APSIPA Transactions on Signal and Information Processing, 2022, 11, e37

Original Paper

Integrating Human Decisions in the Presence of Byzantines: An Evolutionary Game Theoretical Approach

Yiqing Lin[1]*, Hong Hu[1], H. Vicky Zhao[1] and Yan Chen[2]

[1] *Department of Automation, Tsinghua University, Beijing, China*

[2] *School of Cyberspace Security, University of Science and Technology of China, Hefei, Anhui, China*

ABSTRACT

It is an established fact that malicious users in networks are able to mislead other users since the presence of herding behaviors, which will further amplify the hazards of these malicious behaviors. 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 attention. In this paper, we propose an evolutionary game theoretical framework to model the human decision making process, which is based on the statistical signal processing framework. Specifically, we derive the analytical formulation of the evolutionary dynamics and the corresponding numerical evolutionary stable states, which can be utilized to infer the hazard of Byzantines on the network. Based on the above model and the Markov nature of the evolutionary dynamics, the fusion mechanism with maximum a posteriori estimation is proposed. Finally, simulation experiments are conducted to analyze the performance of the proposed human decision-

*Corresponding author: Yiqing Lin, linyq20@mails.tsinghua.edu.cn. This work is supported in part by Tsinghua-Toyota Joint Research Fund.

Received 01 June 2022; Revised 07 September 2022
ISSN 2048-7703; DOI 10.1561/116.00000035

making model and the effectiveness of the fusion mechanism under a variety of parameter settings.

Keywords: Adversarial signal processing, Decision fusion, Byzantine nodes, Graphical evolutionary game theory.

1 Introduction

Decision fusion in the presence of malicious users, often referred to as Byzantines [26], has been a classical signal processing problem. In most scenarios, the users on a multi-sensor network are required to make a local binary decision report based on his/her observations and send it to the fusion center. The fusion center fuses these reports according to some preferred fusion rule to minimize the fusion error probability, while the byzantine users deliberately provide false information to mislead the center. This scenario is first abstracted from distributed spectrum sensing in cognitive radio networks [5], where the fusion center infers the spectrum state from the data returned by sensors.

Much attention has been drawn to decision fusion with byzantine data due to its practicality in many applications, including wireless sensor networks, environmental monitoring, multimedia forensics, etc. Traditional decision fusion problems mainly focus on a system with multiple sensors, which consists entirely of machines. However, with the rapid development of human-computer interaction technology, most systems now contain humans and are greatly influenced by human factors. In emerging applications like [7], it is often humans who become the most critical factor in decision-making, where humans may serve as “sensors” to contribute information towards an inference task. Especially due to the presence of byzantine data, the security issues of such distributed networks become increasingly important and the distributed nature of this system makes the network vulnerable to attacks. It is an established fact that Byzantines in the networks are able to mislead ordinary users with the help of herding behaviors [19], which will further amplify the hazards of these malicious behaviors. 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 fake reviews could have a heavy impact on consumers’ shopping choices, which will lead to the disruption of normal market competition and cause many undesirable consequences [22]. Another example is the presidential election in a democratic country, where voters often post their opinions or comments on some social media, and the election team often develops corresponding propaganda strategies based on the trend of public opinion. This process can be very disruptive to the election team’s decision-making if there are hackers who belong to the opposing side or just want to interfere with the election trying to post fake information and fuel the spread of rumors on the Internet. In this case, it is very significant for the election team to be able to infer the true will of the voters. Unlike simple sensors towards distributed inference, humans are subjective in their decision-making process, which leaves a gap for malicious users. Therefore, it is of great importance to model the behavior of Byzantines and figure out the optimum fusion rule.

Early research on decision fusion did not take the Byzantines into consideration, and the system local observation error is the main factor to affect the fusion results. In this case, the works in [4] and [24] determined the optimum algorithm to combine the local reports based on the Bayesian approach, which is referred to as Chair-Varshney rule [18]. This lemma is intended to maximize the detection probability under the constraint of the false alarm probability by calculating the likelihood ratio test of the two hypotheses. After starting to consider the presence of certain Byzantine users in the network, the authors in [20] modeled the interplay between the Byzantines and the fusion center as a zero-sum game. They proposed a simple but effective mitigation scheme to identify the Byzantines by collecting reports from different time windows and comprehensively analyzed the reports to assign a reputation measure, which is used to isolate Byzantines whose reputation is below a certain threshold, also is known as *Hard Isolation*. In this process, the real system status is inferred using the K-out-of-N rule. Another mitigation strategy based on an adaptive three-tier scheme was described in [25], where the observed behavior of the users was compared with the expected behavior of honest users to identify the Byzantines. Then the three-tier scheme estimated the behavioral parameters of Byzantine users, which are used in the subsequent adaptive fusion rule. In particular, the model in [25] can work for any fraction of Byzantines. But it required very long states to observe, which limited its capabilities. [2] and [1] used a game-theoretic framework to study the equilibrium point of the strategy adopted by the attacker and the defender(FC). The work in [2] proposed a *Soft Isolation* scheme to identify Byzantines based on [20], and used game theory to verify the optimal strategy for Byzantines, where the probability of lying (P_{mal}) is always equal to 1. In [12], a simplified and widely adopted version of decision fusion was considered, which is a discrete model contains only two system states 0 and 1. Besides, a decision fusion method based on the maximum posterior probability criterion was proposed in [1], and its performance with different types of prior knowledge was also analyzed. And this work considered that each user was independent and did not have a mutual influence relationship. Furthermore, in recent studies [6, 8, 28], this statistical signal processing framework has been used to study human decision making processes or some human-machine networks. For example, the work in [6] used prospect Theoretic Utility theory to consider human behaviors in the decision fusion process. And the work in [28] used this framework to describe a human-machine system that integrates human decisions with signals from physical sensors.

Previous works have all assumed that users report their observations independently and are not influenced by others' decisions. However, things can change when users are not simply sensors, but humans with autonomous consciousness. With more information flooding into the population, individual decisions now rarely depend solely on themselves, and the opinions of others are playing an increasingly dominant role. In addition, this influence is not a simple aggregation but depends on crowd interaction. At this point, the relationship between users can be represented by a network and the impact of user interaction on the final fusion result must be taken into account. But few scholars have studied the corresponding influence on the decision fusion results in this distributed detection system with Byzantines under complex networks, such as social media networks. In the real world, when there are a large number of talks about an event, an ordinary person is likely to follow the crowd rather than make independent decisions, which causes him/her to sometimes ignore their own information or preferences. This phenomenon is also known as herding behavior [19]. In other words, the originally honest users may be influenced by malicious users around and report false information despite their original conclusions being correct, thus making the decision fusion process more challenging for the Fusion Center (FC). For example, the work in [27] quantified herding effects in crowd wisdom and in [29] illustrated with real data that the herding effect exists in e-commerce evaluation systems. All of these works have shown that it is necessary to consider herding effects in distributed detection systems but it is a vacancy before our work in this paper.

To address this challenge, we study the interplay between different users and analyze its impact on the fusion center. We use graphical evolutionary game theory [10, 11] to analyze the microscopic interactions among users and to study the impact of Byzantines on other users as well as the fusion center. Graphical evolutionary game theory has been used to study herding behavior in many scenarios, such as information diffusion over social networks [30], crowd dynamic analysis in emergency evacuation [17], and antagonistic crowd behaviors in cases of serious conflict [14]. The results in this literature suggest that graphical evolutionary game theory is a powerful tool for studying the impact of Byzantines on other users' behavior in such distributed detection problems.

Our contributions include:

1. Different from all prior works in decision fusion, we consider the scenario where users may influence each other's decisions and propose a graphical evolutionary game theoretic framework to study their interactions. We analyze the evolutionary dynamics and quantify the impact of Byzantines on other users. In addition, we conducted a large number of experiments

to analyze the influence of different parameters on the steady state of the network and the influence of different attack strategies of malicious users on the network in the presence of the herding effect.

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 probability (MAP) criterion. We compute the posterior probabilities for malicious and ordinary users respectively and consider the Markov nature of the reporting process for ordinary users. We show that our proposed fusion mechanism is more effective in resisting byzantine attacks than any existing ones in the presence of the herding behavior.

A shorter conference version of this paper appeared in [16]. Most of the analysis of user behavior in the initial paper was numerical and lacked further theoretical investigations. This paper derives additional analytic formulations and revises some issues of the model accordingly. For the fusion algorithm, the initial paper approximated the values of MAP for each epoch with the results at steady state, however in this version we exploit the Markov nature of the evolutionary dynamics to obtain the exact values of these intermediate states and also obtain better fusion results. Finally, in the experiment sections, this paper analyses additional network structures (e.g. ER networks) and examines the influences of a wider range of parameters on the results, including lying probability for Byzantines, network size, and payoff matrix.

The rest of the paper is organized as follows. In Section 2, the problem studied is formalized, adopting an evolutionary game theoretical framework. In Section 3, we model the human decision making process and the corresponding evolutionary stable states (ESSs), which could predict the dynamic changes of the user's decision and measure the hazard of Byzantines. In Section 4, we propose a fusion mechanism to integrate the humans' decision towards an inference task based on the MAP criterion. We conduct simulation results in Section 5 and conclude our work in Section 6.

2 Problem Formulation

In the rest of the paper, we specify the following rules for symbolic representation: capital letters will be used to denote random variables, while lowercase letters will be used to represent corresponding instantiations, and bold in them stands for vectors (Greek letters) or matrices (English letters), while ordinary fonts are for scalars. In addition, the superscripts indicate the epoch, while the subscripts give the corresponding user's id.

2.1 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 users' network, and the fusion center. In the settings considered in our work, the system state is represented by a sequence of independent and identically distributed (i.i.d.) random variables $\mathbf{\Theta} = (\Theta^1, \Theta^2 \dots \Theta^T)$. The t-th elements of Θ^t may correspond to system states at different epochs. To simplify the problem, we assume that $\mathbf{\Theta}$ is a binary vector, which means that $\Theta^t \in \{0, 1\}$. And let all states are equiprobable $P(\Theta^t = 0) = P(\Theta^t = 1) = 0.5$, which is widely adopted in previous research [1]. And $\boldsymbol{\theta} = (\theta^1, \theta^2 \dots \theta^T)$ is a specific instantiation of $\mathbf{\Theta}$.

