Li XY, Gui XL. A comprehensive and adaptive trust model for large-scale P2P networks. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 24(5): 868–882 Sept. 2009

A Comprehensive and Adaptive Trust Model for Large-Scale P2P Networks

Xiao-Yong Li (李小勇) and Xiao-Lin Gui (桂小林), *Senior Member, CCF*

Department of Computer Science and Technology, Xi'an Jiaotong University, Xi'an 710049, China

E-mail: lxyxjtu@163.com; xlgui@mail.xjtu.edu.cn

Revised May 13, 2009.

Abstract Based on human psychological cognitive behavior, a Comprehensive and Adaptive Trust (CAT) model for large-scale P2P networks is proposed. Firstly, an adaptive trusted decision-making method based on HEW (Historical Evidences Window) is proposed, which can not only reduce the risk and improve system efficiency, but also solve the trust forecasting problem when the direct evidences are insufficient. Then, direct trust computing method based on IOWA (Induced Ordered Weighted Averaging) operator and feedback trust converging mechanism based on DTT (Direct Trust Tree) are set up, which makes the model have a better scalability than previous studies. At the same time, two new parameters, confidence factor and feedback factor, are introduced to assign the weights to direct trust and feedback trust adaptively, which overcomes the shortage of traditional method, in which the weights are assigned by subjective ways. Simulation results show that, compared to the existing approaches, the proposed model has remarkable enhancements in the accuracy of trust decision-making and has a better dynamic adaptation capability in handling various dynamic behaviors of peers.

Keywords P2P networks, dynamic trust model, IOWA operator, adaptability

1 Introduction

P2P networks are decentralized applications where heterogeneous peers, which are autonomous and have intermittent presence in the network and a high level of anonymity, inter-operate for purposes such as file sharing, distributed computing and e-Commerce transactions without the need of a centralized server. The decentralized nature of P2P systems poses the need for enhanced trust between peers that will enable the reliable communication and exchange of services between them. Peers in P2P systems need to make trust decisions for choosing peers they will transact with or resources they have asked for among the offered ones. There is, thus, the need of at least a minimal trust system to ensure a satisfying level of robustness against various kinds of attacks that have been monitored in P2P systems[1−5]. Such a trust system should be decentralized so that each peer can make autonomous trust decisions based on other peers' trust degree. By "trust degree" we refer to a measure that indicates the trustworthiness of a peer in a particular context. This measure is estimated based on both direct experiences and other peers' feedback.

In literatures, a number of trust evaluating models for P2P networks have been proposed[6−19]. Some of them are very creative and elaborate, but most of these studies still have some limitations need to be addressed:

1) Regarded as a crucial phenomenon by social sciences, the dynamic nature of trust creates the biggest challenge in measuring trust relationship[1−2]. The dynamic nature of trust refers to the trust value of a peer on another peer changing over time due to newer interactions. The existing method of trust management focuses on defining the trust attenuation function with the assumption that there is only a simple and linear reduction, and the method is lack of adaptability. Once the value of attenuation function is identified, it will be difficult to adjust by system dynamically in a practical P2P environment. Even in some literatures, trust management only considers one trust value and the value does not change without considering the dynamic nature of trust and the variability of trust values with time.

2) Feedback (also known as recommendation) provides an efficient and effective way to build reputation-based trust relationship amongst peers in open and dynamic P2P environment. The system collects feedbacks from other peers and aggregates them to yield the global reputation value. One of the key technologies to the success of P2P trust system is the feedback aggregating mechanism. However, most previous studies

Regular Paper

This work is supported by the National Natural Science Foundation of China under Grant No. 60873071 and the National High-Tech Research and Development 863 Program of China under Grant No. 2008AA01Z410.

either paid little attention to the distribution of peer's feedback or behaviored in a broadcast manner based on polling algorithm to collect feedbacks, which leads to worse scalability and less sensitivity of the reputation system when assessing the risk of a peer in performing a task.

3) In many previous studies, the subjective method for assigning weights to trust decision factors cannot reflect the scientific nature of trust decision process, and may lead to misjudgment of trust decision-making. For example, in [7–9, 12–13], they define Overall Trust Degree (OTD) as: $T = W \times E + (1 - W) \times R$, where E is the value of direct trust, R is the value of indirect trust (feedback trust), W is the weight of E (correspondingly, $(1 - W)$ is the weight of R). Direct Trust Degree (DTD) E and Feedback Trust Degree (FTD) R can be computed through mathematical methods. But, what value of W is reasonable and accurate? Most of the previous studies are lack of scientific or reasonable ways.

Focusing on these problems, the CAT model is proposed for large-scale P2P networks. Firstly, an adaptive trusted decision-making method based on HEW is proposed, which can not only reduce the risk and improve system efficiency, but also solve trust measuring and forecasting problems when the direct evidences are insufficient. Then, direct trust computing mechanism based on IOWA operator and feedback trust converging mechanism based on DTT are set up, which makes our model exhibit a higher practicability and a better scalability than previous studies. At the same time, two new parameters, confidence factor and feedback factor, are introduced to adjust the weights of direct trust and feedback trust adaptively, in which the weights are assigned in a subjective manner. Simulation results show that, compared to the existing trust models, the new model has remarkable enhancements in the accuracy of trust decision and has a better dynamic adaptation capability in handling various dynamic behaviors of peers.

The remaining parts of this paper are organized as follows. Section 2 briefly gives an overview of the related work; Section 3 outlines the innovative designs of the CAT model, including HEW-based trust decision-making method, IOWA-based trust attenuation function and DTT-based feedbacks aggregating mechanism, etc. The simulation results are presented in Section 4, and Section 5 concludes the paper and suggests future directions for improvement.

2 Related Work

Many researchers are working on research projects involving trust management in P2P applications. At Georgia Tech., Xiong and Liu have developed the Peer-Trust model[6]. Their model is based on a weighted sum of five peer feedback factors: peer records, scope, credibility, transaction context, and community context. Peer-Trust is fully distributed, uses overlay for trust propagation, public-key infrastructure for securing remote scores, and prevents peers from some malicious abuses. Shi and Liang at Wayne State University have proposed the Trust-Ware system[7−8], a trusted middle-ware for P2P applications. Their approach consists of two models: the Multiple CUrrency Based Economic (M-CUBE) and the PErsonalized Trust (PET). M-CUBE provides a general and flexible substrate to support high-level P2P resource management services. PET derives peer trustworthiness from long-term reputation evaluation and short-term risk evaluation. This paper contributes to modeling the risk as the opinion of short-term trustworthiness and combining with traditional reputation evaluation to derive the trustworthiness in this field.

