Blockchain-based Reputation Evaluation Using Game Theory in Social Networking

Wenjie Guo
Beijing Institute of Technology
Beijing, China
3120215536@bit.edu.cn

Jingfeng Xue
Beijing Institute of Technology
Beijing, China
xuejf@bit.edu.cn

Yong Wang*
Beijing Institute of Technology
Beijing, China
wangyong@bit.edu.cn

Zhixiong Zhou
Capital University of Physical Education And Sports
Beijing, China
zhouzhixiong@cupes.edu.cn

ABSTRACT

Reputation evaluation is one of vital elements in contemporary social network-based applications. A common type of evaluation method is relying on the third party or a community-based rating. However, this type of method is encountering an issue caused by fake comments posted by adversaries. In this paper, we propose a blockchain-based reputation evaluation approach that uses game theory to address the ghost commentator issue. Over the blockchain network, our approach establishes a reputation-based personal credit evaluation system, in which participants' strategies and payoffs are modeled. A smart contract-based deposit incentive mechanism is developed. We have implemented an experimental evaluation to show the evidence to the performance of our approach.

CCS CONCEPTS

• **Security and privacy** → **Social network security and privacy**;
• **Networks** → *Network protocols.*

KEYWORDS

Blockchain, game theory, social network, trust management.

ACM Reference Format:
Wenjie Guo, Jingfeng Xue, Yong Wang, and Zhixiong Zhou. 2022. Blockchain-based Reputation Evaluation Using Game Theory in Social Networking. In *Proceedings of the Fourth ACM International Symposium on Blockchain and Secure Critical Infrastructure (BSCI '22), May 30, 2022, Nagasaki, Japan.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3494106.3528681

1 INTRODUCTION

Currently, reputation ratings are considered a crucial element for estimating user credits in social networkings. Most existing evaluations use a third-party-based or a community-based scheme. However, However, this type of method is encountering an issue caused by fake comments posted by adversaries. Recent advance of network technologies have been reforming features of reputation evaluation systems. On one side, personal credits are available in different sources, which results in less requirements of the traditional credit service; on the other side, it is hard to track data used in the evaluation and uncertain participants in the evaluation may weaken the trustworthiness of the evaluation [8]. For instance, most current reputation platforms only offers evaluation results rather than evidencing the truth. Hence, there is an urgent demand for constructing a scheme of reputation evaluation.

As a decentralized approach, blockchain technology is an alternative for establishing a trustworthy reputation evaluation system [4, 5]. The analysis of the behavior of each participant in the blockchain system is one of the latest research directions, and game theory provides a new solution to this problem [19]. Based on mathematical models, game theory has been applied in strategic decision-making. Game theory in blockchain technology has a dual role, inward and outward. Introversion is the application of theoretical principles to blockchain protocols. Extroversion is the integration of strategic decision models with business processes. The work [20] proposed a reputation-based approach using repeat game model to solve the problems caused by intense competitions among miners and consequently dishonest mining strategies, such as block withholding attack, selfish mining, eclipse attack and stubborn mining, etc.

Reputation-based trust management using game theory can be used to deal with privacy issues, a typical privacy risk comes from data shared by multiple parties in social networks. Xu et. al [24] propose a trust-based mechanism to achieve collaborative privacy management. The user decides whether to publish a data item based on the aggregate opinions of all relevant users. The user updates these values according to the user's privacy loss. Moreover, users can make trade-offs between data sharing and privacy protection by adjusting the parameters of the proposed mechanism. Authors formulate the selection of parameters as a multi-armed bandit problem, and apply an upper-limit confidence strategy to solve the problem.

Consortium blockchain is a specific blockchain with multiple pre-selected nodes to establish the distributed shared database with moderate cost [10]. In our proposed system, the pre-selected nodes are participants associated with trust evaluations. We utilize smart contracts to achieve the evaluation of behavior games, which are

BSCI '22, May 30, 2022, Nagasaki, Japan.

ACM ISBN 978-1-4503-9175-7/22/05.
https://doi.org/10.1145/3494106.3528681

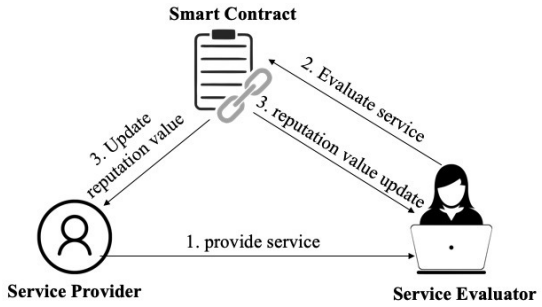

Figure 1: The high architecture of the proposed scheme.

self-executing scripts residing on blockchain with distributed multi-step processes. These smart contract-based solutions make the automation of data trust evaluation highly efficient and defend against malicious assessment. Moreover, the credibility of evaluation data is a core element. For instance, the service evaluator may turn into an insider threat due to self-purpose. The incentive mechanism designed based on game theory can courage evaluators to evaluate honestly as their optimal strategy.

The architecture of the proposed approach is shown in Fig. 1. A service evaluator refers to the participants who evaluates the transaction or blockchain-stored affairs. Smart contract considers the authenticity of the evaluation and reward or punish the service evaluator. The updated reputation values will be securely stored in the reputation blockchain. The decision on rewards and punishments is determined by the proposed game theory-based method.