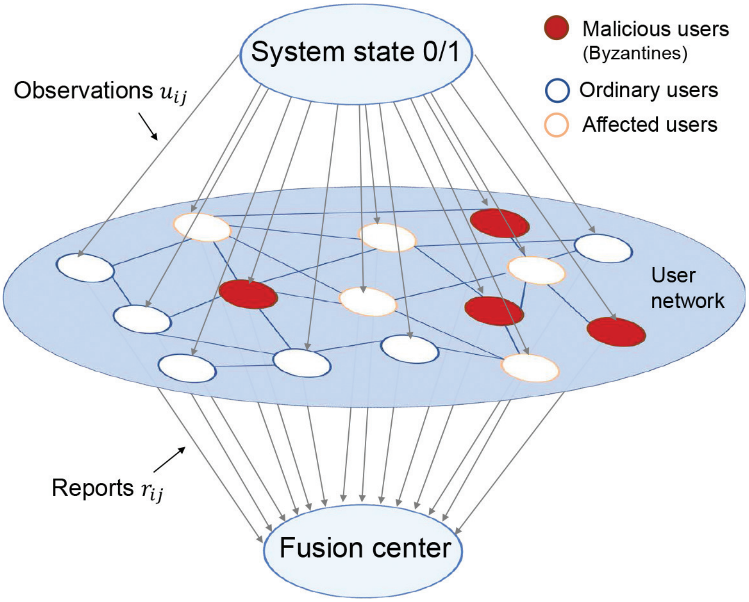


Figure 1: Decision fusion under adversarial conditions. The orange circles represent the ordinary users who are affected by the surrounding Byzantines.

As shown in Figure 1, each user in the network makes an observation of the system state at each epoch to obtain its local result, which can be expressed as an observation random matrix $\mathbf{U} = \{U_i^t\}, i = 1 \dots N, t = 1 \dots T$ and at t-th epoch the i-th users' observation random variable $U_i^t \in \{0, 1\}$. 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 $P(U_i^t \neq \Theta_i^t) = \varepsilon$, and the errors in each observation are i.i.d.

Similarly, each user returns a report to the fusion center, which can be expressed as a report random matrix $\mathbf{R} = \{R_i^t\}$ and $\mathbf{r} = \{r_i^t\}$ is one of its specific instantiations. In this step, the report returned to the center may be intentionally modified with a fixed probability $P(R_{mal}^t \neq U_{mal}^t) = P_{mal}$ 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.

Note that our model translates to many real-world scenarios, such as communities of game or movie fans. The friendships in the community correspond to our user network, while the quality of a product (recommendable or not) can be taken as the system state Θ^t. Some malicious users' comments will not only directly affect the product rating, but will also amplify it through the herding effect. In order to make the community healthy, the platform usually tries to identify and mitigate such impacts, which corresponds to the behavior of the fusion center.

2.2 Graphical Evolutionary Game Theory

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

2.2.1 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 probability attack strategy. For the convenience of math derivation, we use β to represent the ratio of Byzantines to ordinary users.

2.2.2 Strategies

Different strategies are used for ordinary users and malicious users. For ordinary users, 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. And he/she may change his/her strategy at any epoch. Let p_l^t 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 $(1 - p_l^t)$. To avoid redundancy, we abbreviate it to p_l before Section 4 because there is no discussion in the time dimension. For Byzantines, we assume that there is a fixed probability P_{mal} to adopt the S_l strategy and the malicious behaviors in each reporting process are i.i.d.

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.

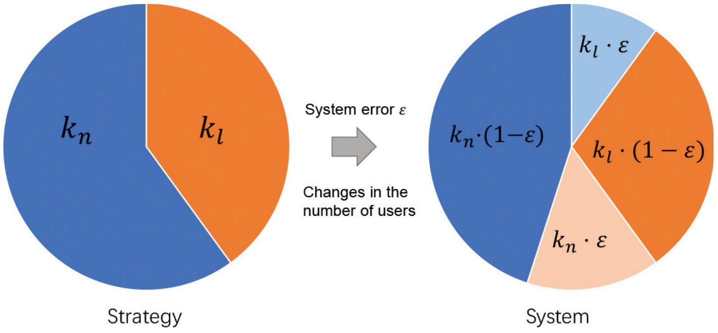


Figure 2: An illustration of how the system error influences the users' decisions.

Among the k neighbors of a focal user, assume k_l of them use strategy S_l and flip their observation results, and $k_n = k - k_l$ they 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 values 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

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

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

2.2.3 Fitness

In EGT, the fitness represents the utility of the players. The strategy of the players who have higher fitness tends to have more advantage to be adopted. Generally, the evolutionary game theory defines users' fitness as follows

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

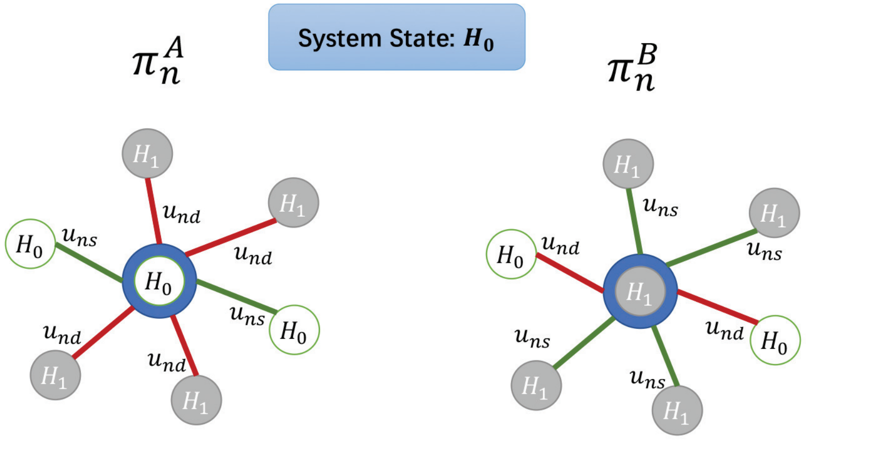
where B is the baseline fitness. The baseline fitness B represents the player's inherent property. For example, if a player is less susceptible to others, then his/her baseline fitness will also be greater. In homogeneous networks, the baseline is considered to be identical for all users, and we let $B = 1$ in our work. α is a weak selection coefficient. In the literature of graphical evolutionary game theory [3, 9, 15], 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. So here we define two symbols: S_s stands for the central user's and neighbor's report values are the same in the last epoch; on the contrary, S_d stands for different. Depending on whether their reported values are the same and the focal user's strategy adopted in the last epoch, he/she receives different payoffs as shown below

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

The evolutionary game theory is similar to the traditional game theory in the interaction of players and the getting of payoffs. In Equation (4), at epoch t, when user i adopts strategy S_l and $r_i^t = \bar{u}_i^t$, 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. Specifically, we assume that the payoffs of ordinary users must satisfy the following conditions: $u_{ns} > u_{ls}/u_{nd} > u_{ld}$ in Equation (4). The payoff matrix will be applied between the focal user and all his/her neighbors. And each of these edges will help the focal user obtains the payoff. Note that the payoff here is not limited to money, but may also be something abstract as a reputation value (number of followers, etc.). We calculate the total fitness of the user in one time slot, and update user's strategy afterwards. Considering the herding effect, we believe the benefit of keeping consistent with others without lying u_{ns} is the greatest choice for ordinary users, and the benefit u_{ld} of being inconsistent with others by lying is the smallest. And the value of u_{ls} and u_{nd} are between the above values, while the specific relation is related to the actual situation.

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 $r_i^t = \bar{u}_i^t \neq \theta^t$, 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 Equation (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 Equation (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 Equations (7) and (8), respectively.

As shown in Figure 3, assuming that the system state is H_0, there are two scenarios regarding the state observed by the user: Scenario A is where his/her observation is correct (H_0), and Scenario B is the opposite (H_1). When the focal user observes H_0 and chooses S_n, he/she will gain a payoff u_{ns} interacting with the neighbor who sends a report of H_0 and gain a payoff u_{nd} interacting with the neighbor who sends a report of H_1. By contrast, if he/she observes H_1 and chooses S_n, then the aforementioned discussion of payoff would be the opposite.



Observed correctly , adopted S_n.

Observed incorrectly and adopted S_n.

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

Therefore, we divide the scenarios 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**

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

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

2.2.4 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 [3], we adopt the Death-Birth update rule and adjust it to our scenario. For the 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 these settings, 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 on in Section 3. And the analysis of the other update rules is similar and omitted here.

2.2.5 ESS

ESS is defined as an evolutionary stable state [23]. 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.

3 Evolutionary Dynamics of the User Network with Byzantines

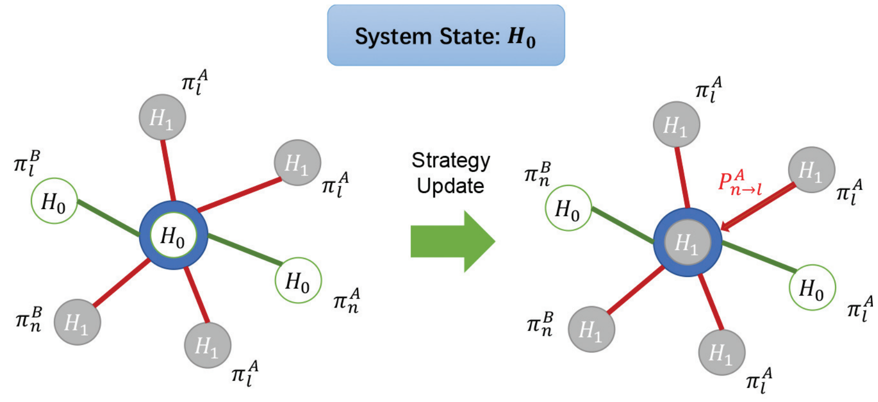
In this section, we will find the dynamics of lying strategy proportion p_l in users' network and the corresponding evolutionary stable states (ESS). The obtained evolutionary dynamic equation and ESS link the user's strategy-making process and the final evolutionary stable state with the user's payoff 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 only knows all of his/her neighbors' previous reports.

At each epoch during the evolution process, each ordinary user in the network will be 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 users' 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 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 inconsistent are considered to have adopted strategy S_l.

As shown in the left part of Figure 4, suppose the focal user has no error in his/her observation H_0 (Scenario A) at the previous epoch and adopts the S_n strategy (reported as H_0). At this point, the user who reported as H_1 in his/her neighbors will be regarded as adopting the S_l strategy by the focal user. It is possible for the focal user to switch from the S_n to the S_l strategy by the influence of these users, as shown in the right part of Figure 4. However, if the focal user has errors in his/her observation (Scenario B), those who report as H_0 in his/her view then are instead the users adopting S_l, which makes everything the opposite.

