An Evolutionary Game Theoretical Framework for Decision Fusion in the Presence of Byzantines

Yiqing Lin*, Hong Hu*, H.Vicky Zhao* and Yan Chen†
* Department of Automation, Beijing National Research Center for Information Science and Technology, Tsinghua University, Beijing, P. R. China
† School of Cyberspace Security, University of Science and Technology of China, Hefei, Anhui, P. R. China.

***Abstract*—It is an established fact that malicious users in networks are able to mislead other users since the presence of herd behaviors, which will further amplify the hazards of these malicious behavior. Due to the aforementioned scenarios in many practical applications, the study of decision fusion in the presence of such malicious users (often called Byzantines) is receiving increasing attentions. In this paper, we propose an evolutionary game theoretical model for decision fusion in the presence of Byzantines (EGT-DFB) to measure the hazard of Byzantines and to perform decision fusion. Specifically, we derive the evolution dynamics and the corresponding evolutionary stable states (ESS), which can be utilized to develop an optimum fusion strategy for the fusion center (FC) based on maximum a posterior probability criterion (MAP). Finally, simulation experiments are conducted to validate the performance of the proposed model and the effectiveness of decision fusion mechanism.**

***Index Terms*—Adversarial signal processing, decision fusion, Byzantine nodes, graphical evolutionary game theory.**

I. Introduction

Decision fusion in the presence of malicious nodes, often referred to as Byzantines [1], has received increasing attention due to its practicality in several applications, including cognitive radio, sensor networks, social networks, etc. In most scenarios, a fusion center is required to make a decision based on the reports given by local nodes on a multi-sensor network, while the byzantine nodes deliberately provide false information to mislead the center. A real-life example is quality evaluation for online products, which, for consumers, mainly depends on the reviews (the reports in the above description) given by previous buyers. The Byzantines who create false reviews could have a great impact on consumers' shopping choices, which will lead to the disruption of normal market competition and cause many undesirable consequences. Therefore, it is of great importance to model the behavior of Byzantines and study the optimum decision fusion in such an adversarial setting.

Early research on decision fusion did not take the Byzantines into consideration, and the systematic local observation error is the main factor affecting the decision results. In this case, the works in [2] and [3] determined the optimum algorithm to combine the local reports according to the Bayesian approach, which is called Chair-Varshney rule. This method degenerates into a simple majority-based decision when local error probabilities are equal for all nodes. The problem of distributed detection in the presence of Byzantines is considered in [4], which formulates decision fusion as a Neyman-Pearson setup and determines the fraction of Byzantines impeding any correct decision. Also, the work in [4] assumes that the Byzantines cooperate with each other to infer the exact status of the system and to attack the system accordingly. Analysis in [4] is extended in [5], which models the interplay between the Byzantines and fusion center as a zero-sum game. The authors in [5] also determine the minimum fraction of Byzantines impeding any correct decision with both cooperative and non-cooperative Byzantines. One way to get better results is to collect reports from different time windows and gradually identify and eliminate Byzantines in the process, which is also adopted in [5]. In [5], the report results at different moments are comprehensively analyzed to assign a reputation measure to each node, which is used to isolate Byzantines whose reputation is below a certain threshold, also is known as *Hard Isolation*. There is also a *Soft Isolation* method based on adaptive learning described in [6], where the observed behavior of the nodes is compared with the expected behavior of honest nodes. What makes this method special is that it works even when the majority of the nodes are byzantine. However, it requires very long state vectors to achieve good performances, which limits its capabilities. The authors in [7] analyze the decision fusion when some additional knowledge about the byzantine behavior is available. The results show that with the knowledge about how often byzantine report false information (P_{mal}) and the number of Byzantines in the networks, better decision results can be obtained. In [8], a game-theoretic approach is used to find the optimum strategies for the Byzantines and the fusion center. It is found that the optimal strategy for byzantines is to always report false information, which is consistent with the conclusion of [7]. A decision fusion method based on the maximum posterior probability criterion is proposed in [9], and its performance with different types of prior knowledge is also analyzed.

Although many previous works have achieved relatively satisfactory results, they all assume that all nodes report their observations independently and they do not influence each other's decision. In the real world, when there are a large number of talks about an event, an ordinary person is likely

This work is supported by the National Key Research and Development Program of China (2017YFB1400100).

to choose to follow the crowd and change his/her mind even if his/her original conclusions are correct. The phenomenon is also known as the herd behavior [10], where people sometimes ignore their own information or preferences and follow others when making decisions. In other words, the originally honest nodes may be influenced by malicious nodes and report false information, thus making the decision fusion process more challenging for the Fusion Center (FC).

To address this challenge, in this work, we study the interplay between different users, and analyze its impact on the fusion center. We use graphical evolutionary game theory [11], [12] to analyze the microscopic interactions among users and to study the impact of byzantines on other nodes as well as the fusion center. Graphical evolutionary game theory has been used to study crowd behavior in many scenarios, such as information diffusion over social networks [13], crowd dynamic analysis in emergency evacuation [14], and antagonistic crowd behaviors in cases of serious conflict [15]. Thus, in this work, we use graphical evolutionary game to analyze the impact of byzantines on other users' behavior.

Our contributions include:

1) Different from all prior works in decision fusion, we consider the scenario where users may influence each other's decision and propose a graphical evolutionary game theoretic framework to study their interactions. We analyze the evolution dynamics and quantify the impact of byzantines on other users.
2) We then study the impact of such " herding" behavior among users on the fusion center, and introduce a fusion method based on the maximum a posterior (MAP) criterion. We consider two different scenarios, where the fusion center has prior knowledge of the mean and the upper bound of the number of Byzantines, respectively, we show that our proposed fusion mechanism is more effective in resisting byzantine attack.

II. Problem Formulation

A. Decision Fusion Problem Formulation

The problem studied in this paper can be formulated into a scenario described in Figure 1, which consists of three parts: the system state, the user network, and the fusion center. In the settings considered in our work, the system state is represented by a sequence $\boldsymbol{\theta^t} = (\theta^1, \theta^2 ... \theta^t)$. The tth elements of $\boldsymbol{\theta^t}$ may correspond to system states at different epoch. To simplify the problem, we assume that θ^t is a binary vector, which means that $\theta^t \in \{0, 1\}$.