At University of Southern California, Hwang and Zhou have developed a robust and scalable P2P reputation system, Power-Trust[9], to leverage the power-law feedback characteristics. Using a distributed ranking mechanism, the Power-Trust system dynamically selects small number of power nodes that are most reputable. By using a look-ahead random walk strategy and leveraging the power nodes, it significantly improves in global reputation accuracy and aggregation speed. Power-Trust is adaptable to dynamics in peer joining and leaving and robust to disturbance by malicious peers. Hwang and Song also have proposed another trust model, Fuzzy-Trust[10], a fuzzy logic reputation system for P2P e-Commerce applications. Peers perform fuzzy inference on local parameters to generate local scores for the peers with whom they have transacted. These local scores are collected from qualified peers, which meet an aggregation threshold and aggregated into global reputation values. The Fuzzy-Trust system uses a DHT-based P2P overlay network for the global reputation aggregation.

In [11], the authors have proposed a reputation-oriented reinforcement learning algorithm for buying agents in electronic market environments, taking into account the fact that the quality of goods offered by different selling agents may not be the same and that a selling agent may alter the quality of its goods. In [12–13], the authors have proposed a reputation-based approach for P2P file sharing systems (called P2PRep). In P2PRep, a peer polls other peers by broadcasting a request about the opinion of the select peer. In [14], the authors have presented a similar approach, called XRep, which considers the reputations of both peers and resources. Both P2PRep and XRep do not give any metrics to quantify the credibilities of voters, and they can only find trustworthy peers within a given horizon.

In EigenTrust[15], each peer is assigned a unique global reputation value. However, it is not clear if the approach is feasible for large-scale P2P systems, in which some local reputation values are unreachable for the requesting peers. Richardson *et al.*'s approach to trust management for Semantic Web is similar to EigenTrust, but ratings are personalized for each user based on the personal experience[16]. Both approaches simply assume that peers are honest and therefore cannot defend some attacks like deceptions and rumors. Guha and colleagues at IBM Almaden Research Center have proposed an interesting idea about the propagation of distrust[17]. In addition to maintaining positive trust values for peers, the system also allows the proactive dissemination of some malicious peers' bad reputations. Sonja and Boudec design a distributed reputation system using a Bayesian approach, in which the second-hand reputation rating is accepted only when it is not compatible with the primary rating[18].

Wang and Chang at National University of Defense Technology of China have presented a time-frame-based trust model[19]. They incorporate time dimension using time-frame, which captures direct experiences and recommendation time-sensitivity, and they also introduce four trust parameters in computing trustworthiness of peers, namely, trust construction factor, trust destruction factor, supervision period factor and feedback credibility. Together, these parameters are adjusted in time using feedback control mechanism, thus, trust valuation can reflect the dynamics of the trust environment. Li and Gui at Xi'an Jiaotong University of China, based on human cognitive psychology, have proposed a new reputation model, Tree-Trust[20], in which the concept of direct trust tree (DTT) is presented. Based on DTT a practical aggregation algorithm for feedbacks is proposed. In their model, the feedbacks are searched by using DTT instead of in broadcast manner. Simulation results show that, compared to the existing models, the proposed model is more robust in trust dynamic adaptability.

3 Design of the CAT Model

3.1 Overall Framework of OTD Calculation

In a P2P network, a user, a process or a resource which interacts or can interact with other users, processes or resources is called entity. Let $\Omega = \{P_1, P_2, \ldots, P_N\}$ denote the entities in the system, then Ω is called Entity Domain. According to the role of an entity in the trust management system, we use the notion of SP (Service Provider) to represent the entities who provide services for others, notion SR (Service Requester) to stand for the entities who request services, and notion FR (Feedback Rater) to represent the entities who assign their feedbacks for others. A trust rating is an integrated opinion about the outcome of a transaction. The trust model monitors an entity's behaviors by collecting, aggregating and distributing such trust rating, so, in the CAT model, basic computing structure of trust model is defined as follows.

Definition 3.1. *In general, overall trust degree in the CAT model is defined by the following equation*:

$$T(P_i, P_j) = \begin{cases} T_D(P_i, P_j), & \text{if } h \geqslant H, \\ T_I(P_i, P_j), & \text{if } h = 0, \\ W_1 \times T_D(P_i, P_j) + & \\ W_2 \times T_I(P_i, P_j), & \text{if } 0 < h < H, \end{cases} \quad (1)$$

where $T(P_i, P_j) \in [0, 1]$ is called Overall Trust Degree (OTD), the value 0 *of $T(P_i, P_j)$ represents no trust, while the value 1 represents total trust. $T_D(P_i, P_j)$ is called Direct Trust Degree (DTD) and $T_I(P_i, P_j)$ is called Feedbacks (Indirect) Trust Degree (FDT). H is called Historical Evidence Window (HEW) and h is the total amount of current direct evidences of entity P_i for P_j. W_1 is the weight of $T_D(P_i, P_j)$ (correspondingly, W_2 is the weight of $T_I(P_i, P_j)$). In the CAT model, the overall calculating framework and processing flow for OTD based on HEW is described in Fig.*1.

In Fig.1, we can find, in the CAT model, only when $h < H$, the trust model need to compute feedback trust degree $T_I(P_i, P_j)$. If $h \geqslant H$, it indicates that current direct evidences of entity P_i for P_j are enough to judge P_j's trust degree, the CAT model does not need to compute P_j's feedback trust degree $T_I(P_i, P_j)$. So we call the CAT model trust evaluation model based on historical evidence window H, which is more in line with the human psychological cognitive process and behavior habits. That is to say, most people firstly believe their own direct experience and judgments. When people's direct experience for others are enough to determine the others trust degrees, they do not need to ask the recommendation information from third-party entities. In Fig.1, the IOWA-based DTD calculating process will be discussed in Subsection 3.2, the DTT-based FTD calculating process will be discussed in Subsection 3.3, and the assigning algorithm for the weights W_1 and W_2 will be discussed in Subsection 3.4. Now, we first give a formal definition for the trust decision function.