According to the DB update rule, the probability of users changing to a new strategy is proportional to the fitness of all users adopting that strategy. To be elaborated, the probability of the user strategy changing from S_n to S_l will be expressed as follows: the numerators are the sum of the fitness of the neighbors adopting S_l, and the denominators are the sum of the fitness of all neighbors. Therefore, the probability that the central user changes his/her strategy from S_n to S_l is as follows:



Observed correctly , adopted S_n.

Observed correctly , adopted S_l.

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

- **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**

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

$$\begin{aligned} P_{n\to l} &= (1-\varepsilon)\cdot P_{n\to l}^A + \varepsilon\cdot P_{n\to l}^B \\ &= \frac{\varepsilon\cdot\left[k_n(1-\varepsilon)\cdot\pi_n^A + k_l\varepsilon\cdot\pi_l^B\right] + (1-\varepsilon)\cdot\left[k_l(1-\varepsilon)\cdot\pi_l^A + k_n\varepsilon\cdot\pi_n^B\right]}{\left[k_n(1-\varepsilon)\cdot\pi_n^A + \varepsilon k_l\cdot\pi_l^B\right] + \left[k_l(1-\varepsilon)\cdot\pi_l^A + \varepsilon k_n\cdot\pi_n^B\right]} \\ &= \left(2(1-\varepsilon)\varepsilon + (1-2\varepsilon)^2\ \frac{k_l}{k}\right)\cdot\frac{1+\alpha\left(\frac{au_{ns}+bu_{nd}+cu_{ls}+du_{ld}}{(1-2\varepsilon)^2k_l+2\varepsilon(1-\varepsilon)k}-1\right)}{1+\alpha\left(\frac{eu_{ns}+fu_{nd}+gu_{ls}+hu_{ld}}{k}-1\right)} \end{aligned} \tag{11}$$

where the forms of a, b, c, d, e, f, g, h can be viewed in Equation (A.18)

Combining Equations (9) and (10), we yields the probability that any ordinary user in the network changes strategy from S_n to S_l, which is shown as Equation (11) at the bottom of the next page. And taking Equations (5)–(8) into the formula, we could simplify it to derive the second equation. Besides, in the last equation, we invoke the fact that $\frac{1+\mu x}{1+\nu x} = 1 + (\mu-\nu)x + O(x^2)$

for small x. And because α is a weak selection coefficient, which is a small quantity, we will omit the $O(x^2)$ term in the following. And we replaced some coefficients with letters to express conciseness.

Note that in Equations (9) and (10), we need to know k_l and k_n, the number of neighbors adopting strategy S_l and S_n, respectively. Assume an ordinary user has k neighbors, which contains k_o ordinary neighbors and k_b byzantine neighbors. From Section 2, we assume that the ratio of Byzantines to ordinary users is β, and we assume that byzantine users are uniformly distributed throughout the entire network. Therefore, we use the approximation $k_b = \frac{\beta}{1+\beta}k$ in the following analysis. Among the k_o ordinary neighbors, k_{ol} of them adopt strategy S_l and the rest $k_{on} = k_o - k_{ol}$ adopt strategy S_n. Note that p_l is the percentage of ordinary users adopting strategy S_l among all ordinary users. In summary, among the k_o ordinary neighbors and k_b byzantines, $k_l = k_{ol} + k_{bl} = k_{ol} + \beta k_o P_{mal}$ of them adopt strategy S_l and $k_n = k - k_l$ of them adopt strategy S_n.

Afterward, we will derive the dynamics of the proportion of liars for ordinary users based on the probability obtained above. In our model, at each epoch, all users will determine whether they need to update their strategies through all the reports they received at the last epoch. Therefore, each user may have an influence on $\dot{p_l}$. First, we calculate the contribution of the i-th ordinary user to $\dot{p_l}$. For the i-th user, the probability that he/she adopted the strategy S_n at the last epoch is $(1 - p_l)$ and the probability that he/she changes from the strategy S_n to the strategy S_l is $P_{n\to l}\left(k_{oi}, k_{oli}\right)$. Assume that there are N users in the users' network, where the number of ordinary users is N_o and the malicious user is N_b. At this time, the number of ordinary users who adopt the strategy S_l will increase by 1, and p_l will increase by $1/N_o$, which happens with the probability as follow

$$\mathbb{P}\left(\mathbf{\Delta} p_{li} = \frac{1}{N_o}\right) = (1 - p_l)P_{n\to l}\left(k_{oi}, k_{li}\right), \tag{12}$$

where $\mathbf{\Delta}$ indicates the increment. With a similar argument as above, one can compute the probability that the i-th user changes its strategy from S_l to S_n. We thus obtain

$$\mathbb{P}\left(\mathbf{\Delta} p_{li} = -\frac{1}{N_o}\right) = p_l(1 - P_{n\to l}\left(k_{oi}, k_{li}\right)), \tag{13}$$

Hence, we yield the contribution of the i-th ordinary user to expected change p_l

$$\dot{p}_{li} = \frac{1}{N_o}(1 - p_l)P_{n\to l}\left(k_{oi}, k_{li}\right) - \frac{1}{N_o}p_l\left(1 - P_{n\to l}\left(k_{oi}, k_{li}\right)\right), \tag{14}$$

Since here it is all for each epoch t, the time axis superscript t is omitted. And the dynamic of p_l is equal to the sum of all users' contributions

$$\begin{aligned} \dot{p}_l &= \sum_{i=1}^{N_o} \frac{1}{N_o}(1-p_l)P_{n\to l}\left(k_{oi},k_{li}\right) - \frac{1}{N_o}p_l\left(1-P_{n\to l}\left(k_{oi},k_{li}\right)\right), \\ &= (1-p_l)E\left[P_{n\to l}\left(k_o,k_{ol}\right)\right]) - p_l(1-E\left[P_{n\to l}\left(k_o,k_{ol}\right)\right]). \end{aligned} \tag{15}$$

where $E\left[P_{n\to l}\left(k_o,k_{ol}\right)\right])$ represents the expectation of Equation (11) which will be calculated in the following. And we assume that N_0 is sufficiently large since the number of users is usually huge for social networks.

Therefore, given k_o, k_{ol} is a binomial random variable with the probability mass function

$$\theta\left(k_o,k_{ol}\right) = \left(\begin{array}{c} k_o \\ k_{ol} \end{array}\right) p_l^{k_{ol}}\left(1-p_l\right)^{k_o-k_{ol}}. \tag{16}$$

and we have the following:

$$\begin{aligned} E\left[k_{ol}|k_o\right] &= k_o p_l, \\ E\left[k_{ol}^2|k_o\right] &= k_o^2 p_l^2 - k_o p_l^2 + k_o p_l, \\ E\left[k_{ol}^3|k_o\right] &= k_o(k_o-1)(k_o-2)p_l^3 + 3(k_o-1)k_o p_l^2 + k_o p_l. \end{aligned} \tag{17}$$

The neighbor nodes of ordinary users who adopt the strategy S_l can be divided into two types: one is the Byzantines and the number is k_{bl}; the other is the ordinary people who use the strategy S_l, and the number is k_{ol}. Obviously $k_l = k_{bl} + k_{ol}$. We assume that the number of ordinary users in its neighbors is k, and the proportion of ordinary users to Byzantines is β, thus $k_{bl} = \beta \cdot k$. Since the proportion of ordinary users using the strategy S_l is p_l in the entire network, each ordinary neighbor has a probability p_l of adopting strategy S_l.

Summarizing the above analysis, taking the expectation of Equation (11) (note that k, k_o are also random variables and we need to take the expectation of it further) , we can get the expression of $E\left[P_{n\to l}\left(k_o,k_{ol}\right)\right]$. In a non-uniform network structure, we need to analyze the network structure first, and then calculate the average value of the relevant degree parameters of the network. Substituting the expressions of $E\left[P_{n\to l}\left(k_o,k_{ol}\right)\right]$ into Equation (15), we thus obtain the expression in $\dot{p}_l$ as Equation (19) at the bottom of the next page.

It is worth noting that Equation (19) is an important conclusion of this paper, which shows the dynamic changes of the network. Under different parameter settings, this formula could directly predict the next trend of the network. Therefore, the value of p_l can be obtained by the difference formula

$$p_l^{t+1} = p_l^t + \dot{p}_l^t, \quad \forall t = 1,2\ldots \tag{18}$$

which is the proportion of liars p_l in the ordinary users over epoch t.

Through the above analysis, we have obtained the most critical parameter p_l and its dynamics in the model, which means that we could predict the harm caused by malicious users to ordinary users in the evolution process. Due to the influence of group effects, their behavior made them unconsciously become 'Byzantines'. Then 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) using Wolfram Mathematica 12.1 software. And we will mainly analyze the numerical solution of this part in the simulation section. It can be seen that the dynamics of p_l is a cubic polynomial with respect to p_l. Therefore ESS has three roots, but in this scenario, two of the roots are always imaginary numbers, so only one real root is retained. The solution of ESS is the expected proportion of users who lie in the network, which will be fixed at a certain proportion when it stabilizes over time.