As shown in Figure 1, each user in the network makes an observation of the system state at each fixed epoch to obtain its local result, which can be expressed as an observation vector $(u_1^t, u_2^t, u_3^t ... u_n^t)$. We take the system errors in the observation process into consideration, which means that the user may observe a wrong system state (the user cannot know whether the observed result is correct). In this paper, we assume that the system error occurs with a fixed probability ε, and the errors for different users at different epochs are i.i.d. After that,

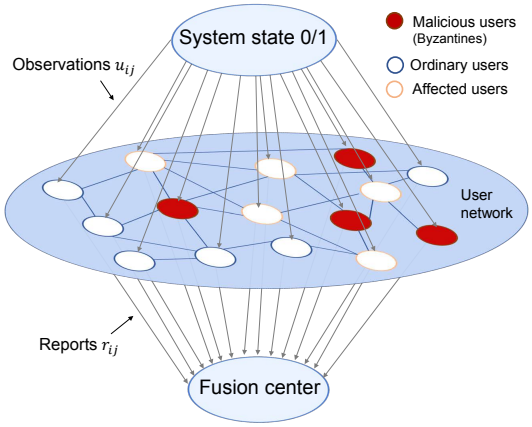


Fig. 1. Decision fusion under adversarial conditions. The orange circles represent the ordinary users who are affected by the surrounding Byzantines.

each user returns a report to the fusion center, which can be expressed as a report vector $\mathbb{R}^t = (r_1^t, r_2^t, r_3^t ... r_n^t)$. In this step, the report returned to the center may be intentionally modified by Byzantines whose purpose is to mislead the center. Another case where the returned report is different from the observed value is that an ordinary user is affected by the aforementioned herding effect and makes a decision to lie. For example, if a user finds that the reports of all surrounding users are different from his/hers, he/she will doubt the authenticity of his/her observations and may choose to modify his/her report to be consistent with those around him/her for his/her own benefit. The fusion center needs to perform decision fusion based on the reports received, so as to infer the true system status as much as possible. For the second case of reports modification mentioned above, we use graphical evolutionary game theory to describe the interaction between ordinary users and their neighbors, including ordinary and malicious ones.

B. Basic Elements of Graphical Evolutionary Game Theory

Generally, graphical evolutionary game theory contains the following basic elements: users, graph structure, strategy, fitness, and evolutionary stable states (ESS).

Users and Graph Structure: The user network is represented using an undirected and connected graph, where each node represents a user, and each edge represents the mutual relationship between a pair of users. The graph consists of ordinary users, who adopt a specific strategy updating rule, and Byzantines, who use a fixed malicious strategy. For the convenience of math derivation, we use β to represent the ratio of Byzantines to ordinary users, that is, β is the ratio of the number of Byzantines to the number of ordinary users.

Strategies: At epoch t, the true system state is θ_t, and user i receives an observation u_i^t. Each user has two strategies to choose from when reporting to FC: to lie (S_l) or to be honest (S_n). Specifically, under the definition of our binary system state, adopting the lying strategy means user i's reported value $r_i^t = \bar{u_i^t}$ where $\bar{\cdot}$ is the NOT logic operator; while adopting the S_n strategy means $r_i^t = u_i^t$ and user i reports his/her

original observation to the fusion center. Let p_l represents the percentage of ordinary users who adopt the S_l strategy. Thus, the percentage of ordinary users who adopt the S_n strategy is $(1-p_l)$.

Due to the existence of system error, a user's reported value is not only related to his/her own strategy but also related to whether there are errors ($u_i^t \neq \theta^t$) in its observation. Considering both system errors and users' possible lying behavior, we observe the following change in users' reported values, as illustrated in Figure 2.

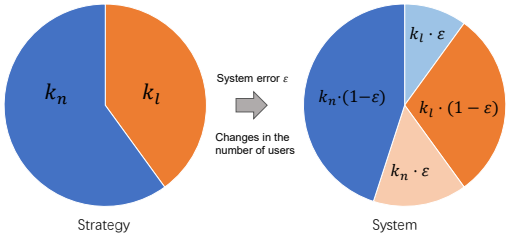


Fig. 2. An illustration of how the system error influences the users' decision.

Among the k neighbor's of a focal user, assume k_l of them use strategy S_l and flip their observation results, and $k_n = k - k_l$ them choose strategy S_n and report their observations to the fusion center. Given the system error probability ε, let k_n^S be the number of neighbors whose reported value are the same as the true system state, and $k_l^S = k - k_n^S$ is the number of neighbors whose reported values are different from the true system state. Then, we have:

$$k_l^S = (1-\varepsilon)k_l + \varepsilon k_n, \tag{1}$$

$$k_n^S = (1-\varepsilon)k_n + \varepsilon k_l. \tag{2}$$

Fitness: Generally, the evolutionary game theory defines users' fitness as follows:

$$\pi = (1-\alpha)B + \alpha U, \tag{3}$$

where B is the baseline fitness, and we let $B = 1$ in our work. α is a weak selection coefficient. In the literature of graphical evolutionary game theory [16]–[18], α is usually considered to be very small and we also make this assumption in our work. U is the payoff matrix quantifying the payoff users receive by interacting with their neighbors. In our work, we assume that users do not know their neighbors' adopted strategies but can observe their neighbors' reported values. Depending on whether their reported values are the same, they receive different payoffs as shown below.

$$\begin{array}{c} \\ S_l \\ S_n \end{array} \begin{array}{c} \begin{array}{cc} S_s & S_d \end{array} \\ \begin{pmatrix} u_{ls} & u_{ld} \\ u_{ns} & u_{nd} \end{pmatrix} \end{array}. \tag{4}$$

In Eq. (4), at epoch t, when user i adopts strategy S_l and $r_i^t = s_i^t$, if neighbor j's reported value is the same as his/her flipped observation, that is, $r_i^t = r_j^t$, then user i receives payoff u_{ls} during this interaction with user j; while when $r_i^t \neq r_j^t$ user i receives payoff u_{ld} during this interaction. Similarly, when user i adopts strategy S_n and reports the original observation, he/she receives payoff u_{ns} and u_{nd} when $r_i^t = r_j^t$ and $r_i^t \neq r_j^t$, respectively. At this time, the user receives the payoff u_{ls}; Similarly, S_d represents the neighbor is different from its own, and the payoff received by the user is u_{ld}; it is similar when the user adopts the S_n strategy.