Definition 3.2. *Suppose an SP P_i has m levels of services, being defined as the set of service policy $S = \{s_1, s_2, \ldots, s_m\}$. Then, between SP P_i and SR P_j, the trust decision function ρ is a mapping from OTD*

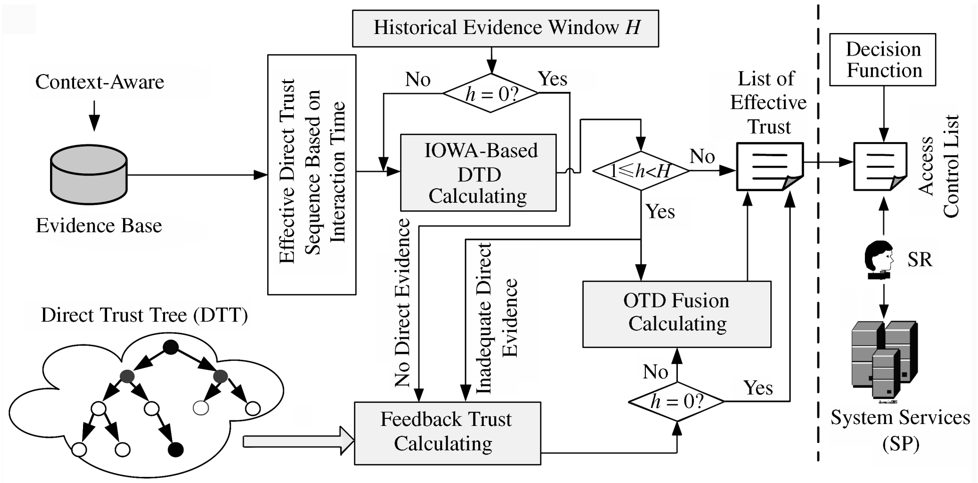


Fig.1. OTD calculating framework and processing flow based on HEW.

$T(P_i, P_j)$ at stamp time L to P_i's service policy S:

$$\rho(T(P_i,P_j)) = \begin{cases} s_m, & c_k \leqslant T(P_i,P_j) \leqslant 1, \\ s_{m-1}, & c_{k-1} \leqslant T(P_i,P_j) < c_k, \\ \vdots & \\ s_2, & c_1 \leqslant T(P_i,P_j) < c_2, \\ s_1, & 0 \leqslant T(P_i,P_j) < c_1, \end{cases} \tag{2}$$

where $c_1, c_2, \ldots, c_k \in [0,1]^R$. In the example of (2), an SP P_0 can define its trust decision function as:

$$\rho(T(P_0,P_1)) = \begin{cases} s_3, & 0.5 \leqslant T(P_0,P_1) \leqslant 1, \\ s_2, & 0.2 \leqslant T(P_0,P_1) < 0.5, \\ s_1, & 0 \leqslant T(P_0,P_1) < 0.2, \end{cases}$$

where $\{s_1, s_2, s_3\} = \{$no service, download, download & upload$\}$. If $T(P_0,P_1)$ equals 0.19, then $\rho(T(P_0,P_1)) = \rho(0.19) = s_1$. It implies that P_1 has no access privilege for the resources of P_0. If $T(P_0,P_1) = 0.4$, then $\rho(T(P_0,P_1)) = \rho(0.4) = s_2$, and P_1 has download privilege for the resources of peer P_0. If $T(P_0,P_1) = 0.79$, then $\rho(T(P_0,P_1)) = \rho(0.79) = s_3$, and P_1 has both download and upload privileges.

3.2 IOWA-Based DTD Forecast

1) Basic Problem Description

The DTD is computed by the knowledge of the entity's past interactive experiences, without requesting information from a trusted third party (TTP). The DTD is generated every time-stamp when an interaction takes place. According to human social psychological cognition and behavior habits, in the CAT model, we define that direct trust degree $T_D(P_i, P_j)$ must obey two fundamental properties.

Property 3.1 (Dynamic Nature). *The $T_D(P_i, P_j)$ of a peer P_i on P_j for service S changes over time due to newer interactions.*

$$\begin{cases} T_D(P_i,P_j)_{\text{new}} > T_D(P_i,P_j)_{\text{old}}, & \textit{if } \varepsilon > 0, \\ T_D(P_i,P_j)_{\text{new}} < T_D(P_i,P_j)_{\text{old}}, & \textit{if } \varepsilon < 0, \end{cases}$$

where $\varepsilon > 0$ indicates that the satisfaction level for the latest interaction is positive, that is to say, P_j's actions are in accord with P_i's expectation; $\varepsilon < 0$ indicates that the satisfaction level for the latest interaction is negative, that is to say, P_j's actions are out of accord with P_i's expectation.

Property 3.2 (Time-Based Aging Nature). *The $T_D(P_i, P_j)$ of a peer P_i on P_j for service S decreases with the passage of time.*

$$T_D(P_i,P_j)_{L-\Delta L} > T_D(P_i,P_j)_L$$

where L is the time stamp of current time and $L-\Delta L$ is the latest time stamp. According to people's experience, old knowledge has less infection and new knowledge has more contribution to trust decision. So, the time-based aging nature is another important property of DTD.

In the CAT model, the main goal is that the computing model of DTD should obey Property 3.1 and Property 3.2. Suppose entity P_i has rated the satisfaction degree of the latest h interactions with P_j as a time series of probabilistic ratings:

$$E^{(t_n)}_{P_i,P_j} = \{e^{(t_{n-h})}_{P_i,P_j}, e^{(t_{n-h-1})}_{P_i,P_j}, \ldots, e^{(t_L)}_{P_i,P_j}, \ldots, e^{(t_n)}_{P_i,P_j}\} \tag{3}$$

where t_n is the last time-stamp from current time, $0 \leqslant e^{(t_L)}_{P_i,P_j} \leqslant 1, t_L \in [t_{n-h}, t_n]$ and h is bounded by the allowed max history records H, H is History Evidence Window (HEW). Then, the basic problem of DTD computing or forecasting can be described as: we already know $E^{(t_n)}_{P_i,P_j}$, a series of direct probabilistic ratings of the past h time-stamps, and we need to predict the DTD of the next time point t_{n+1}.

Definition 3.3. *DTD's forecasting value or the rating of satisfaction degree of peer P_i for P_j at time-stamp t_{n+1} can be computed as the following fusion computing function*:

$$T_D(P_i, P_j) = F_{n+1}(E^{(t_n)}_{P_i,P_j}). \tag{4}$$

From Definition 3.3, it can be found that the problem of DTD calculation is a forecasting process of data fusion based on time series $\{e^{(t_{n-h})}_{P_i,P_j}, \ldots, e^{(t_L)}_{P_i,P_j}, \ldots, e^{(t_n)}_{P_i,P_j}\}$. Time series forecasts are dependent on the availability of historical data. Forecasts are estimated by extrapolating the past data into the future. Time series data typically have four patterns[21−22]: 1) trend variations, 2) cyclical variations, 3) seasonal, and 4) random variations. Time series forecasting is one of the most widely used techniques. From the literatures, it indicates that the top three quantitative forecasting techniques used are simple moving average (MA), weighted moving average (WMA), and exponential smoothing (ES), in which WMA is the most used data fusion method for the forecasting models, and its basic definition is described as follows.

Definition 3.4. *Weighted moving average* (*WMA*) *forecasting function is defined*:

$$F_{n+1}(A_{n+1}) = \sum_{i=t_{n-h}}^{t_n} \omega_i A_i \tag{5}$$

where $F_{n+1}(A_{n+1})$ is the forecasting function for period t_{n+1} , h is the number of periods used to calculate moving average, A_i is the actual data at time-stamp t_i, and ω_i is the weight assigned to A_i (*with* $\sum \omega_i = 1$).