After studying the impact of malicious users on the network in the presence of the herding effect, we can use the model to better study the adversarial strategies in such multi-agent systems, such as an effective fusion mechanism to integrate human decisions for fusion center.

$$
\begin{aligned}
\dot{p}_l &= (1-p_l)E\left[P_{n\to l}\left(k_o, k_{ol}\right)\right]) - p_l(1-E\left[P_{n\to l}\left(k_o, k_{ol}\right)\right]) \\
&= -p_l + 2(1-\varepsilon)\varepsilon + (1-2\varepsilon)^2\ \frac{p_l+\beta P_{mal}}{1+\beta} + \alpha(1-2\varepsilon)^2 \\
&\cdot \Bigg[\frac{A_\Delta}{(1+\beta)^3}\Big((2\overline{\frac{1}{k_o}} - 3 + \overline{k_o})p_l^3 + 3(-\overline{\frac{1}{k_o}} + \overline{k_o}\beta P_{mal} + 1 - \beta P_{mal})p_l^2 \\
&+ (\overline{\frac{1}{k_o}} + 3\overline{k_o}\beta^2 P_{mal}^2 + 3\beta P_{mal})p_l + \overline{k_o}\beta^3 P_{mal}^3\Big) \\
&+ \frac{B_\Delta}{(1+\beta)^2}\Big((-1+\overline{k_o})p_l^2 + (1+2\overline{k_o}\beta P_{mal})p_l + \overline{k_o}\beta^2 P_{mal}^2\Big) \\
&+ \frac{C_\Delta}{1+\beta}\Big(\overline{k_o}p_l + \overline{k_o}\beta P_{mal}\Big) + D_\Delta\Bigg] + O(\alpha^2),
\end{aligned}
\tag{19}
$$

where

$$
\begin{gathered}
A_\Delta = (1-2\varepsilon)^2\ \Delta, \quad B_\Delta = -2\left(1-2\varepsilon+2\varepsilon^2\right)\Delta - \Delta_l + 4(1-\varepsilon)\varepsilon\Delta_n, \\
C_\Delta = \left(1-\varepsilon+\varepsilon^2\right)\Delta + \Delta_l - 4(1-\varepsilon)\varepsilon\Delta_n, \quad D_\Delta = (1-\varepsilon)\varepsilon\Delta_n, \\
\Delta = u_{ld} - u_{ls} + u_{nd} - u_{ns}, \quad \Delta_l = u_{ls} - u_{nd}, \quad \Delta_n = u_{nd} - u_{ns}.
\end{gathered}
$$

4 Decision Fusion in the Affected Network by Byzantines

In this section, we derive the optimal fusion rule which takes into account the correlation between users' reports based on the evolutionary game theoretical analysis on the harm of the Byzantines. Previous works [1, 16] have considered each user's report independently when calculating the posterior probability. Instead, we consider the influence of a user's surroundings on his decision making in an asynchronous manner and introduce relevant variables in the time dimension to rederive the maximum posterior probability fusion mechanism.

4.1 The Optimum Decision Rule

The optimum decision rule by adopting the maximum a posterior probability criterion has been first proposed in [1]. We also conduct MAP criterion while considering that ordinary users may change their strategies due to the herding effect and behave like a malicious user, who will send $r_i^t \neq u_i^t$. Given the received reports vector $\mathbf{r} = \{r_i^t\}, i = 1 \dots N, t = 1 \dots T$, the optimum decision fusion rule $\boldsymbol{\theta}^*$ minimizing the error probability is shown as follows

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} P\left(\boldsymbol{\theta}|\mathbf{r}\right), \tag{20}$$

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

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} P\left(\mathbf{r}|\boldsymbol{\theta}\right), \tag{21}$$

Similar to [1], let $\boldsymbol{\xi} = (\xi_1, \xi_2 \dots \xi_N)$ 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 $P(\boldsymbol{\xi})$ be the probability distribution of Byzantines across the entire network. We expand the equation of $\boldsymbol{\theta}^*$ on $\boldsymbol{\xi}$ as follows

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \sum_{\boldsymbol{\xi}} P\left(\mathbf{r} \mid \boldsymbol{\theta}, \boldsymbol{\xi}\right) P\left(\boldsymbol{\xi}\right), \tag{22}$$

4.2 Measuring the Hazard of Byzantines

In this subsection, we calculate the posterior probability $P\left(\mathbf{r} \mid \boldsymbol{\theta}, \boldsymbol{\xi}\right)$ using the evolutionary game theoretical model proposed in Sections 2 and 3. Note that in the scenario this paper considers, Byzantines can not only directly compromise the information fusion process by submitting false reports, but amplify their attack by misleading the decisions of ordinary users around them in the next round thanks to the herding effect. Since the two groups have different update strategies, they need to be calculated separately.

4.2.1 The Malicious Users

We measure the direct hazard to FC of the false reports submitted by the Byzantines here. As described, it is assumed that the reports sent by the

Byzantines are independent of each other and affected only by the real system state, observation error, and the attack probability P_{mal}. Thus, Equation (22) can be expressed as follows

$$\boldsymbol{\theta}^{*} = \arg\max_{\boldsymbol{\theta}} \sum_{\boldsymbol{\xi}} \left(\prod_{i=1}^{N} \prod_{t=1}^{T} P\left(r_i^t | \theta^t, \xi_i\right) \right) P\left(\boldsymbol{\xi}\right). \tag{23}$$

Combined with the system error ε and malicious users' probability of lying is P_{mal}, the probability δ that the FC receives a wrong report is

$$\delta = \varepsilon\left(1 - P_{mal}\right) + (1 - \varepsilon) P_{mal}. \tag{24}$$

In order to go on, we conclude the value of $P\left(r_i^t | \theta^t, \xi_i\right)$ in two values according to whether the report is consistent with the real system state θ^t

$$P\left(r_i^t | \theta^t, \xi_i = 1\right) = \begin{cases} \delta & \text{with error,} \\ 1 - \delta & \text{error free.} \end{cases} \tag{25}$$

4.2.2 The Ordinary Users

Note that with the existence of herding effect and the mutual influence between ordinary users, users' reports are not independent which is different from the previous works in [1, 16]. To elaborate, since the user's decision is only related to the reports of the surrounding users at the last epoch and their own observed system status at this epoch. It can be seen that the process is Markov. Considering the Markov property of the stochastic process, $P\left(\mathbf{r} \mid \boldsymbol{\theta}, \boldsymbol{\xi}\right)$ can be expanded in the time dimension to

$$P\left(\mathbf{r} \mid \boldsymbol{\theta}, \boldsymbol{\xi}\right) = \prod_{t=1}^{T} P\left(\mathbf{r}^{t+1} \mid \mathbf{r}^{t}, \theta^{t}, \boldsymbol{\xi}\right) P\left(\mathbf{r}^{1} \mid \theta^{1}, \boldsymbol{\xi}\right) \tag{26}$$

where $\mathbf{r}^t$ represents the report vector of all users at epoch t. This formula leads to two situations from the epoch t: At the initial epoch, since there are no previous reports, the reports sent by ordinary users are only related to their observed system status. At non-initial epochs, in addition to their observations, ordinary users will also compare the reports of the surrounding users with their last observations to infer the decision-making strategy of surrounding users. Finally, the ordinary users comprehensively consider these two aspects to make their decision on what to report at this epoch. The detailed decision-making process of the ordinary user has been described in Section 2. According to whether epoch t is the initial epoch, the state transition probability of ordinary users can be divided into the following two situations:

At the initial epoch, ordinary users will not be affected by surrounding users and depend on whether there is a systematic error ε in the observation at this epoch

$$P\left(r_i^1|\xi_i=0,\theta^1\right)=\begin{cases}\varepsilon & \text{with error,}\\ 1-\varepsilon & \text{error free.}\end{cases} \tag{27}$$

At non-initial epochs, ordinary users are affected by the lying users around. Specifically, ordinary users may use lying strategies to achieve consistency with those around them, which is also known as the herding effect. In this case, whether the ordinary user's report is consistent with the real system state depends on two factors: the systematic error in the observation, and the strategy he/she adopted. Therefore, at epoch t, the probability that the FC receives a wrong report from an ordinary user is:

$$\gamma^t=(1-\varepsilon)p_l^t+\varepsilon(1-p_l^t). \tag{28}$$

where the calculation of specific values p_l^t can be obtained from Section 2. In this way, we can derive the posterior probability of sending reports from the ordinary users at a non-initial epoch t according to whether the report is consistent with the real system state θ^t

$$P\left(r^t\mid r^{t-1},\theta^t,\xi_i=0\right)=\begin{cases}\gamma^t & \text{with error,}\\ 1-\gamma^t & \text{error free.}\end{cases} \tag{29}$$

Finally, we merge the calculations of the two types of users and introduce a binary hidden matrix $Q^{N\times T}$. If $q_i(t)=1$, it represents that the report of the i-th user at epoch t is consistent with the real state of the system at epoch t, that is $r_i^t=\theta^t$. Conversely, if $q_i(t)=0$, it means that $r_i^t\neq\theta^t$. Combined with the above discussion, the optimum decision rule in this scenario can be written as:

$$\boldsymbol{\theta}^*=\arg\max_{\boldsymbol{\theta}}\sum_{\boldsymbol{\xi}}\prod_{t=1}^{T}\left(\prod_{i:\xi_i=1}(1-\delta)^{q_i(t)}\delta^{1-q_i(t)}\prod_{i:\xi_i=0}(1-\gamma^t)^{q_i(t)}(\gamma^t)^{1-q_i(t)}\right)P\left(\boldsymbol{\xi}\right), \tag{30}$$

Same as in [1], we consider the distribution of Byzantines and find the optimal fusion mechanism accordingly. To simplify the analysis, we assume the FC knows the expected fraction of Byzantines users in the network. Let's first take an assistant function $f(\xi_i)$:

$$\begin{aligned}f(\xi_i)&=\prod_{t=1}^{T}\left(\prod_{i:\xi_i=1}(1-\delta)^{q_i(t)}\delta^{1-q_i(t)}\prod_{i:\xi_i=0}(1-\gamma^t)^{q_i(t)}(\gamma^t)^{1-q_i(t)}\right)\\&=\begin{cases}\prod_{t=1}^{T}(1-\delta)^{q_i(t)}\delta^{1-q_i(t)}, & \xi_i=1\\ \prod_{t=1}^{T}(1-\gamma^t)^{q_i(t)}(\gamma^t)^{1-q_i(t)}, & \xi_i=0\end{cases}\end{aligned} \tag{31}$$

Noticing that ξ_is are i.i.d, we can get the probability that a user is malicious is $\frac{\beta}{1+\beta}$ and the probability that a user is ordinary is $\frac{1}{1+\beta}$. Then we can obtain the expectation for each $f(\xi_i)$:

$$\mathbb{E}[f(\xi_i)] = \frac{\beta}{1+\beta}\prod_{t=1}^{T}(1-\delta)^{q_i(t)}\delta^{1-q_i(t)} + \frac{1}{1+\beta}\prod_{t=1}^{T}(1-\gamma^t)^{q_i(t)}(\gamma^t)^{1-q_i(t)} \quad (32)$$

Exploiting the property of expectation of i.i.d variables, we can obtain:

$$\mathbb{E}[f(\xi_1), f(\xi_2), \ldots, f(\xi_N)] = \prod_{i=1}^{N}\mathbb{E}[f(\xi_i)] \quad (33)$$

