IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 13, 2020, accepted March 2, 2020, date of publication March 6, 2020, date of current version March 17, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2979022

A Reliable IoT Edge Computing Trust Management Mechanism for Smart Cities

BO WANG[1,2,3], MINGCHU LI[1,3], XING JIN[1,3,4], AND CHENG GUO[1,3]

[1]School of Software Technology, Dalian University of Technology, Dalian 116620, China
[2]School of Applied Technology, University of Science and Technology Liaoning, Anshan 114051, China
[3]Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian 116620, China
[4]School of Cyberspace, Hangzhou Dianzi University, Hangzhou 310018, China

Corresponding author: Bo Wang (wangboustl@126.com)

This work was supported in part by the National Nature Science Foundation of China under Grant 61572095 and Grant 61877007.

ABSTRACT With the development of the Internet of Things (IoT) technology, many end-users participate in the smart city through their own intelligent mobile devices (such as personal wearable devices, smartphones.) or sensors. The main challenge of the device sensing layer in the edge computing system of the IoT in the smart city is to select the trusted participants. Because not all the intelligent devices of the IoT are trustworthy, some intelligent devices of the IoT may maliciously damage the network or services and affect the service quality of the system. On this basis, an intelligent device selective recommendation mechanism based on the dynamic black-and-white list was proposed to solve the problem of selecting trusted participants to improve the service quality of the edge computing system of the IoT in the smart city. We introduced the evolutionary game theory to theoretically qualitatively study the validity and stability of the trust management mechanism proposed in this paper. The Lyapunov theory was used to prove the validity and stability of the trust management mechanism. The effectiveness of the trust management mechanism was verified by the actual scenario of the personal health monitoring management system and the air-quality monitoring and analysis system in the smart city environment. Experiments showed that the trust management mechanism proposed in this paper has a significant role in promoting the cooperation of multi intelligent devices in the IoT edge computing system. It more reliably resists the malicious attacks to service providers and is suitable for the large-scale IoT edge computing system in the smart city.

INDEX TERMS Edge computing, Internet of Things, malicious attack, smart city, trust management mechanism.

I. INTRODUCTION

With the development of the Internet of Things (IoT) technology [1], [2] and 5G communication technologies, mobile edge computing (MEC) [3]–[5], [43] has become a hot spot in the research of industry and academia. MEC is developed from mobile cloud computing technology to process data at the edge of the network, which reduces service request response time, reduces mobile device energy consumption, reduces network bandwidth, and ensures data security. MEC has a wide range of applications, such as computation offloading [6]–[8], big data storage [9], face recognition-based video analysis, intelligent transportation [10], smart cities [11], [12], healthcare [13], collaborative mobile edge computing [14], [15], and so on.

The associate editor coordinating the review of this manuscript and approving it for publication was Kashif Sharif.

A smart city [19] uses advanced technologies such as IoT, big data, cloud computing, and other technologies to realize smart management and operation of the city, and thus create a better lifestyle for people in the city. It includes smart homes, intelligent transportation, intelligent healthcare, intelligent weather.

In the IoT edge computing system of the smart city, many smart devices in the sensing layer exist in more challenging and complex scenarios. These intelligent devices interact with each other in various areas of the city, such as roads, buildings, or stadiums. Each end-user participates in the smart city system through his or her intelligent mobile devices (e.g., personal wearable devices, smartphones.) or sensors. These smart terminal devices submit the data to the edge service provider for processing. Due to the limited resources of the edge service provider, when the user sent a large amount of data to the edge service provider at the same time,

it may cause the delay of the edge service provider. Therefore, the smart terminal device offloads the task to another smart device with idle resources, reducing service latency due to extensive data submitted. The main challenge of the device sensing layer in the edge computing environment of the IoT in the smart city is to choose the trusted participants because not all the intelligent devices of the IoT are trustworthy, some smart devices of the IoT may maliciously damage networks or services and affect the service quality of the system. Although these smart devices significantly improve the quality of life of the people and provide more convenience for our lives. However, there are various security risks and malicious attacks in the edge computing system of the IoT in the real smart city [34]–[41]. Therefore, the service provider of the edge computing system of the IoT in the smart city must adopt the trust management mechanism to ensure the quality of the assistance behavior of the intelligent devices and improve the satisfaction of the users. How to build an effective trust management mechanism for smart device collaboration behavior and ensure the service quality of the system is the critical content of this paper.

Some existing work has studied the trust management mechanism. However, there are also some problems with the trust management mechanism of edge computing. Most trust value calculations do not adequately consider the trust value between smart devices and edge service providers and the trust value between smart devices. Besides, many previous studies have assumed that all participants in the system adopted the proposed models, which ignores the participant's ability to choose independently. We review and compare the actual work in Section II.

In this paper, we propose a smart device selective recommendation mechanism based on the dynamic black-and-white list to solve the problem of selecting trusted participants. We verified through experiments that the proposed management mechanism has excellent performance. In this paper, we introduce evolutionary game theory to theoretically study the validity and stability of the trust management mechanism. We use the Lyapunov theory to prove the validity and stability of the trust management mechanism proposed in this paper. The effectiveness of the trust management mechanism is verified by the scenario of the personal healthcare monitoring management system and the air-quality monitoring and analysis system in the smart city.

This paper has the following contributions.

(1) We adopt the trust calculation method based on multi-intelligent devices and multi-edge centers under the framework of the smart city. Consider the trust between smart devices and the trust relationship between edge service providers and smart devices.

(2) We propose an evaluation mechanism of personalized context content perception.

(3) We propose a personalized device selective recommendation mechanism based on the dynamic black-and-white list to improve the service quality of the IoT edge computing systems in the smart city.

(4) We introduce evolutionary game theory to theoretically study the validity and stability of the trust management mechanism. We use the Lyapunov theory to prove the validity and stability of the trust management mechanism proposed in this paper. The effectiveness of the trust management mechanism is verified by the scenario of the personal health monitoring management system and the air-quality monitoring and analysis system in the smart city.

The paper has been organized as follows: Section II presents a review of related studies. Section III describes the IoT edge computing trust management framework for smart cities based on the cloud platform and presents the system architecture, while Section IV outlines the details of the trust management mechanism. Section V evaluates the model by deploying a scenario and performs tests. Finally, the main conclusions about the research and future lines of work are presented.

II. RELATED WORK

With the development of IoT technology, many end-users participate in the edge computing system of IoT in the smart city through intelligent mobile devices (such as personal wearable devices, smartphones.) and sensors. One of the significant challenges in the edge computing system of the IoT in the smart city is choosing trusted devices because not all devices are trustworthy, and some IoT intelligent devices may maliciously damage networks or services and affect system service quality. Many kinds of literature have studied the security and trust problems in different architectures and proposed corresponding solutions, for example, authentication [44]–[48], security defense [49], [50], trust management [51]–[55].

In particular, Kamvar *et al.* [23] proposed assigning a unique global trust value to each peer based on the upload history of the peer. A distributed security method based on Power iteration is proposed to calculate the global trust value. Peers use these global trust values to select peers. The network can effectively identify malicious peers and isolate them from the system. Su *et al.* [28] proposed a flexible trust management tool ServiceTrust that provides network service quality sensitivity and protection against attacks. ServiceTrust consists of three unique features. First, it encapsulates feedback that is sensitive to network service quality through a multi-scale scoring scheme and translates user behavior into a local trust algorithm. Second, ServiceTrust uses the similarity of two user feedback behaviors to measure local trust values and uses pairwise feedback similarity to aggregate local trust values into global trust algorithms. ServiceTrust measures the contribution score of the local trust value of the participant's global trust. Finally, the use of pairwise feedback similarity weighted trust propagation further enhances the robustness of global trust calculations to malicious or sparse feedback.

A personalized pre-trust management model, PersonalTrust, was proposed by Li *et al.* [29] PersonalTrust leverages individualized pre-trust delivery ability. Assume that in addition to those pre-trust peers selected by the system, each peer

can choose its personalized pre-trust peer based on its interactions with other peers on the network. By leveraging pre-trust passability, obtain a pre-trust matrix containing information to ensure that peer's trust will only spread within the circle of trusted friends. Also, PersonalTrust automatically updates the pre-trust matrix. Fan *et al*. [31] proposed reliable trust management, GroupTrust, to provide secure trust management in the case of dishonest ratings, malicious masquerading, and malicious collusion. The GroupTrust solution is based on the feedback credibility of pairwise similarity to improve the resiliency of trust value calculations for the case of dishonest ratings, defining trust propagation thresholds to control how trust is propagated.

Yuan *et al.* [32] proposed a hybrid trust calculation method for IoT edge device evaluation. The framework consists of three different parts, namely global trust calculation based on multi-source feedback, trust evaluation based on IoT edge device cooperation, and trust factor calculation algorithm based on objective information entropy theory. To effectively calculate the trust, the entire process is done on the edge device. When calculating the trust and incentives of a particular IoT edge device, the two factors considered are the interaction with other devices and the quality of service provided by the IoT edge device.

He *et al*. [16] proposed a social trust scheme that enhances security. When considering trust-based mobile social networks with mobile edge computing, caching, and device-to-device, a new in-depth reinforcement learning approach was adopted to make decisions to optimize the allocation of network resources automatically. Huang *et al*. [17] proposed a distributed vehicle edge computing reputation management system. The system employs a vehicle edge computing server to perform local reputation management tasks for the vehicle. The multi-weight subjective logic method is used to update the reputation value of the distributed vehicle edge computing reputation management system. The system collects, weights, and aggregates all reputation's values in a region to form a final local reputation update value. The service provider optimizes the resource allocation in the computing offloading by considering the reputation value of the vehicle, which plays a role in security protection and network efficiency. Goh *et al*. [18] designed three schemes to solve the integrity challenges in edge computing. Each scheme provides different security functions, and these schemes are suitable for different application requirements and resource configurations. Yuan and Li [20] proposed a multi-source feedback-based edge computing trust computing mechanism. Due to the use of the multi-source feedback mechanism for global trust computing, this trust computing scheme was more reliable, resisting malicious attacks on malicious feedback providers. The authors also used a lightweight trust evaluation mechanism to achieve collaborative work between network devices. This mechanism is beneficial to low-cost trust calculation algorithms and is suitable for large-scale edge computing. Xu *et al*. [21] proposed realizing trust by exploiting the non-repudiation and non-tampering attributes of the blockchain and developed a blockchain-based big data-sharing framework to support various applications across the limited edges of resources. The author designed a consensus-based mechanism with low computational complexity, which was especially beneficial for edge devices with low computing power. Blockchain transaction filtering and unloading schemes can significantly improve system efficiency. Xu *et al*. [42] presented a reverse auction game to encourage edge nodes to provide caching services with incentives cooperatively. With the model, mobile users can determine the candidate of the edge nodes to cache content based on the interaction between mobile users and edge nodes. A trust management method is designed to evaluate the reliability of the selected candidate of the edge nodes by considering the direct trust evaluation. Xu *et al*. [30] proposed a novel secure caching scheme in heterogeneous networks for multihoming users. A trust mechanism is designed to verify the reliability of each edge computing-enabled small cell base station.

Chen *et al*. [56] proposed adaptive and extensible trust management to support service applications in SOA-based IoT systems. A novel adaptive filtering technique was developed to determine the best way to combine direct and indirect trust dynamically. This method minimizes convergence time and trust estimation bias in the case of opportunistic service and collusion attacking malicious nodes. For large-scale mobile-cloud IoT systems, Chen *et al.* [57] proposed and analyzed a service management protocol based on hierarchical trust called IoT-HiTrust. Hierarchical trust-based service management protocol allows IoT users to report their service experience to IoT service providers and query their subjective service trust scores from IoT service providers by scalable reporting and query designs.

The above research has not solved some problems well, such as Reference [23] and [28]–[31] are the proposed mechanism that cannot be directly applied to the IoT edge computing environment. Reference [42] shows the trust value calculation of edge computing. Most trust value calculations do not adequately consider the trust value between intelligent devices and edge service providers and the trust value between smart devices. References [30] and [32] assume that the mechanism proposed by itself will be adopted by all participants in the system, which ignores the participants' ability to choose independently. These situations may result in inaccurate or unfair results of the trust mechanism. The comparison between this paper and other relevant literature is shown in Table 1.

III. SYSTEM MODEL

In this section, we first introduce the smart city edge computing framework of multi-terminal and multi-edge centers. Secondly, present the structure of the trust mechanism proposed in this paper, and finally, add the attack model used in this paper.

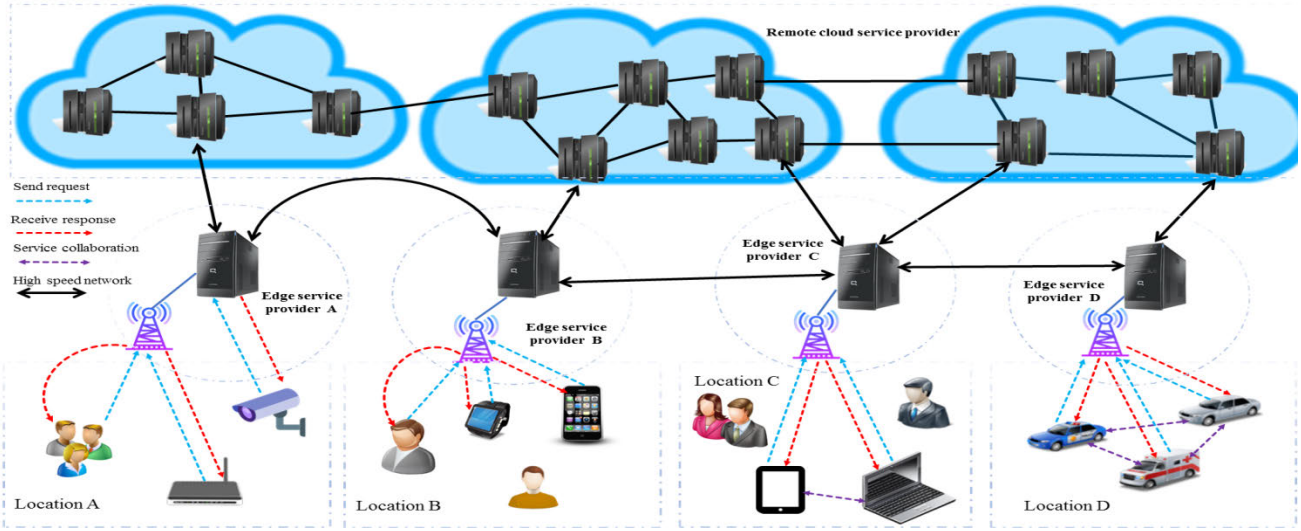


FIGURE 1. Edge computing framework for multi-intelligence terminals and multi-edge centers based on the cloud in the smart city.

TABLE 1. comparison of versatility for trust management.

Publication	Direct trust	Indirect trust	Recommendation mechanism	Acceptance of recommendation mechanism
Our Work	√	√	√	√
[16]	√	√		
[17]	√			
[20]	√	√		
[23]	√	√		
[28]	√	√		
[29]	√	√		
[30]	√	√		
[31]	√	√		
[32]	√	√		
[42]			√	
[52]	√			
[53]	√	√		
[56]	√		√	
[57]	√		√	

A. SMART CITY EDGE COMPUTING FRAMEWORK

Fig. 1 shows the edge computing framework of the smart city based on multi-terminal and multi-edge centers of the cloud platform. Edge computing deploys the devices on the infrastructure, which is physically or logically close to the end devices or users, thereby leveraging the various resources owned by these nearer infrastructures to accomplish all kinds of tasks. Because these nearer infrastructures generally have lower latency, the response time of the job can be significantly reduced, which in turn can improve the user quality of experience (QoE) and alleviate the dependence of cloud computing on network bandwidth and delay, reduce to denial of service attacks and improve service availability.

As shown in Fig. 1, the multi-terminal and multi-edge centers smart city edge computing framework based on the cloud platform consists of three parts: end-users, edge service providers, and remote cloud service providers.

1) END-USERS

Devices access edge service providers via wired or wireless networks and perform computing or storage capabilities can be defined as end-users. Since the end-user will choose the edge service provider closest to itself to access, the processing time of all submitted service request tasks will significantly reduce, and the end-user quickly obtains the execution result. When the end-user completes the job, it provides feedback on the evaluation of the service to the edge service provider. Before the new task starts, the end-user sends a request to the edge service provider to obtain the credibility of the relevant collaborator.

2) EDGE SERVICE PROVIDERS

Edge service providers are composed of servers and related devices distributed in the same area. They receive and process service requests from multiple end-users near edge service providers, and provide computing, storage, and other services to end-users quickly and flexibly. Edge service providers monitor the service behavior of devices and aggregate evaluation feedback from devices and users and then send the results to cloud data centers. In the real edge computing environment, there may be a large number of unreliable (or malicious) devices and users' aggressive behavior, and feedback from these deceptive devices and users will lead to dishonest evaluation feedback results. We extend the traditional evaluation feedback mechanism to reduce edge computing risks and improve the reliability of the system.

3) REMOTE CLOUD SERVICE PROVIDERS

Remote cloud service providers are responsible for managing and monitoring the resources and services of edge service

providers, and coordinating edge service providers to complete the cloud computing services required by end-users. Remote cloud service providers control the operation status and resource usage of each edge service provider in real-time by monitoring the services and resources provided by each edge service provider. By collecting the services and resources provided by each edge service provider, remote cloud service providers coordinate and schedule resources and services in each edge service provider to produce more compelling cloud computing services for end-users. We assume that remote cloud service providers are reliable and always available, and attacks on remote cloud service providers' servers are beyond the scope of this paper.

B. AUTHENTICATION

Smart device authentication is required when sensing devices and data are shared to ensure that illegal devices cannot access the system in edge computing. Identity authentication can be solved through symmetric or asymmetric password schemes. It is necessary to select the trusted, smart device based on identity authentication to establish the trust management mechanism.