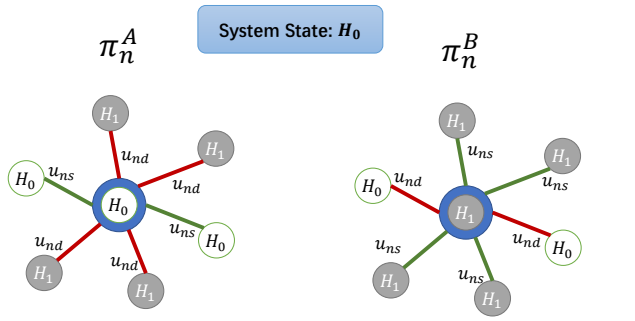


Fig. 3. Calculation of the fitness π in two scenarios (with or without system error).

Given the above definition, the next step is to define the fitness function. Note that users do not know whether their observations include system errors. Therefore, we first consider the scenario where user i's observation is error free and is the same as the true system state, that is, $u_i^t = \theta^t$. Therefore, if user i adopts strategy S_l and $r_i^t = u_i^t \neq \theta^t$, then user i receives payoff u_{ld} when interacting with each of the k_n^S neighbors whose reported values are the same as θ^t, and user i receives payoff u_{ls} when interacting with each of the k_l^S neighbors whose reported values are different from θ^t. Therefore, user i's fitness is Eq. (5). Similarly, when user i adopts strategy S_n and reports the original observation with $r_i^t = u_i^t = \theta^t$, and his/her fitness is Eq. (6). In the second scenario, user i's observation includes error, and $u_i^t = \bar{\theta}^t$. Using the same analysis as above, user i's fitness when adopting strategy S_l and S_n are eq. (7), and eq. (8), respectively.

Therefore, we divide the situation into two types according to whether system error occurs, and calculate the fitness of users who adopt the strategy S_l and the strategy S_n when system errors exist and the fitness of users who adopt the strategy S_l and the strategy S_n when the observation is error-free respectively. The above process can be derived as follows:

- **Scenario A: observation is correct**

$$\pi_l^A = 1 - \alpha + \alpha\left[k_l^S u_{ls} + \left(k - k_l^S\right) u_{ld}\right], \tag{5}$$

$$\pi_n^A = 1 - \alpha + \alpha\left[k_n^S u_{ns} + \left(k - k_n^S\right) u_{nd}\right], \tag{6}$$

- **Scenario B: observation is incorrect**

$$\pi_l^B = 1 - \alpha + \alpha\left[k_n^S u_{ls} + \left(k - k_n^S\right) u_{ld}\right], \tag{7}$$

$$\pi_n^B = 1 - \alpha + \alpha\left[k_l^S u_{ns} + \left(k - k_l^S\right) u_{nd}\right]. \tag{8}$$

Strategy Update Rule: In the long iteration process, ordinary users may be affected by their neighbors to update

their strategies. In evolutionary game theory, there are three most prevalent strategy update rules, namely birth-death (BD), death-birth (DB), and imitation (IM). Same as [16], we adopt the Death-Birth update rule and adjust it to our scenario. For DB strategy update rule, a random player is chosen to abandon his/her current strategy (Death process). Then, the chosen player adopts one of his/her neighbors' strategies with the probability being proportional to their fitness (Birth process). In this work, users can only observe others' reported values but not their strategies. Therefore, in our research, each user can only infer the strategies adopted by others through comprehensively considering the reports of others and their own observations. The specific details of this process will be elaborated in Section III. The other update rules are similar and omitted here. And analysis of the other update rules are similar and omitted here.

ESS: ESS is defined as an evolutionary stable state [19]. After the evolutionary process reaches ESS, even if some mutant populations appear (mutants can be understood as decision-makers taking new different strategies), the system can automatically eliminate these small disturbances and return to the stable state. At the ESS, the evolution dynamics satisfy $\dot{p}_l = 0$, that is, the proportion of ordinary users with strategy S_l does not change. Let (p_l^*, p_n^*) be the percentage of users adopting strategy S_l and S_n, respectively, at the ESS.

III. Evolutionary Dynamics of the User Network with Byzantines

In this section, we will find the dynamics of network strategy proportion p_l and the corresponding evolutionary stable states (ESS). The obtained evolutionary dynamic equation and ESS link the user's strategy-making process and the final evolutionary stable state with the user's payoff matrix, system error, and proportion of Byzantines.

The study of the dynamic evolution process in this section is based on the following two assumptions: (a) each user does not know whether other users are Byzantines and (b) the user knows all of his/her neighbors' previous reports.

At each epoch during the evolution process, An ordinary user is randomly selected from the network as the focal user to update the strategy. According to the DB update rule, the focal user will adopt the strategy of its neighbors, and the probability of adopting is proportional to the user fitness. However, since the user does not know whether there is an observation error, it has no way of knowing whether the neighbor has an observation error. What needs to be clarified is that the user's fitness calculation is performed locally based on his/her observations. Specifically, each user believes that his/her observation is definitely correct and uses it as a reference to calculate the fitness of their neighbors. In the fitness calculation process of the focal user, the neighbors whose reports are consistent with his/her own observation results are considered to have adopted the strategy S_n, and neighbors whose reports are different are considered to have adopted strategy S_l.

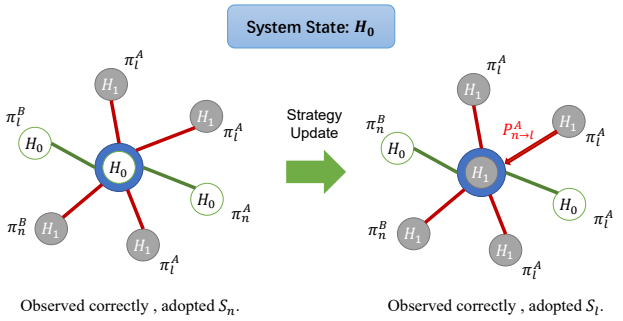


Fig. 4. An example of strategy updating process for the central user who has a correct observation and adopts strategy S_l.