The main contributions of this paper are summarized as follows:

- We propose a novel approach that utilizes consortium blockchain to establish a distributed reputation-based trust evaluation system.
- We develop a non-cooperative non-zero sum behavior game model mechanism, which encourages participants to honestly evaluate the reliability of transactions in order to improve the credibility of the information.

The rest of this paper is organized as follows. We have reviewed and summarized related work, as displayed in Section 2. Next, Section 3 presents problems, model design, and major algorithms. Finally, Sections 4 and 5 provide our evaluation results and conclusions, respectively.

2 RELATED WORK

2.1 Blockchain-based Reputation Management

Consortium blockchain has achieved great potential in “5G + IoT" fields. The authors [11] proposed exploits advantages of edge computing and blockchain to establish a privacy-preserving mechanism while considering other constraints, such as energy cost. The work [10] presented a consortium blockchain-oriented approach to solve the problem of privacy leakage without restricting trading functions. The proposed method detects the relationship between various energy transaction volumes that can be mined and other information (e.g., geographic location and energy usage), and solves the privacy problem of energy transaction users in the smart grid. In order to balance resource consumption and transaction execution efficiency, vicious researchers consider using cloud computing or edge computing [9]. Executing complex smart contracts is also a huge burden on devices with limited resources.

Fog/edge computing is the next frontier of cloud computing because it can calculate and store large amounts of data generated by IoT devices near its source. However, its functionality and deployment flexibility make fog computing vulnerable to security and privacy attacks.[9]. Therefore, to ensure security and privacy, [2] propose a trust management system based on two-way subjective logic, which allows service requesters to verify whether the service provider can provide reliable and secure services, and let The service provider checks the trustworthiness of the service provider. The proposed distributed and event-driven trust management system considers both QoS and social trust metrics to determine the trust of a fog node, and final trust value is calculated by dynamically combining information obtained from self-observation and recommendations of neighboring nodes.

IoT network integrates various heterogeneous nodes, e.g., connected devices, smart homes, etc. These smart objects communicate and collaborate with each other in a distributed and dynamic environment, facing some security challenges. Traditional trust management solutions cannot meet the new requirements of IoT, such as heterogeneity, mobility, and scalability. The authors in [17] propose a hierarchical and scalable blockchain trust management protocol that has mobility support in large-scale distributed IoT systems. The key point is that the mobile smart object can propagate trust information about the service provider to the blockchain. Therefore, all objects will have a global view of each service provider in the architecture, thus speeding up trust decisions. In addition, protocol in [17] is resistant to known malicious attacks such as bad-mouthing, ballot-stuffing and cooperative Attack, and researchers confirmed the validity of their proposal through theoretical analysis and extensive simulations.

The development of IoT has promoted the development of the Internet of Vehicles (IoV) with autonomous vehicles and roadside infrastructure as the main components. However, untrusted environment hinders the broadcast of information to improve traffic safety and efficiency. Therefore, trust establishment in IoV is a key security issue, which is constantly limited by scalability challenges. [13] proposes a blockchain-based protocol for IoV using smart contracts, physical unclonable functions, certificates, and dynamic proof of work consensus algorithms. Researchers provides a secure framework for registering trusted vehicles and preventing malicious vehicles. physical unclonable functions is used to assign a unique identity to each vehicle that establishes trust. The certificate is issued by the roadside unit to protect the privacy of the vehicle, and the dynamic proof of work consensus algorithms allows the protocol to be extended based on the incoming traffic generated by the vehicle. A security and performance analysis is proposed in order to prove the feasibility and scalability of the proposed protocol, and a case study was also discussed that confirmed that the protocol can provide excellent decentralized trust management for IoV.

In [25], vehicles can validate the received messages from neighboring vehicles using Bayesian Inference Model. Based on the validation result, the vehicle will generate a rating for each message

source vehicle. With the ratings uploaded from vehicles, roadside units (RSUs) calculate the trust value offsets of involved vehicles and pack these data into a block. Then, each RSU will try to add their blocks to the trust blockchain which is maintained by all the RSUs. By employing the joint proof-of-work (PoW) and proof-of-stake consensus mechanism, the more total value of offsets (stake) is in the block, the easier RSU can find the nonce for the hash function (PoW). In this way, all RSUs collaboratively maintain an updated, reliable, and consistent trust blockchain. Simulation results reveal that the proposed system is effective and feasible in collecting, calculating, and storing trust values in vehicular networks.

2.2 Game Theory-based Behavior Analysis

Reputation plays an important role in many communities. Game theory has shown great potential in various fields.

Kang et. al in [15] propose a two-stage soft security enhancement solution to address the challenge that voting collusion between the candidates and compromised high-stake participants in the filed of Internet of Vehicles (IoV). Their method includes two stages, miner selection and block verification, in the first stage, the author designed a reputation-based voting scheme to ensure safe miner selection. The program uses past interactions and recommendations from other tools to assess the reputation of candidates. Candidates with high reputation are selected as active miners and standby miners. In the second stage, in order to prevent internal collusion between active miners, the newly generated blocks will be further verified and reviewed by standby miners. In order to incentive standby miners to participate in block verification, this paper uses game theory to model the interaction between active miners and standby miners.

The authors also presented a secure P2P data sharing system in vehicular computing and networks in [16], researchers utilized consortium blockchain and smart contract technologies to achieve secure and efficient data storage and data sharing, in which the authors proposed a reputation based data sharing scheme with the three-weight subjective logic model considering interaction frequency, event timeliness, and trajectory similarity. The authors have demonstrated the usage of smart contracts in the dimensions of data storage and information sharing.