According to Definition 3.3 and Definition 3.4, we can get the WMA-based DTD computing equation:

$$\begin{aligned} T_D(P_i, P_j) &= F_{n+1}(E^{(t_n)}_{P_i,P_j}) \\ &= \begin{cases} \displaystyle\sum_{L=t_{n-h}}^{t_n} (e^{(L)}_{P_i,P_j} \times \gamma(L)), & h \neq 0, \\ 0, & h = 0, \end{cases} \end{aligned} \tag{6}$$

where $\gamma(L) \in [0, 1]$ and $\sum \gamma(L) = 1$, $\gamma(L)$ determines the weights given to the past observations, and we called it as attenuation function. According to Property 3.1, the trust dynamic nature refers to the trust value of an entity on another entity changes over time due to newer interactions. But in the existing methods, managing trust is focused on defining the trust attenuation function with the assumption that there is only a simple and linear reduction and these methods are lack of adaptability. Once the value of attenuation function is identified, it will be difficult to dynamically adjust by system in a practical distributed application environment. Obeying to people's experience of cognitive psychology, old knowledge has less infection and new knowledge has more contribution to trust decision. The simple averaging and exponential averaging have similar results if the entities behave in a consistent manner. However, the estimate of the current rating in the simple averaging will tend to lag behind the true value of the current rating for a malicious entity if it explores the reputation mechanisms. For example, the simple averaging is not sensitive to the attacks of (malicious) entities, where entities may accumulate a high trust value and then attack the P2P network systems. In the following, we first give the basic concept of IOWA operation, then, we introduce a new calculating method of attenuation function $\gamma(L)$ based on IOWA operation, which has a higher flexibility and a better dynamic adaptability than the existing models.

2) IOWA Operator

The OWA operators were introduced in 1988 by Yager[23]. These operators aim at finding the most suitable aggregation for a number of criteria or variables.

Definition 3.5. *An OWA operator of dimension n is a function $\phi : \mathbb{R}^N \rightarrow \mathbb{R}$, that has associated a set of weights or weighting vector $\boldsymbol{M}^* = [\omega_1, \omega_2, \ldots, \omega_n]^{\mathrm{T}}$ with it, so that $\omega_i \in [0, 1]$ and $\sum_{i=1}^n \omega_i = 1$, and it is defined to aggregate a list of value $\{p_1, p_2, \ldots, p_n\}$ according to the following expression*:

$$\phi(p_1, p_2, \ldots, p_n) = \sum_{i=1}^{n} \omega_i p_{\sigma(i)} \tag{7}$$

being $\sigma : 1, 2, \ldots, n \rightarrow 1, 2, \ldots, n$ a permutation such that $p_{\sigma(i)} \geqslant p_{\sigma(i+1)}$, $\forall i = 1, 2, \ldots, n-1$, i.e., $p_{\sigma(i)}$ is the i-th highest value in the set $\{p_1, p_2, \ldots, p_n\}$.

An issue in the definition of the OWA operator is how to obtain the associated weighting vector. In [23], Yager proposed two ways to obtain it. The first approach is to use some kind of learning mechanism using some sample data; and the second approach is to try to give some semantics or meaning to the weights.

In [24] Mitchell and Estrakh described a modified OWA operator in which the input arguments are not rearranged according to their values but rather using a function of the arguments. Inspired by this work, Yager introduced in [25] a more general type of OWA operator, which is named Induced Ordered Weighted Averaging (IOWA) operator.

Definition 3.6. *An IOWA operator of dimension n is a function $\Phi_M : \mathbb{R} \times \mathbb{R}^N \rightarrow \mathbb{R}$, to which a set of weights or weighting vector $\boldsymbol{M}^* = [\omega_1, \omega_2, \ldots, \omega_n]^{\mathrm{T}}$ is associated to it, such that $\omega_i \in [0, 1]$ and $\sum_{i=1}^n \omega_i = 1$, it* *is defined to aggregate the set of the second arguments of a list of n 2-tuples* $\{\langle u_1, p_1\rangle, \langle u_2, p_2\rangle, \ldots, \langle u_n, p_n\rangle\}$ *according to the following expression*:

$$\varPhi_M(\langle u_1, p_1\rangle, \langle u_2, p_2\rangle, \ldots, \langle u_n, p_n\rangle) = \sum_{i=1}^{n} \omega_i p_{\sigma(i)} \quad (8)$$

being $\sigma : 1, 2, \ldots, n \to 1, 2, \ldots, n$ *a permutation such that* $p_{\sigma(i)} \geqslant p_{\sigma(i+1)}$, $\forall i = 1, 2, \ldots, n-1$, *i.e.*, $\langle u_{\sigma(i)}, p_{\sigma(i)}\rangle$ *is the* 2*-tuple with* $u_{\sigma(i)}$ *the i-th highest value in the set* $\{u_1, u_2, \ldots, u_n\}$.

In Definition 3.6, the reordering of the set of values to aggregate $\{p_1, p_2, \ldots, p_n\}$, is induced by the reordering of the set of values $\{u_1, u_2, \ldots, u_n\}$ associated to them, which is based upon their magnitude. Due to this use of the set of values, Yager called them the values of an order inducing variable, and $\{p_1, p_2, \ldots, p_n\}$ are the values of the argument variable. As to [23–26], the main difference between the OWA operator and the IOWA operator resides in the reordering step of the argument variable. In the case of OWA operator this reordering is based upon the magnitude of the values to be aggregated, while in the case of IOWA operator an order inducing variable is used as the criterion to induce that reordering. Obviously, an immediate consequence of Definition 3.6 is that if the order inducing variable is the argument variable then the IOWA operator is reduced to the OWA operator.

3) Algorithm of IOWA-based DTD Forecasting

IOWA can be applied to DTD forecast based on inference. In (3), we can reform the series as the following tuples:

$$\begin{aligned} E_{P_i,P_j}^{(t_n)} = &\{\langle u_1, e_{P_i,P_j}^{(t_{n-h})}\rangle, \langle u_2, e_{P_i,P_j}^{(t_{n-h-1})}\rangle, \ldots, \\ &\langle u_L, e_{P_i,P_j}^{(t_L)}\rangle, \ldots, \langle u_h, e_{P_i,P_j}^{(t_n)}\rangle\}. \end{aligned} \quad (9)$$

Then, the IOWA-based DTD forecasting expression can be defined by:

$$T_D(P_i, P_j) = \varPhi_{\boldsymbol{M}}(E_{P_i,P_j}^{(t_n)}) = \sum_{j=1}^{h} \gamma(j) \times \boldsymbol{b}_j \quad (10)$$

where $\boldsymbol{M}^* = \gamma(i) = [\gamma(1), \gamma(2), \ldots, \gamma(h)]^{\mathrm{T}}$ is a weight vector, such that $\gamma(i) \in [0, 1]$ and $\sum_{i=1}^{h} \gamma(i) = 1$, N-dimensional ordered argument vector $\boldsymbol{b}_j = \{e_{P_i,P_j}^{(t_{n-h})}, e_{P_i,P_j}^{(t_{n-h-1})}, \ldots, e_{P_i,P_j}^{(t_L)}, \ldots, e_{P_i,P_j}^{(t_n)}\}$.