According to the DB update rule, the probability that the central user changes his/her strategy from S_n to S_l is as follows:

- **Scenario A: Focal user has no observation error**

$$P_{n\to l}^{A} = \frac{k_l(1-\varepsilon)\cdot\pi_l^A + k_n\varepsilon\cdot\pi_n^B}{\left[k_n(1-\varepsilon)\cdot\pi_n^A + \varepsilon k_l\cdot\pi_l^B\right] + \left[k_l(1-\varepsilon)\cdot\pi_l^A + \varepsilon k_n\cdot\pi_n^B\right]}. \tag{9}$$

- **Scenario B: Focal user has observation error**

$$P_{n\to l}^{B} = \frac{k_n(1-\varepsilon)\cdot\pi_n^A + k_l\varepsilon\cdot\pi_l^B}{\left[k_n(1-\varepsilon)\cdot\pi_n^A + \varepsilon k_l\cdot\pi_l^B\right] + \left[k_l(1-\varepsilon)\cdot\pi_{l1} + \varepsilon k_n\cdot\pi_n^B\right]}. \tag{10}$$

Combining Eq. (9) and Eq. (10) yields the probability that any ordinary user in the network changes strategy from S_n to S_l , which is shown as Eq. (11) at the bottom of next page. Note that in Eq. (9) and Eq. (10), we need to know k_l and k_n, the number of neighbors adopting strategy S_l and S_n, respectively. In the following, we will study how to analyze k_l and k_n.

Assume an ordinary user has k ordinary neighbors and k_{bl} byzantine neighbors. From Section II, we assume that the ratio of byzantines to ordinary users are β, and we assume that byzantine users are uniformly distributed throughout the entire network. Therefore, we use the approximation $k_{bl} = \frac{\beta}{1+\beta}k$ in the following analysis. Among the k ordinary neighbors, k_{ol} of them adopt strategy S_l and the rest $k_{on} = k - k_{ol}$ adopt strategy S_n. Note that p_l is the percentage of ordinary users in the network adopting strategy S_l. Therefore, given k, k_{ol} is a binomial random variable with probability mass function

$$\theta\left(k, k_{ol}\right) = \binom{k}{k_{ol}} p_l^{k_{ol}} \left(1-p_l\right)^{k-k_{ol}}. \tag{12}$$

In summary, among the k ordinary neighbors and k_{bl} byzantines, $k_l = k_{ol} + k_{bl}$ of them adopt strategy S_l and $k_n = k_{on} = k - k_{ol}$ of them adopt strategy S_n.

Summarizing the above analysis, we can get $E\left[P_{n\to l}\left(k, k_{ol}\right)\right]$, the expected probability of ordinary user's strategy changing from S_n to S_l. At this time, the

number of ordinary users who adopt the strategy S_l will increase by 1, and p_l will increase by $1/N$, which happens with probability:

$$\mathbb{P}\left(\Delta p_l = \frac{1}{N}\right) = (1-p_l)E\left[P_{n\to l}\left(k,k_l\right)\right], \quad (13)$$

where Δ indicates the increment. With a similar argument as above, one can compute the probability that a user changes its strategy from S_l to S_n. We thus obtain:

$$\mathbb{P}\left(\Delta p_l = -\frac{1}{N}\right) = p_l(1-E\left[P_{n\to l}\left(k,k_l\right)\right]), \quad (14)$$

Summarizing Eq. (13) and Eq. (14), we deduce the dynamic of p_l:

$$\dot{p}_l = -\frac{1}{N}\mathbb{P}\left(\Delta p_l = -\frac{1}{N}\right) + \frac{1}{N}\mathbb{P}\left(\Delta p_l = \frac{1}{N}\right). \quad (15)$$

Hence, substituting the expressions of users' fitness π in Eqs. (5) - (8), Eq. (11) and Eqs. (13) - (14) into Eq. (15), we let $\dot{p}_l = 0$ to get the proportion of ordinary users who use the strategy S_l when the user network is in a stable state (ESS). In other words, we get the proportion of ordinary users who are affected by Byzantines. Due to the influence of group effects, their behavior made them unconsciously become "Byzantines". The above process is calculated using Wolfram Mathematica 12.1 software, and the analytical solution is very complex and cannot be displayed in the paper. Therefore, we will mainly analyze the numerical solution of this part in the simulation section of this paper.

IV. Decision Fusion in the Presence of Byzantines and Herd Behavior

In this section, based on the evolutionary game theoretic study of how byzantines influence ordinary users in the previous section, we propose the optimal fusion strategy.

A. The Optimum Decision Rule

The optimum decision rule by adopting a maximum the posterior probability criterion has been proposed in [9], assuming that ordinary users will always honest report their observations with $r_i^t = u_i^t$ and byzantines flip their observations with probability P_{mal}. the ordinary users will always honestly report the observed system status. For the honest nodes $r_i^t = u_i^t$ always holds, and for malicious nodes, they flip u_i^t with a certain probability P_{mal} . Given the received reports vector $\boldsymbol{R}^t = (r_1^t, r_2^t, r_3^t ... r_n^t)$, the optimum decision fusion results $\boldsymbol{\theta^{t,*}}$ in [9] minimizing the error probability is shown as follows:

$$\boldsymbol{\theta^{t,*}} = \arg\max_{\theta^t} P\left(\boldsymbol{\theta^t}|\boldsymbol{R}^t\right), \quad (16)$$

By applying Bayes rule and using the fact that all state sequences have equal probabilities, we get:

$$\boldsymbol{\theta^{t,*}} = \arg\max_{\theta^t} P\left(\boldsymbol{R}^t|\boldsymbol{\theta^t}\right), \quad (17)$$

Similar to the work in [9], let $\boldsymbol{\xi_n} = (\xi_1, \xi_2 ... \xi_n)$ be a binary random sequence in which $\xi_i = 0$ if node i is an ordinary user, and $\xi_i = 1$ when user i is a byzantine user, and let $P(\xi)$ be the probability distribution of byzantines across the entire network. Using the same method in [9], the optimal $\boldsymbol{\theta^{t,*}}$ given ξ_i and θ^j is:

$$\boldsymbol{\theta^{t,*}} = \arg\max_{\theta^t} \sum_{\xi^n} \left(\prod_{i=1}^{n} \prod_{j=1}^{t} P\left(r_i^j|\xi_i,\theta^j\right) \right) P\left(\xi^n\right), \quad (18)$$

which will be used in our future analysis.