In traditional data sharing applications, users need to transmit data to a third party for verification. However, users are reluctant to share cybersecurity information due to concerns about mistrust, possible false information, privacy loopholes and lack of incentives. [22] proposed a blockchain-based data sharing framework “iShare”. Similar to the mining pool in Bitcoin, the attack model is part of the participants. Sub-organizations are generated, and more profits are obtained by not publishing information to another group, similar to a pool block retention attack. In the case of two groups, each group determines the number of organizations that penetrate into the other group to maximize its profit. Therefore, non-cooperative games can be used for modeling and analysis. The only Nash equilibrium is not to launch an attack, which is the best response for each group. When the number of infiltrating organizations meets different constraints, the Nash equilibrium will change. More general situations such as multi-party organization scenarios can be considered in future work.

The risk of false information and lack of incentives are also common in traditional cloud computing scenarios. Cloud users may not fully trust the calculation results returned by the cloud provider. A prior study provides evidences [18] of using a smart contract-based method to address this issue. In this solution, the cloud user uses the smart contract to pay the same Calculate the task, and then collect and compare the results from the two clouds to verify correctness. However, the smart contract-based method described above is limited to processing data on the blockchain, and trusted entities can launch attacks by modifying the data to obtain additional profits [21]. [1] proposed a distributed entity-based scheme to prevent attacks. In this solution, the coordination game can be used to analyze the interaction between voters and verifiers, and it is easy to prove that the game has two Nash equilibriums, that is, the strategies of the participants are the same.

3 MODEL DESIGN

3.1 Overview of the System

This part will briefly introduce the trust management system architecture of this paper from the five parts, trust composition, trust propagation, trust aggregation, trust update and trust formation, using the method described from [14]. In our paper, if the consumer of the service does not evaluate the service after receiving the service, it will not have any influence on the system. Aimed to facilitate the analysis of this paper, we assume that all service consumers participate in service evaluation, service estimators and service consumers in the following represents the same meaning.

In order to meet the needs of service requesters and maximize application performance, it is important to evaluate the trustworthiness of service providers in the IoT environment. On one hand, the trustworthiness of service providers cannot be separated from their own good reviews, and on the other hand, they cannot be separated from true evaluations. Therefore, the reputation proposed in this paper also includes two aspects, including quality of service (QoS) trust and social trust. QoS trust in general refers to performance and is measured by competence, cooperativeness, reliability, task completion capability, etc. Social trust derives from social relationship between owners of IoT devices and is measured by intimacy, honesty, centrality [3], and connectivity.

In our blockchain-based reputation system, both metadata and trust values are stored, and are shared among participants with identity authenticated. Based on the blockchain to establish a trust management system, our trust propagation method is decentralized, and all the storage, concentration, and update of trust data are stored in the system in a distributed manner. At the same time, all calculations and subsequent rewards and punishments will be based on smart contracts, which largely resists traditional attacks on centralized control centers.

Trust aggregation refers to the collection of trust evidence through self-observation or feedback from participants. In this paper, we use a weighted sum method to aggregate the reputation value, and the weight is dynamic and will change with past experience and user preferences.

Trust update concerns when trust is updated. In general, there are two schemes - event-driven scheme and time-driven scheme. In this work, We consider a more practical situation, that is, the storage

and update of the blockchain itself requires greater consumption. Therefore, we consider two dimensions of trust update, frequency and time, which are described in details in Section 3.5.

3.2 Design Goals

Trust, as a soft security measure, is particularly suitable for service-oriented IoT systems, because humans' inherent IoT devices may be malicious for their own benefit. Malicious users can also collude to dominate the service provider market. We enumerate the possible threats to IoT systems [6, 12]. The target of this article is to protect against the following attacks:

(1) Self-promotion attacks: a malicious service provider can promote its own service by providing good recommendations for itself.
(2) Bad-mouthing attacks: a malicious service provider can ruin the trust of a well-behaved participant by providing bad recommendations against it so as to decrease the chance of that node being selected for service.
(3) Ballot-stuffing attacks: a malicious service provider can boost the trust of malicious participants so as to increase the chance of themselves being selected. This is a form of collusion recommendation attacks, i.e., it can collaborate with other malicious nodes to boost the trust of each other.
(4) Opportunistic service attacks: a malicious service provider can provide good service to gain high reputation opportunistically especially when it senses its reputation is dropping because of providing bad service. With good reputation, it can effectively collude with other bad node to perform bad-mouthing and ballot-stuffing attacks.

The "ghost commentator problem" is a concrete manifestation of the above problem. The attacker prepares a certain amount of funds to buy out the service evaluation party participating in the transaction, so that the service evaluation party will evaluate according to the attacker's wishes. We present this attacking method also as the "bribe".