Authentication is an essential process of establishing the identity of both sides of communication in the edge computing system of the IoT. During the end-user registration phase, the authentication system assigns a unique identity ID to each intelligent device. A shared symmetric key generated between each end-user and each edge center. The registration timestamp of each smart device is also different from preventing related attacks. Finally, the authentication system stores information about the previous operation of the intelligent device. During the registration phase of the edge center, the authentication system assigns a unique identification to each edge center and stores the related operation information in the authentication system. In the authentication phase, when the intelligent device wants to interact with the edge center, the smart device first encrypts related settings and obtains the current timestamp, and then sends the device ID, the relevant parameters after the encryption calculation and the timestamp to the edge center. When the edge center receives the information submitted by the intelligent device, it checks whether the timestamp is valid. If the timestamp is correct, the relevant parameters are calculated through decryption and judged at the same time. If the conditions met and accurate, the edge center passes the authentication of the intelligent device. Otherwise, the edge center terminates the interaction with the smart device. After that, the edge center encrypts the relevant parameters and obtains the current timestamp, and then sends the relevant parameters after the encryption calculation, the arrival time of the information submitted by the smart device and the current timestamp to the intelligent device. When the intelligent device receives the information sent by the edge center, it checks whether the timestamp is valid. If the timestamp is accurate, the relevant parameters are calculated by decryption and judged. If the conditions are met and legal, then the intelligent device has passed the authentication of the edge center. Both the smart device and the edge center maintain the same shared session key during the authentication process.

C. COMPOSITION OF TRUST MECHANISM

The trust mechanism studied in this paper consists of direct trust, indirect trust, and selective recommendation mechanism. Direct trust is the feedback between the end-user and other end-users, which are the most basic trust relationship that encourages cooperation between end-users. Indirect trust is feedback between the edge service provider and the end-user, which is the crucial factor in reducing the risk of malicious device attack, especially critical for the successful deployment of edge computing services. The selective recommendation mechanism is based on the dynamic black-and-white list to improve the service quality of edge computing through a particular recommendation algorithm. We define a trust mechanism.

Definition 1: Direct trust is the quantized value of the end-user completing the requested task. This value is based on the interaction history between end-users.

Definition 2: Indirect trust is a score based on the quantitative calculation of the edge service providers. After the task unloading (or forwarding) completed, the edge service provider will calculate the real-time trust of the end-user. When another end-user submits a request to it, the edge service provider sends the value to the requester.

Definition 3: The selective recommendation mechanism is a selection and recommendation process based on a white list and blacklist. Edge service provider dynamically adds or removes records. The purpose of sifting devices is to select suitable devices and improve the quality of service. At the same time, resist the related malicious attacks.

D. ATTACK MODEL

There are different types of end-users in various practical scenarios of a smart city. Without loss of generality, we assume that there are two types of end-users in the system: ordinary end-users and malicious end-users. Ordinary end-users provide effective services and feedback on correct evaluation information. Malicious end-users provide invalid services and feedback for misleading evaluation information. This paper considers the following four malicious attack models.

1) BAD-MOUTHING ATTACKS

In this attack model, malicious end-users exist independently, and in a round of interaction, they provide invalid services and dishonest trust evaluation, reduce the chance that this suitable terminal device will be selected as a service provider.

2) GOOD-MOUTHING ATTACKS

In this attack model, malicious end-users exist independently, and in a round of interaction, they provide invalid services and provide good trust ratings for other malicious devices. They are thereby increasing the chance that this malicious terminal device is selected as a service provider.

3) BALLOT-STUFFING ATTACK

In this attack model, there is a collusion relationship between malicious terminal devices, they will give each other a proper evaluation, and when they are selected as service providers, they will provide invalid services.

4) SELECTIVE BEHAVIOR ATTACK

In this attack model, malicious terminal devices provide effective services with a probability of $f\%$ to obtain higher trust values to increase the chances of transmitting invalid services.

IV. DESIGN OF TRUST MANAGEMENT MECHANISM

This section introduces the trust management mechanism proposed in this paper from three aspects: the trust relationship between end-users, the trust relationship between the edge service provider and end-users, and the terminal device selective recommendation mechanism.

A. THE TRUST RELATIONSHIP BETWEEN END-USERS

After a task completed, the requester end-user i evaluates the provider end-user j based on the quality of service. We record the evaluation as $tr(i,j)$. $tr(i,j) = 1$ means that the end-user i is satisfied with the service provided by the end-user j, otherwise $tr(i,j) = -1$. We use $s_{ij} = \sum tr(i,j)$ to represent the overall evaluation of the end-user i to end-user j, which is direct trust.

The establishment of trust often requires many trust interactions among participants, while untrustworthy interaction is enough to destroy the established trust relationship between participants. This paper introduces an evaluation mechanism of personalized context content perception, in which we fix the influence of negative evaluation on direct trust as 1, which is recorded as $tr(i,j) = -1$. Then set a weight parameter w_i to indicate the impact of a positive evaluation on direct trust. When the end-user j provides a satisfactory service to end-user i, we use $tr(i,j) = w_i$ to indicate the end-user i's trust evaluation of the end-user j, where $w_i \in [0,1]$. The smaller the value of w_i, the easier it is for the end-user i to establish a trust relationship. Under this mechanism, the end-user i's direct trust s_{ij} to end-user j can be expressed as Equation (1).

$$s_{ij} = w_i \times sat(i,j) - unsat(i,j) \tag{1}$$

where $sat(i,j)$ and $unsat(i,j)$ are respectively expressed as the number of satisfactory and unsatisfactory services provided by the end-user j for the end-user i.

Different services have different degrees of importance. For example, in an actual smart city scenario, the importance of services in a personal healthcare monitoring system is often higher than that in an air-quality monitoring and analysis system. Therefore, Equation (1) can be modified by Equation (2).

$$s_{ij} = w_i \sum_{x \in sat(i,j)} DI(x) - \sum_{x' \in unsat(i,j)} DI(x') \tag{2}$$

where $DI(\bullet)$ indicates the importance of the services.

To avoid false expansion of direct trust caused by malicious end-users, we need to perform the normalization operation shown in Equation (3) for direct trust value among end-users.

$$c_{ij} = \begin{cases} \dfrac{\max(0, s_{ij})}{\sum_j \max(0, s_{ij})} & if \ \sum_j \max(0, s_{ij}) > 0 \\ p_j & otherwise \end{cases} \tag{3}$$

where c_{ij} is the normalized direct trust that the end-user i places on the end-user j. If the trusted end-user set P can be obtained in advance, it assumed that each end-user in the set P is trustworthy. For the end-user i who has no interaction with other end-users, $\sum_j \max(0, s_{ij}) = 0$ can be obtained. if the end-user j belongs to the trusted set P, then $p_j = 1/|P|$ is established, otherwise $p_j = 0$. If the trusted end-user set cannot be obtained in advance, there is $p_j = 1/N$, where N is the total number of end-users.

B. THE TRUST RELATIONSHIP BETWEEN EDGE SERVICE PROVIDERS AND END-USERS

When the end-user completes the task, the trust feedback information as shown in Equation (3) is submitted to the edge service provider. In the edge service provider, we construct a $N \times N$'s Markov matrix $C = [c_{ij}]$ to represent normalized direct trust between any two end-users. The Markov matrix is shown in Equation (4).

$$C = \begin{bmatrix} c_{11} & \cdots & c_{1j} & \cdots & c_{1N} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ c_{i1} & \cdots & c_{ij} & \cdots & c_{iN} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ c_{N1} & \cdots & c_{Nj} & \cdots & c_{NN} \end{bmatrix} \tag{4}$$

Direct trust represents the trust relationship established between end-users based on mutual interactions. Because end-users have mobility features, in a large-scale smart city IoT edge computing system, interactions often do not occur only between familiar end-users. How to predict the potential trust relationship between unfamiliar end-users based on the existing direct trust relationship is a problem that needs to be solved. This problem is also the task that the edge service provider needs to accomplish. In human society, trust is often transitive [24]. If the end-user i trusts end-user j and end-user j trusts end-user k, even if there is no direct interaction between i and k, we still infer that end-user i will trust end-user k with a high probability. Equation (5) is used to formalize the idea as mentioned above of trust transfer.

$$t_{ik} = \sum_j c_{ij} \times c_{jk} \tag{5}$$

End-user i establishes an indirect trust with k by referring to the direct trust of all its friend's j to k. Since Equation (5) only considers the neighbor friends of the end-user, hence we call it one-degree trust propagation. We introduce vector $\overrightarrow{c_i} = [c_{ij}]$ to represent end-user i's normalized direct trust to all other end-users. Vector $\overrightarrow{t_i} = [t_{ij}]$ represents the indirect trust of the end-user i for all other end-users. Therefore, the one-degree

trust propagation shown in Equation (5) can be written in matrix form as shown in Equation (6).

$$\overrightarrow{t_i} = C^T \times \overrightarrow{c_i} \tag{6}$$

In an IoT edge computing system with sparse evaluation data, the indirect trust evaluation obtained through one-degree trust propagation often fails to meet the interaction requirements of the end-user. To achieve a broader range of trust cognition, end-user i learns from the evaluation information of friends of his or her friends, which is called two-degree trust propagation. Two-degree trust propagation can be expressed formally as Equation (7).

$$\overrightarrow{t_i} = \left(C^T\right)^2 \times \overrightarrow{c_i} \tag{7}$$

Similarly, m-degree trust propagation can be expressed as Equation (8).

$$\overrightarrow{t_i} = \left(C^T\right)^m \times \overrightarrow{c_i} \tag{8}$$

According to the convergence of the Markov matrix, we know that when m is large enough, the arbitrary vector $\overrightarrow{t_i}$ ($i \in \{1, 2, \ldots n\}$) will converge to the left principal eigenvector $\overrightarrow{t}$ of the matrix C, which represents the indirect trust value of each end-user in the trust mechanism.

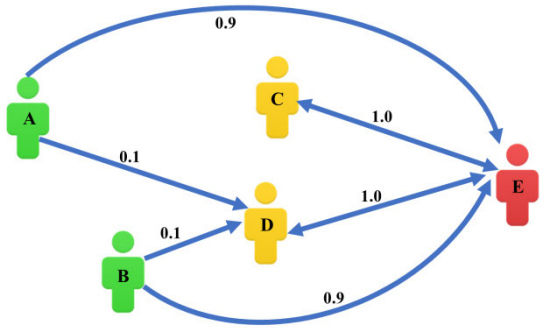


FIGURE 2. Evaluation of information network.

However, in the actual smart city system, malicious end-users tend to choose to provide untrue trust feedbacks to increase their chances of transmitting invalid services, which poses a considerable challenge for accurate trust evaluation. In the evaluation information network shown in Fig. 2, end-user A and B are good end-users, which provide effective services and real feedbacks. End-user C and D are malicious end-users who provide invalid services and untrue feedbacks. End-user E is a masquerading end-user, which will provide useful services for end-user A and B with a certain probability to obtain a high trust value. He or she will offer a high direct trust evaluation for malicious end-user C and D. At this time, end-user A and B will generate an indirect trust relationship of 0.9 with end-user C, which is unreasonable. On receiving a valid service, the trustworthy end-user will place a positive rating on the service provider, while the untrustworthy end-user will set a negative rating. Therefore, trustworthy and untrustworthy end-users always have dissimilar evaluation information. Similarly, trustworthy and untrustworthy end-users will also provide disparate ratings on receiving invalid services. Sometimes, with a certain probability, smart or strategic malicious end-users may feedback honest ratings the same as the trustworthy end-users.

However, from a macro point of view, the evaluation similarity between two trustworthy end-users usually is higher than the evaluation similarity between a trustworthy end-user and an untrustworthy end-user. For example, as shown in Fig. 2, end-user A and E fed back a very different trust evaluation for end-user D. Inspired by this observation; this paper uses the cosine similarity method to measure the credibility of end-user evaluation information.

For the end-user i and j, we use r_{ij} to indicate the evaluation credibility of the end-user i for the end-user j, as shown in Equation (9).

$$r_{ij} = \begin{cases} \dfrac{\sum_{k \in comn(i,j)} c_{ik} \times c_{jk}}{\sqrt{\sum_{k \in comn(i,j)} (c_{ik})^2} \times \sqrt{\sum_{k \in comn(i,j)} (c_{jk})^2}} \\ \quad if\ |comn(i,j)| > 0 \\ 0.5 \quad otherwise \end{cases} \tag{9}$$

where $comn(i, j)$ is the set of end-users who have interacted with end-user i and j, Equation (9) shows that the more similar the evaluation information of the end-user j and i are, the more reliable the evaluation information of the end-user j is. By further considering evaluation credibility r_{ij} between end-users, we rewrite the direct trust s_{ij} between end-users into Equation (10).

$$s_{ij} = r_{ij} \times s_{ij} \tag{10}$$

After the normalization operation of Equation (3) and the trust transfer operation of Equation (8), we get a more accurate end-user indirect trust vector $\overrightarrow{t}$. In this paper, the vector $\overrightarrow{t}$ is used as the trust evaluation of each end-user by the edge service provider.

C. THE PERSONALIZED DEVICE SELECTION MECHANISM

Many existing algorithms only consider the indirect trust value of the end-user, ignoring the personalized direct trust of the end-user. As shown in Fig. 3, end-user A feeds back the lower direct trust value to end-user C. However, C may obtain a higher indirect trust value than E. Furthermore, in the case of a service request for end-user A, end-user C will have a higher recommendation priority than end-user E, but this ignores the end-user A's personalized trust. To consider the personalized direct trust, this paper proposes a personalized end-user selection mechanism based on the black-and-white list.

1) BLACKLIST MECHANISM

In the selection and recommendation mechanism, end-users add untrustworthy end-users to their blacklists and reduce the opportunities for interactions with them. Use set B_i to

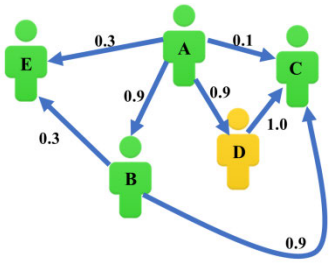


FIGURE 3. Example of a trust evaluation network.

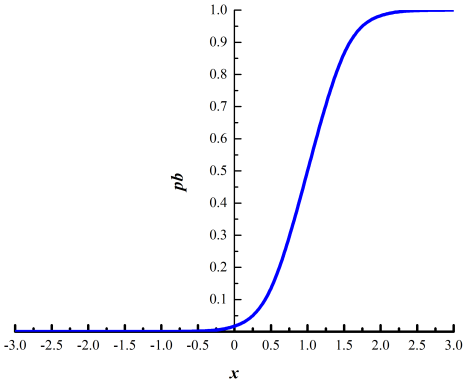


FIGURE 4. Diagram of the relationship between pb_{ij} and x_{ij}.

represent the blacklist of the end-user i. Given $|sat(i,j)|$ and $|unsat(i,j)|$ respectively represent the number of satisfactory and unsatisfactory services provided by the end-user j for the end-user i. The probability pb_{ij} that the end-user i adds end-user j to the blacklist can be expressed by Equation (11).

$$pb_{ij} = \frac{1}{1 + e^{(1-x_{ij})\delta_i}} \tag{11}$$

where $x_{ij} = |unsat(i,j)| - |sat(i,j)|$, δ_i denotes end-user i's tolerance of unsatisfactory services. The larger value δ_i is, the higher the probability that the end-user who provides an invalid service will be added to the end-user i's blacklist. When δ_i is fixed to 5, the relationship between pb_{ij} and x_{ij} is shown in Fig. 4.

2) WHITELIST MECHANISM

In the selection and recommendation mechanism, end-users add honest end-users to their whitelists and increase opportunities for interactions with them. Use set W_i to represent the whitelist of the end-user i. Given $|sat(i,j)|$ and $|unsat(i,j)|$ respectively represent the number of satisfactory and unsatisfactory services provided by end-user j for the end-user i. The probability pw_{ij} that the end-user i adds end-user j to the whitelist can be expressed by Equation (12).

$$pw_{ij} = \frac{1}{1 + e^{(1+x_{ij})\eta_i}} \tag{12}$$

where η_i denotes end-user i's satisfaction with satisfactory services. The larger value η_i is, the higher the probability that

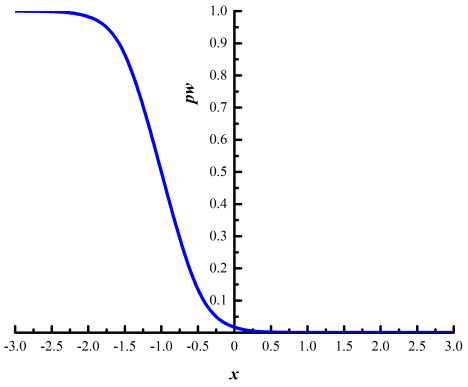


FIGURE 5. Diagram of the relationship between pw_{ij} and x_{ij}.

the end-user who provides a valid service will be added to the end-user i's whitelist. When η_i is fixed to 5, the relationship between pw_{ij} and x_{ij} is shown in Fig. 5.

After introducing the black-and-white list mechanism, we propose an end-user selection mechanism that respects the end-user personalized trust to improve the quality of service in the smart city system. The end-user selection mechanism is shown in Algorithm 1.