B. Measuring the Hazard of Byzantines

In our work, Byzantines can not only attack the decision fusion process by directly submitting false reports but also indirectly affect the fusion process by influencing ordinary users. Therefore, we use the model in Section III to predict and measure the hazard of Byzantines. After that, we utilize the expected proportion of ordinary users who are affected by Byzantines p_l to recalculate the optimum decision rule that minimizes the fusion error probability.

First of all, we calculate the direct hazard of the Byzantines, and the probability δ that the FC receives a wrong report is:

$$\delta = \varepsilon\left(1 - P_{mal}\right) + (1-\varepsilon)P_{mal}. \quad (19)$$

Note that with the existence of herd effects, we have calculated the proportion of ordinary users affected by Byzantines in ESS, which is the solution of the model in the previous section. We define this proportion as p_l, that is, the proportion of ordinary users who will flip the observation report. Therefore, the probability that the FC receives a wrong report from an ordinary user is:

$$\gamma = (1-\varepsilon)\cdot p_l + \varepsilon\cdot(1-p_l). \quad (20)$$

From the above analysis, we can conclude that the value of $P\left(r_i^j|\xi_i,\theta^j\right)$ can be divided into the following four types according to ξ_i and θ^j:

$$P\left(r_i^j|\xi_i,\theta^j\right) = \begin{cases} \gamma & \text{Ordinary users, with error} \\ 1-\gamma & \text{Ordinary users, error free} \\ \delta & \text{Byzantine users, with error} \\ 1-\delta & \text{Byzantine users, error free,} \end{cases} \quad (21)$$

$$P_{n\to l} = (1-\varepsilon)\cdot P_{n\to l}^A + \varepsilon\cdot P_{n\to l}^B = \frac{\varepsilon\cdot\left[k_n(1-\varepsilon)\cdot\pi_n^A + k_l\varepsilon\cdot\pi_l^B\right] + (1-\varepsilon)\cdot\left[k_l(1-\varepsilon)\cdot\pi_{l1} + k_n\varepsilon\cdot\pi_n^B\right]}{\left[k_n(1-\varepsilon)\cdot\pi_n^A + \varepsilon k_l\cdot\pi_l^B\right] + \left[k_l(1-\varepsilon)\cdot\pi_{l1} + \varepsilon k_n\cdot\pi_n^B\right]}. \quad (11)$$

Substituting (21) into (18), the optimum decision rule in this scenario can be written as:

$$\boldsymbol{\theta}^{t,*} = \arg\max_{\theta^t} \sum_{\xi^n} \left(\prod_{i:\xi_i=0} (1-\gamma)^{t_{eq(i)}} \gamma^{t-t_{eq(i)}} \prod_{i:\xi_i=1} (1-\delta)^{t_{eq(i)}} \delta^{t-t_{eq(i)}} \right) P(\xi^n). \tag{22}$$

where $t_{eq(i)}$ is the number of j's for which $r_i^j = \theta^j$. In (22), $\prod_{i:\xi_i=0}(1-\gamma)^{t_{eq(i)}}\gamma^{t-t_{eq(i)}}$ corresponds to ordinary users, and $\prod_{i:\xi_i=1}(1-\delta)^{t_{eq(i)}}\delta^{t-t_{eq(i)}}$ corresponds Byzantines.

It can be observed from the above equation that the complexity of this problem grows exponentially with the number of states of the system t, so the decision fusion will be limited by the number of the state of the system.

To simply the analysis, same as in [9], we consider the distribution of different Byzantines and find the optimal fusion strategy accordingly, as shown below.

C. Addressing Different Distribution of Byzantines

Similar to the work in [9], in this work, we use "distribution of byzantines" to refer to the prior knowledge of bzyantines known to the fusion center. Here, two cases are considered in our work. In the first one, the FC knows the expected fraction of byzantines users in the network. In the second case, the FC knows only an upper bound of the number of Byzantines.

Maximum Entropy With Given $E[\beta]$: In the first case, a simple distribution is considered that the ratio of ordinary users to Byzantines is known to FC. Then the probability that a user is byzantine is $\frac{\beta}{1+\beta}$ and these probabilities for different users are independent of each other. Therefore, eq. (22) can be rewritten as:

$$\boldsymbol{\theta}^{t,*} = \arg\max_{\theta^t} \prod_{i=1}^{n} [\frac{1}{1+\beta}(1-\gamma)^{t_{eq(i)}}\gamma^{t-t_{eq(i)}} + \frac{\beta}{1+\beta}(1-\delta)^{t_{eq(i)}}\delta^{t-t_{eq(i)}}]. \tag{23}$$

Here, as we assume that the probability of each node being a byzantine is independent of other nodes', the complexity of the algorithm only increases linearly with the number of nodes n.

Maximum Entropy With Given Upper Bound on the Possible Number of Byzantines: In many realistic scenarios, we do not know the number or proportion of malicious users in the network, but we can roughly estimate the upper limit of malicious users. Therefore, in the second case, we focus on analyzing this scenario.