3.3 Reputation-based Trust Composition

In this system, when participant i has a transaction with participant j (indicated by the symbol T_{ij}), participant i will score participant j based on the satisfaction of this transaction. Participant i will score participant j based on the satisfaction of this transaction, satisfaction means that participant i's satisfaction with participant j increases by 1; Unsatisfied means that participant i is dissatisfied with participant j and $UnSa_i j$ plus one. After multiple rounds of transactions, the evaluation value of participant i to participant j is $Ev_{ij} = Sa_{ij} - UnSa_{ij}$, and Ev_{ij} is standardized to obtain the partial trust value of i to j. The partial trust value is shown in Eq. (1):

$$L_{\mathrm{ij}} = \frac{Max\left(\mathrm{Ev_{ij}}, 0\right)}{\sum_{k=1}^{\mathrm{n}} Max\left(\mathrm{Ev_{ik}}, 0\right)} (\mathrm{k} \neq \mathrm{i}). \tag{1}$$

where n represents the total number of participants in the system. For ease of description, in this paper, partial trust value and local trust value have the same meaning. From this, we can obtain the partial trust vector of participant i $L_i = \{L_{ij} \mid j \in (1, \ldots, n)\}$, obviously $L_{ii} = 1$. In this evaluation system, each participant will regularly aggregate its own partial trust vector to the rating platform server. A partial trust matrix TL is formed, where $TL_{ij}(i, j \in (1, \ldots, n))$ represents the partial trust value of participant i to participant j, obviously $TL_{ij} = L_{ij}$.

In addition, as in real world, participant i not only needs to consider its own trust value to participant j before transacting with participant j, but also consider related parties (participants who have had transactions with participant i) Evaluation of party j. Let participant i's associated participant k's partial trust value L_{kj} to participant jj's influence on participant i be determined by participant i's local trust value LocVaik to participant k.

From this, we can obtain the global trust value G_{ij} of participant i to participant j, as shown in Eq. (2).

$$G_{ij} = func(k) = \sum_{k=1}^{n} L_{ik} \cdot L_{kj} = \sum_{k=1}^{n} TL_{ik} \cdot TL_{kj}. \tag{2}$$

The global trust vector of participant i to other participants $T_i = G_{ij} \mid j \in (1, \ldots, n).T_i = (G_{ij} \mid j \in (1, \ldots, n)$.

It can be seen from this that participant i will comprehensively calculate its transaction trust value Re_{ij} for participant j before trading with participant j, as shown in Eq. (3):

$$\begin{aligned} Re_{ij} &= \alpha \cdot L_{ij} + (1-\alpha) \cdot G_{ij} \\ &= \alpha \cdot TL_{ij} + (1-\alpha) \cdot \sum_{k=1}^{n} TL_{ik} \cdot TL_k. \end{aligned} \tag{3}$$

where α is the self-confidence coefficient of participants , and $\alpha \in (0, 1)$. When participant i needs to purchase a service, it will choose the one with the highest transaction trust value from the participants that provide similar services for the transaction.

In addition, for a new participant i who has no transaction record, the platform server initializes its partial trust value to others to $\frac{1}{2}$.

3.4 Behavioral Game between the Service Provider and Estimator

Under the background of information asymmetry, the perceived credit system cannot fully guarantee the authenticity of the evaluation information. On one hand, the information provider may provide false evaluation information and mislead the users' transaction decision; On the other hand, the service provider will identify the information based on its own experience and make the correct transaction decision as much as possible. In this process, it is the a non-zero sum game relationship between the gains and losses of the two sides of the game. The behavioral game model based on the proposed credit system is shown in Fig.1.

The objects participating in the game in the model are divided into two categories, one is the service provider; the other is the service estimator. Both game objects in the model adopt mixed strategies. Any service provider i may accept the evaluation information provided by the credit platform with probability p_i, or reject the evaluation information provided by the platform with probability $(1 - p_i)$. At the same time, any information provider j may publish a false evaluation with a probability of q_j, or may publish a true evaluation with a probability of $(1 - q_j)$. From above, we can get the strategy matrix of the proposed model, namely $P_i = (p_i, 1 - p_i)$ and $Q_j = (q_j, 1 - q_j)$. In order to construct the payoff function of the participating objects of this model, we quantify

Table 1: Game payoff matrix of model participants.

Strategies	Evaluate honestly	Evaluate dishonestly
Serve honestly	(Us_{hh}, Ue_{hh})	(Us_{hd}, Ue_{hd})
Serve dishonestly	(Us_{dh}, Ue_{hh})	(Us_{dd}, Ue_{dd})

the profit results of both parties in the game, and focus on the key factors that affect the value of the payoff.

The formal representation of the game model is as follows in Eq. (4):

$$\begin{aligned} N &= \{ServiceProvider(i), ServiceEstimator(j)\} \\ S &= \{Serve\ honestly, Serve\ dishonestly, \\ &\quad Evaluate\ honestly, Evaluate\ dishonestly\} \\ U &= \{(Us_{hh}, Ue_{hh}), (Us_{hd}, Ue_{hd}), \\ &\quad (Us_{hh}, Ue_{hh}), (Us_{hh}, Ue_{hh})\}. \end{aligned} \tag{4}$$

The game payoff matrix of the participants in this model is shown in Tab. 1.

For the convenience of analysis, we adopt the classic 1 *dollar* game model in this part, that is, the final maximum utility is a fixed value. The specific meaning can be seen in [23]. The following describe each utility separately.