If adding a time-stamp u_L $(1 \leqslant L \leqslant h)$ for each value of b_j, we can get the OWA pair $\langle u_L, e_{P_i,P_j}^{(t_L)}\rangle$. Because u_L in $\langle u_L, e_{P_i,P_j}^{(t_L)}\rangle$ is referred to as the order inducing variable and $e_{P_i,P_j}^{(t_L)}$ as the argument variable, we can define $(u_1, u_2, \ldots, u_h)$ as a time series $(t_{n-h}, t_{n-h-1}, \ldots, t_L, \ldots, t_n)$. Then:

$$\begin{aligned} T_D(P_i, P_j) &= \sum_{j=1}^{h} \gamma(j) \times \boldsymbol{b}_j \\ &= \begin{cases} \displaystyle\sum_{L=t_{n-h}}^{t_n} (e_{P_i,P_j}^{(L)} \times \gamma(L)), & h \neq 0, \\ 0, & h = 0. \end{cases} \end{aligned} \quad (11)$$

According to [23–24, 26], weights $\gamma(L)$ can be computed through Algorithm 1.

Algorithm 1. Computing Weight Using the IOWA Operator

```
Input $(\lambda, E_{P_i,P_j}^{(t_n)})$; /* for different $\lambda$ and $h$, we can get different IOWA weight, $\lambda$ is the situation parameter[25].*/
$m = h$;
if $\lambda < 0.5$ then
    $\lambda = 1 - \lambda$;
end if
if $\lambda \geqslant 0.5$ then
    $\gamma(1)[(m-1)\lambda + 1 - m\gamma(1)]^m = [(m-1)\lambda]^{m-1}[((m-1)\lambda - m)\gamma(1) + 1]$   (12)
    /*calculate $\gamma(1)$*/
    $\gamma(m) = \frac{((m-1)\lambda - m)\gamma(1) + 1}{(m-1)\lambda + 1 - m\gamma(1)}$   (13)
    /*calculate $\gamma(m)$*/
    for $L = 2$ to $m - 1$ do
        $\gamma(L) = \sqrt[m-1]{\gamma(1)^{(m-j)}\gamma(m)^{(j-1)}}$   (14)
    /*calculate $\gamma(L)$*/
end if
Output $[\gamma(1), \gamma(2), \ldots, \gamma(h)]$;
```

In Algorithm 1, the parameter λ can be treated as a magnifying lens for the optimistic decision makers to determine the most important attribute based on the sparest information (i.e., optimistic and λ =0 or 1) situation[25]. On the other hand, when $\lambda = 0.5$ (moderate situation), this method can get the attribute weights (equal weights of attributes) for the pessimistic decision makers based on maximal information (maximal entropy). Hence, the optimal value of $\gamma(i)$ should satisfy (12). When $\gamma(i)$ is computed, we can determine $\gamma(m)$ by (13), and then the other weights are obtained from (14). Through the above analysis, we find that IOWA algorithm can deal with the dynamical weighting problem more rationally and flexibly than subjective weighting assignment. So, we can use Algorithm 1 to computing $\boldsymbol{M}^* = \gamma(i) = [\gamma(1), \gamma(2), \ldots, \gamma(h)]^{\mathrm{T}}$ for our trust evaluation model.

4) Example

The IOWA-based DTD forecasting process can be understood through the following illustrative example. Given a time series:

$$\begin{aligned} E_{P_i,P_j}^{(t_n)} &= \{\langle 1, e_{P_i,P_j}^{(t_{n-h})}\rangle, \cdots, \langle h, e_{P_i,P_j}^{(t_n)}\rangle\} \\ &= \{\langle 1, 0.85\rangle, \langle 2, 0.84\rangle, \langle 3, 0.6\rangle, \langle 4, 0.75\rangle\} \end{aligned}$$

then, in Algorithm 1, $m = h = 4$, we can get $\boldsymbol{M}^* = \gamma(i) = [\gamma(1), \gamma(2), \gamma(3), \gamma(4)]^{\mathrm{T}}$, being listed in Table 1.

Table 1. Values of $\boldsymbol{M}^*$ for Different Situation Parameter λ

λ	$\gamma(4)$	$\gamma(3)$	$\gamma(2)$	$\gamma(1)$
0.0	0.000 000	0.000 000	0.000 000	1.000 000
0.1	0.010 365	0.043 457	0.182 129	0.764 099
0.2	0.045 018	0.106 445	0.251 953	0.596 466
0.3	0.054 918	0.113 770	0.237 305	0.493 805
0.4	0.073 547	0.130 859	0.233 398	0.416 657
0.5	0.250 000	0.250 000	0.250 000	0.250 000
0.6	0.416 657	0.233 398	0.130 859	0.073 547
0.7	0.493 805	0.237305	0.113 770	0.054 918
0.8	0.596 466	0.251 953	0.106 445	0.045 018
0.9	0.764 099	0.182 129	0.043 457	0.010 365
1.0	1.000 000	0.000 000	0.000 000	0.000 000

From Table 1, we find that the values of $\boldsymbol{M}^*$ are of symmetrical distribution in both sides of $\lambda = 0.5$. According to people's experience of cognitive psychology, old knowledge has less infection and new knowledge has more contribution to trust decision. That is to say, trust value has the attribute of dynamic attenuation over time decay, so, attenuation function $\gamma(L)$ is a decreasing function, which is the reason that we set $\lambda = 1 - \lambda$ while $\lambda < 0.5$ in Algorithm 1. So, in our example, we set $\lambda \in [0.5, 1]$, if the value of λ approaches to 0.5, the trust is averagely influenced by the last experience, if the value of λ approaches to 1, the trust value is heavily influenced by the latest experience. If we set $\lambda = 0.8$, then according to (11), the computing expression of DTD is

$$\begin{aligned} T_D(P_i, P_j) = &\sum_{L=t_{n-h}}^{t_n} (e_{P_i,P_j}^{(L)} \times \gamma(L)) \\ = &\,0.5964 \times 0.75 + 0.2519 \times 0.6 + \\ &\,0.1064 \times 0.84 + 0.054 \times 0.8 \\ = &\,0.7337. \end{aligned}$$

3.3 DTT-Based FTD Calculation

Feedbacks Trust Degree (FTD) also can be called transitive trust, which means that trust is derived from an existing trust relationship between entities. A peer aggregates feedbacks about another peer and combines the direct trust degree (if any) with testimonies received from any FR. Aggregating feedbacks can be used for deciding whether the other peer is trustworthy. Transitivity is the fundamental attributes of reputation-based trust relationship. But, according to people's habit of psychology cognition, people only trust the recommendation information from acquaintance. In general, people do not believe the recommendation information from strangers.