In this case, we assume the FC knows that the number of Byzantines N_B will be lower than a certain upper bound h (eg: $h = N/2$). Since we are considering the maximum entropy distribution, in the fusion process we assume that all possible values of N_B with $N_B < h$ are equally likely. To solve this problem, let $\mathbb{I}$ be the indexing set $\{1, 2...n\}$. We denote with I_k the set of all the possible k-subsets of $\mathbb{I}$. Enumerate every $I \in I_k$, and treat the users at the subscripts in I as Byzantines each time, which means a user i is byzantine if $i \in I$, ordinary otherwise. With this notation, (22) can be modified to:

$$\boldsymbol{\theta}^{t,*} = \arg\max_{\theta^t} \sum_{k=0}^{h-1} \sum_{I\in\mathcal{I}_k} \left(\prod_{i\in I} (1-\delta)^{t_{eq(i)}} \delta^{t-t_{eq(i)}} \prod_{i\in\mathcal{I}\setminus I} (1-\gamma)^{t_{eq(i)}} \gamma^{t-t_{eq(i)}} \right). \tag{24}$$

Obviously, the complexity of this algorithm grows exponentially not only with the number of system states t, but also with the number of users n, which makes the algorithm difficult to implement, so in the experiment, we used the dynamic programming algorithm in [9] to find the optimal solution of (24).

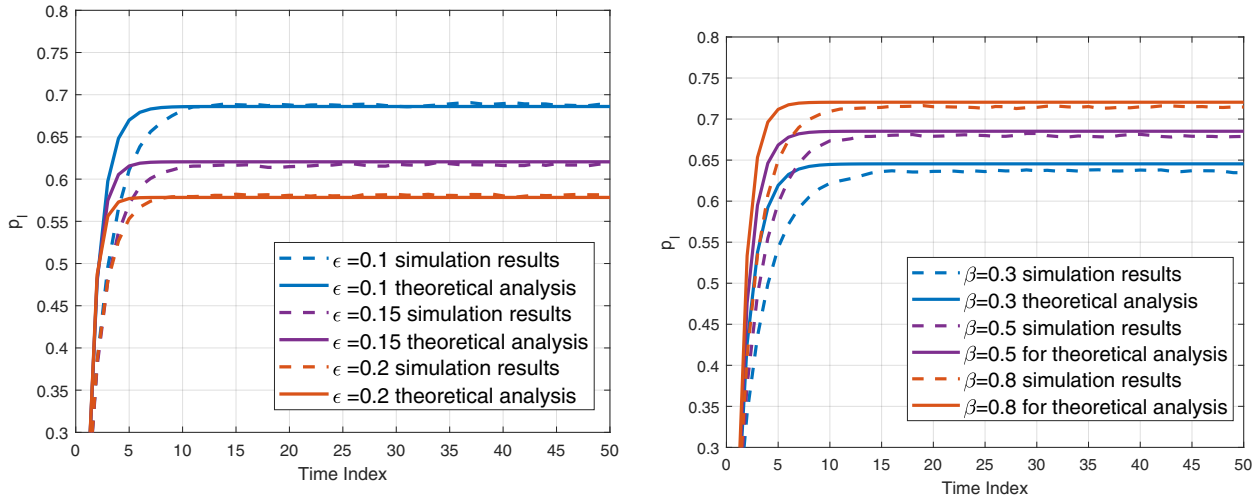
V. Simulation Results

In this section, our simulation is divided into two parts. First, we verify and analyze the theoretical analysis in Section III to measure the hazard of Byzantines. Then we use the analytical results of the proposed graph evolutionary game to perform decision fusion on a random network that takes herd behavior into consideration.

A. Evolutionary Dynamics of Byzantine Users

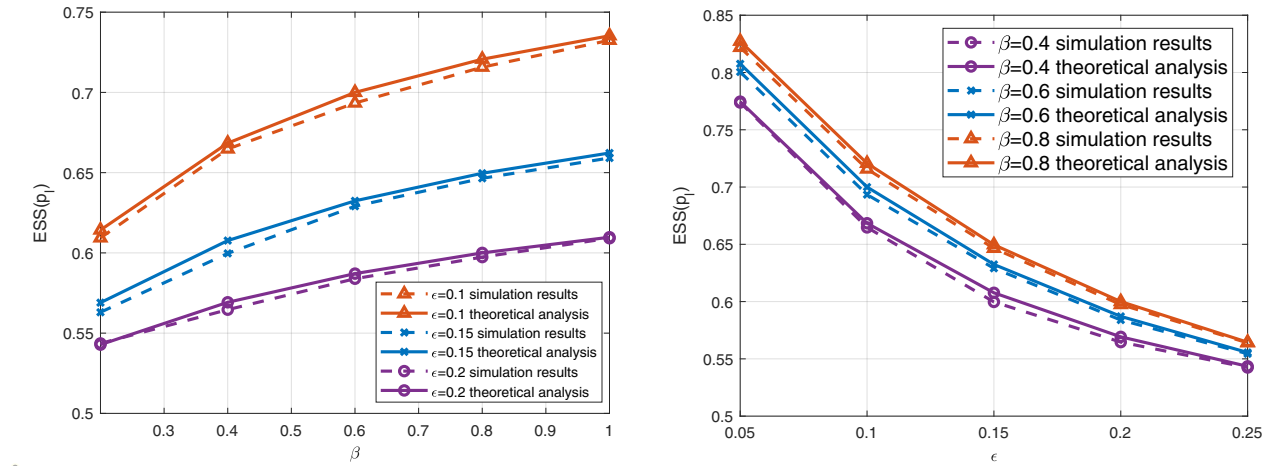
We first verify the effectiveness of the theoretical analysis of the hazards of Byzantine users through Monte Carlo simulation experiments. Two commonly used network structures are considered in the experiment: uniform degree (regular) network and the Barabási-Albert (BA) scale-free network. The parameters set in our simulations are as follows: the size of the network is 1000, the degree for regular networks and average degree for scale free networks are $k = 10$, and the weak selection coefficient α is 0.0001. The payoff matrix is set to $u_{ns} = 0.8, u_{nn} = 0.6, u_{ls} = 0.6, u_{ln} = 0.4$. And the initial proportion is $p_l = 0.2$. For each type of network, 5 graphs are randomly generated, and 96 simulation runs are conducted for each graph. Besides, the number of generations for graphical EGT is set to 300.