Algorithm 1 End-User Selection Mechanism Based on the Black-and-White List

input:
i: End-user service request
B_i: Blacklist set of end-user i
W_i: Whitelist set of end-user i
R: End-user set responding to i service requests
t_k: Indirect trust value of end-user k, $k \in R$
out:
j: Recommended service provider end-user
1: $R \leftarrow R - B_i$
2: if $R == \emptyset$
3: **Return**-1
4: else
5: $RW \leftarrow R \cap W_i$
6: **if** $RW \neq \emptyset$
7: end-user j is randomly returned with a probability $t_j = t_j / \sum_{k=0}^{k=|RW|} t_k$
8: **else**
9: end - user j is randomly returned with a probability $t_j = t_j / \sum_{k=0}^{k=|R|} t_k$

3) BLACK-AND-WHITE LIST DYNAMIC CONVERSION MECHANISM

In the selective recommendation mechanism, the behavior of the device will exhibit changing characteristics. For example, devices in the white list may use the high trust value they have obtained to propagate invalid or malicious services. Devices in the blacklist may provide excellent services to increase their trust value. To capture the dynamic behavior of the

Algorithm 2 Black-and-White List Dynamic Conversion Mechanism

input:
i : End-user service request
B_i : Blacklist set of end-user i
W_i : Whitelist set of end-user i
O_i : Common set
out:
B_i' : Blacklist of end-user i after conversion
W_i' : White list of end-user i after conversion
O_i' : Common list of end-user i after conversion

1: $B_i' \leftarrow B_i$, $W_i' \leftarrow W_i$, $O_i' \leftarrow O_i$
2 if $W_i \neq \emptyset$
3: **Calculate** pw_{ij}^m **according to Equation (13)**
4: $rmd = rand()$
5: **if** $rmd \leq pw_{ij}^m$
6: $W_i' \leftarrow W_i - \{i\}$
7: $O_i' \leftarrow O_i \cup \{i\}$
8: **else**
9: **Device** j**remains in Set** W_i
10: if $B_i \neq \emptyset$
11: **Calculate**pb_{ij}^m**according to Equation (14)**
12: **if**$pb_{ij}^m = 1$
13: $B_i' \leftarrow B_i - \{i\}$
14: $O_i' \leftarrow O_i \cup \{i\}$
15: **else**
16: **Device** j**remains in Set** B_i
17: Return B_i', W_i', O_i'

device, this paper proposes a dynamic conversion mechanism for black-and-white lists. The mechanism consists of two parts: a white list removal mechanism and a blacklist removal mechanism.

a: WHITE LIST REMOVAL MECHANISM

Assume that device j is a device in the device i's white list, if the device i obtains an invalid or malicious service from the device j, then it will remove the device j from the white list to the ordinary user list with probability pw_{ij}^m. The probability pw_{ij}^m is related to the following four attributes: (1) The ratio of the number of dissatisfied services to the total number of services is $\frac{|unsat(i,j)|}{|unsat(i,j)+sat(i,j)|}$. If the larger value is, the higher the frequency of invalid service provided by the device j is. The probability that the corresponding device j is removed from the white list is more elevated. (2) In order to prevent the device from strategically providing unsatisfactory services while maintaining a highly effective service provision ratio, we also need to consider the number of invalid services $|unsat(i,j)|$ supplied by the device j. If the more significant value is, the probability that the device j will be removed from the white list is higher. (3) The importance of service $DI(x) \in [0, 1]$. If the device j provides false feedback on a highly valuable service, the probability that the device j will be removed from the white list is higher. (4) The sensitivity $\phi_i \in R$ of the device j to invalid or malicious services. If the more significant value ϕ_i is, it means that the tolerance of the device i to invalid or malicious service smaller is. The probability of the device j being moved out is higher. In summary, we define pw_{ij}^m as shown in Equation (13).

$$pw_{ij}^m = \frac{2}{1+e^{-\phi_i \times |unsat(i,j)| \times \frac{\sum_{\chi \in unsat(i,j)} DI(\chi)}{\sum_{\chi \in unsat(i,j)} DI(\chi) + \sum_{\Psi \in sat(i,j)} DI(\Psi)}}} - 1 \tag{13}$$

where $unsat(i,j)$ represents the service transaction set that device i is not satisfied with the device j, $sat(i,j)$ represents the service transaction set that device i is satisfied with the device j. χ and ψ represent a service between devices, and function $DI(\bullet)$ indicates the importance of service.

b: BLACKLIST REMOVAL MECHANISM

It is assumed that the device j is a blacklisted device from the device i, considering that the device j may change its original untrusted behavior in order to improve its trust value. In order to capture the behavior characteristics of the device j dynamically, the device i periodically calculates a probability value pb_{ij}^m that moves the device j from the blacklist to the common list. The probability pb_{ij}^m is related to the following three attributes: (1) The attitude of neighbor k towards the device j is $attitude(k,j)$. In this paper, devices that have interacted with each other are defined as neighbors. We assume that $attitude(k,j) = a_1$ indicates the device k puts the device j into the blacklist. $attitude(k,j) = a_2$ indicates device k puts the device j into the common list, and $attitude(k,j) = a_3$ indicates device k put the device j into the white list, where $a_1, a_2, a_3 \in N_+$ and $a_3 > a_2 > a_1$. If the attitude of the device k to the device j is better, the probability that the device j will be removed from the blacklist by the device i will be higher. (2) For the device i, the evaluation credibility r_{ik} of neighbor k. If r_{ik} is higher, the influence of neighbor k on pb_{ij}^m is more significant. (3) The abhorrence of the device i against invalid or malicious services is ξ_i. If the value ξ_i is higher, the possibility that the device j is removed from the blacklist is less. In summary, we define pb_{ij}^m as shown in Equation (14).

$$pb_{ij}^m = \begin{cases} 1 & if \sum\limits_{k \in N(i)} \dfrac{attitude(k,j) \times r_{ik}}{a_3 \times |N(i)|} \geq \xi_i \\ 0 & otherwise \end{cases} \tag{14}$$

where $N(i)$ represents the neighbor device set of the device i. The black-and-white list dynamic conversion mechanism is shown in Algorithm 2.

V. THEORETICAL ANALYSIS OF TRUST MANAGEMENT MECHANISM

Evolutionary game theory provides a powerful mathematical framework for studying the interaction between rational individuals [22], [33]. Therefore, we introduce evolutionary game theory to qualitatively analyze the validity and stability of the trust management mechanism. A trust management

mechanism can be modeled as a non-cooperative game. The participants in the game model are intelligent devices. In this section, we will introduce the game model proposed in this paper from three parts: strategy set, strategy expectation fitness, and evolutionary strategy.

A. STRATEGY SET

In this paper, there are three types of users in the system. One is a malicious user, one is a user who adopts the mechanism proposed in this paper, and one is a user who does not use the mechanism proposed in this paper. Our game model has a three-strategy set $S = \{s_1, s_2, s_3\}$. s_1 represents the behavior used by good devices, and the strategy will always provide effective services and feedback on real evaluation. s_2 is behavior used by malicious devices. This strategy will always provide invalid services and untrue feedback evaluation. s_3 represents the device behavior utilized by the selective recommendation mechanism proposed in this paper. This strategy will provide effective services for the trusted devices determined by the system and feedback from the actual evaluation. Without loss of generality, we assume that the total number of users in the game model at a time t is $M(t)$, and the number of devices using the strategies s_1, s_2, and s_3 are $M_1(t)$, $M_2(t)$, and $M_3(t)$, respectively. $M_1(t) + M_2(t) + M_3(t) = M(t)$. We introduce the vector $X(t) = [x_1(t), x_2(t), x_3(t)]$ to represent the proportion of the strategy distribution in the system, where $x_i(t) = {M_i(t)}/{M(t)}$. For convenience, we simplify the representation of $M_i(t)$ and $x_i(t)$ as M_i and x_i. It should be noted that (1) In order to facilitate theoretical analysis and reduce the impact of other factors on user strategies, this paper only considers the above three strategies. (2) In this paper, we assume that the strategy s_3 takes effective services and provides real feedback. The strategy s_3 takes the case of providing invalid services and feeding back untrue evaluations, which we discuss in another paper.

B. STRATEGY EXPECTATION FITNESS

1) ASSUMPTIONS

Due to the influence of data noise and other factors, when the trusted device selection and recommendation mechanism proposed in this paper is deployed, the system will not only judge the well-performing devices (such as the devices using the strategy s_1 and the strategy s_2 in the game model) as trusted devices. Sometimes, malicious devices (such as devices using the strategy s_3 in the game model) are mistakenly judged as trusted devices. Therefore, we assume that the probability of such error occurring is ε, with $\varepsilon \in [0, 1]$. For the following two considerations: (1) To avoid the influence of network topology on device behavior. (2) In order to facilitate theoretical analysis, the game model in this paper considers a mixed uniform interaction scenario. The device will use probability x_1, x_2, and x_3 to provide services for other devices using the strategy s_1, s_2, and s_3, respectively. We model the system as a time-discrete model and assume that in a time step, each device expects to initiate a service request.

2) EXPECTED FITNESS OF STRATEGY

a: EXPECTED FITNESS OF THE STRATEGY s_1

Mechanism calculates the expected profit $E^R_{s_1}(t)$ of the device using the strategy s_1. When a device adopting strategy s_1 initiates a request, due to the interactive setting of well mixed, the device will provide services for the devices utilizing strategies s_1, s_2, and s_3 with the expected number x_1, x_2, and x_3, respectively. Therefore, $E^R_{s_1}(t)$ can be expressed as shown in Equation (15).

$$E^R_{s_1}(t) = (x_1 + x_3)b_p + x_2 b_n \tag{15}$$

where b_p and b_n represent the positive revenue when the requester device receives a valid service and the negative revenue when an invalid service is received, so there is $b_p > 0 > b_n$. We calculate the expected loss $E^P_{s_1}(t)$ of the device using the strategy s_1. When the device adopting the strategy s_1 is used as the service provider, the device will provide the device with the strategy s_1 and the device with the strategy s_2 as the expected number x_1 and x_2, respectively, in a one-time step due to the mixed interaction setting. Because the requester using strategy s_3 uses the trust selective recommendation mechanism proposed in this paper, it interacts with the device using the strategy s_1 with a higher probability. Under the setting of the recommended error probability ε, the number of services that the devices adopting the strategy s_1 expect to provide for the devices utilizing the strategy s_3 in a one-time step is $\frac{M_3}{M_1+\varepsilon M_2+M_3} = \frac{x_3}{x_1+\varepsilon x_2+x_3}$. $E^P_{s_1}(t)$ can be expressed as shown in Equation (16).

$$E^P_{s_1}(t) = \left(x_1 + x_2 + \frac{x_3}{x_1 + \varepsilon x_2 + x_3}\right)c_p \tag{16}$$

where $c_p \in \Re^+$ represents the positive loss when the device contributes a useful service. The expected fitness $\bar{f}_{s_1}(t)$ using the strategy s_1 in the time step t can be expressed as described in Equation (17).

$$\bar{f}_{s_1}(t) = E^R_{s_1}(t) - E^P_{s_1}(t) = (x_1 + x_3)b_p + x_2 b_n - \left(x_1 + x_2 + \frac{x_3}{x_1 + \varepsilon x_2 + x_3}\right)c_p \tag{17}$$

b: EXPECTED FITNESS OF THE STRATEGY s_2

When a device adopting strategy s_2 initiates a request, due to the interactive setting of well mixed, the device will provide services for the devices adopting strategies s_1, s_2, and s_3 with the expected number x_1, x_2, and εx_3, respectively. Therefore, $E^R_{s_2}(t)$ can be expressed as shown in Equation (18).

$$E^R_{s_2}(t) = (x_1 + \varepsilon x_3)b_p + x_2 b_n \tag{18}$$

When the device adopting the strategy s_2 is used as the service provider, the device will provide invalid services for the device with the strategy s_1 and the device with the strategy s_2 as the expected number x_1 and x_2, respectively. When the recommended error probability is ε, the size of the device

interaction set using strategy s_3 is $M_1+\varepsilon M_2+M_3$. If the current device using strategy s_2 can be judged as a trusted device with the probability of ε, the expected number that the device provides invalid services for the device using the strategy s_3 is $\frac{M_3}{M_1+\varepsilon M_2+M_3}=\frac{x_3}{x_1+\varepsilon x_2+x_3}$. On the contrary, the number of invalid services offered by the device for the device adopting strategy s_3 is 0. In general, devices adopting the strategy s_2 provide invalid services for devices adopting the strategy s_3 with the expected number $\frac{\varepsilon x_3}{x_1+\varepsilon x_2+x_3}$. Therefore, $E^P_{s_2}(t)$ can be expressed as shown in Equation (19).

$$E^P_{s_2}(t)=\left(x_1+x_2+\frac{\varepsilon x_3}{x_1+\varepsilon x_2+x_3}\right)c_n \tag{19}$$

where $c_n\in\Re^-$ represents the negative loss when the device contributes an invalid service. The expected fitness $\bar{f}_{s_2}(t)$ using the strategy s_2 in the time step t can be expressed as described in Equation (20).

$$\bar{f}_{s_2}(t)=E^R_{s_2}(t)+E^P_{s_2}(t)=(x_1+\varepsilon x_3)b_p+x_2b_n+\left(x_1+x_2+\frac{\varepsilon x_3}{x_1+\varepsilon x_2+x_3}\right)c_n \tag{20}$$

c: EXPECTED FITNESS OF THE STRATEGY s_3

Since the device adopting the strategy s_3 selects an interactive object based on the trust selective recommendation mechanism when the device acts as a service requester, it obtains valid services and invalid services with the expected number $\frac{x_1+x_3}{x_1+\varepsilon x_2+x_3}$ and $\frac{\varepsilon x_2}{x_1+\varepsilon x_2+x_3}$, respectively. Therefore, $E^R_{s_3}(t)$ can be expressed as shown in Equation (21).

$$E^R_{s_3}(t)=\frac{x_1+x_3}{x_1+\varepsilon x_2+x_3}b_p+\frac{\varepsilon x_2}{x_1+\varepsilon x_2+x_3}b_n \tag{21}$$

When the device adopting the strategy s_3 is used as the service provider, it takes the expected number x_1, εx_2, and $\frac{\varepsilon x_3}{x_1+\varepsilon x_2+x_3}$ as the device adopting the strategy s_1, and the device adopting the strategy s_2 and the device adopting the strategy s_3 to provide effective services. Therefore, $E^P_{s_3}(t)$ can be expressed as shown in Equation (22).

$$E^P_{s_3}(t)=\left(x_1+\varepsilon x_2+\frac{\varepsilon x_3}{x_1+\varepsilon x_2+x_3}\right)c_p \tag{22}$$

The expected fitness $\bar{f}_{s_3}(t)$ using the strategy s_3 in the time step t can be expressed as described in Equation (23).

$$\bar{f}_{s_3}(t)=E^R_{s_3}(t)-E^P_{s_3}(t)=\frac{x_1+x_3}{x_1+\varepsilon x_2+x_3}b_p+\frac{\varepsilon x_2}{x_1+\varepsilon x_2+x_3}b_n-\left(x_1+\varepsilon x_2+\frac{\varepsilon x_3}{x_1+\varepsilon x_2+x_3}\right)c_p-c_r \tag{23}$$

where c_r ($c_r>0$) indicates that the device adopting the strategy s_3 needs extra cost when using the trust selective recommendation mechanism.

C. EVOLUTIONARY STRATEGY

Because the actual scenario of the smart city IoT edge computing system is very complex, it can't be pre-set with the optimal strategy and intelligent devices will constantly adopt learning methods to update their behavior. In general, a strategy with a high degree of fitness is expected to increase its proportion in the game model and vice versa. Therefore, we use the replication dynamics equation that has been widely used in the evolutionary game to formalize the evolution process of strategy. The replication dynamics equation of the game model is shown in Equation (24).

$$\dot{x}_{s_i}(t)=x_i(t)\left(\bar{f}_{s_i}(t)-\bar{f}(t)\right) \tag{24}$$

where $\bar{f}(t)=\sum_{i=1}^{N}x_i(t)\bar{f}_{s_i}(t)$ represents the expected profit of the whole game model. We can find the stability point of the system.

Definition 4: According to the Lyapunov stability theory, let O be the optimal solution of the dynamic system Θ. If the real part of each eigenvalue of O's Jacobian matrix is not positive, and the eigenvalue with zero real part is the single root of the minimum polynomial, then O is the asymptotic stability point of the dynamic system Θ.

Proposition 1: When $bp=7$, $bn=-15$, $cp=1$, $cn=-10$, $cr=4$, (0.205791 0.140849 0.653360) is the evolution stable state of the model. The proof of the theory is given in the appendix. We have verified the game model in detail in the experimental part of this paper.

VI. EXPERIMENT AND ANALYSIS

A. EXPERIMENT SETUP

The experimental setup is shown in Table 2. There are 1000 different end-users in the experiment. The number of preset trusted end-users is 5. Without losing generality, this paper assumes that all end-users have the same tolerance δ_i and the same recognition η_i. According to literature [25]–[27], $\delta_i=\eta_i=10$ is usually used. There are 2000 different services in the system. At the beginning of the system, each end-user has an average of 15 services. We introduce 1000 rounds of warm-up interactions to build trust relationships between end-users initially. The percentage of collaborative devices (PCD) is set to 10%, 20%, 40%, and 100%, respectively, indicating that the IoT edge computing system is idle, busy, highly busy, and extremely busy. We use *num_trans* to indicate the number of service requests initiated by the ordinary end-user, and *num_valid* to represent the number of valid services obtained by the ordinary end-user. In the experiment, we measure the performance of the trust model by using the effective service acquisition rate as $r=num\_valid/num\_trans$. To facilitate the representation of bad-mouthing attacks, good-mouthing attacks, ballot-stuffing attacks, and selective behavior attacks, we have abbreviated the attack models as Model A, Model B, and Model C, respectively. Since bad-mouthing attacks and good-mouthing attacks are similar, we classify them into one category. In this paper, we consider an edge computing scenario of the Internet of things in a smart city to complete simple tasks. In the domain of trust management, simulations [23], [31], [32], [34], [41] based on synthetic data have

TABLE 2. Simulation setting.

Symbol	Description	Value
N	Number of end-users	1000
δ_i	End-user tolerance	10
η_i	End-user satisfactory	10
PCD	Percentage of collaborative end-users	10%,20%,40%,100%
θ	The number of preset trusted end-users	5
ζ	Total number of services	2000
μ	Number of services per end-user initially	15
b_p	Profit from a good service	7
c_p	The cost of providing a good service	-1
b_n^h	Loss of an invalid service to the personal health monitoring system	-15
b_n^a	Loss of air quality analysis system getting an invalid service	-2
c_n^h	The cost of an invalid service provided by the personal health monitoring system	-10
c_n^a	The cost of an air quality analysis system to provide an invalid service	0

been widely used. In the future, we will check the performance of the proposed model once the relevant real dataset is released. All the experiments and simulations were run on a Win7 machine with Intel Core I5-4590 3.3GHz processor and 8 GB memory.

B. WEIGHT INFLUENCE

We assume that all end-users have the same weight setting w for valid service. We set the experimental scenario to be 70% of the system as ordinary end-users, and 30% of end-users of the attack model C. The PCD is set to be 100%. The experimental results are shown in Fig. 6. We find that the value of w has a significant influence on the effective service acquisition rate. At the time of $w = 0$, since the positive evaluation cannot be obtained by providing an effective service, the system receives the worst performance, and the effective service ratio is less than 90%. At the time of $w = 1$, the effective service acquisition rate was 92.86%, because there is no distinction between positive and negative evaluation. Through experiments, it is found that in the experimental environment given in Table 2, when $w = 0.8$, the system performs best,

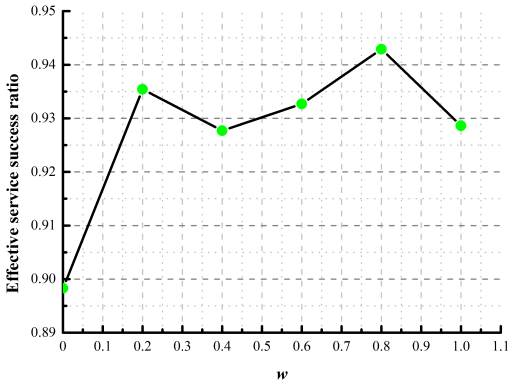


FIGURE 6. Weight influence.

and the end-user obtains effective service with a probability of 94.29%. Experiments have achieved similar results with different end-user settings and collaboration ratio settings.

