and Q. Zhu, "Dynamic resilient network games with applications to multiagent consensus," *IEEE Trans. Control Netw. Syst.*, vol. 8, pp. 246–259, 2021.

[7] ——, "Cluster formation in multiagent consensus via dynamic resilient graph games," in *Proc. IEEE Conf. Dec. Contr.*, 2020, pp. 3779–3784.

[8] G. De Pasquale and M. Elena Valcher, "Consensus problems on clustered networks," in *Proc. IEEE Conf. Dec. Contr.*, 2020, pp. 3675–3680.

[9] M. L. Katz and C. Shapiro, "Systems competition and network effects," *Journal of Economic Perspective*, vol. 8, pp. 93–115, 1994.

[10] Y. Li, C. A. Courcoubetis, L. Duan, and R. Weber, "Optimal pricing for peer-to-peer sharing with network externalities," *IEEE/ACM Trans. Netw.*, vol. 29, no. 1, pp. 148–161, 2021.

[11] X. Gong, L. Duan, X. Chen, and J. Zhang, "When social network effect meets congestion effect in wireless networks: Data usage equilibrium and optimal pricing," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 2, pp. 449–462, 2017.

[12] M. Zhu and S. Martinez, "On the performance analysis of resilient networked control systems under replay attacks," *IEEE Trans. Autom. Contr.*, vol. 59, no. 3, pp. 804–808, 2014.

[13] D. Yang, G. Xue, J. Zhang, A. Richa, and X. Fang, "Coping with a smart jammer in wireless networks: A Stackelberg game approach," *IEEE Trans. Wireless Commun.*, vol. 12, pp. 4038–4047, 2013.

[14] Y. Nugraha, A. Cetinkaya, T. Hayakawa, H. Ishii, and Q. Zhu, "Cluster formation in multiagent consensus via dynamic resilient graph games," in *Proc. IEEE Conf. Control Tech. App.*, 2021, pp. 735–740.

[15] S. M. Dibaji, H. Ishii, and R. Tempo, "Resilient randomized quantized consensus," *IEEE Trans. Autom. Contr.*, vol. 63, no. 8, pp. 2508–2522, 2018.

[16] Y. Nugraha, A. Cetinkaya, T. Hayakawa, H. Ishii, and Q. Zhu, "Dynamic resilient graph games for state-dependent jamming attacks analysis on multi-agent systems," in *Proc. IFAC World Congress*, 2020, pp. 3483–3488.

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