Fig. 5 shows the evolutionary dynamics of p_l on the BA network where all users use the DB update rule. Our simulation results show that the network structure does not affect our simulation results on p_l, and thus, we omit the results on regular networks here. We can see that the theoretical results can fit well with simulation results with different experimental parameter settings, and the number of ordinary users adopting the S_l strategy gradually increases to a stable value (ESS) with time due to the influence of byzantine users. Besides, we evaluate the performance under different parameters on networks. Fig. 5(a) shows that as the system observation error increases, the number of users who adopt the two strategies in the group tends to be similar, that is, the proportion of users who adopt the S_l strategy tends to 0.5. Fig. 5(b) shows that as the number of Byzantines increases, the number of users adopting S_l strategies in the group will also increase. These findings are also shown in Figure 6.



(a) $\beta = 0.5$ (b) $\varepsilon = 0.1$

Fig. 5. The evolutionary dynamics of p_l on BA scale-free networks ($\bar{k} = 10$) with (a) different system errors $\varepsilon = 0.1, 0.15, 0.20$, and (b) different percentages of byzantines $\beta = 0.3, 0.5, 0.8$.



(a) (b)

Fig. 6. The ESS (p_l^*) on BA scale-free networks ($\bar{k} = 10$) with (a) different system errors $\varepsilon = 0.1, 0.15, 0.20$, and (b) different percentages of byzantines $\beta = 0.4, 0.6, 0.8$.

Then we evaluate the ESS under different parameters on networks. Fig. 6(a), 6(b) show the impact of the system error rates and the proportion of Byzantines on the ordinary user group, respectively. From Fig. 6(a), we can see that as the system error rate increases, the proportion of ordinary users influenced by byzantines (p_l^*) gradually approaches 0.5. This is because when the system error rate is too large, users can no longer make any meaningful decisions and thus adopt a strategy similar to "tossing a coin". Then, from Fig. 6(b), it can be seen that when the system error rate is small, the number of Byzantines has a greater impact on the network and vice versa, which is also consistent with the conclusions of previous experiments. Besides, through experiments, we also find that the network structure has little effect on the model.

In addition, the experimental results show that Byzantines are very harmful to the network. When the system error is small, only a few Byzantines are needed to affect the majority of ordinary users, which is also consistent with the conclusion in [13]. Meanwhile, the increase in system error rate will reduce this phenomenon, but the proportion of ordinary users will still be affected by more than 50%, so it is valuable to study effective methods to resist byzantine attacks.

B. Decision Fusion with Herd Behavior

After validating the correctness of our analysis in our model in Section III, we further use this result to verify the performance of the proposed decision fusion algorithm.

In the presence of herd behavior, we first compare the accuracy of the original fusion strategy with our newly proposed fusion strategy. Similar to the previous experiment, we first compare the impact of different system error rates on algorithm performance under the same proportion of Byzantines, as shown in Table I. Here, we still run 300 simulations each time, but the fusion time window t is 10, and the average value of the fusion accuracy rate is obtained by repeating 1000 trials in the Monte Carlo method. Previous works have either shown that $P_{mal} = 1$ is a dominant strategy [20], [21], so we similarly set parameters $P_{mal} = 1$ like this. Since it was found that the graph structure has little effect on the network, so for simplicity, we use a random uniform network to perform fusion experiments. In addition, we consider two scenarios in this work: in scenario 1 (SC1) ordinary users will only submit honest reports and scenario 2 (SC2) is the main study object of this paper, where ordinary users will be affected by byzantine users to flip reports. Comparing the accuracy of the OPT method in [9] and after the scene change, we prove the effectiveness of the proposed EGT-DFB method proposed in this paper.

TABLE I
ACCURACY OF THE FUSION ALGORITHMS IN TWO SCENARIOS WITH $\varepsilon = 0.1$

Method	Scenario \ β	0.3	0.5	0.7	0.9
OPT [9]	SC1	0.9985	0.9982	0.9981	0.9980
OPT [9]	SC2	0.0073	**0.0033**	**0.0033**	**0.0033**
EGT-DFB	SC2	0.9953	0.9967	0.9967	0.9968

TABLE II
ACCURACY OF THE FUSION ALGORITHMS IN TWO SCENARIOS WITH $\varepsilon = 0.2$

Method	Scenario \ β	0.3	0.5	0.7	0.9
OPT [9]	SC1	0.9985	0.9985	0.9983	**0.9268**
OPT [9]	SC2	0.0145	0.0046	0.0053	0.0033
EGT-DFB	SC2	0.9940	0.9959	0.9964	**0.9967**

From Table I and Table II, we can see that the original method (OPT) proposed in [9] had remarkable fusion accuracy when ordinary users were not influnced by others and submit their observations independently in scenario (SC1), but it fails to get accurate estimates of the system states when users can see others' reported values and when they influence each other's decision. The proposed fusion strategy (EGT-DFB) can still achieve high estimation accuracy even when byzantines may greatly influence others' decisions. under the situation of ordinary users being affected. The worst case when $\varepsilon = 0.1$ fusion accuracy rate is only $3.3e-3$. When ε is larger in TableII, our model even has a higher fusion accuracy than the original model in the original scene when $\beta = 0.9$. The results show that although Byzantines can be very detrimental to the network through herd behavior, we can still predict the dynamic changes of the network through our graphical EGT model and achieve better decision fusion results.

TABLE III
ACCURACY OF THE FUSION ALGORITHMS IN TWO SCENARIOS WITH $\beta = 0.3$

Method	Scenario \ ε	0.05	0.10	0.15	0.20
OPT [9]	SC1	0.9985	0.9985	0.9982	0.9985
OPT [9]	SC2	0.0093	0.0037	0.0071	0.0145
EGT-DFB	SC2	0.9904	0.9953	0.9926	0.9949

Under the same experimental settings, we explored the performance of the algorithm under different system error rates, and the results are shown in Table III. From Table III we can see that the system error rate has almost no effect on the fusion accuracy. In addition, as the number of Byzantines increases, the accuracy rate of the proposed fusion rules also increases. Then, we conduct experiments on different prior knowledge of Byzantine distribution whose results are shown in Table IV.