(1) When the service provider chooses to provide real services and the service evaluator evaluates according to his own wishes, the utilities of the two persons is their deposit, ($Us_{hh} = 1, Ue_{hh} = 1$). This is the ideal target we want, that is, we need the Nash equilibrium to converge to this solution.
(2) When the service provider chooses to provide dishonest services and the service evaluator truly evaluates it according to his wishes, the service provider will suffer losses, that is, lose his deposit, and the service evaluator will recover his deposit, ($Us_{dh} < 1, Ue_{dh} = 1$). This situation can be understood as a failure of the attack.
(3) When the service provider chooses not to provide the honest service, and the service evaluator does not evaluate according to the real situation either, the utilities of the two persons is greater than their deposit($Us_{dd} > 1, Ue_{dd} > 1$). In this case, the attack is successful and we need to prevent this kind of situation.
(4) When the service provider chooses to provide honest services, but the service evaluator fails to evaluate according to his own wishes, the service provider will lose its deposit. In this case, ($Us_{hd} < 1, Ue_{hd} < 1$), the service provider will seriously damage the market development, where the attacker should be severely punished.

We assume that all participants are intelligent. When neither service providers nor evaluators provide the honest service, the benefit is the greatest. If the game is played only once, the current situation can be regarded as a “prisoner’s dilemma". However, the transaction will obviously continue. From a long-term perspective, the current Nash equilibrium is not the global optimal solution. We need to model the long-term benefits and establish an incentive mechanism based on smart contracts, which will be described in detail in next part.

3.5 Deposit-based Incentive Mechanism

In order to resit the attack mentioned in section 3, our work proposes an deposit based incentive mechanism.A system parameter analysis is proposed to measure the probability that an attacker with limited funds can change the result in Section 4. The context in this and next section will show that there is a Nash equilibrium under these conditions, that is, all rational participants are forced to act honestly.

Each transaction will be considered as a proposal submitted to the proposal pool. Submitter shows an transaction with the funding, presented by Dv. Before all information providers conduct their transaction evaluations, participants need to pay a certain deposit.

The submitter submits the proposal and is ready to accept the evaluation of the evaluator, and then the evaluator will bet on the bet. Anyone can join or exit the system anytime and anywhere. Commonly, the certification result is the cumulative sum of the number of certifications. Each service evaluator will compare it with the average evaluation of other subjects. Specifically, we propose a deviation to describe the distance between different service evaluators and the evaluation average. For example, if Alice’s score is 0.9, Bob’s score is 0.2, and the overall average score is 0.6, then Alice’s deviation is $(|0.9 - 0.8|)/0.8 = 0.125$, and Bob’s deviation is $(|0.2 - 0.8|)/0.8 = 0.75$, different users have different deviations, starting from reality, we designed a layered feedback mechanism. Divide the deviation into three levels, lower than the tolerance value, tolerable value to trigger, and higher than trigger. These three levels correspond to honest evaluation, accidental deviation and malicious evaluation respectively.

In response to malicious estimators, we adopted the "nuclear option" strategy proposed in repeat game [7]. Our solution is directly reducing its reputation by half, which will cause a huge obstacle to its future participation in business and protect the normal operation of the entire system.

On the other hand, if the service provided by the service party is unstable and the user’s evaluation deviation does not exceed the trigger, it is tolerable to withhold the deposit, and it will not have much impact if the honest evaluation is maintained. Intuitively, the proposed system encourages estimator to bet on proposal with high confidence.

The process is briefly summarized as follows:

1. A user in the system generates transaction information, creates a proposal and submits it to the blockchain network through the client, and places a funding D_v.

2. Other users make judgments based on their own information and submit deposits to participate in evaluation.

3. The smart contract collects the deposits of each player until it reaches the condition for outputting the game result.

4. The output based on game theory will be given by smart contract.

5. Obtain the outcome of transaction according to the player’s betting situation, then return the rewards or punishments respectively.

The main process of deposit-based incentive mechanism algorithm is shown in Algorithm. 1.

In detailed, the system description are as follows:

Algorithm 1: Deposit-based incentive mechanism algorithm.

Input: estimatorID id, proposal p, deposit d, grade e, estimatorReputation r
Output: estimatorReputation r_{upd}

Init:set depositTotal = 0, Timer = thresholdTime and gradeTotal = 0;
while *True* **do**
  receive(d,e);
  gradeTotal += e;
  depositTotal += d;
  Timer- -;
  cnt++;
  if *depositTotal > D or timer == 0* **then**
    break;
  end
end
gradeAverage = gradeTotal/cnt;
foreach *g in g[]* **do**
  if $\frac{|g-gradeAverage|}{gradeAverage} < deviation_{endure}$ **then**
    r_upd = r+1
  else
    if $\frac{|g-gradeAverage|}{gradeAverage} < deviation_{trigger}$ **then**
      r_{upd} = r
    else
      r_{upd} = r * $\frac{1}{2}$ // nuclear option
    end
  end
end

First of all, there are many proposals constructed by submitters are waited to be certified, which is denoted by P, and our work assume that its size $n = |P|$, and each proposal is associated with a Dv used to motivate users to verify his/her proposal. The detailed description of the construction of P is not discussed in this paper. It can be constructed in many different ways, and this will be our follow-up work.

Then one player submit its own deposit through the smart contract. $s_{i,r,v}$ denotes the amount that player i placed a stake to get the opportunity for evaluate the proposal p_j, here r is not yet decided, because it will be assignment by smart contract randomly. Note that r is chosen from $[1, |P|]$. How to obtain a real random number in smart contract on blockchain network is a active research in recent days, and this work do not discuss this issue. For ease of analysis, the deposit evaluators placed is all d, At the same time, the evaluator submits a sealed estimation grade $e \in (0, 1)$. This can be accomplished using a commit-reveal scheme in which the voter commits to a hash of their vote concatenated with a nonce and later reveals the vote and nonce.