Taking Equations (32) and (33) together we can rewrite Equation (30) as:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \prod_{i=1}^{N}\left(\frac{\beta}{1+\beta}\prod_{t=1}^{T}(1-\delta)^{q_i(t)}\delta^{1-q_i(t)} + \frac{1}{1+\beta}\prod_{t=1}^{T}(1-\gamma^t)^{q_i(t)}(\gamma^t)^{1-q_i(t)}\right). \quad (34)$$

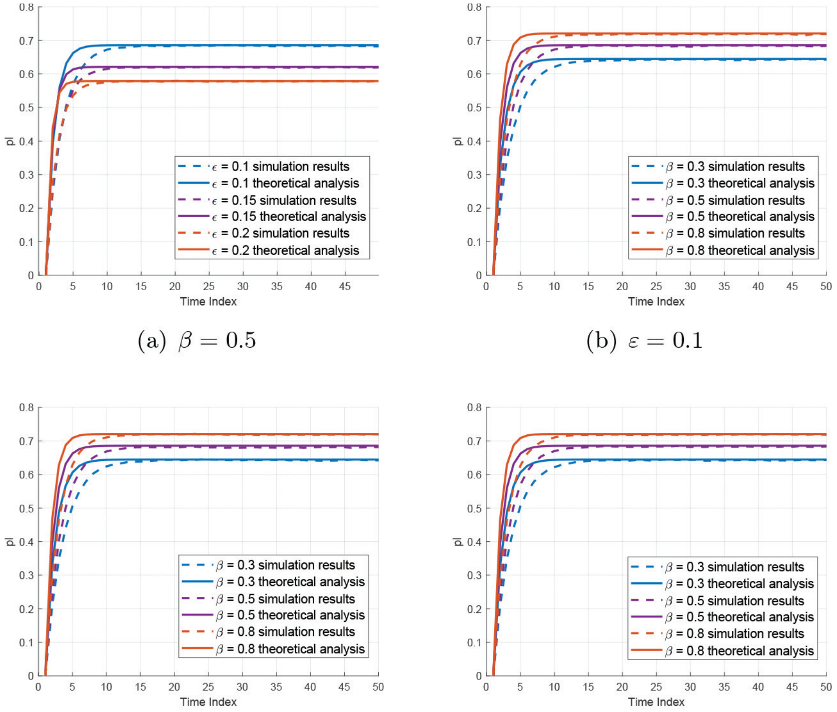
Other discussions on the distribution of Byzantines are similar to those in [1], so we will not repeat them here. Besides, since the complexity of the algorithm increases exponentially with time T, we calculate the total time separately according to the period of every τ epoch, so that it can also become a real-time processing algorithm. In addition, due to the large scale of the general social network, when the number of users is large, the algorithm may have computational accuracy problems because the posterior probability may be very small. It is only necessary to nest the logarithmic operation in the outermost layer of the calculation by computer. Given the above, we yield the optimum decision fusion rule in the affected network by Byzantines due to herding behaviors.

5 Simulation Results

In this section, our simulation is divided into two parts. First, we verify and analyze the theoretical analysis in Section 3 to measure the hazard of Byzantines. In addition, we conduct a large number of experiments to analyze the influence of different parameters on the steady state of the network and the influence of different attack strategies of malicious users on the network in the presence of herding effect. Then we use the analytical results of the proposed fusion method in Section 4 to perform decision fusion on a random network that takes herding behavior into consideration.

5.1 Evolutionary Dynamics of Byzantine Users

We first verify the effectiveness of the theoretical analysis on the hazards of Byzantine users through Monte Carlo simulation. Three commonly used network structures are considered in the experiment: the uniform degree (regular) network, the Barabási-Albert (BA) scale-free network, and Erdős-Rényi (ER) network. The default 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 and ER networks are $\overline{k} = 30$, and the weak selection coefficient α is 0.001. The payoff matrix is set to $u_{ns} = 0.8, u_{nd} = 0.6, u_{ls} = 0.6, u_{ld} = 0.4$. And the initial lying proportion p_l is set to 0. For each type of network, 5 graphs are randomly generated, and 128 simulations are conducted for each graph. Besides, the number of iterations for graphical EGT is set to 300.



(a) $\beta = 0.5$

(b) $\varepsilon = 0.1$

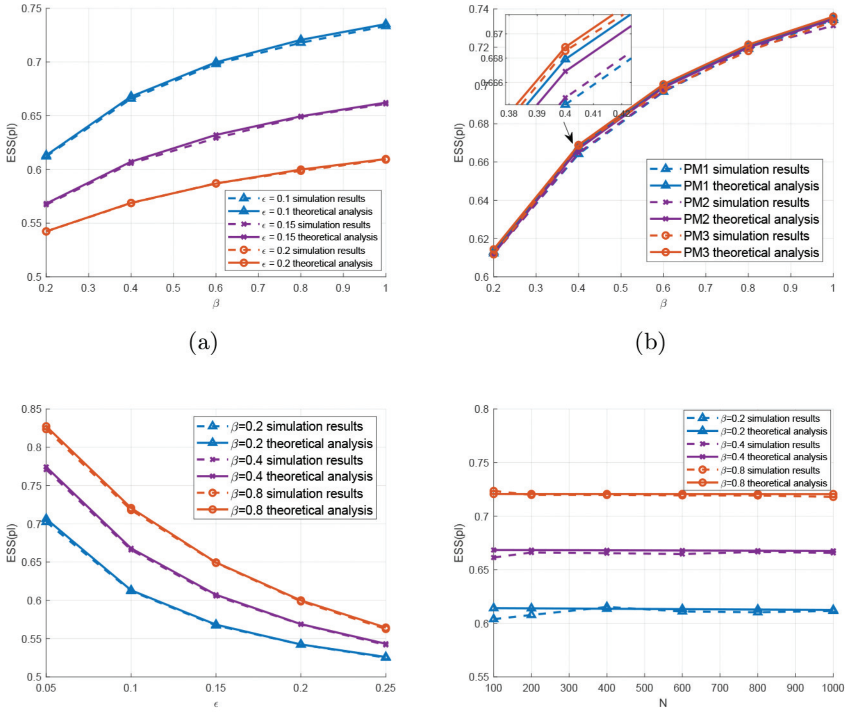
(c) Scale-free Network

(d) ER Network

Figure 5: The evolutionary dynamics of p_l on regular networks ($\bar{k} = 30$) with (a) different system errors $\varepsilon = 0.1, 0.15, 0.20$, and (b) different percentages of byzantines $\beta = 0.3, 0.5, 0.8$ and on (c) BA scale-free networks ($\bar{k} = 30$) and (d) ER random networks ($\bar{k} = 30$).

Figure 5 shows the evolutionary dynamics of p_l on the regular network where all users adopt the DB update rule. We can see that the theoretical results can fit well with simulation results under different experimental parameter settings, and the number of ordinary users adopting the S_l strategy gradually increases to a stable value (ESS) over time due to the influence of byzantine users. Besides, we evaluate the performance under different parameters on networks. Figure 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 be 0.5. Figure 5(b) shows that as the number of Byzantines increases, the number of users adopting S_l strategies will also increase. These findings are also reflected in Figure 6.

Figure 5(c) and (d) show the evolutionary dynamics of p_l on the BA scale-free network and the ER network, and the parameter settings are consistent



(a)

(b)

(c)

(d)

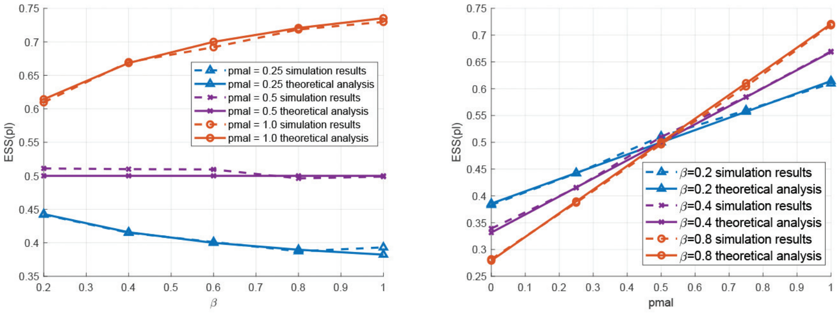
Figure 6: The ESS (p_l^*) on regular networks ($\bar{k} = 30$) with (a) different system errors $\varepsilon = 0.1, 0.15, 0.20$, (b) different payoff matrix setting, (c) different percentages of byzantines $\beta = 0.4, 0.6, 0.8$, and (d) different users network size.

with (b). It can be seen that (b), (c), and (d) have almost the same results, which indicates the network structure has little effect on our simulation results on p_l. Therefore, without loss of generality, we conduct the experiments only on regular networks in the rest of this section.

Then we evaluate the ESS of p_l under different parameters respectively, including system error, payoff matrix, the proportion of Byzantines on the ordinary user group, and the users' network size. From Figure 6(a), we can conclude that as the system error rate increases, the proportion of ordinary users influenced by Byzantines (p_l^*) gradually approaches 0.5. This is due to the fact that users cannot make any meaningful decisions when the system error is too large and thus adopt a strategy similar to "tossing a coin". In Figure 6(b), we set up three different payoff matrices, and believe that the payoffs of ordinary users satisfy the following conditions: $u_{ns} > u_{ls}/u_{nd} > u_{ld}$ in Equation (4). The implication is that the benefit of keeping consistent with others without lying u_{ns} is the greatest for ordinary users, and the benefit of being inconsistent with others by lying u_{ld} is the smallest. Then we discuss the relationship between the remaining u_{ls} and u_{nd}, and set up three different sets of payoff matrices: PM1 $u_{ns} = 0.8, u_{ls} = 0.6, u_{nd} = 0.6, u_{ld} = 0.4$; PM2 $u_{ns} = 0.8, u_{ls} = 0.5, u_{nd} = 0.7, u_{ld} = 0.4$; PM3 $u_{ns} = 0.8, u_{ls} = 0.7, u_{nd} = 0.5, u_{ld} = 0.4$. It can be seen from Figure 6(b) that there is very little difference between the results of three different payoff matrices when $u_{ns} > u_{ls}/u_{nd} > u_{ld}$ is satisfied. In this way, it can be shown that the prediction result of our model is robust to the payoff matrix parameter when the relationship is satisfied, so it can be roughly estimated when setting the parameters.