TABLE IV
ACCURACY OF THE FUSION RESULTS WITH DIFFERENT DISTRIBUTION OF BYZANTINES

Byzantines	h	30%			50%		
Distribution	β	0.2	0.3	0.4	0.3	0.5	0.8
Known upper bound		0.9998	1.0000	1.0000	0.9998	1.0000	1.0000
Known Byzantine ratio		0.9892	0.9953	0.9956	0.9953	0.9967	0.9967

It can be seen from Table IV that when the FC has some prior knowledge of Byzantines, such as the upper limit of the Byzantine population as a percentage of the total population $h \geq \frac{\beta}{1+\beta}100\%$, the accuracy of decision fusion is higher than before. Particularly, for the case where the upper limit of the number of Byzantines is known, the closer the known upper limit is to the real number, the higher the accuracy. In addition, the algorithm performs better when the upper limit of the number is known than the specific ratio of Byzantines is known. This is because when we only know the upper limit, all possibilities are considered in our fusion strategy, but this also greatly increases the computational complexity. Overall, the fusion mechanism can achieve a well fusion effect in both cases.

VI. CONCLUSIONS

In this paper, we delved into a new scenario of decision fusion, where ordinary users will be affected by Byzantines due to the existence of herd behavior phenomena and may use the same strategy as the Byzantines. We propose to utilize graphical EGT to analyze the user's behavior and measure the hazard impact of Byzantines. Then we derive the evolution dynamics and the corresponding evolutionary stable states (ESSs). Next, we propose an effective fusion mechanism for

the FC based on the prediction results of the graphical EGT model and a decision fusion method based on the maximum a posterior criterion. Our simulation results show that the graphical EGT model can predict the number of lies in a group and the hazards of Byzantines. In addition, we show that our proposed fusion strategy can effectively resist byzantine attacks even when malicious users may greatly influence others' decisions.

REFERENCES

[1] A. Vempaty, L. Tong, and P. Varshney, "Distributed inference with byzantine data: State-of-the-art review on data falsification attacks," *Signal Processing Magazine, IEEE*, vol. 30, pp. 65–75, 09 2013.

[2] Z. Chair and P. K. Varshney, "Optimal data fusion in multiple sensor detection systems," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-22, no. 1, pp. 98–101, 1986.

[3] P. K. Varshney, *Distributed detection and data fusion*. Springer Science & Business Media, 2012.

[4] S. Marano, V. Matta, and L. Tong, "Distributed detection in the presence of byzantine attacks," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 16–29, 2009.

[5] A. S. Rawat, P. Anand, H. Chen, and P. K. Varshney, "Collaborative spectrum sensing in the presence of byzantine attacks in cognitive radio networks," *IEEE Transactions on Signal Processing*, vol. 59, no. 2, pp. 774–786, 2011.

[6] A. Vempaty, K. Agrawal, P. Varshney, and H. Chen, "Adaptive learning of byzantines' behavior in cooperative spectrum sensing," in *2011 IEEE wireless communications and networking conference*. IEEE, 2011, pp. 1310–1315.

[7] A. Abrardo, M. Barni, K. Kallas, and B. Tondi, "Decision fusion with corrupted reports in multi-sensor networks: A game-theoretic approach," in *53rd IEEE Conference on Decision and Control*, 2014, pp. 505–510.

[8] B. Kailkhura, S. Brahma, Y. S. Han, and P. K. Varshney, "Optimal distributed detection in the presence of byzantines," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 2925–2929.

[9] A. Abrardo, M. Barni, K. Kallas, and B. Tondi, "A game-theoretic framework for optimum decision fusion in the presence of byzantines," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 6, pp. 1333–1345, 2016.

[10] R. M. Raafat, N. Chater, and C. Frith, "Herding in humans," *Trends in Cognitive Sciences*, vol. 13, no. 10, pp. 0–428, 2009.

[11] C. Jiang, Y. Chen, and K. J. R. Liu, "Modeling information diffusion dynamics over social networks," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 1095–1099.

[12] C. Jiang, Y. Chen, and K. R. Liu, "Graphical evolutionary game for information diffusion over social networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 4, pp. 524–536, 2014.

[13] H. Zhang, Y. Li, Y. Hu, Y. Chen, and H. V. Zhao, "Measuring the hazard of malicious nodes in information diffusion over social networks," in *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2019, pp. 476–481.

[14] A. Mohd Ibrahim, I. Venkat, and P. De Wilde, "The impact of potential crowd behaviours on emergency evacuation: An evolutionary game theoretic approach," *Journal of Artificial Societies and Social Simulation, The*, vol. 22, 01 2019.

[15] C. Li, P. Lv, D. Manocha, H. Wang, Y. Li, B. Zhou, and M. Xu, "Acsee: Antagonistic crowd simulation model with emotional contagion and evolutionary game theory," *IEEE Transactions on Affective Computing*, pp. 1–1, 2019.

[16] X. Cao, Y. Chen, C. Jiang, and K. J. Ray Liu, "Evolutionary information diffusion over heterogeneous social networks," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 2, no. 4, pp. 595–610, 2016.

[17] C. Jiang, Y. Chen, and K. J. R. Liu, "Evolutionary dynamics of information diffusion over social networks," *IEEE Transactions on Signal Processing*, vol. 62, no. 17, pp. 4573–4586, 2014.

[18] Y. Li, Y. Li, H. Hu, H. V. Zhao, and Y. Chen, "Graphical evolutionary game theoretic analysis of super users in information diffusion," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 5650–5654.

[19] P. D. Taylor and L. B. Jonker, "Evolutionary stable strategies and game dynamics," *Mathematical biosciences*, vol. 40, no. 1-2, pp. 145–156, 1978.

[20] A. S. Rawat, P. Anand, H. Chen, and P. K. Varshney, "Collaborative spectrum sensing in the presence of byzantine attacks in cognitive radio networks," *IEEE Transactions on Signal Processing*, vol. 59, no. 2, pp. 774–786, 2010.

[21] B. Kailkhura, S. Brahma, Y. S. Han, and P. K. Varshney, "Optimal distributed detection in the presence of byzantines," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013, pp. 2925–2929.