For example, in Fig.2, if P_0 knows (or trusts) P_1, P_1 knows P_2, then, through P_1's recommendation, P_0 will trust P_2. If P_0 does not know P_3, but P_3 knows P_2, then, P_0 will not trust P_3's recommendation, that is, P_0 will not trust P_2. Moreover, the trust degree between P_0 and P_2 is not equal to the trust degree between P_1 and P_2, it will be attenuated. That is to say, trust degree of the first hand is higher than that of the second hand. So, we consider that feedbacks aggregation algorithm should take into account the transitive attenuation of trust relationship. Feedback trust degree $T_I(P_i, P_j)$ has two important properties:

Property 3.3 (Partial Transitive Nature). *If P_i trusts P_j and P_j trusts P_k for a particular context, then this chain relationship creates a partial reputation $T_I(P_i, P_k)$ for P_i on P_k.*

$$T_I(P_i, P_k) \leqslant T_I(P_i, P_j), T_I(P_i, P_k) \leqslant T_I(P_j, P_k).$$

Property 3.4 (Distance Attenuation Nature). *If P_i collects feedbacks about P_j from other nodes in the network, the feedbacks collected from closer peers should be counted with more weight compared to the values collected from distant peers. Distance attenuation nature ensures this feature.*

In our CAT model, the main goal of feedback trust degree is that the computing model should obey Property 3.3 and Property 3.4. P2P system is client-oriented, and the peers involved are autonomous. Such a system can grow or shrink dynamically with self-organizing capabilities. In a fully distributed P2P system involving numerous peers, to improve search efficiency and reduce unnecessary traffic in P2P reputation system, the CAT model adopts a DTT-based searching algorithm[20,27]. The algorithm can extend the search region but reduce the search traffic, and also balance the network load, so that it can acquire better scalability than other models for large-scale P2P applications.

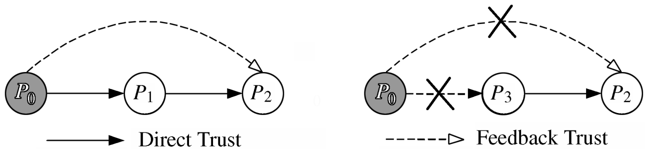


Fig.2. Feedback trust degree.

In DTT, each peer has a set of acquaintances (trustworthy peers), a subset of its neighbors. A peer's Trusted Neighbor Nodes (TNN) are those acquaintances who have direct interactions with it. A peer can maintain a data table for each acquaintance. This data table mainly includes some trust information, such as the acquaintance reliability to provide high-quality services and credibility to provide trustworthy feedbacks to other peers. More importantly, peers can adaptively choose their TNN based on the average of local ratings (DTD), which they do so often from among their current TNN. In order to construct DTT on top of the purely unstructured P2P overlay networks, CAT applies a simple data table (called neighbor table) in every peer's local database. Each peer maintains a neighbor table. If another peer has direct interactions with it, then direct trust degree of the peer is recorded in this table and the peer becomes a TNN. Table 2 is a example of P_0's neighbor table. P_0 has four neighbors, but only P_1 and P_2 are its TNN. According to a peer's neighbor table, system can get all its TNN and corresponding TNN'S direct trust degree, high direct trust degree implies that this neighbor can provide high-quality services or provide trustworthy feedbacks to other peers.

Table 2. P_0's Neighbor Table

Peer	Neighbors	TNN?	DTD	Other Items
P_0	P_i	No	0.0	$\cdots$
P_0	P_1	Yes	0.5	$\cdots$
P_0	P_2	Yes	0.7	$\cdots$
P_0	P_k	No	0.0	$\cdots$

Suppose $W = \{W_1, W_2, \ldots, W_k, \ldots, W_M\}$, $(k \in [1, M))$ as a group of FR towards entity P_j and the testimony $T_D(W_k, P_j)$ is W_k's direct trust degree for entity P_j, then the aggregating algorithm of feedback trust degree from the set of FR is defined as follows:

Definition 3.7. *The feedback trust degree of entity P_i for P_j can be computed as the following expressions.*

$$T_I(P_i, P_j) = \begin{cases} \dfrac{\sum_{k=1}^{M}(\varpi(W_k) \times T_D(W_k, P_j))}{\sum_{k=1}^{M}\varpi(W_k)}, & M \neq 0, \\ 0, & M = 0, \end{cases} \tag{15}$$

where $\varpi(W_k)$ is the weight for the credibility of W_k, the value of $\varpi(W_k)$ reflects the attenuation of transitive trust.

If W_k's LEVEL value is big, it implies that W_k is far from the root node of DTT and the feedback from W_k has less reliability, so W_k's feedback should be given a little weight. (LEVEL equals the hops from an FR to the root on DTT, for detailed construction method of DTT–Direct Trust Tree, please check our previous work[20,27]. From this fact we can compute $\varpi(W_k)$ as:

$$\varpi(W_k) = \begin{cases} 1, & LEVEL = 0, \\ \prod_{m=1}^{LEVEL} T_D(P_m, P_k), & LEVEL > 0. \end{cases} \tag{16}$$

As an example, we consider the instance in Fig.3. When an SR P_{14} requests a certain service from P_0, P_0 needs to collect feedbacks from DTT and aggregates these feedbacks to form a global reputation value for P_{14}. In Fig.3, P_0 looks into its neighbor table and finds two TNN P_1 and P_2, then only sends query messages to P_1 and P_2 instead of to all its neighbors P_i, P_1, P_2 and P_k. Thus, a 2-LEVEL DTT is formed, $\varpi(P_1) = 0.5$ and $\varpi(P_2) = 0.7$. Repeat this process, and if the largest searching depth configured by system is LEVEL 4, the searching process will be stopped. Thus, a LEVEL 4 DTT is set up. In the DTT, if some peers have a rating record about P_{14}, they will send this record to P_0 directly in a unicast manner. In Fig.3, if LEVEL = 0, and $\varpi(P_0) = 1$, it implies self-trust of the root node. If LEVEL = 1, then $\varpi(P_1) = 0.5$ and $\varpi(P_2) = 0.7$, it implies the first level weight of the neighbor nodes. Whereas LEVEL = 2, $\varpi(P_3) = 0.5 \times 0.6 = 0.30$. As for LEVEL = 3, it implies the third level neighbor nodes, their trust weight is $\varpi(P_9) = 0.5 \times 0.6 \times 0.8 = 0.24$. In Fig.3, suppose the set of FR searched by DTT is $W = \{P_8, P_9, P_6\}$. Feedbacks trust information from $W = \{P_8, P_9, P_6\}$ for peer P_{14} are $T_D(P_8, P_{14}) = 0.5$, $T_D(P_6, P_{14}) = 0.6$ and $T_D(P_9, P_{14}) = 0.6$, then:

$$\begin{aligned} T_I(P_0, P_{14}) &= \frac{\sum_{k=1}^{3}(\varpi(W_k) \times T(W_k, P_{14}))}{\sum_{k=1}^{3}\varpi(W_k)} \\ &= \frac{0.5 \times 0.21 + 0.5 \times 0.24 + 0.6 \times 0.35}{0.21 + 0.24 + 0.35} \\ &= 0.54375. \end{aligned}$$

3.4 Weights Allocation Function

According to (1), if $0 < h < H$, it indicates that current direct evidences of entity P_i for P_j are not enough

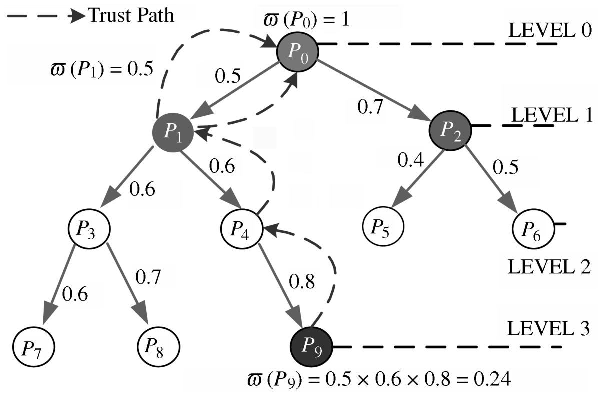


Fig.3. Trust path and feedback aggregation.

to judge P_j's trust degree, and CAT need to compute P_j's FDT $T_I(P_i,P_j)$ and DTD $T_D(P_i,P_j)$. Then entity P_j's OTD by entity P_i is calculated as follows:

$$T(P_i,P_j) = W_1 \times T_D(P_i,P_j) + W_2 \times T_I(P_i,P_j) \quad (17)$$

where W_1 is the weight of direct trust degree correspondingly, W_2 is the weight of indirect trust degree. Now, the key question is what value of W_1 and W_2 are reasonable and accurate? As mentioned above, in the vast majority of the previous studies, W_1 and W_2 are assigned through three subjective ways: experts opinion method, random allocation method and average weight method. However, all of the three methods have two key deficiencies: these methods do not reflect trust decision scientific nature and rationality, and may lead to misjudgment of trust decision; these methods are lack of adaptability, once the values of W_1 and W_2 are identified, it will be difficult to adjust by system dynamically in a practical distributed application environment. In this subsection, we will introduce two new adaptive weights assigning functions, confidence factor and feedback factor, which can solve the difficulties of the weights assignment.

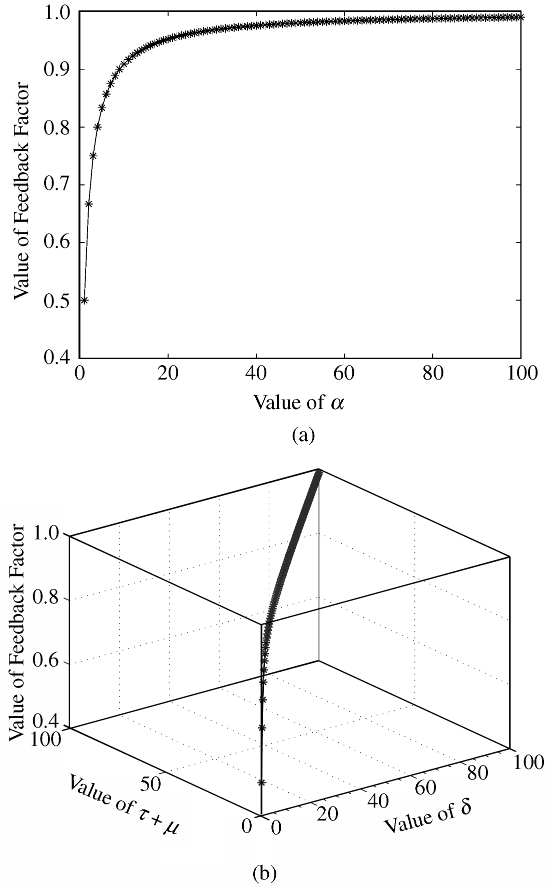
Intuitively, according to people's psychology cognition, the value of $T_I(P_i,P_j)$ calculated by (15) should have a higher weight if the number of entities common to both of the interacting entities is higher. Likewise entities with more interactions with a particular entity should have a higher say in recommendation. Now we use a new function: feedback factor, to reflect these objective phenomenons. The feedback factor should be a maximum if the number of common entities and the number of individual interactions of these entities are greater than a threshold value; we defined the threshold value as a positive constant K.

Definition 3.8. *The feedback factor $\varUpsilon_{P_i,P_j}$ is defined as*:

$$\varUpsilon_{P_i,P_j} = \begin{cases} \dfrac{1}{2}(\varPsi(\tau+\mu)+\varPsi(\delta)), & \tau+\mu+\delta < K, \\ 1, & \tau+\mu+\delta \geqslant K, \end{cases} \quad (18)$$

where $\varPsi(x) = 1 - \frac{1}{x+\alpha}$, δ is the total number of interaction entities with P_j, τ is the total number of P_i's direct neighbors, and μ is the total number of P_i's indirect neighbors.

The function $\varPsi(x)$ has the desirable property that with increasing x (x could be any positive integer) the function quickly approaches 1 and can be used to calculate this mechanism. (Notice that instead of the above function $\varPsi(x)$ we could have used any other function that has the property of quickly approaching 1 with increase in the argument. Our choice of the above function is there for brevity and ease of calculation.) α is an positive constant and can be tuned by the system accordingly.



(a)

(b)

Fig.4. Value of $\varUpsilon_{P_i,P_j}$ with different context parameters.