C. PERFORMANCE RESULTS AND ANALYSIS

1) SINGLE ATTACKS

In order to study the performance of this model in quality of service assurance, we aim at (1) Different attack models, from attack model A to attack model C. (2) Different malicious end-user ratio, from 10% to 50%. (3) Different end-user cooperation degree, from 10% to 100%. We measure the availability of effective services in the IoT edge computing system for the above three different experimental settings.

In Fig. 7 and Fig. 8, end-user collaboration degrees are set to be 100% and 10%, respectively. The former is an extremely busy IoT edge computing system, while the latter is an idle IoT edge computing system. Fig. 7 (a) shows the impact of the attack model A on the system under the different proportions of malicious end-users. We find that (1) The accumulation of trust evaluation information further helps the model to evaluate the credibility of each end-user accurately. Therefore, as time goes by, the rate of effective service acquisition in the system will gradually improve. Especially in the high proportion of malicious end-users' scenarios, the performance improvement will be more obvious. (2) In 10%, 20%, 30%, 40%, and 50% malicious end-users' scenarios, the model in this paper achieve 99.3%, 97.9%, 97.7%, 95.5%, and 92.6% effective service access rates respectively in the 10000 rounds of interaction. This result illustrates the effective performance of the proposed model in guaranteeing the quality of service of the IoT edge computing system.

Fig. 7(b) and Fig. 7(c) show the impact of attack model B and model C on the system under the different proportions of malicious end-user settings, respectively. Similar to the results in Fig. 7 (a), the performance of the model also be improved progressively over time. Under the attack model B and C with 50% malicious end-user distribution, this model achieves about 90% effective service acquisition rate in 10000 rounds of interaction, which also shows the robust performance of the model in resisting different malicious attacks.

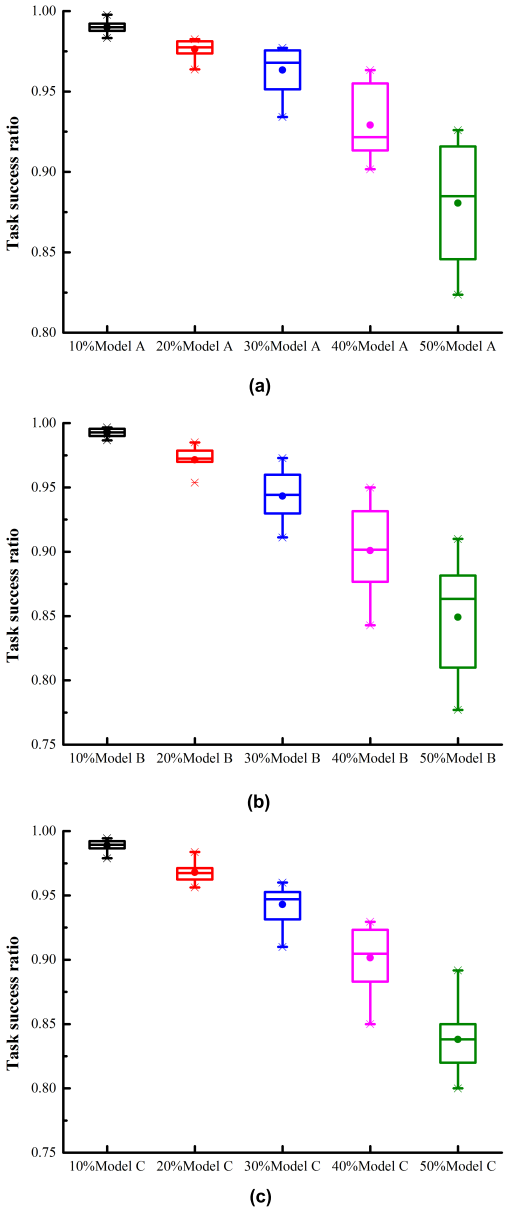


FIGURE 7. Task success ratio performance in contrast with different attack models. (a) PCD=100% and the attack model is Model A. (b) PCD=100% and the attack model is Model B. (c) PCD=100% and the attack model is Model C.

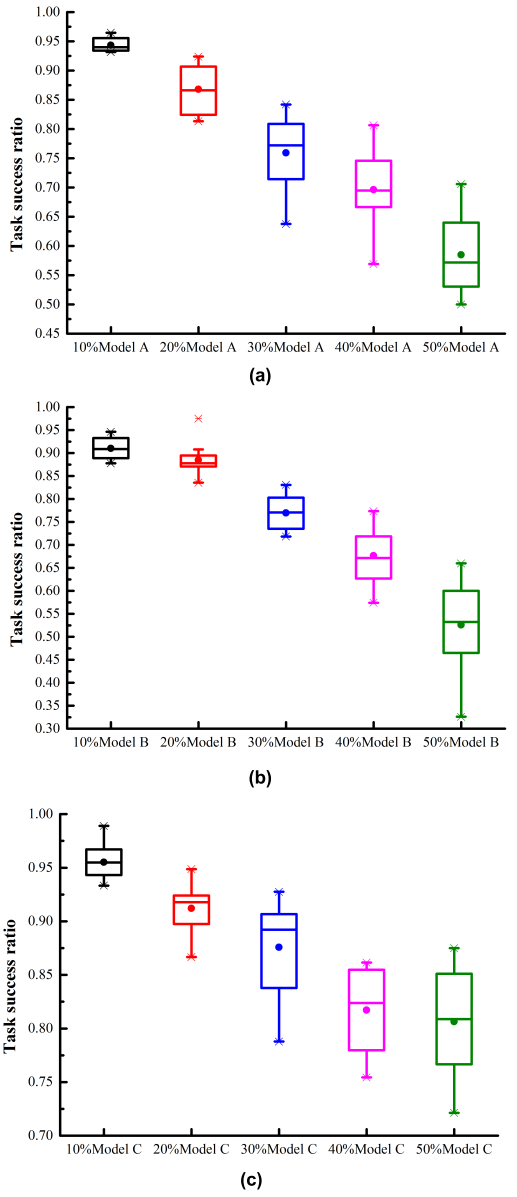


FIGURE 8. Task success ratio performance in contrast with different attack models. (a) PCD=10% and the attack model is Model A. (b) PCD=10% and the attack model is Model B. (c) PCD=10% and the attack model is Model C.

In a 10% idle IoT edge computing system, the model cannot accurately estimate the credibility of each end-user due to limited trust feedback information. As shown in Fig. 8(a), Fig. 8(b), and Fig. 8(c), the performance of the system will repeatedly oscillate over time, and progressive improvement cannot be achieved. How to improve the end-users trust evaluation under the condition of less feedback information is the goal of this paper's future work. In the IoT edge computing system with 20% and 40% collaborative systems, the performance of the model is between PCD=10% and PCD=100%. In order to save space, it is not shown here.

2) COMPLEX ATTACKS

In the real-world edge computing scenario, there may be multiple types of malicious users at the same time. Therefore, in order to further study the effectiveness and robustness of the trust model in this paper, we need to consider complex attacks (that is, complex attacks that include multiple attack models at the same time) on the experimental performance. The experimental settings for the complex attack scenarios considered in this article are as follows: (1) The distribution ratio of malicious end-users are 10%, 20%, 30%, 40%, and 50%, respectively; (2) The values of PCD are 10%, 20%, 40%, and 100%, respectively; (3) To study the impact of different proportions of complex attack models on the success

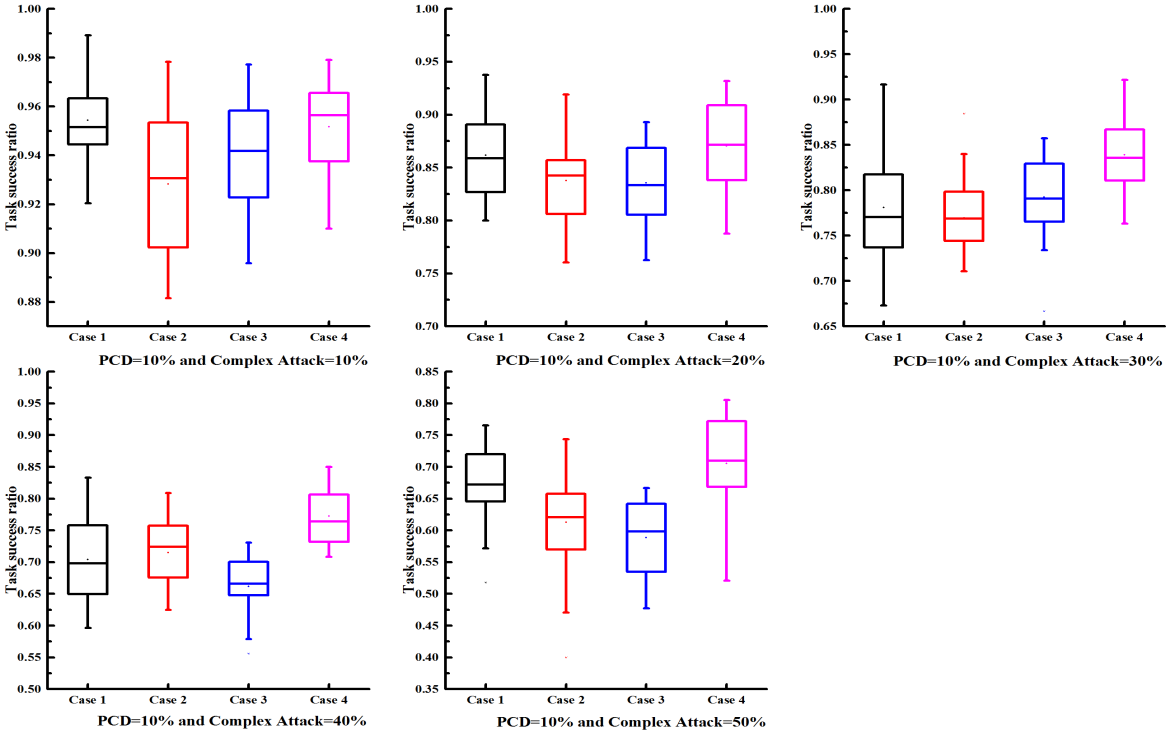


FIGURE 9. Task success ratio performance in contrast with different complex attack models under PCD=10%.

ratio of the task, we consider four complex attack scenarios: Case1 (malicious end-users randomly select Models A, B, and C with a 1 / 3 probability respectively), Case2 (malicious end-users randomly select Models A, B, and C with a probability of 15%, 60%, and 25%. At this time, Model B is dominant.), Case 3 (malicious end-users select Models A, B, and C with a probability of 60%, 25%, and 15%, respectively. At this time, Model A is dominant.) and Case 4 (malicious end-users choose Models A, B, and C with a probability of 25%, 15%, and 60%, respectively. At this time, Model C is dominant.). The results are shown in Fig. 9 and Fig.10, and we can get the following three findings: (1) As the distribution of malicious end-users increases, it is expected that the task success ratio will decrease, but the decrease rate is relatively slow, which indicates that the trust model in this paper has robust performance; (2) With the increase of PCD (that is, the increase of transaction data), the task success ratio will also increase, especially when PCD = 100%, even under the 50% of malicious users, the average success ratio of the task can reach 80%, which shows the effectiveness of the model proposed in this paper; (3) Under different proportions of complex attack models, the trust model proposed in this paper can achieve similar results, which verifies the mechanism's ability to resist attacks against complex models. To sum up: under the complex attack model, the trust model proposed in this paper can still maintain good performance.

D. THE EFFECT OF VALUE f ON THE PERFORMANCE OF THE MODE

We further study the impact of the "selective behavior attack" misbehaves (i.e., f) on the performance of the proposed trust model. Notably, the parameter f is arranged from 10% to 90% with an interval of 10%. The experimental results are shown in Fig. 11 and Fig. 12. It can be seen from the figure that as the value of f increases, the success ratio of the task increases. Fig. 11 and Fig. 12 show that with the same value f, the more malicious nodes, the lower the task success ratio. Because there are many experiments, only parts of the content are shown here, and other experimental results are similar to the results shown.

E. COMPARATIVE EXPERIMENT

In Fig. 13, the performance of this model will be compared with TCM [32], Eigen Trust [23], and Group Trust [31]. The reason we chose EigenTrust, GroupTrust is because distributed P2P technology has been similar to the edge computing model, but the latter has extended the former to extend the concept of P2P to network edge devices. So we want to compare whether the algorithm proposed in this paper is better than these excellent algorithms. We choose TCM because TCM is a new IoT system trust management mechanism. It is an adaptive model based on information entropy theory, which calculates the indirect trust value of the device from the direct evaluation information. In the experimental scenario with PCD=10% and attack model B, as shown in Fig. 13(a) and Fig. 13(b), we find that: (1) Set with 10%, 20%, 30%, and 40% malicious end-user ratio, the proposed model achieves the best performance with 91.026%, 88.501%, 76.990%, and 67.677% of the task completion rate. However, under the 50% malicious end-user ratio setting, the performance of the proposed model is slightly worse than the performance of the GroupTrust model due to the randomness problem caused

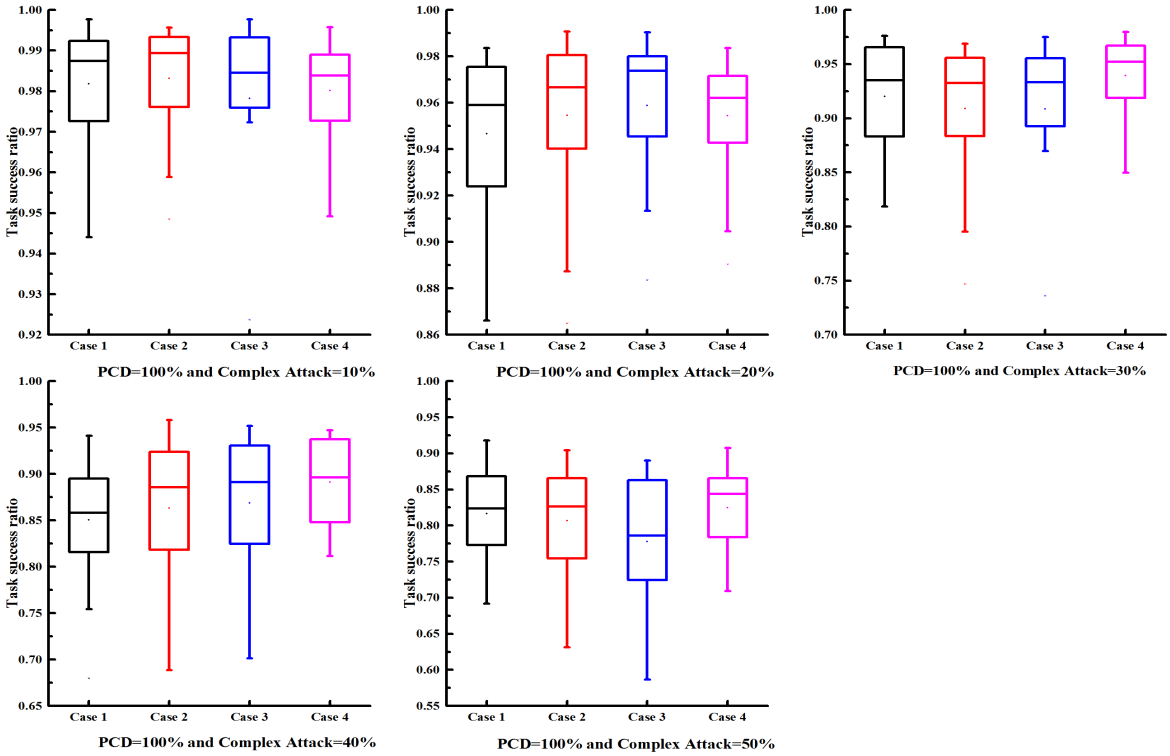


FIGURE 10. Task success ratio performance in contrast with different complex attack models under PCD=100%.

by the lack of evaluation information. (2) With the increase of the malicious end-user ratio, the system performance shows a gradual downward trend. In the experimental scenario with PCD=20% and attack model B, there is an absolute increase in the number of interactions between end-users. As shown in Fig. 13(b) and Fig. 13(c), we find that: (1) The proposed model achieves optimal performance under 20%, 30%, 40%, and 50% malicious end-user distribution ratios. However, because the trust evaluation information is still not enough, the model proposed in this paper cannot evaluate the credibility of the end-user very accurately and then achieves a task completion rate slightly worse than the Group Trust model under the 10% malicious end-user ratio setting. (2) Compared with the scenario of PCD=10%, there is an absolute increase in trust evaluation information. The task completion rate of the system in the scenario of PCD=20% has been improved.

In the experimental scenario with PCD=40% and attack model B, the interaction between end-users is more frequent. As shown in Fig. 13(c) and Fig. 13(d), we find that: (1) With the increase of trust evaluation information, the model proposed in this paper more accurately evaluates the credibility of end-users and then achieves the optimal performance under different malicious end-user distribution ratios. (2) Compared with the scenario of PCD=20%, the task success rate of the system has been further improved.

In the experimental scenario of PCD = 100% and attack model B, the edge computing system is extremely busy. As shown in Fig. 13(d) and Fig. 13(e), we find that: (1) Similar to the results in PCD=40%, the proposed model achieves the optimal performance under different malicious end-user distribution ratios. (2) Compared with PCD = 40%, the success rate of the system has been greatly improved. Even under the 50% malicious end-user distribution ratio, the proposed model achieves a success rate of 84.907%. Fig. 13 shows only the experimental comparison results in the scenario of the attack model B. In the scenario of attack Model A and Model C, we still get similar results.