Notice that the result of PM3 is slightly larger than PM1 and PM2 if we take a closer look, and this is because the u_{ls} of PM3 is relatively large. From Figure 6(c), 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. Finally, we study the influence of different network sizes on the model and the results are shown in Figure 6(d), where N is the total number of users of the network size. And here it is assumed that the degree of the network has a relationship with the total number of users in the network, where $\overline{k} = 0.05N$. From Figure 6(d), we also find that the network size has little effect on the model except for some minor deviations between the theoretical and simulation results when the size of the network is small.

Figure 7 shows the ESS of p_l on the regular network which reflects the hazards on the network by malicious users. First, we can see that the theoretical results can fit well with the simulation results under different experimental parameter settings. Because at the initial epoch the proportion of liars p_l is set to 0, it can be seen from the plotted trend in Figure 7 that regardless of how the attack strategy varies over a fairly large range, the average user will still be influenced by the malicious node to adopt a lying strategy. Figure



(a) (b)

Figure 7: The hazard (p_l^*) on regular networks ($\bar{k} = 30$) with (a) different attack strategies on lying probability $P_{mal} = 0.25, 0.5, 1.0$, and (b) different attack strategies on proportion of byzantines and ordinary users $\beta = 0.2, 0.4, 0.8$.

7(a) also shows that the trend of steady-state p_l with respect to β varies with P_{mal} value. Specifically, when the lying probability of malicious users $P_{mal} > 0.5$, the proportion of lying users p_l in the ESS will increase with the increase of malicious users. Conversely, when the lying probability of a malicious user $P_{mal} < 0.5$, the number of lying users in the network will decrease as the number of malicious users increases. When the lying probability of a malicious user $P_{mal} = 0.5$, no matter how the number of malicious users changes, approximately there will always be a half-to-halt situation between the lying user and the honest user in the network. From Figure 7(b), it can be seen that as the lying probability P_{mal} of malicious users increases, the proportion of lying users p_l will increase regardless of the proportion of malicious users β. Meanwhile, comparing the slopes of the plots in Figure 7(b), we can conclude that when the β increases, the probability of lying caused by the attack becomes more severe as the P_{mal} increases.

In addition, the experimental results show that Byzantines are very harmful to the network. When the system error is small, it takes only a small proportion of attackers to affect the majority of ordinary users, which is also consistent with the conclusion in [30]. Meanwhile, the increase in system error rate will mitigate this phenomenon, but the proportion of ordinary users will still be affected by more than 50%, hence it is valuable to study effective methods to resist byzantine attacks.

5.2 Decision Fusion with Herding Behavior

After validating the correctness of our analysis in our model in Section 3, we further use this model to verify the performance of the proposed decision fusion algorithm. The default experimental settings are the same as before, and since it was demonstrated in the last subsection that the graph structure has little effect on the network, for simplicity we use a random uniform network to perform the following fusion experiments. Besides, we still run 300 iterations each trial, but the fusion time window T is 6, and the average value of the fusion accuracy rate is obtained by repeating 1000 trials in the Monte Carlo method.

In the presence of herding behavior, we first compare the accuracy of the existing commonly used fusion strategy [1] with our newly proposed fusion strategy. Previous works have shown that $P_{mal} = 1$ is a dominant strategy [21] for Byzantines in game theory framework experiments [13], so we have adopted it as well in our experiments. In addition, we consider two scenarios to compare our work and the previous works: in Scenario 1 (SC1) ordinary users will always submit honest reports; Scenario 2 (SC2) is the main study object of this paper, where we introduce the herding effect, a phenomenon in which users' decisions are affected by the other users around them. Comparing the accuracy of the OPT method in [1] and the EGTM-DFB method in different scenes, we prove the effectiveness of the proposed EGTM-DFB method proposed in this paper.

From Tables 1 and 2, we can see that the original method (OPT) proposed in [1] had remarkable fusion accuracy when ordinary users were not influenced by others and submitted their observations independently in scenario (SC1), but it fails to get accurate estimates of the system states in SC2, which indicates that the erosion of the herding effect on the average user will significantly reduce the accuracy of the OPT method. The proposed fusion strategy (EGTM-DFB) can achieve high estimation accuracy under the situation of ordinary users

Table 1: Accuracy of the fusion algorithms in two scenarios with $\varepsilon = 0.1$.

Method	β Scenario	0.3	0.5	0.7	0.9
OPT [1]	SC1	1.0000	1.0000	1.0000	1.0000
OPT [1]	SC2	0.020	**0.0033**	**0.0033**	**0.0033**
EGTM-DFB	SC2	0.9998	0.9967	0.9967	0.9967

Table 2: Accuracy of the fusion algorithms in two scenarios with $\varepsilon = 0.2$.

Method	β Scenario	0.3	0.5	0.7	0.9
OPT [1]	SC1	1.0000	1.0000	1.0000	**0.9999**
OPT [1]	SC2	0.0172	0.0033	0.0033	0.0033
EGTM-DFB	SC2	0.9967	0.9967	0.9967	**0.9967**

being affected by Byzantines. In Tables 1 and 2, our models can reach a very high fusion rate **0.9967** and are quite insensitive to the parameters β and ε. The results show that although Byzantines can be very detrimental to the network through herding behavior, we can still predict the hazards of the network through our graphical EGT model and adjust the fusion strategy to achieve productive decision fusion results.

Under the same experimental settings except that β is fixed to 0.3, we explored the performance of the algorithm under different system error rates, and the results are shown in Table 3. From Table 3 we can conclude that the system error rate has little effect on the fusion accuracy. Overall, the fusion mechanism we proposed can achieve a good fusion effect in the scenario where there is the herding effect.

Table 3: Accuracy of the fusion algorithms in two scenarios with $\beta = 0.3$.

Method	ε Scenario	0.05	0.10	0.15	0.20
OPT [1]	SC1	1.0000	1.0000	1.0000	1.0000
OPT [1]	SC2	0.0200	0.0200	0.0199	0.0172
EGTM-DFB	SC2	1.0000	0.9998	0.9967	0.9967

6 Conclusions

In this paper, we delve into a new scenario of decision fusion, where ordinary users will be affected by Byzantines due to the existence of herding behaviors and may use the same strategy as the Byzantines. Firstly, we utilize graphical evolutionary game theory to analyze users' behavior and measure the hazard of Byzantines. And we derive the analytical formulation of the evolutionary dynamics and the corresponding numerical evolutionary stable states. Secondly, we propose an effective fusion mechanism for the FC based on our human decision making model and the maximum a posteriori estimation. Finally, simulation results show that our theoretical model works well to characterize and predict users' behavior and the hazards of Byzantines. In addition, we conduct experiments to verify our fusion strategy can effectively resist Byzantines even when malicious users may greatly influence others' decisions in the presence of herding behaviors.

Appendix

A Details of Derivations in $\boldsymbol{P_{n-l}}$, $\boldsymbol{E\left[P_{n\to l}(k_o, k_{ol})\right]}$, and $\boldsymbol{\dot{p}_l}$

Taking Equations (1)–(2) into Equations(5)–(8), we get:

$$\pi_l^A = 1-\alpha+\alpha\Big\{\,[(1-\varepsilon)k_l+\varepsilon k_n]\,u_{ls}+[k-(1-\varepsilon)k_l-\varepsilon k_n]\,u_{ld}\Big\}, \tag{A.1}$$

$$\pi_n^A = 1-\alpha+\alpha\Big\{\,[(1-\varepsilon)k_n+\varepsilon k_l]\,u_{ns}+[k-(1-\varepsilon)k_n-\varepsilon k_l]\,u_{nd}\Big\}, \tag{A.2}$$

$$\pi_l^B = 1-\alpha+\alpha\Big\{\,[(1-\varepsilon)k_n+\varepsilon k_l]\,u_{ls}+[k-(1-\varepsilon)k_n-\varepsilon k_l]\,u_{ld}\Big\}, \tag{A.3}$$

$$\pi_n^B = 1-\alpha+\alpha\Big\{\,[(1-\varepsilon)k_l+\varepsilon k_n]\,u_{ns}+[k-(1-\varepsilon)k_l-\varepsilon k_n]\,u_{nd}\Big\}. \tag{A.4}$$

The equation we are addressing is:

$$P_{n\to l} = (1-\varepsilon)\cdot P_{n\to l}^A + \varepsilon\cdot P_{n\to l}^B \tag{A.5}$$

Taking Equations (9)–(10) into Equation (A.5), we get:

$$P_{n\to l} = \frac{\varepsilon\cdot\left[k_n(1-\varepsilon)\cdot\pi_n^A + k_l\varepsilon\cdot\pi_l^B\right] + (1-\varepsilon)\cdot\left[k_l(1-\varepsilon)\cdot\pi_l^A + k_n\varepsilon\cdot\pi_n^B\right]}{\left[k_n(1-\varepsilon)\cdot\pi_n^A + \varepsilon k_l\cdot\pi_l^B\right] + \left[k_l(1-\varepsilon)\cdot\pi_l^A + \varepsilon k_n\cdot\pi_n^B\right]} \tag{A.6}$$

To facilitate simplification, we introduce the following notations:

$$\begin{aligned} A &= \varepsilon \\ B &= 1-\varepsilon \\ C &= k_l \\ D &= k_n = k-k_l \end{aligned} \tag{A.7}$$

Taking Equation (A.7) into Equation (A.6) and expanding the contents of the brackets, we get:

$$P_{n\to l} = \frac{B^2C\pi_l^A + A^2C\pi_l^B + ABD\pi_n^B + ABD\pi_n^A}{BD\pi_n^A + AD\pi_n^B + BC\pi_l^A + AC\pi_l^B} \tag{A.8}$$