Considering that the blockchain consumes a lot of energy and time when rewriting data, we do not adopt a real-time update method for the update of reputation. Taking into account different types of transactions, we have considered the frequency of transactions and the cycle of reputation updates. Specifically, in the first case, if a large number of evaluations are generated for a certain transaction in a short period of time, that is, the volume of deposit in a short period of time exceeds the Deposit threshold D, the system will trigger an update of the reputation. In the second case, there are new transactions but the transaction volume is relatively small, but a long time has passed. In order to provide the user of the service with the latest reference in time, the system will also trigger the update of the reputation.

For the penalty mechanism, we refer to the "nuclear option" in the repeated game and refer to work [7]. If an appraiser's score is not much different from the average of everyone else (degree of deviation), we just deduct part of its deposit. If there is a big deviation, we decide to implement the "nuclear option" to halve its overall reputation. The halving process will directly make the service appraiser unable to purchase the service later. Consider that the service provider may be unstable. This will also directly affect the value of its social trust, thereby affecting other service users.

3.6 Threat Model

Consider a malicious participant attempting to launch the attacks mentioned in Section 3. Assuming that a malicious participant attempts to control the result of proposal j, and the malicious participant owns $Bribe$ currency and attempts to buy a certain amount of estimator for manipulation the result.

Every time estimator receives a proposal, the action is processed by a smart contract. Due to the decentralized nature of the blockchain, it is not possible for an attacker to only attack the intermediate platform to achieve its goal. Assuming that the proposal randomly allocated by the smart contract belongs to the Bernoulli distribution. For ease of analysis, we assume that all estimators will place a deposit of d when participating in the evaluation, and now we calculate the probability of successful implementation of the attack.

It is known that it would be $\frac{D}{d}$ estimations on each proposal in total. Mark $X \sim B(\frac{D}{d}, p_r)$, $\frac{D}{d} = t$, $B(\cdot)$ presents a binomial random variable. It can be known from the principle of binomial distribution that: $P(X = x) = C_t^x {p_r}^x (1 - p_r)^{t-x}$. So the probability of successful attacking implementation is shown in Eq. (5), where p_r is the probability that the estimation belong to adversary.

$$\begin{aligned} P_{manipulate} &= P\left[B(\frac{D}{d}, p_r) > \frac{1}{2} \times \frac{D}{d}\right] \\ &= \sum_{i=0}^{\frac{t}{2}} C_t^i {p_r}^i (1 - p_r)^{t-i}. \end{aligned} \tag{5}$$

Assume the adversary's $Bribe$ is an integer multiple of d, then at most $\frac{Bribe}{d}$ estimations belong to the malicious participant. Since the smart contract randomly sends a proposal to the estimator, so that the probability of a random estimation is belong to adversary is as Eq. (6):

$$p_r = \frac{\frac{Bribe}{d}}{\frac{D}{d}} = \frac{Bribe}{D}. \tag{6}$$

4 EXPERIMENT AND EVALUATIONS

All the experiments were run on the machine with Intel(R) Core(TM) i7-4710HQ CPU @ 2.50GHz and 16GB memory.

For the threat model described in the previous section, we simulated the probability of a malicious participant's success in an attack. We consider the two situations in this part:

One situation is that, the service provider knows the $deposit_{amount}$ set by the system, which means the attacker knows the total volume of evaluation deposit required at first. Then the malicious participant can calculate how much money he needs, to launch an attack in order to obtain a greater probability of successful manipulating. To simulate this situation, We set the input of the experiment including $deposit_{amount}$, d, $Bribe$, and $|P|$, where $deposit_{amount}$, d and $|P|$ are fixed. The output refers to the probability of the attacker successfully tampering with the evaluation result. The $deposit_{amount}$ represents the total amount of deposit required when evaluating a service; d represents the deposit that each evaluator needs to pay when participating in the evaluation. For ease of calculation, we set the value of d to 1 unit; $Bribe$ refers to the principal used by malicious participants to launch an attack, and $|p|$ represents the number of services to be evaluated in the current service evaluation pool. Here we analyze a single service and set the value of $|p|$ to 1. The experimental results are shown in the fig. 2.

The figure 2 describes the relationship between the node's successful probability of doing evil and the node's investment in the case of a fixed total deposit. The deposit that a node normally needs to pay is 1 unit. It can be seen that when the node's investment is less than about five times the total deposit, the probability of the node's successful evil is almost zero. In fact, as long as the malicious node invests more than 1 time, he will lose money, and he needs to have a 50% probability of success, and he needs to deposit amout 9 times the unit of funds, which is 180 units, so the penalty mechanism we designed is very Effective. In addition, the comparison between the solid line and the dashed line in the figure shows that the larger the total amount of deposits, the higher the capital requirements for the node to successfully commit evil.

Another situation is that, the system administrator decides to set the total amount of the deposit after knowing the funds of the malicious party, to make the probability of the attacker launching a "ghost comments" attack to tamper with the result approaches as close to zero as possible. That is, the system administrator knows the $Bribe$ first. To facilitate analysis, we set $Bribe$ to 1, and we also made a comparison simulation when $Bribe$ is 2. Then the administrator can calculate how much he needs to set the threshold of the deposit to be able to resist attacks by malicious participants. To simulate this situation, We set the input of the experiment including $deposit_{amount}$, d, $Bribe$, and also $|P|$. The output is the same as above, refers to the probability of the attacker successfully tampering with the evaluation result. Differently, in this situation $Bribe$ is fixed, represents the total amount of funds the attacker has, and the deposit threshold $deposit_amount$ becomes the independent variable. The settings and assumptions of d and $|P|$ are the same as the previous experiment. The experimental results are shown in the fig. 3.