F. EVOLUTIONARY GAME EXPERIMENT

In this part, we use experimental methods to verify the game model proposed in this paper. We experimented with two scenarios. The first scenario is the personal healthcare monitoring management system scenario in a smart city, and the other is an air-quality monitoring and analysis system. The personal healthcare monitoring management system is a monitoring system for various personal health data such as heartbeat, blood pressure, blood oxygen saturation, human posture. It provides real-time monitoring and early warning for patients. The air environment analysis system is an intelligent system for collecting, analyzing, and forecasting indicators such as ozone, carbon dioxide, nitrogen dioxide, sulfur dioxide, carbon monoxide, and PM2.5 in the monitored area. In the experiment, we set the experimental parameters as shown in Table 2. It should be noted that in real life, due to the importance of human life, the loss of invalid services

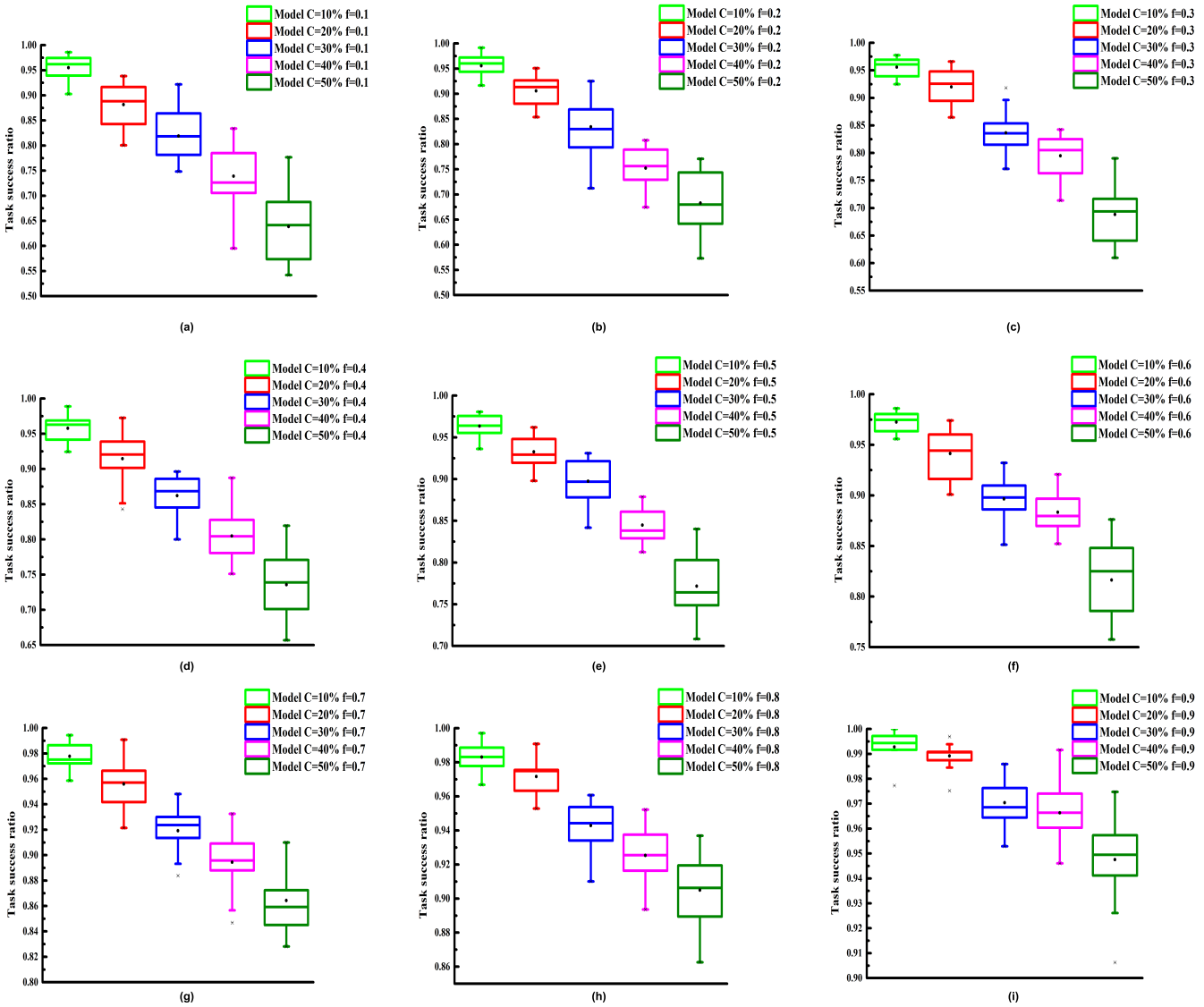


FIGURE 11. Task success ratio performance (a) PCD=40% and f=0.1. (b) PCD=40% and f=0.2. (c) PCD=40% and f=0.3. (d) PCD=40% and f=0.4. (e) PCD=40% and f=0.5. (f) PCD=40% and f=0.6. (g) PCD=40% and f=0.7. (h) PCD=40% and f=0.8. (i) PCD=40% and f=0.9.

in the personal healthcare monitoring system must be higher than that in the air-quality analysis system, and the cost of invalid services in the former is higher than that in the latter.

1) THE IMPACT OF THE COST OF CHOOSING THE RECOMMENDATION SERVICE ON THE GAME MODEL

We studied the effect of c_r on the game model of the personal healthcare monitoring system under $\varepsilon = 0$ and $\varepsilon = 0.1$ settings. The value of c_r is set to a positive integer between 0 and 12 when $\varepsilon = 0$. From the experimental results, it can be found that strategy s_3 finally dominates the whole system when $c_r <= 1$, because the strategy s_3 obtains the highest expected fitness. The system is in the state of oscillation when $c_r = 2$. Based on Definition 4, we prove this phenomenon. The three strategies coexist when $2 < c_r <= 8$. When $c_r = 4$, the proportion of each strategy when the system is stable is $x_1 = 0.205791, x_2 = 0.140849$, and $x_3 = 0.653360$, which indicates that the acceptance rate of the recommended system is 0.65336 and verifies the evolutionary stability shown in Proposition 1. With the increasing value of c_r, the profit of the strategy s_3 gradually decreases, so the proportion of the strategy s_2 will be higher in the steady-state. The coexistence of strategy s_1 and strategy s_2 occurred when $8 < c_r <= 11$. This phenomenon is because the expected fitness of the strategy s_3 promotes the growth of the strategy s_2, while the increase of the strategy s_2 inhibits the proliferation of the strategy s_1 population, which eventually leads to its disappearance from the system. When $c_r = 12$, the strategy s_2 dominates the whole system because the strategy s_2 achieves the highest expected fitness. In summary, we find that the proportion of the strategy s_2 in the system is positively correlated with the value c_r. The results are shown in Fig. 14.

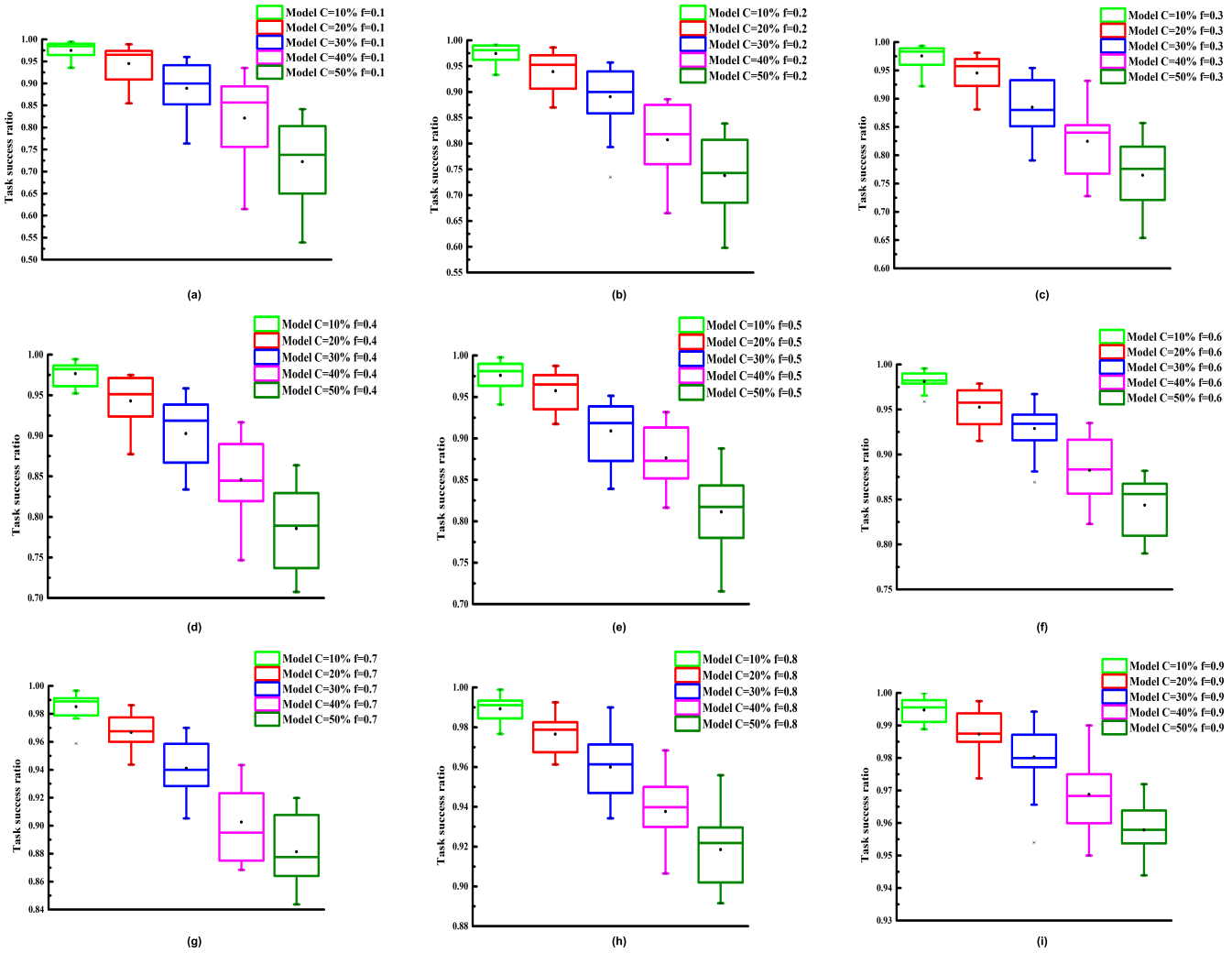


FIGURE 12. Task success ratio performance (a) PCD=100% and f=0.1. (b) PCD=100% and f=0.2. (c) PCD=100% and f=0.3. (d) PCD=100% and f=0.4. (e) PCD=100% and f=0.5. (f) PCD=100% and f=0.6. (g) PCD=100% and f=0.7. (h) PCD=100% and f=0.8. (i) PCD=100% and f=0.9.

The value of c_r is set to a positive integer between 0 and 12 when $\varepsilon = 0.1$. From the experimental results, it can be found that strategy s_3 finally dominates the whole system when $c_r = 0$ because the strategy s_3 obtains the highest expected fitness. The system is in the state of oscillation when $0 < c_r = 2$. The three strategies coexist when $2 < c_r <= 5$. With the increasing value of c_r, the profit of the strategy s_3 gradually decreases, so the proportion of the strategy s_2 will be higher in the steady-state. When $c_r = 6$, the strategy s_2 dominates the whole system. In summary, we find that the proportion of the strategy s_2 in the system is positively correlated with the value c_r. We find that in the case of $\varepsilon = 0$, the strategy s_2 dominates the entire system when $c_r = 12$, and in the case of $\varepsilon = 0.1$, as long as $c_r = 6$, the strategy s_2 dominates the whole system. In the case of a recommended error rate, we should appropriately reduce the recommended cost c_r to ensure the effective operation of the system. The results are shown in Fig. 15.

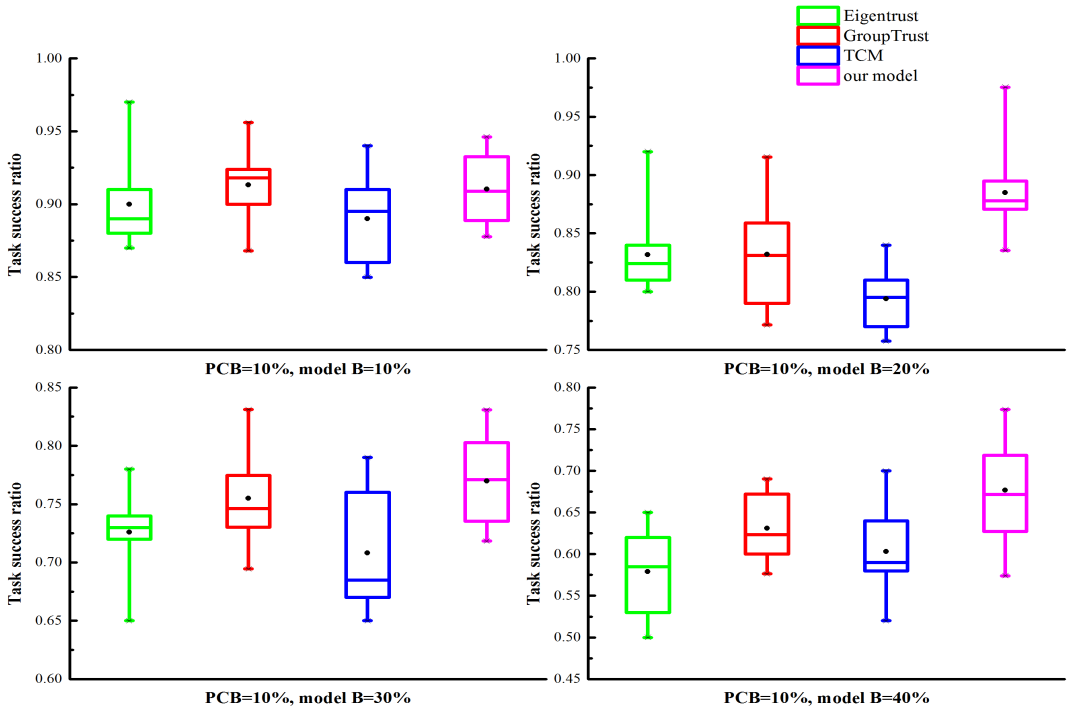
We studied the effect of c_r on the game model of the air-quality monitoring and analysis system under $\varepsilon = 0$ and $\varepsilon = 0.1$ settings. The value of c_r is set to a positive integer between 0 and 9 when $\varepsilon = 0$. From the experimental results, it can be found that strategy s_2 finally dominates the whole system when $c_r = 9$. The results are shown in Fig. 16.

The value of c_r is set to a positive integer between 0 and 1 when $\varepsilon = 0.1$. From the experimental results, it can be found that the strategy s_3 finally dominates the whole system when $0 < c_r <= 0.5$. The system is in the state of oscillation when $0.6 < c_r <= 0.8$. When $c_r > 0.8$, the strategy s_2 dominates the entire system. The results are shown in Fig. 17.

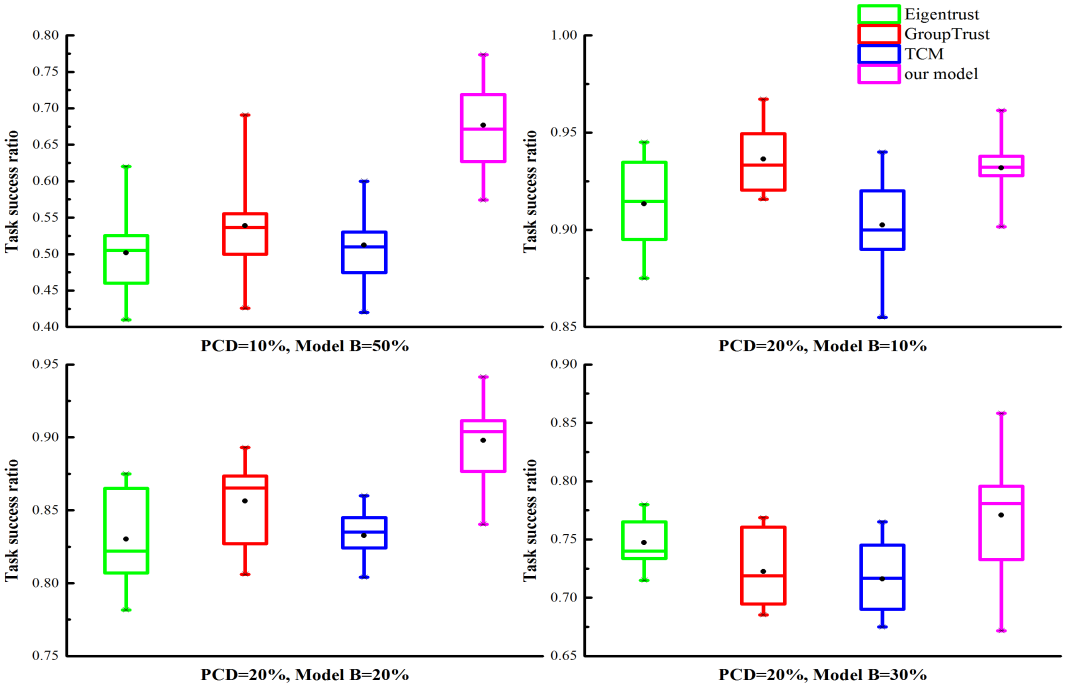
Based on the experimental results in Fig. 14, Fig. 15, Fig. 16, and Fig. 17, we find that in the data-sensitive scenario of a personal healthcare monitoring system, users have a high degree of acceptance of the recommendation cost. While in the insensitive data scenario of air quality monitoring and analysis system, users are often unable to accept the higher recommendation cost.

2) THE IMPACT OF THE ERROR RATE ON THE GAME MODEL

Due to data sparsity, data noise, and other reasons, the selective recommendation system is often less than 100% accurate, so it is necessary to study the impact of the recommended



(a)



(b)

FIGURE 13. Task success ratio performance in contrast with different trust models.

error rate on the game model. We studied the effect of ε on the game model of the personal healthcare monitoring system under $c_r = 0.5$(low cost) and $c_r = 1$(high cost)settings.

As shown in Fig. 18, when $c_r = 0.5$ and the error rate is $\varepsilon <= 0.25$ since strategy s_3 obtains the highest expected fitness. Therefore, strategy s_3 dominates the entire model.