Note that Equations (A.1)–(A.4) have a common part $1-\alpha$, which we can separate out to obtain:

$$P_{n\to l} = \frac{(1-\alpha)\left(B^2C + A^2C + 2ABD\right) + \alpha\left\{\textcircled{1}\right\}}{(1-\alpha)\left(BD + AD + BC + AC\right) + \alpha\left\{\textcircled{2}\right\}} \tag{A.9}$$

where ① and ② are expressions as follows, which will be addressed later:

$$\begin{aligned}
\textcircled{1} = B^2C\Big\{ &[(1-\varepsilon)k_l + \varepsilon k_n]\, u_{ls} + [k - (1-\varepsilon)k_l - \varepsilon k_n]\, u_{ld} \Big\} \\
+ A^2C\Big\{ &[(1-\varepsilon)k_n + \varepsilon k_l]\, u_{ls} + [k - (1-\varepsilon)k_n - \varepsilon k_l]\, u_{ld} \Big\} \\
+ ABD\Big\{ &[(1-\varepsilon)k_l + \varepsilon k_n]\, u_{ns} + [k - (1-\varepsilon)k_l - \varepsilon k_n]\, u_{nd} \Big\} \\
+ ABD\Big\{ &[(1-\varepsilon)k_n + \varepsilon k_l]\, u_{ns} + [k - (1-\varepsilon)k_n - \varepsilon k_l]\, u_{nd} \Big\}
\end{aligned} \tag{A.10}$$

$$\begin{aligned}
\textcircled{2} = BC\Big\{ &[(1-\varepsilon)k_l + \varepsilon k_n]\, u_{ls} + [k - (1-\varepsilon)k_l - \varepsilon k_n]\, u_{ld} \Big\} \\
+ AC\Big\{ &[(1-\varepsilon)k_n + \varepsilon k_l]\, u_{ls} + [k - (1-\varepsilon)k_n - \varepsilon k_l]\, u_{ld} \Big\} \\
+ AD\Big\{ &[(1-\varepsilon)k_l + \varepsilon k_n]\, u_{ns} + [k - (1-\varepsilon)k_l - \varepsilon k_n]\, u_{nd} \Big\} \\
+ BD\Big\{ &[(1-\varepsilon)k_n + \varepsilon k_l]\, u_{ns} + [k - (1-\varepsilon)k_n - \varepsilon k_l]\, u_{nd} \Big\}
\end{aligned} \tag{A.11}$$

For now we extract the factors containing A, B, C, D from the numerator and denominator, and obtain:

$$P_{n\to l} = \frac{B^2C + A^2C + 2ABD}{BD + AD + BC + AC} \cdot \frac{1 + \alpha\left(\frac{\{\textcircled{1}\}}{B^2C + A^2C + 2ABD} - 1\right)}{1 + \alpha\left(\frac{\{\textcircled{2}\}}{BD + AD + BC + AC} - 1\right)} \tag{A.12}$$

From Equation (A.7), we can get:

$$\begin{aligned}
&BD + AD + BC + AC = k \\
&B^2C + A^2C + 2ABD = (1-2\varepsilon)^2\, k_l + 2\varepsilon(1-\varepsilon)k
\end{aligned} \tag{A.13}$$

Hence, Equation (A.12) can be simplified as:

$$P_{n\to l} = \left(2(1-\varepsilon)\varepsilon + (1-2\varepsilon)^2 \frac{k_l}{k}\right) \cdot \frac{1 + \alpha\left(\frac{\{\textcircled{1}\}}{(1-2\varepsilon)^2 k_l + 2\varepsilon(1-\varepsilon)k} - 1\right)}{1 + \alpha\left(\frac{\{\textcircled{2}\}}{k} - 1\right)} \tag{A.14}$$

Now let's deal with expressions ① and ②.

For ①, let the coefficients of $u_{ns}, u_{nd}, u_{ls}, u_{ld}$ be a, b, c, d, which means:

$$\textcircled{1} = a u_{ns} + b u_{nd} + c u_{ls} + d u_{ld} \tag{A.15}$$

Similarly, for ② we can set the coefficients as follows:

$$\textcircled{2} = eu_{ns} + fu_{nd} + gu_{ls} + hu_{ld} \tag{A.16}$$

Replace A, B, C, D in Equations (A.10) and (A.11) with equation Equation (A.7), and collect the terms according to $u_{ns}, u_{nd}, u_{ls}, u_{ld}$, we can calculate that:

$$\begin{aligned}
a &= (1-\varepsilon)\varepsilon k^2 - (1-\varepsilon)\varepsilon k k_l \\
b &= a \\
c &= (1-\varepsilon)\varepsilon k k_l \\
d &= \left(1 - 3\varepsilon + 3\varepsilon^2\right) k k_l - (1-2\varepsilon)^2 k_l^2 \\
e &= k^2(1 - 2\varepsilon + 2\varepsilon^2) - 2kk_l(1 - 3\varepsilon + 3\varepsilon^2) + (1-2\varepsilon)^2 k_l^2 \\
f &= 2(1-\varepsilon)\varepsilon k^2 + (1 - 6\varepsilon + 6\varepsilon^2) k k_l \\
g &= 2(1-\varepsilon)\varepsilon k k_l + (1-2\varepsilon)^2 k_l^2 \\
h &= (1 - 2\varepsilon + 2\varepsilon^2) k k_l - (1-2\varepsilon)^2 k_l^2
\end{aligned} \tag{A.17}$$

Noting that f and g are identical to other variables, we re-organize the above results and bring them into ① and ② to obtain:

$$P_{n\to l} = \left(2(1-\varepsilon)\varepsilon + (1-2\varepsilon)^2 \ \frac{k_l}{k}\right) \cdot \frac{1 + \alpha\left(\frac{au_{ns} + bu_{nd} + cu_{ls} + du_{ld}}{(1-2\varepsilon)^2 k_l + 2\varepsilon(1-\varepsilon)k} - 1\right)}{1 + \alpha\left(\frac{eu_{ns} + fu_{nd} + gu_{ls} + hu_{ld}}{k} - 1\right)}$$

where

$$\begin{aligned}
a &= (1-\varepsilon)\varepsilon k^2 - (1-\varepsilon)\varepsilon k k_l \\
b &= a \\
c &= (1-\varepsilon)\varepsilon k k_l \\
d &= \left(1 - 3\varepsilon + 3\varepsilon^2\right) k k_l - (1-2\varepsilon)^2 k_l^2 \\
e &= k^2(1 - 2\varepsilon + 2\varepsilon^2) - 2kk_l(1 - 3\varepsilon + 3\varepsilon^2) + (1-2\varepsilon)^2 k_l^2 \\
f &= 2(1-\varepsilon)\varepsilon k^2 + (1 - 6\varepsilon + 6\varepsilon^2) k k_l \\
g &= 2(1-\varepsilon)\varepsilon k k_l + (1-2\varepsilon)^2 k_l^2 \\
h &= (1 - 2\varepsilon + 2\varepsilon^2) k k_l - (1-2\varepsilon)^2 k_l^2.
\end{aligned} \tag{A.18}$$

Exploiting the fact that $\frac{1+\mu x}{1+\nu x} = 1 + (\mu - \nu)x + O(x^2)$ for small x, we can simplify Equation (A.18) as follows:

$$\begin{aligned}
P_{n\to l} &= \left(2(1-\varepsilon)\varepsilon + (1-2\varepsilon)^2 \frac{k_l}{k}\right) \\
&\quad \cdot \left(1 + \alpha\left(\frac{\textcircled{1}}{(1-2\varepsilon)^2 k_l + 2\varepsilon(1-\varepsilon)k} - \frac{\textcircled{2}}{k}\right) + O(\alpha^2)\right) \\
&= 2(1-\varepsilon)\varepsilon + (1-2\varepsilon)^2 \frac{k_l}{k} + \frac{\alpha}{k^2}\left(k\textcircled{1} - \left((1-2\varepsilon)^2 k_l + 2\varepsilon(1-\varepsilon)k\right)\textcircled{2}\right) \\
&\quad + O(\alpha^2) \qquad \text{(A.19)}
\end{aligned}$$

For the complex part of Equation (A.19), we perform straightforward calculations and collect the terms according to $u_{ns}, u_{nd}, u_{ls}, u_{ld}$:

$$\begin{aligned}
\textcircled{3} &= \frac{\alpha}{k^2}\left[k\textcircled{1} - \left((1-2\varepsilon)^2 k_l + 2\varepsilon(1-\varepsilon)k\right)\textcircled{2}\right] \\
&= \frac{\alpha(1-2\varepsilon)^2}{k^2}\Big\{ \\
&\quad + k_l\left[(1-\varepsilon)\varepsilon k^2 + (1-2\varepsilon)^2 k_l k - (1-2\varepsilon)^2 k_l^2\right] u_{ls} \\
&\quad + k_l\left[(1-\varepsilon+\varepsilon^2)k^2 - 2(1-2\varepsilon+2\varepsilon^2)k_l k + (1-2\varepsilon)^2 k_l^2\right] u_{ld} \\
&\quad + (k-k_l)\left[(\varepsilon-1)\varepsilon k^2 - (1-2\varepsilon)^2 k_l + (1-2\varepsilon)^2 k_l^2\right] u_{ns} \\
&\quad + (k-k_l)\left[(1-\varepsilon)\varepsilon k^2 - 4(1-\varepsilon)\varepsilon k_l k - (1-2\varepsilon)^2 k_l^2\right] u_{nd}\Big\}
\end{aligned} \qquad \text{(A.20)}$$

To facilitate further calculations, we structure the results in powers of $\frac{k_l}{k}$ as follows.

$$\begin{aligned}
\textcircled{3} &= \alpha(1-2\varepsilon)^2 k\Big\{ -(-1+\varepsilon)\varepsilon(u_{nd} - u_{ns}) \\
&\quad + (1-2\varepsilon)^2 \frac{k_l^3}{k^3}\left(u_{ld} - u_{ls} + u_{nd} - u_{ns}\right) \\
&\quad + \frac{k_l^2}{k^2}\left\{\left[-2 - 4\left(-1+\varepsilon\right)\right] u_{ld} + u_{ls} - u_{nd} + 2u_{ns}\right. \\
&\quad \left. +4(-1+\varepsilon)\varepsilon(u_{ls} - 2u_{nd} + 2u_{ns})\right\} \\
&\quad + \frac{k_l}{k}\left\{\left[1 + (-1+\varepsilon)\varepsilon\right] u_{ld} - u_{us} - (-1+\varepsilon)\varepsilon\left(u_{ls} - 5u_{nd} + 5u_{ns}\right)\right\}\Big\}
\end{aligned} \qquad \text{(A.21)}$$

To further simplify, replace $u_{ns}, u_{nd}, u_{ls}, u_{ld}$ with the following variables:

$$\begin{aligned}
\Delta &= u_{ld} - u_{ls} + u_{nd} - u_{ns} \\
\Delta_l &= u_{ls} - u_{nd} \\
\Delta_n &= u_{nd} - u_{ns}
\end{aligned} \qquad \text{(A.22)}$$