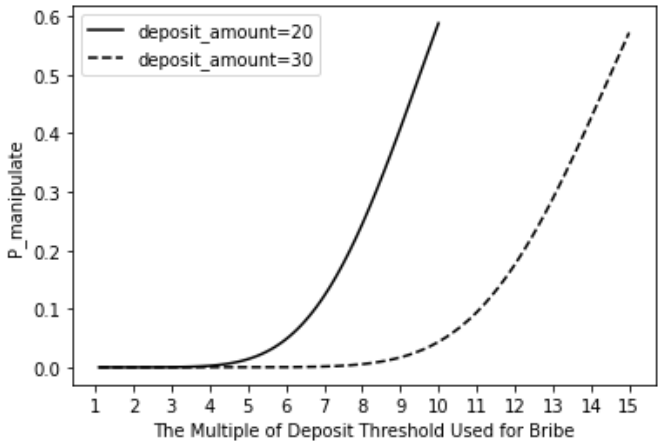

Figure 2: The relationship between successful tampering and the multiple of the deposit amount used for Bribe under limited total amount of deposits.

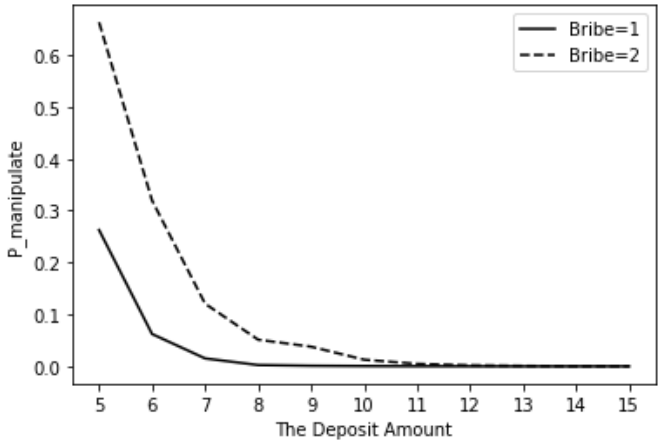

Figure 3: The probability of successful attack varies with the total amount of deposit under limited bribe.

The figure 3 shows the relationship between the probability of the attacker's success and the total amount of deposit when the amount of funds invested by the malicious party is limited.

It can be seen from the figure that the higher the total deposit required by the system for credit evaluation, the lower the probability of an attacker successfully carrying out a malicious attack.

This means that when he launches an attack, that is, when he is the only one who makes a false evaluation, as long as there are 8 people participating in the evaluation at the same time, the probability of success will drop to less than 10%. When there are 11 people participating in the evaluation at the same time, the probability of success will drop to less than 1%, which proves that the mechanism we designed can resist the attacks in section 3.

5 CONCLUSIONS

This work proposed a scheme of blockchain-based trust management using game theory for 5G-enabled distributed IIoT system to solve the ghost commentator problem. We developed the blockchain-enabled reputation-based evaluation system that modelized evaluation strategies by using behavioral game theory. The incentive mechanism in our work was a deposit-based scheme that relied on the implementation of smart contract. We showed the effect of parameter setting on the system and our experiments illustrated that using the deposit-based smart contract mechanism could crackdown malicious participants who intended launch ghost commentator attacks. Future work would further address the dynamic execution scenarios in which a consistent changing proposal list would be researched.

ACKNOWLEDGMENTS

This work was supported by the National Key Research & Development Program of China (No. 2020YFC2006204) and the National Natural Science Foundation of China (No. 62172042)

REFERENCES

[1] J. Adler, R. Berryhill, A. Veneris, Z. Poulos, N. Veira, and A. Kastania. 2018. Astraea: A Decentralized Blockchain Oracle. In *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. 1145–1152. https://doi.org/10.1109/Cybermatics_2018.2018.00207

[2] E. Alemneh, S. Senouci, P. Brunet, and T. Tegegne. 2020. A two-way trust management system for fog computing. *Future Generation Computer Systems* 106 (2020), 206–220.

[3] V. Batagelj. 1996. Centrality in Social Networks. *AMS Subj.Class* 90 (01 1996), 92–30.

[4] D. Berdik, S. Otoum, N. Schmidt, D. Porter, and Y. Jararweh. 2021. A survey on blockchain for information systems management and security. *Information Processing & Management* 58, 1 (2021), 102397.

[5] U. Bodkhe, S. Tanwar, K. Parekh, P. Khanpara, S. Tyagi, N. Kumar, and M. Alazab. 2020. Blockchain for Industry 4.0: A Comprehensive Review. *IEEE Access* 8 (2020), 79764–79800. https://doi.org/10.1109/ACCESS.2020.2988579

[6] I. Chen, J. Guo, and F. Bao. 2016. Trust Management for SOA-Based IoT and Its Application to Service Composition. *IEEE Transactions on Services Computing* 9, 3 (2016), 482–495. https://doi.org/10.1109/TSC.2014.2365797

[7] P. Daian, S. Goldfeder, T. Kell, Y. Li, X. Zhao, I. Bentov, L. Breidenbach, and A. Juels. 2020. Flash Boys 2.0: Frontrunning in Decentralized Exchanges, Miner Extractable Value, and Consensus Instability. In *2020 IEEE Symposium on Security and Privacy, SP 2020, San Francisco, CA, USA, May 18-21, 2020*. IEEE, 910–927. https://doi.org/10.1109/SP40000.2020.00040

[8] E. Friedman, P. Resnick, and R. Sami. 2007. Manipulation-resistant reputation systems. *Algorithmic Game Theory* 677 (2007).