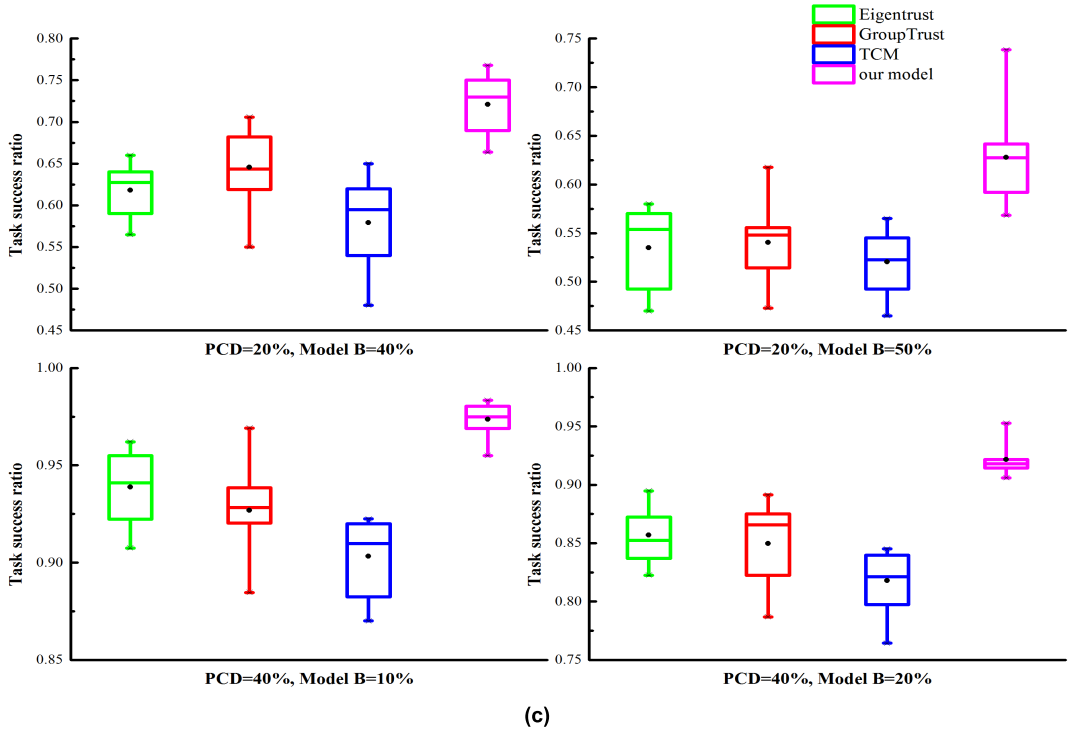


FIGURE 13. *(Continued.)* **Task success ratio performance in contrast with different trust models.**

As the error rate increases, the expected fitness of strategy s_3 gradually decreases. When $\varepsilon >= 0.3$, strategy s_2 dominates the whole model.

As shown in Fig. 19, when $c_r = 1$ and the error rate is $\varepsilon <= 0.25$, the game model is always in three strategic oscillation states. The Lyapunov principle can demonstrate this phenomenon in *Definition* 4. When $\varepsilon >= 0.3$, because strategy s_2 achieves the highest expected fitness, so strategy s_2 dominates the whole model.

By comparing the experimental results in Fig. 18 and Fig. 19, we find that in the case of $c_r = 0.5$, as long as $\varepsilon <= 0.25$, the strategy s_3 dominates the whole model.

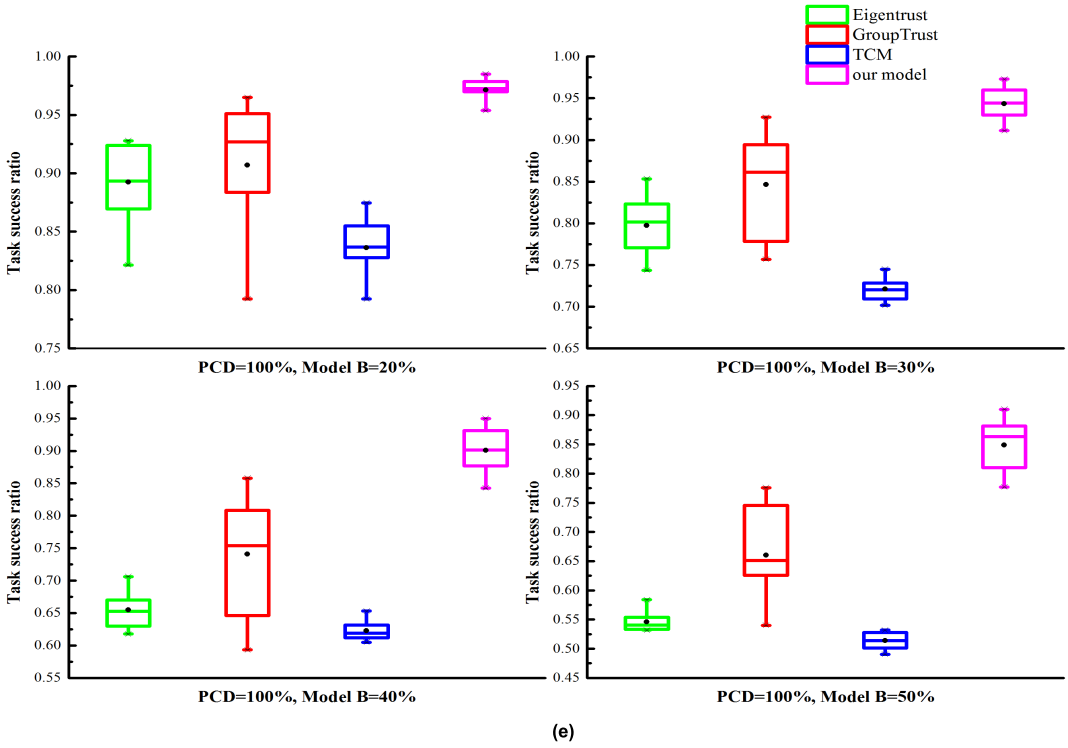


FIGURE 13. *(Continued.)* **Task success ratio performance in contrast with different trust models.**

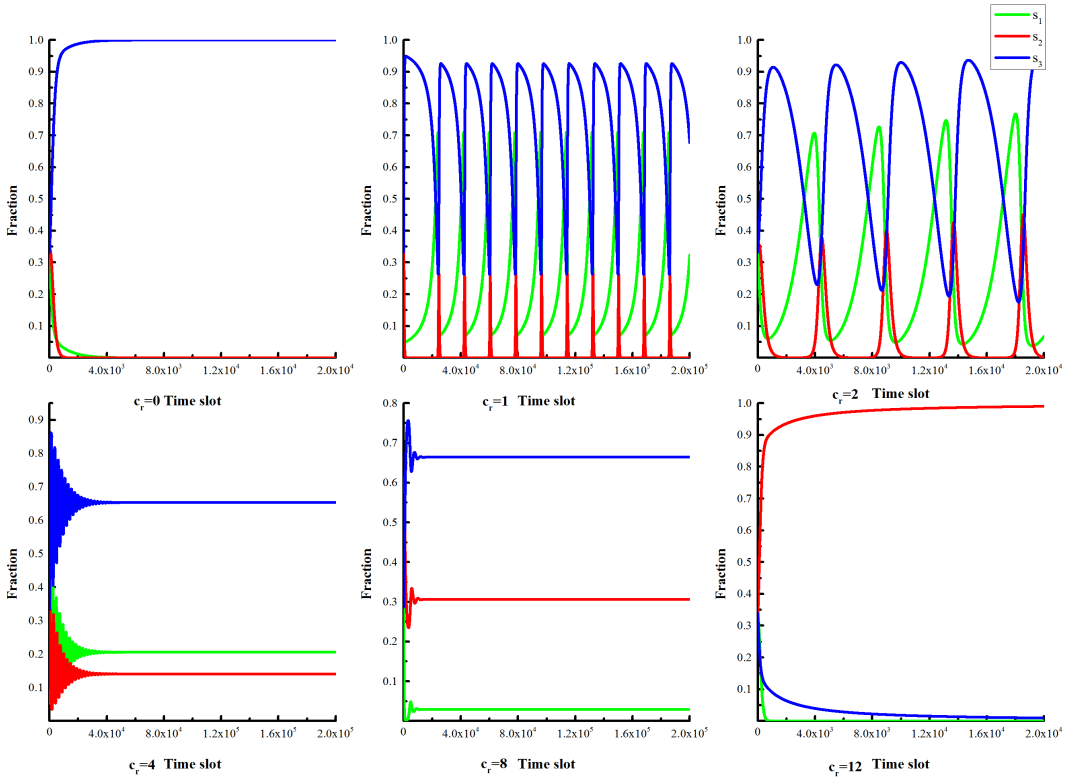


FIGURE 14. **The effect of c_r on the game model of the personal healthcare monitoring system under $\varepsilon = 0$.**

However, in the case of $c_r = 1$, the strategy s_3 cannot achieve the highest expected fitness. Therefore, the smaller the value c_r is, the higher the tolerance of the model to the error rate is.

We studied the effect of ε on the game model of the air-quality monitoring and analysis system under $c_r = 0.5$ and $c_r = 1$ settings. When $c_r = 0.5$, the result is similar to that

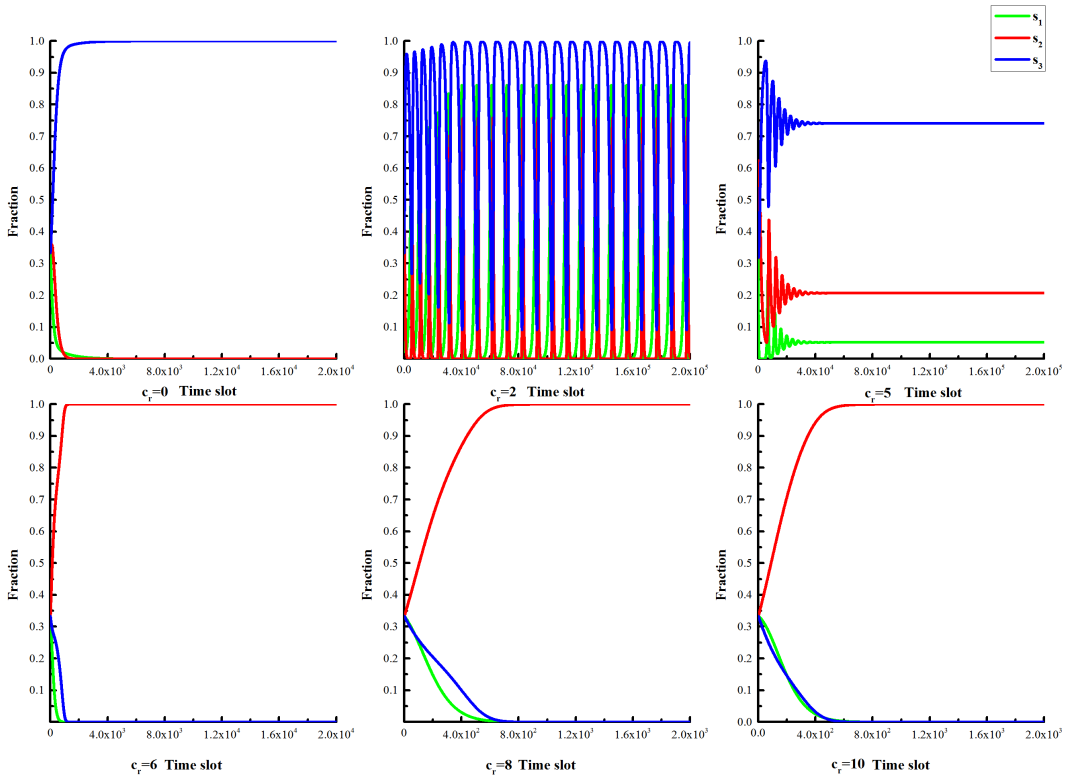


FIGURE 15. The effect of c_r on the game model of the personal healthcare monitoring system under $\varepsilon = 0.1$.

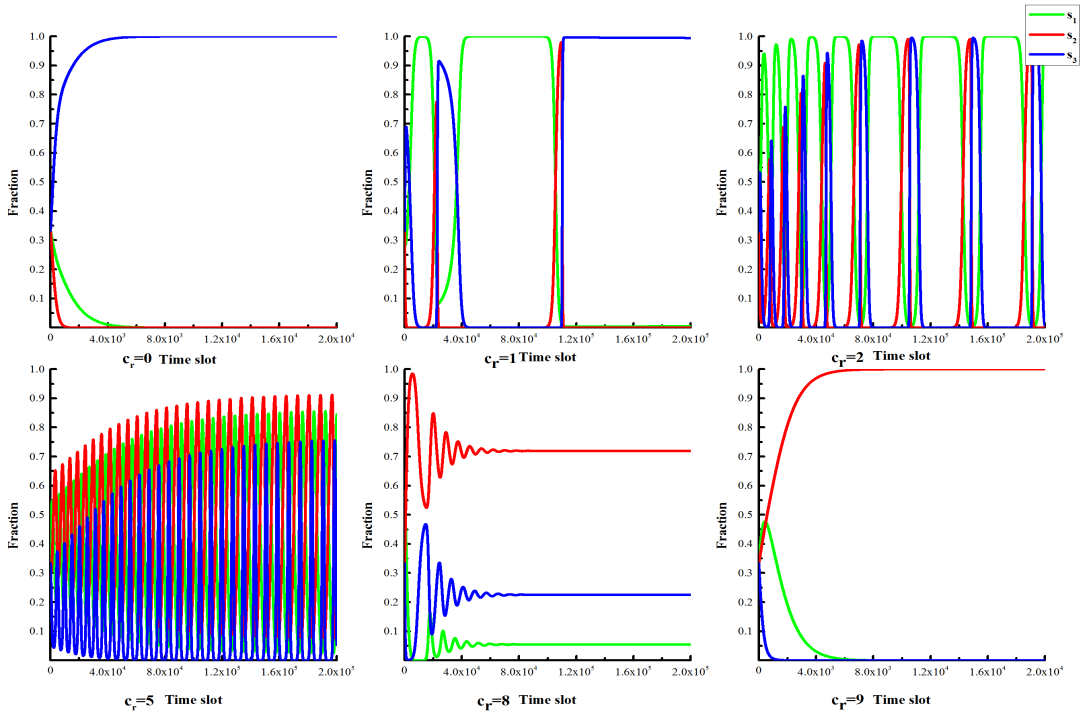


FIGURE 16. The effect of c_r on the game model of the air-quality monitoring and analysis system under $\varepsilon = 0$.

in Fig. 16. The strategy s_3 dominates the whole model when $\varepsilon <= 0.3$. The strategy s_2 dominates the whole model when $\varepsilon >= 0.4$. The results are shown in Fig. 20.

Fig. 21 shows the experimental results with $c_r = 1$, which is similar to the results in Fig. 19. The model is always in the three-strategy oscillation state when $\varepsilon <= 0.15$. The strategy

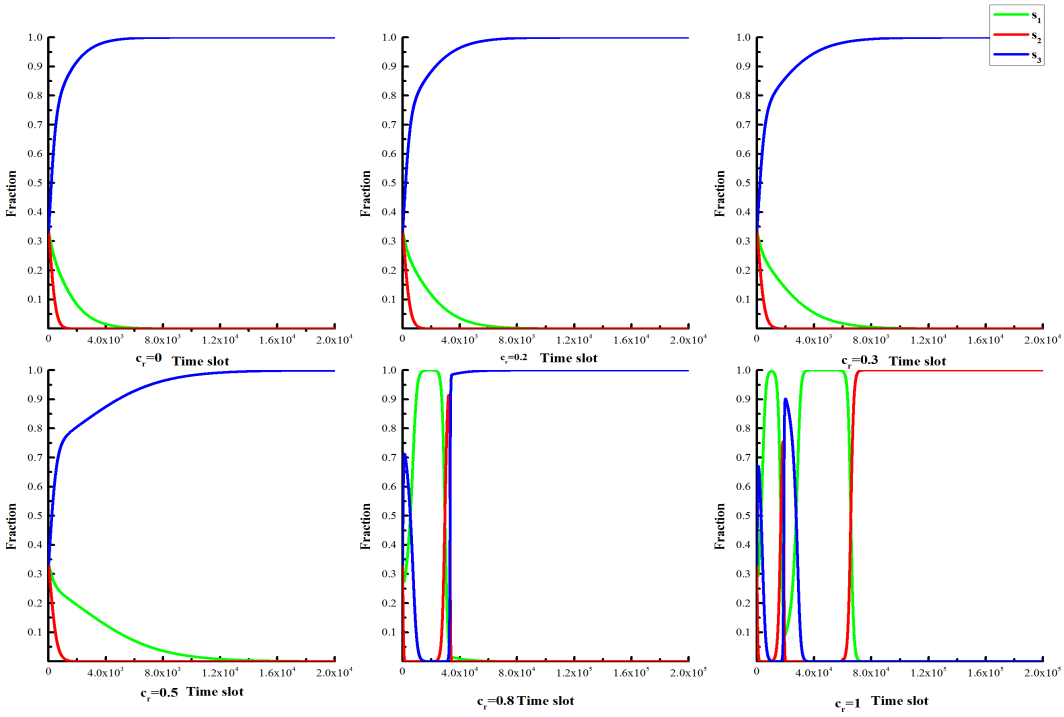


FIGURE 17. The effect of c_r on the game model of the air-quality monitoring and analysis system under $\varepsilon = 0.1$.