Taking Equation (A.22) into Equation (A.21), we get:

$$\begin{aligned}
\textcircled{3} =& \alpha(1-2\varepsilon)^2 k \Big\{ \\
& \left(\frac{k_l}{k}\right)^3 \left[(1-2\varepsilon)^2 \Delta\right] + \\
& \left(\frac{k_l}{k}\right)^2 \left[-2\left(1-2\varepsilon+2\varepsilon^2\right)\Delta - \Delta_l + 4(1-\varepsilon)\varepsilon\right] + \\
& \left(\frac{k_l}{k}\right) \left[\left(1-\varepsilon+\varepsilon^2\right)\Delta + \Delta_l - 4(1-\varepsilon)\varepsilon\Delta_n\right] + \\
& (1-\varepsilon)\varepsilon\Delta_n \Big\}
\end{aligned} \tag{A.23}$$

Taking it a step further, we introduce the following variables to denote the coefficients of $\frac{k_l}{k}$:

$$\begin{aligned}
A_\Delta &= (1-2\varepsilon)^2\Delta \\
B_\Delta &= -2\left(1-2\varepsilon+2\varepsilon^2\right)\Delta - \Delta_l + 4(1-\varepsilon)\varepsilon\Delta_n \\
C_\Delta &= \left(1-\varepsilon+\varepsilon^2\right)\Delta + \Delta_l - 4(1-\varepsilon)\varepsilon\Delta_n \\
D_\Delta &= (1-\varepsilon)\varepsilon\Delta_n
\end{aligned} \tag{A.24}$$

Using Equation (A.24) to replace the variables in Equation (A.23) and taking them into Equation (A.19), we end up with:

$$\begin{aligned}
P_{n\to l} =& 2(1-\varepsilon)\varepsilon + (1-2\varepsilon)^2 \frac{k_l}{k} + \\
& \alpha(1-2\varepsilon)^2 k \Big\{ \left(\frac{k_l}{k}\right)^3 A_\Delta + \left(\frac{k_l}{k}\right)^2 B_\Delta + \left(\frac{k_l}{k}\right) C_\Delta + D_\Delta \Big\} + O(\alpha^2)
\end{aligned} \tag{A.25}$$

According to the Section 3, we can obtain:

$$\begin{aligned}
k &= k_o + k_b = (1+\beta)\,k_o \\
k_l &= k_{ol} + k_{bl} = k_{ol} + \beta k_o P_{mal} \\
k_l^2 &= k_{ol}^2 + 2k_{ol}\beta k_o P_{mal} + \beta^2 k_o^2 P_{mal} \\
k_l^3 &= k_{ol}^3 + 3k_{ol}^2 \beta k_o P_{mal} + 3k_o\left(\beta k_o P_{mal}\right)^2 + \left(\beta k_o P_{mal}\right)^3
\end{aligned} \tag{A.26}$$

Then we derive the expectation for $P_{n\to l}$ by bringing $E(k_{ol}|k)$, $E(k_{ol}^2|k)$, and $E(k_{ol}^3|k)$ from Equations (17) to (A.26):

$$
\begin{aligned}
&E\left[P_{n\to l}(k_o, k_{ol})\right] \\
&\quad = 2(1-\varepsilon)\varepsilon + (1-2\varepsilon)^2\ \frac{p_l + \beta P_{mal}}{1+\beta} + \alpha(1-2\varepsilon)^2 \\
&\qquad \cdot \Bigg[\frac{A_\Delta}{(1+\beta)^3}\Big((2\overline{\frac{1}{k_o}} - 3 + \overline{k_o})p_l^3 + 3(-\overline{\frac{1}{k_o}} + \overline{k_o}\beta P_{mal} + 1 - \beta P_{mal})p_l^2 \\
&\qquad + (\overline{\frac{1}{k_o}} + 3\overline{k_o}\beta^2 P_{mal}^2 + 3\beta P_{mal})p_l + \overline{k_o}\beta^3 P_{mal}^3\Big) \\
&\qquad + \frac{B_\Delta}{(1+\beta)^2}\Big((-1+\overline{k_o})p_l^2 + (1+2\overline{k_o}\beta P_{mal})p_l + \overline{k_o}\beta^2 P_{mal}^2\Big) \\
&\qquad + \frac{C_\Delta}{1+\beta}\Big(\overline{k_o}p_l + \overline{k_o}\beta P_{mal}\Big) + D_\Delta\Bigg] + O(\alpha^2)
\end{aligned}
\tag{A.27}
$$

Then it is straightforward to obtain Equation (19):

$$
\begin{aligned}
\dot{p}_l &= (1-p_l)E\left[P_{n\to l}\left(k_o, k_{ol}\right)\right]) - p_l(1 - E\left[P_{n\to l}\left(k_o, k_{ol}\right)\right]) \\
&\quad = -p_l + 2(1-\varepsilon)\varepsilon + (1-2\varepsilon)^2\ \frac{p_l + \beta P_{mal}}{1+\beta} + \alpha(1-2\varepsilon)^2 \\
&\qquad \cdot \Bigg[\frac{A_\Delta}{(1+\beta)^3}\Big((2\overline{\frac{1}{k_o}} - 3 + \overline{k_o})p_l^3 + 3(-\overline{\frac{1}{k_o}} + \overline{k_o}\beta P_{mal} + 1 - \beta P_{mal})p_l^2 \\
&\qquad + (\overline{\frac{1}{k_o}} + 3\overline{k_o}\beta^2 P_{mal}^2 + 3\beta P_{mal})p_l + \overline{k_o}\beta^3 P_{mal}^3\Big) \\
&\qquad + \frac{B_\Delta}{(1+\beta)^2}\Big((-1+\overline{k_o})p_l^2 + (1+2\overline{k_o}\beta P_{mal})p_l + \overline{k_o}\beta^2 P_{mal}^2\Big) \\
&\qquad + \frac{C_\Delta}{1+\beta}\Big(\overline{k_o}p_l + \overline{k_o}\beta P_{mal}\Big) + D_\Delta\Bigg] + O(\alpha^2),
\end{aligned}
$$

where

$$
\begin{aligned}
&A_\Delta = (1-2\varepsilon)^2\ \Delta, \quad B_\Delta = -2\left(1-2\varepsilon+2\varepsilon^2\right)\Delta - \Delta_l + 4(1-\varepsilon)\varepsilon\Delta_n, \\
&C_\Delta = \left(1-\varepsilon+\varepsilon^2\right)\Delta + \Delta_l - 4(1-\varepsilon)\varepsilon\Delta_n, \quad D_\Delta = (1-\varepsilon)\varepsilon\Delta_n, \\
&\quad \Delta = u_{ld} - u_{ls} + u_{nd} - u_{ns}, \quad \Delta_l = u_{ls} - u_{nd}, \quad \Delta_n = u_{nd} - u_{ns}.
\end{aligned}
$$

References

[1] 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*, 11(6), 2016, 1333–45.

[2] 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, 505–10.

[3] 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*, 2(4), 2016, 595–610.

[4] Z. Chair and P. K. Varshney, "Optimal Data Fusion in Multiple Sensor Detection Systems," *IEEE Transactions on Aerospace and Electronic Systems*, AES-22(1), 1986, 98–101.

[5] R. Chen, J.-M. Park, and K. Bian, "Robust Distributed Spectrum Sensing in Cognitive Radio Networks," in *IEEE INFOCOM 2008-The 27th Conference on Computer Communications*, IEEE, 2008, 1876–84.

[6] B. Geng, S. Brahma, T. Wimalajeewa, P. K. Varshney, and M. Rangaswamy, "Prospect Theoretic Utility based Human Decision Making in Multi-agent Systems," *IEEE Transactions on Signal Processing*, 68, 2020, 1091–104.

[7] B. Geng, Q. Li, and P. K. Varshney, "Prospect Theory based Crowdsourcing for Classification in the Presence of Spammers," *IEEE Transactions on Signal Processing*, 68, 2020, 4083–93.

[8] B. Geng and P. K. Varshney, "On Decision Making in Human-Machine Networks," in *2019 IEEE 16th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, IEEE, 2019, 37–45.

[9] C. Jiang, Y. Chen, and K. J. R. Liu, "Evolutionary Dynamics of Information Diffusion Over Social Networks," *IEEE Transactions on Signal Processing*, 62(17), 2014, 4573–86.

[10] 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, 1095–9.

[11] 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*, 8(4), 2014, 524–36.

[12] 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, 2925–9.

[13] B. Kailkhura, S. Brahma, and P. K. Varshney, "Optimal Byzantine Attacks on Distributed Detection in Tree-based Topologies," in *2013 International Conference on Computing, Networking and Communications (ICNC)*, IEEE, 2013, 227–31.

[14] 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*, 2019, 1–1.

[15] 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, 5650–4.

[16] Y. Lin, H. Hu, H. V. Zhao, and Y. Chen, "An Evolutionary Game Theoretical Framework for Decision Fusion in the Presence of Byzantines," in *2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2020, 161–9.

[17] 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*, 22, 2019, DOI: 10.18564/jasss.3837.

[18] J. Neyman and E. S. Pearson, "IX. On the Problem of the Most Efficient Tests of Statistical Hypotheses," *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 231(694–706), 1933, 289–337.

[19] R. M. Raafat, N. Chater, and C. Frith, "Herding in Humans," *Trends in Cognitive Sciences*, 13(10), 2009, 0–428.

[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*, 59(2), 2011, 774–86.

[21] 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*, 59(2), 2010, 774–86.

[22] Y. Sun and Y. Liu, "Security of Online Reputation Systems: The Evolution of Attacks and Defenses," *IEEE Signal Processing Magazine*, 29(2), 2012, 87–97.

[23] P. D. Taylor and L. B. Jonker, "Evolutionary Stable Strategies and Game Dynamics," *Mathematical Biosciences*, 40(1–2), 1978, 145–56.

[24] P. K. Varshney, *Distributed Detection and Data Fusion*, Springer Science & Business Media, 2012.

[25] 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, 1310–5.

[26] 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*, 30, 2013, 65–75, DOI: 10.1109/MSP.2013.2262116.

[27] T. Wang, D. Wang, and F. Wang, "Quantifying Herding Effects in Crowd Wisdom," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014, 1087–96.

[28] T. Wimalajeewa, P. K. Varshney, and M. Rangaswamy, "On Integrating Human Decisions with Physical Sensors for Binary Decision Making," in *2018 21st International Conference on Information Fusion (FUSION)*, IEEE, 2018, 1–5.

[29] H. Xie, Y. Li, and J. C. Lui, "Understanding Persuasion Cascades in Online Product Rating Systems," in *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33, No. 01, 2019, 5490–7.

[30] 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, 476–81.