[9] K. Gai, J. Guo, L. Zhu, and S. Yu. 2020. Blockchain Meets Cloud Computing: A Survey. *IEEE Communications Surveys Tutorials* 22, 3 (2020), 2009–2030. https://doi.org/10.1109/COMST.2020.2989392

[10] K. Gai, Y. Wu, L. Zhu, M. Qiu, and M. Shen. 2019. Privacy-Preserving Energy Trading Using Consortium Blockchain in Smart Grid. *IEEE Transactions on Industrial Informatics* 15, 6 (2019), 3548–3558. https://doi.org/10.1109/TII.2019.2893433

[11] K. Gai, Y. Wu, L. Zhu, Z. Zhang, and M. Qiu. 2020. Differential Privacy-Based Blockchain for Industrial Internet-of-Things. *IEEE Transactions on Industrial Informatics* 16, 6 (2020), 4156–4165. https://doi.org/10.1109/TII.2019.2948094

[12] K. Hoffman, D. Zage, and C. Nita-Rotaru. 2009. A Survey of Attack and Defense Techniques for Reputation Systems. *ACM Comput. Surv.* 42, 1, Article 1 (Dec. 2009), 31 pages. https://doi.org/10.1145/1592451.1592452

[13] U. Javaid, M. N. Aman, and B. Sikdar. 2020. A Scalable Protocol for Driving Trust Management in Internet of Vehicles With Blockchain. *IEEE Internet of Things Journal* 7, 12 (2020), 11815–11829. https://doi.org/10.1109/JIOT.2020.3002711

[14] A. Jøsang, R. Ismail, and C. Boyd. 2007. A survey of trust and reputation systems for online service provision. *Decision support systems* 43, 2 (2007), 618–644.

[15] J. Kang, Z. Xiong, D. Niyato, D. Ye, D. I. Kim, and J. Zhao. 2019. Toward Secure Blockchain-Enabled Internet of Vehicles: Optimizing Consensus Management Using Reputation and Contract Theory. *IEEE Transactions on Vehicular Technology* 68, 3 (2019), 2906–2920. https://doi.org/10.1109/TVT.2019.2894944

[16] J. Kang, R. Yu, X. Huang, M. Wu, S. Maharjan, S. Xie, and Y. Zhang. 2019. Blockchain for Secure and Efficient Data Sharing in Vehicular Edge Computing and Networks. *IEEE Internet of Things Journal* 6, 3 (2019), 4660–4670. https://doi.org/10.1109/JIOT.2018.2875542

[17] D. E. Kouicem, Y. Imine, A. Bouabdallah, and H. Lakhlef. 2020. A Decentralized Blockchain-Based Trust Management Protocol for the Internet of Things. *IEEE Transactions on Dependable and Secure Computing* (2020), 1–1. https://doi.org/10.1109/TDSC.2020.3003232

[18] P. Liu and W. Zhang. 2018. A New Game Theoretic Scheme for Verifiable Cloud Computing. In *2018 IEEE 37th International Performance Computing and Communications Conference (IPCCC)*. IEEE, 1–8.

[19] Z. Liu, N. Luong, W. Wang, D. Niyato, P. Wang, Y. Liang, and D. Kim. 2019. A survey on applications of game theory in blockchain. *arXiv preprint arXiv:1902.10865* (2019).

[20] M. Nojoumian, A. Golchubian, L. Njilla, K. Kwiat, and C. Kamhoua. 2018. Incentivizing blockchain miners to avoid dishonest mining strategies by a reputation-based paradigm. In *Science and Information Conference*. Springer, 1118–1134.

[21] J. Peterson, J. Krug, M. Zoltu, A. Williams, and S. Alexander. 2019. Augur: a Decentralized Oracle and Prediction Market Platform (v2. 0). *Whitepaper, https://augur. net/whitepaper. pdf* (2019).

[22] D. B. Rawat, L. Njilla, K. Kwiat, and C. Kamhoua. 2018. iShare: Blockchain-Based Privacy-Aware Multi-Agent Information Sharing Games for Cybersecurity. In *2018 International Conference on Computing, Networking and Communications (ICNC)*. 425–431. https://doi.org/10.1109/ICCNC.2018.8390264

[23] M. Shubik. 1971. The dollar auction game: A paradox in noncooperative behavior and escalation. *Journal of conflict Resolution* 15, 1 (1971), 109–111.

[24] L. Xu, C. Jiang, N. He, Z. Han, and A. Benslimane. 2019. Trust-Based Collaborative Privacy Management in Online Social Networks. *IEEE Transactions on Information Forensics and Security* 14, 1 (2019), 48–60. https://doi.org/10.1109/TIFS.2018.2840488

[25] Z. Yang, K. Yang, L. Lei, K. Zheng, and V. C. M. Leung. 2019. Blockchain-Based Decentralized Trust Management in Vehicular Networks. *IEEE Internet of Things Journal* 6, 2 (2019), 1495–1505. https://doi.org/10.1109/JIOT.2018.2836144