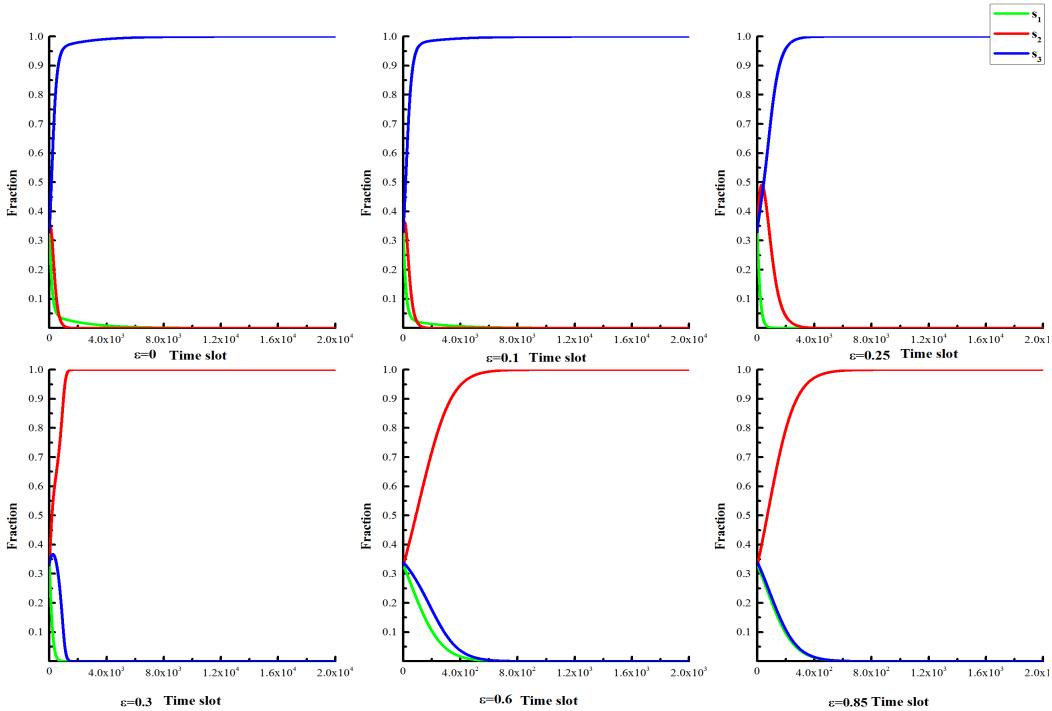


FIGURE 18. The effect of ε on the game model of the personal healthcare monitoring system under $c_r = 0.5$.

s_2 dominates the whole model when $\varepsilon >= 0.2$. Based on the experimental results of Fig. 18, Fig. 19, Fig. 20, and Fig. 21, we found that in the data-sensitive scenario of the personal healthcare monitoring system, users are more inclined to use the selective recommendation mechanism, so they have a greater acceptance of the recommendation error rate. To the extent that the air-quality analysis system is not sensitive to data, users do not have a strong willingness to use the

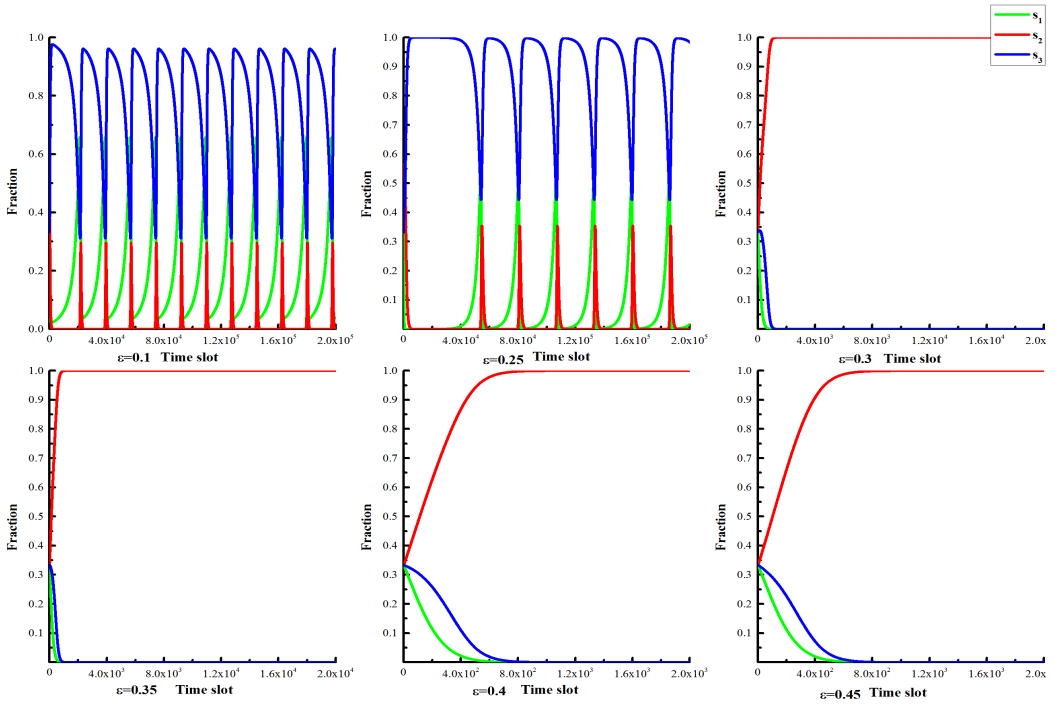


FIGURE 19. The effect of ε on the game model of the personal healthcare monitoring system under $c_r = 1$.

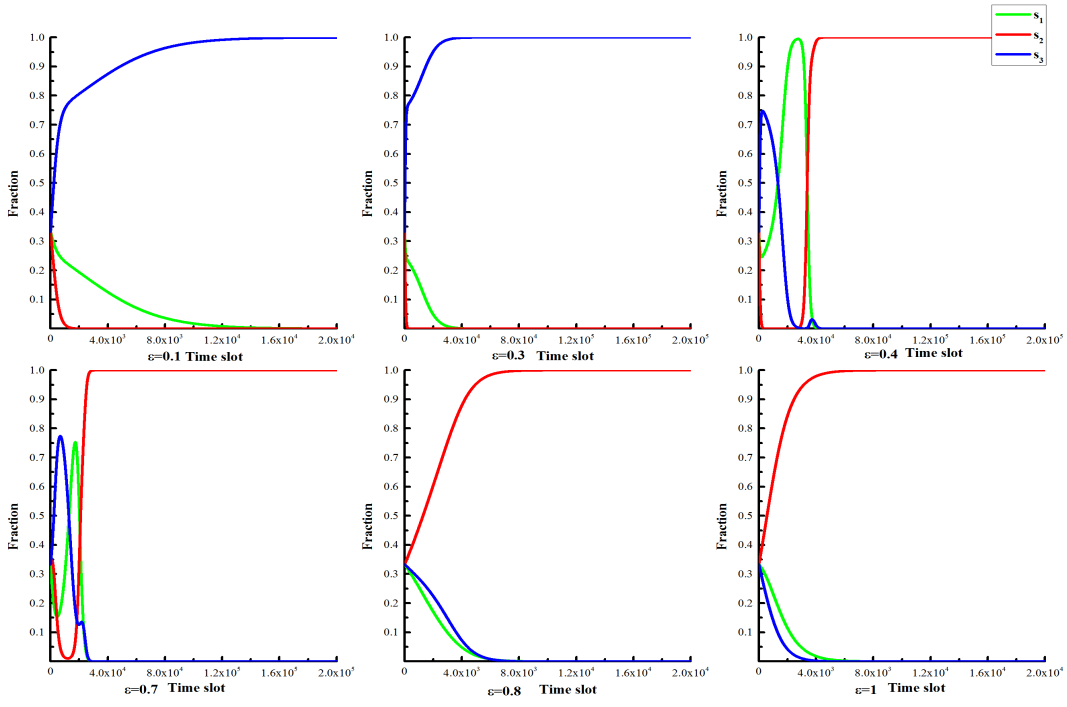


FIGURE 20. The effect of ε on the game model of the air-quality monitoring and analysis system under $c_r = 0.5$.

selective recommendation mechanism, so they have a small acceptance of the recommended error rate.

G. INFLUENCE OF TOLERANCE VALUE

In order to study the impact of different tolerance values on the trust model in this paper, we set the user tolerance value to a random value from 1 to 100 and compared the experimental results with the experimental results of similar tolerance values as shown in Fig. 22, Fig. 23, and Fig. 24. In Fig. 22, we set the malicious user to adopt the attack model A. By comparing with the homogeneous tolerance value scenario ($\delta_i = 10$), we can find that under the user heterogeneous

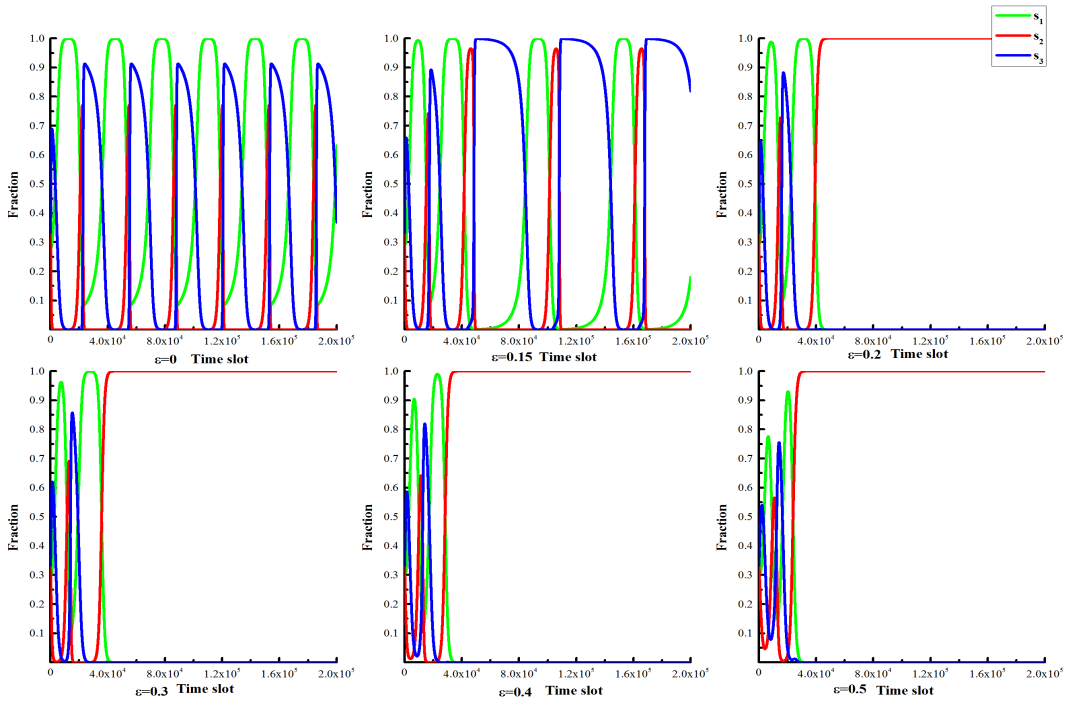


FIGURE 21. The effect of ε on the game model of the air-quality monitoring and analysis system under $c_r = 1$.

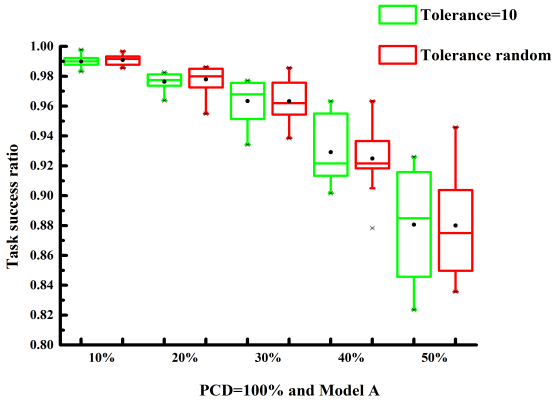


FIGURE 22. The effect of tolerance value on the trust model of under Model A.

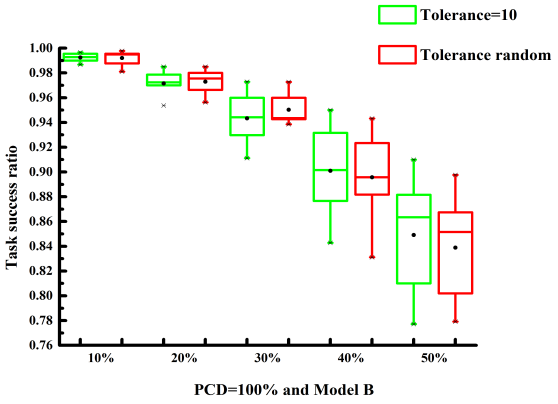


FIGURE 23. The effect of tolerance value on the trust model of under Model B.

tolerance value setting, the trust model in this paper obtains similar results with $\delta_i = 10$, which shows that when the attack model is A, the heterogeneous tolerance value will not have a significant impact on the trust model proposed in this paper. Similarly, as shown in Fig. 23, we can find that in the case of the attack model B, the heterogeneous tolerance value will not have a significant impact on the trust model proposed in this paper. However, as shown in Fig. 24, we find that when the attack model is C, compared with the setting of the homogeneous tolerance value, the different tolerance values can further improve the performance of the trust model proposed in this paper. The reason for this phenomenon is: In the experiment, the user tolerance value is set to a random value from 1 to 100. At this time, the tolerance value of most users will be higher than 10, and the strict tolerance value setting will well inhibit the camouflage users from obtaining high trust value. Therefore, when the attack model

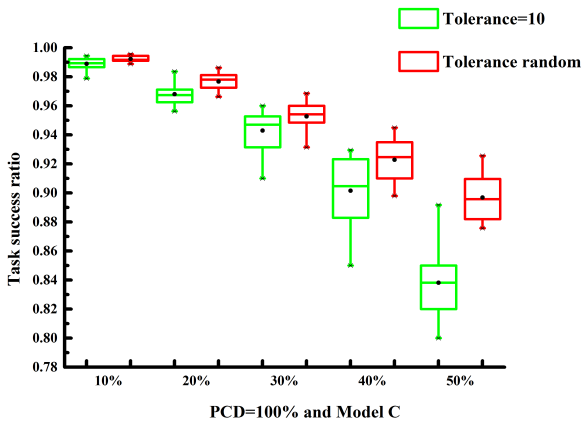


FIGURE 24. The effect of tolerance value on the trust model of under Model C.

is C, the trust model proposed in this paper can achieve better performance under the setting of different tolerance values.

VII. CONCLUSION

This paper presents a reliable IoT edge computing trust management mechanism for smart cities. In this paper, an intelligent device selective recommendation mechanism based on the dynamic black-and-white list is proposed to solve the problem of selecting trusted participants to improve the service quality of the edge computing system of the IoT in the smart city. Experiments show that the trust management mechanism proposed in this paper has a significant role in promoting the cooperation of multi intelligent devices in the IoT edge computing system. It more reliably resists the malicious attacks to service providers and is suitable for the large-scale IoT edge computing system in the smart city. Future work includes how to improve the accuracy of the end-user trust assessment with less feedback and how to solve the trust problem of a large number of complex malicious users in IoT edge computing.

APPENDIX

According to the Equation (17), Equation (20), and Equation (23), $\bar{f}(t)$ is obtained as shown in the Equation (25).

$$\begin{aligned}\bar{f}(t) = &\left(x_1^2 + x_1x_3 + x_1x_2 + \varepsilon x_2x_3 + \frac{x_1x_3 + x_3^2}{x_1 + \varepsilon x_2 + x_3}\right)b_p \\ &+ \left(x_1x_2 + x_2^2 + \frac{\varepsilon x_2x_3}{x_1 + \varepsilon x_2 + x_3}\right)b_n \\ &+ (x_1x_2 + x_2^2 + \frac{\varepsilon x_2x_3}{x_1 + \varepsilon x_2 + x_3})c_n \\ &- \left(x_1^2 + x_1x_2 + \frac{x_1x_3}{x_1 + \varepsilon x_2 + x_3} + x_1x_3 + \varepsilon x_2x_3 \right. \\ &\left. + \frac{\varepsilon x_3^2}{x_1 + \varepsilon x_2 + x_3}\right)c_p \\ &- x_3c_r \end{aligned} \tag{25}$$

The dynamic replication equations of the strategies s_1, s_2, and s_3 are shown in Equation (26), Equation (27), and Equation (28), respectively. In this paper, the Lyapunov stability theory is used to prove the stability of the game model. For the replication dynamic equations of Equation (26), Equation (27), and Equation (28), the Jacobian matrix as shown in Equation (29) can be obtained.

$$\begin{aligned}&\dot{x}_{s_1} \\ &= x_1(t)\left(\bar{f}_{s_1}(t) - \bar{f}(t)\right) \\ &= \left(x_1^2 + x_1x_3 - x_1^3 - x_1^2x_3 - x_1^2x_2 - \varepsilon x_1x_2x_3 + \frac{-x_1^2x_3 - x_1x_3^2}{x_1 + \varepsilon x_2 + x_3}\right)b_p \\ &\quad + \left(-x_1^2 - x_1x_2 + x_1^3 + x_1^2x_2 + x_1^2x_3 + \varepsilon x_1x_2x_3 \right. \\ &\quad \left. + \frac{\varepsilon x_1x_3^2}{x_1 + \varepsilon x_2 + x_3} - \frac{x_1x_3}{x_1 + \varepsilon x_2 + x_3} + \frac{x_1^2x_3}{x_1 + \varepsilon x_2 + x_3}\right)c_p \\ &\quad + \left(x_1x_2 - x_1^2x_2 - x_1x_2^2 - \frac{\varepsilon x_1x_2x_3}{x_1 + \varepsilon x_2 + x_3}\right)b_n \\ &\quad - (x_1^2x_2 + x_1x_2^2 + \frac{\varepsilon x_1x_2x_3}{x_1 + \varepsilon x_2 + x_3})c_n + x_1x_3c_r \end{aligned} \tag{26}$$

$$\begin{aligned}&\dot{x}_{s_2} \\ &= x_2(t)\left(\bar{f}_{s_2}(t) - \bar{f}(t)\right) \\ &= \left(x_1x_2 + \varepsilon x_2x_3 - x_1^2x_2 - x_1x_2x_3 - x_1x_2^2 \right. \\ &\quad \left. - \varepsilon x_2^2x_3 + \frac{-x_1x_2x_3 - x_2x_3^2}{x_1 + \varepsilon x_2 + x_3}\right)b_p \\ &\quad + \left(x_2^2 - x_1x_2^2 - x_2^3 - \frac{\varepsilon x_2^2x_3}{x_1 + \varepsilon x_2 + x_3}\right)b_n \\ &\quad + \left(x_1^2x_2 + x_1x_2^2 + \frac{x_1x_2x_3}{x_1 + \varepsilon x_2 + x_3} + x_1x_2x_3 \right. \\ &\quad \left. + \varepsilon x_2^2x_3 + \frac{\varepsilon x_2x_3^2}{x_1 + \varepsilon x_2 + x_3}\right)c_p \\ &\quad + (-x_1x_2^2 - x_2^3 - \frac{\varepsilon x_2^2x_3}{x_1 + \varepsilon x_2 + x_3} \\ &\quad + x_1x_2 + x_2^2 + \frac{\varepsilon x_2x_3}{x_1 + \varepsilon x_2 + x_3})c_n + x_2x_3c_r \end{aligned} \tag{27}$$

$$\begin{aligned}&\dot{x}_{s_3} \\ &= x_3(t)\left(\bar{f}_{s_3}(t) - \bar{f}(t)\right) \\ &= \left(\frac{x_1x_3 + x_3^2}{x_1 + \varepsilon x_2 + x_3} - x_1^2x_3 - x_1x_3^2 - x_1x_2x_3 \right. \\ &\quad \left. - \varepsilon x_2x_3^2 + \frac{-x_1x_3^2 - x_3^3}{x_1 + \varepsilon x_2 + x_3}\right)b_p \\ &\quad + \left(\frac{\varepsilon x_2x_3 - \varepsilon x_2x_3^2}{x_1 + \varepsilon x_2 + x_3} - x_1x_2x_3 - x_2^2x_3\right)b_n \\ &\quad - (x_1x_2x_3 + x_2^2x_3 + \frac{\varepsilon x_2x_3^2}{x_1 + \varepsilon x_2 + x_3})c_n + \left(x_3^2 - x_3\right)c_r \\ &\quad + \left(x_1^2x_3 + x_1x_2x_3 + \frac{x_1x_3^2}{x_1 + \varepsilon x_2 + x_3} + x_1x_3^2 + \varepsilon x_2x_3^2 \right. \\ &\quad \left. + \frac{\varepsilon x_3^3}{x_1 + \varepsilon x_2 + x_3} - x_1x_3 - \varepsilon x_2x_3 - \frac{\varepsilon x_3^2}{x_1 + \varepsilon x_2 + x_3}\right)c_p \end{aligned} \tag{28}$$

$$J = \begin{pmatrix} \frac{\partial \dot{x}_{s_1}}{\partial x_{s_1}} & \frac{\partial \dot{x}_{s_1}}{\partial x_{s_2}} & \frac{\partial \dot{x}_{s_1}}{\partial x_{s_3}} \\ \frac{\partial \dot{x}_{s_2}}{\partial x_{s_1}} & \frac{\partial \dot{x}_{s_2}}{\partial x_{s_2}} & \frac{\partial \dot{x}_{s_2}}{\partial x_{s_3}} \\ \frac{\partial \dot{x}_{s_3}}{\partial x_{s_1}} & \frac{\partial \dot{x}_{s_3}}{\partial x_{s_2}} & \frac{\partial \dot{x}_{s_3}}{\partial x_{s_3}} \end{pmatrix} = \begin{pmatrix} J_{11} & J_{12} & J_{13} \\ J_{21} & J_{22} & J_{23} \\ J_{31} & J_{32} & J_{33} \end{pmatrix} \tag{29}$$

Proposition 1 is obtained by *Definition* 4.

Proposition 1: When $bp = 7, bn = -15, cp = 1, cn = -10, cr = 4$, (0.205791 0.140849 0.653360) is the stable evolution state of the model.

Proof: When $x_1 = 1$, $x_2 = 0$ and $x_3 = 0$ are substituted into the Jacobian matrix, the eigenvalues are $\lambda_1 = -12$, $\lambda_2 = 11$, and $\lambda_3 = -6$. According to

Definition 4, since the eigenvalue λ_2 is positive, the value is not a stable point. When $x_1 = 0$, $x_2 = 1$, and $x_3 = 0$ are substituted into the Jacobian matrix because there is a case where the denominator is 0. According to *Definition* 4, the value is not a stable point. When $x_1 = 0$, $x_2 = 0$, and $x_3 = 1$ are substituted into the Jacobian matrix, the eigenvalues are $\lambda_1 = -3$, $\lambda_2 = 3$, and $\lambda_3 = -3$. According to *Definition* 4, since the eigenvalue λ_2 is positive, the value is not a stable point. When $x_1 = 0.205791$, $x_2 = 0.140849$, and $x_3 = 0.653360$ are substituted into the Jacobian matrix, the eigenvalues are $\lambda_1 = -1.39000232+1.69496275j$, $\lambda_2 = -1.39000232-1.69496275j$, and $\lambda_3 = -0.09492218$. According to *Definition* 4, since the eigenvalue λ_2 is not positive, the value is a stable point.

REFERENCES

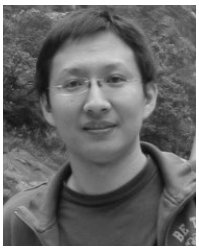
[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2347–2376, Jun. 2015.
[2] A. Botta, W. de Donato, V. Persico, and A. Pescapé, "Integration of cloud computing and Internet of Things: A survey," *Future Gener. Comput. Syst.*, vol. 56, pp. 684–700, Mar. 2016.
[3] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016.
[4] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, 4th Quart., 2017.
[5] S. Wang, X. Zhang, Y. Zhang, L. Wang, J. Yang, and W. Wang, "A survey on mobile edge networks: Convergence of computing, caching and communications," *IEEE Access*, vol. 5, pp. 6757–6779, 2017.
[6] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Trans. Netw.*, vol. 24, no. 5, pp. 2795–2808, Oct. 2016.
[7] A. U. R. Khan, M. Othman, S. A. Madani, and S. U. Khan, "A survey of mobile cloud computing application models," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 393–413, Jul. 2014.
[8] C. You, K. Huang, H. Chae, and B.-H. Kim, "Energy-efficient resource allocation for mobile-edge computation offloading," *IEEE Trans. Wireless Commun.*, vol. 16, no. 3, pp. 1397–1411, Mar. 2017.
[9] H. Cai, B. Xu, L. Jiang, and A. V. Vasilakos, "IoT-based big data storage systems in cloud computing: Perspectives and challenges," *IEEE Internet Things J.*, vol. 4, no. 1, pp. 75–87, Feb. 2017.
[10] H. Menouar, I. Guvenc, K. Akkaya, A. S. Uluagac, A. Kadri, and A. Tuncer, "UAV-enabled intelligent transportation systems for the smart city: Applications and challenges," *IEEE Commun. Mag.*, vol. 55, no. 3, pp. 22–28, Mar. 2017.
[11] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.
[12] B. Ahlgren, M. Hidell, and E. C.-H. Ngai, "Internet of Things for smart cities: Interoperability and open data," *IEEE Internet Comput.*, vol. 20, no. 6, pp. 52–56, Nov. 2016.
[13] S. M. R. Islam, D. Kwak, M. H. Kabir, M. Hossain, and K.-S. Kwak, "The Internet of Things for health care: A comprehensive survey," *IEEE Access*, vol. 3, pp. 678–708, 2015.
[14] T. X. Tran, A. Hajisami, P. Pandey, and D. Pompili, "Collaborative mobile edge computing in 5G networks: New paradigms, scenarios, and challenges," *IEEE Commun. Mag.*, vol. 55, no. 4, pp. 54–61, Apr. 2017.
[15] Z. Ning, X. Kong, F. Xia, W. Hou, and X. Wang, "Green and sustainable cloud of things: Enabling collaborative edge computing," *IEEE Commun. Mag.*, vol. 57, no. 1, pp. 72–78, Jan. 2019.
[16] Y. He, F. R. Yu, N. Zhao, and H. Yin, "Secure social networks in 5G systems with mobile edge computing, caching, and device-to-device communications," *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 103–109, Jun. 2018.
[17] X. Huang, R. Yu, J. Kang, and Y. Zhang, "Distributed reputation management for secure and efficient vehicular edge computing and networks," *IEEE Access*, vol. 5, pp. 25408–25420, 2017.
[18] S.-T. Goh, H. Pang, R. H. Deng, and F. Bao, "Three architectures for trusted data dissemination in edge computing," *Data Knowl. Eng.*, vol. 58, no. 3, pp. 381–409, Sep. 2006.
[19] A. Gharaibeh, M. A. Salahuddin, S. J. Hussini, A. Khreishah, I. Khalil, M. Guizani, and A. Al-Fuqaha, "Smart cities: A survey on data management, security, and enabling technologies," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2456–2501, 4th Quart., 2017.
[20] J. Yuan and X. Li, "A multi-source feedback based trust calculation mechanism for edge computing," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Honolulu, HI, USA, Apr. 2018, pp. 819–824.
[21] C. Xu, K. Wang, P. Li, S. Guo, J. Luo, B. Ye, and M. Guo, "Making big data open in edges: A resource-efficient blockchain-based approach," *IEEE Trans. Parallel Distrib. Syst.*, vol. 30, no. 4, pp. 870–882, Apr. 2019.
[22] J. Hofbauer, and K. Sigmund, *Evolutionary Games and Population Dynamics*. Cambridge, U.K.: Cambridge Univ. Press, 1998.
[23] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The eigentrust algorithm for reputation management in P2P networks," in *Proc. 12th Int. Conf. World Wide Web (WWW)*, Budapest, Hungary, 2003, pp. 640–651.
[24] S. Papadopoulos, K. Bontcheva, E. Jaho, M. Lupu, and C. Castillo, "Overview of the special issue on trust and veracity of information in social media," *ACM Trans. Inf. Syst.*, vol. 34, no. 3, pp. 1–5, Apr. 2016.
[25] G. Szabó and C. Tőke, "Evolutionary prisoner's dilemma game on a square lattice," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 58, no. 1, p. 69, 1998.
[26] Z. Wang, A. Szolnoki, and M. Perc, "Different perceptions of social dilemmas: Evolutionary multigames in structured populations," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 90, no. 3, Sep. 2014, Art. no. 032813.
[27] K. Nagashima and J. Tanimoto, "A stochastic pairwise Fermi rule modified by utilizing the average in payoff differences of neighbors leads to increased network reciprocity in spatial prisoner's dilemma games," *Appl. Math. Comput.*, vol. 361, pp. 661–669, Nov. 2019.
[28] Z. Su, L. Liu, M. Li, X. Fan, and Y. Zhou, "ServiceTrust: Trust management in service provision networks," in *Proc. IEEE Int. Conf. Services Comput.*, Santa Clara, CA, USA, Jun. 2013, pp. 272–279.
[29] M. Li, Q. Guan, X. Jin, C. Guo, X. Tan, and Y. Gao, "Personalized pre-trust reputation management in social P2P network," in *Proc. Int. Conf. Comput., Netw. Commun. (ICNC)*, Kauai, HI, USA, Feb. 2016, pp. 1–5.
[30] Q. Xu, Z. Su, Q. Zheng, M. Luo, B. Dong, and K. Zhang, "Game theoretical secure caching scheme in multihoming edge computing-enabled heterogeneous networks," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4536–4546, Jun. 2019.
[31] X. Fan, L. Liu, M. Li, and Z. Su, "GroupTrust: Dependable trust management," *IEEE Trans. Parallel Distrib. Syst.*, vol. 28, no. 4, pp. 1076–1090, Apr. 2017.
[32] J. Yuan and X. Li, "A reliable and lightweight trust computing mechanism for IoT edge devices based on multi-source feedback information fusion," *IEEE Access*, vol. 6, pp. 23626–23638, 2018.
[33] X. Jin, M. Li, G. Cui, J. Liu, C. Guo, Y. Gao, B. Wang, and X. Tan, "RIMBED: Recommendation incentive mechanism based on evolutionary dynamics in P2P networks," in *Proc. 24th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Las Vegas, NV, USA, Aug. 2015, pp. 1–8.
[34] A. Mosenia and N. K. Jha, "A comprehensive study of security of Internet-of-Things," *IEEE Trans. Emerg. Topics Comput.*, vol. 5, no. 4, pp. 586–602, Oct. 2017.
[35] W. Yu, F. Liang, X. He, W. G. Hatcher, C. Lu, J. Lin, and X. Yang, "A survey on the edge computing for the Internet of Things," *IEEE Access*, vol. 6, pp. 6900–6919, 2018.
[36] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, "A survey on Internet of Things: Architecture, enabling technologies, security and privacy, and applications," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1125–1142, Oct. 2017.
[37] Y. Yang, L. Wu, G. Yin, L. Li, and H. Zhao, "A survey on security and privacy issues in Internet-of-Things?" *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1250–1258, Oct. 2017.
[38] L. Chen, S. Thombre, K. Järvinen, E. S. Lohan, A. Alén-Savikko, H. Leppäkoski, M. Z. H. Bhuiyan, S. Bu-Pasha, G. N. Ferrara, S. Honkala, J. Lindqvist, L. Ruotsalainen, P. Korpisaari, and H. Kuusniemi, "Robustness, security and privacy in location-based services for future IoT: A survey?" *IEEE Access*, vol. 5, pp. 8956–8977, 2017.
[39] A. H. Ngu, M. Gutierrez, V. Metsis, S. Nepal, and Q. Z. Sheng, "IoT middleware: A survey on issues and enabling technologies," *IEEE Internet Things J.*, vol. 4, no. 1, pp. 1–20, Feb. 2017.

[40] I. Farris, T. Taleb, Y. Khettab, and J. Song, "A survey on emerging SDN and NFV security mechanisms for IoT systems," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 812–837, 1st Quart., 2019.
[41] I. U. Din, M. Guizani, B. S. Kim, S. Hassan, and M. K. Khan, "Trust management techniques for the Internet of Things: A survey," *IEEE Access*, vol. 7, pp. 29763–29787, 2019.
[42] Q. Xu, Z. Su, Q. Zheng, M. Luo, and B. Dong, "Secure content delivery with edge nodes to save caching resources for mobile users in green cities," *IEEE Trans Ind. Informat.*, vol. 14, no. 6, pp. 2550–2559, Jun. 2018.
[43] M. Tiwary, D. Puthal, K. S. Sahoo, B. Sahoo, and L. T. Yang, "Response time optimization for cloudlets in mobile edge computing," *J. Parallel Distrib. Comput.*, vol. 119, pp. 81–91, Sep. 2018.
[44] D. Balfanz, D. Smetters, P. Stewart, and H. C. Wong, "Talking to strangers: Authentication in ad-hoc wireless networks," in *Proc. 9th Annu. Symp. Netw. Distrib. Syst. Secur.*, Feb. 2002, pp. 1–3.
[45] S. Bouzefrane, A. F. B. Mostefa, F. Houacine, and H. Cagnon, "Cloudlets authentication in NFC-based mobile computing," in *Proc. 2nd IEEE Int. Conf. Mobile Cloud Comput., Services, Eng.*, Oxford, U.K., Apr. 2014, pp. 72–267.
[46] S. Sicari, A. Rizzardi, L. A. Grieco, and A. Coen-Porisini, "Security, privacy and trust in Internet of Things: The road ahead," *Comput. Netw.*, vol. 76, pp. 146–164, Jan. 2015.
[47] I. Garcia-Magarino, S. Sendra, R. Lacuesta, and J. Lloret, "Security in vehicles with IoT by prioritization rules, vehicle certificates, and trust management," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 5927–5934, Aug. 2019.
[48] P. Gope, A. K. Das, N. Kumar, and Y. Cheng, "Lightweight and physically secure anonymous mutual authentication protocol for real-time data access in industrial wireless sensor networks," *IEEE Trans Ind. Informat.*, vol. 15, no. 9, pp. 4957–4968, Sep. 2019.
[49] H. Hui, C. Zhou, X. An, and F. Lin, "A new resource allocation mechanism for security of mobile edge computing system," *IEEE Access*, vol. 7, pp. 116886–116899, 2019.
[50] F. Lin, Y. Zhou, X. An, I. You, and K.-K.-R. Choo, "Fair resource allocation in an intrusion-detection system for edge computing: Ensuring the security of Internet of Things devices," *IEEE Consum. Electron. Mag.*, vol. 7, no. 6, pp. 45–50, Nov. 2018.
[51] Z. Yan, P. Zhang, and A. V. Vasilakos, "A survey on trust management for Internet of Things," *J. Netw. Comput. Appl.*, vol. 42, pp. 120–134, Jun. 2014.
[52] F. Bao, I.-R. Chen, and J. Guo, "Scalable, adaptive and survivable trust management for community of interest based Internet of Things systems," in *Proc. IEEE 11th Int. Symp. Auton. Decentralized Syst. (ISADS)*, Mexico City, Mexico, Mar. 2013, pp. 1–7.
[53] D. Gessner, A. Olivereau, A. S. Segura, and A. Serbanati, "Trustworthy infrastructure services for a secure and privacy-respecting Internet of Things," in *Proc. IEEE 11th Int. Conf. Trust, Secur. Privacy Comput. Commun.*, Liverpool, U.K., Jun. 2012, pp. 998–1003.
[54] S. Sicari, A. Coen-Porisini, and R. Riggio, "DARE: Evaluating data accuracy using node REputation," *Comput. Netw.*, vol. 57, no. 15, pp. 3098–3111, Oct. 2013.
[55] Z. Yan and C. Prehofer, "Autonomic trust management for a component-based software system," *IEEE Trans. Depend. Sec. Comput.*, vol. 8, no. 6, pp. 810–823, Nov./Dec. 2011.
[56] I.-R. Chen, J. Guo, and F. Bao, "Trust management for SOA-based IoT and its application to service composition," *IEEE Trans. Services Comput.*, vol. 9, no. 3, pp. 482–495, May/Jun. 2016.
[57] I.-R. Chen, J. Guo, D.-C. Wang, J. J. P. Tsai, H. Al-Hamadi, and I. You, "Trust-based service management for mobile cloud IoT systems," *IEEE Trans. Netw. Service Manage.*, vol. 16, no. 1, pp. 246–263, Mar. 2019.

BO WANG received the B.S. degree in computer science and technology from the Anshan University of Science and Technology, in 2005, and the M.S. degree in computer application and technology from the University of Science and Technology Liaoning, China, in 2008. He is currently pursuing the Ph.D. degree with the School of Software Technology, Dalian University of Technology (DLUT), China. His current research interests include edge computing, resource scheduling, trust management, and game theory.

MINGCHU LI received the B.S. degree in mathematics from Jiangxi Normal University, in 1983, the M.S. degree in applied science from the University of Science and Technology Beijing, in 1989, and the Ph.D. degree in mathematics from the University of Toronto, in 1997. He was an Associate Professor with the University of Science and Technology Beijing, from 1989 to 1994. He was involved in research and development on information security at Longview Solution Inc., Compuware Inc., from 1997 to 2002. Since 2002, he has been a Full Professor with the School of Software, Tianjin University. Since 2004, he has also been a Full Professor, a Ph.D. Supervisor, and the Vice Dean with the School of Software Technology, Dalian University of Technology. His main research interests include theoretical computer science and cryptography.

XING JIN received the bachelor's degree in computer science from Lanzhou University, China, in 2012. He is currently pursuing the Ph.D. degree with the School of Software Technology, Dalian University of Technology (DLUT), China. His research interests include trust management, cooperation theory, and evolutionary game theory.



CHENG GUO received the B.S. degree in computer science from the Xi'an University of Architecture and Technology, in 2002, and the M.S. and Ph.D. degrees in computer application and technology from the Dalian University of Technology, Dalian, China, in 2006 and 2009, respectively. From July 2010 to July 2012, he was a Postdoctoral Researcher with the Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan. Since 2013, he has been an Associate Professor with the School of Software Technology, Dalian University of Technology. His current research interests include information security and cryptology.