Fig.4(a) shows the change in the feedback factor $\varUpsilon_{P_i,P_j}$ value with different values of the adjustable constant α (with $\delta = \tau+\mu = 100$). α's value can be made higher if only a few numbers of entities are deemed necessary to increase the $\varUpsilon_{P_i,P_j}$ in the $T_i(P_i,P_j)$ value and vice versa. Fig.4(b) shows the values for $\varUpsilon_{P_i,P_j}$ with different values of the parameters δ and $\tau+\mu$ with $\alpha = 0.2$. The curve reflects that with an increasing number of common entities and number of interactions with these entities, the $\varUpsilon_{P_i,P_j}$ value approaches 1 rapidly. From Fig.4 and (18), we can see that feedback trust degree calculated should have higher weight if the number of entities is bigger. Likewise more interactions with other entities should have a higher weight in recommendation. For example, suppose $\delta = 5$, $\tau+\mu = 15$ and $\alpha = 0.2$, then $\varUpsilon_{P_i,P_j}$ is 0.87.

Definition 3.9. *The confidence factor $\varPhi_{P_i,P_j}$ is defined as*:

$$\varPhi_{P_i,P_j} = 1 - \frac{\sum_H \xi(P_i,P_j)}{H+\beta} \quad (19)$$

where H is HEW in Definition 1, *and $\sum_H \xi(P_i,P_j)$ is* *the total failing transaction number in HEW H, and β is an adjustable positive constant in the system and can be tuned accordingly.*

In (19), H is employed for the risk calculation. Only the behaviors of the entities inside the window are need to consider. With the window shifting forward, the risk value reflects the fresh statistics of the entities' recent behaviors. The window size and adjustable positive constant play an important role in the risk calculation. The smaller the window size and the value of β are, the more the shorter-term assessment is favorite by the trust calculation. To reduce the risk from the cooperation, users can focus more on the confidence factor by assigning it a lower value to β. Yet this will decrease the availability of the resources, because the less risk for the cooperation is requested, the less entities are qualified to be cooperated. The system can make a tradeoff between the risk and the resource availability by adjust the adjustable positive constant β.

For example, suppose $H = 10$, $\sum_H \xi(P_i, P_j) = 2$ and $\beta = 2$, then the value of $\varPhi_{P_i,P_j}$ is 0.833.

When we get the value of $\varUpsilon_{P_i,P_j}$ and $\varPhi_{P_i,P_j}$ through computing (18) and (19), we can use the following equation to calculate the weight of direct trust degree W_1 and the weight of indirect trust degree W_2.

$$W_1 = \frac{\varPhi_{P_i,P_j}}{\varPhi_{P_i,P_j} + \varUpsilon_{P_i,P_j}}, W_2 = \frac{\varUpsilon_{P_i,P_j}}{\varPhi_{P_i,P_j} + \varUpsilon_{P_i,P_j}}. \quad (20)$$

From the calculation processes of W_1 and W_2, $\varPhi_{P_i,P_j}$ and $\varUpsilon_{P_i,P_j}$, we can find that the 4 parameters are completely determined by the system automatically according to the dynamic changes of the environmental context. So we consider that the weights assigning process is an adaptive manner, which overcomes the shortage of traditional method, by which the weights are set up in subjective manners, and makes the model have a better rationality and a higher practicability.

4 Evaluation and Comparison

We will now analyse the effectiveness of the proposed trust evaluation model by means of experiments. Our intention with this section is to confirm if it is accurate and has robust dynamic adaptive capacity in a variety of scenarios where more complex malicious and dynamic strategies are introduced. Referring to Liang and Shi's technique[28], we have implemented a simulator to test the feasibility of the CAT model based on NETLOGO[29], a very popular multi-agent simulation tool implemented based on JAVA in the AI community, which can easily model the parallel and independent agents, to simulate interactions among entities. For comparison purpose, we have also implemented other two notable trust mechanisms aPET model[30] and PT model[6] in the simulator. We have designed several performance mechanisms for our trust model and algorithms in a comprehensive way. Owing to restriction of paper length, we mainly focus on two aspects: forecasting accuracy and dynamic adaptation capability.

4.1 Simulation Setup

In our simulation, we use some of the experimental parameters as the same of Liang's simulator[28]. In order to empirically evaluate our trust mechanism against more complex strategies, we make some changes. Our network simulation proceeds in cycles. For simplicity, we assume that every peer in the network makes one transaction in each query cycle and each peer holds a limited number of direct trust neighbors. According to [28], the behaviors of peers like FR can be one of the four types: honest peer (HFR), malicious peer (MFR), exaggeration peer (EFR), and collusive peer (CFR). HFR always gives correct feedbacks. MFR always gives the opposite opinion $(1 - T)$ to others. EFR exaggerates their ratings by an exaggerating factor, which is 0.5 in our simulation. For this type of FR, the feedback $T + e(T - 0.5) = T + 0.5(T - 0.5)$ will be sent out. For the collusive peers, CFR sends out 1 for the peers in the collusive group, and 0 for the peers outside the group. We use a network community with up to 10^5 peers, and 10 direct trust neighbors per node as a starting point for the experiment.

Table 3 summarizes the main parameters related to the community setting, trust computation and outcomes the evaluation. The default values of these parameters for most experiments are listed. In our simulation, the size of the collusive group takes up 20% of the total number of the network. Three types of behaviors of SP are studied: fixed (FSP), random (RSP) and oscillating (OSP). FSP includes the fixed good SP and fixed bad SP. With the option FSP, SP will not change their qualities once the simulation starts. With the option RSP, SP changes their qualities randomly. While with the option OSP, SP changes their qualities in a fixed oscillation span (20 time-steps in our simulation). Both the random and oscillating SP are dynamic. The percentage of the bad SP (BSP) can be 20% or 70% within the whole system. The percentage of the dynamic SP (DSP) can be 30% or 70%, 30% DSP simulates a relative stable community and 70% simulates a high dynamic community with many dynamic peers. For the percentage of honest FR, we take two choices, 20% or 80%. 20% HFR reflects the community is a terrific community and 80% HFR reflects the community is a relative good community with less BFR. Instead of using the physical running time, we use the notion of time-step, which is introduced in the NETLOGO[29], to calculate the simulation time. Within each step, the peer will finish all the activities including service requesting, service providing, trustworthiness value updating, and feedbacks disseminating.

Table 3. Simulation Setup and Descriptions

Setup	Values	Descriptions
N	10^5	The total number of peers
BSP	20, 70%	BSP=FSP+RSP+OSP
DSP	30, 70%	The percentage of dynamic SP
HFR	20, 80%	The percentage of honest FR
Time-Steps	2000	The total running time-steps
H	2∼12	HEW
λ	[0.5, 1]	The parameter in Algorithm 1
β	2	The parameter in (19)
α	0.2	The parameter in (18)
$\Delta\tau$	20	Updating-time-threshold
max-LEVEL	3	The parameter in DTT

4.2 Accuracy

The ultimate goal of any trust model endeavor is to have an accurate and unbiased forecast for given entity's trust degree. A secure trust system should have a good trust decision-making accuracy, that is to say, by should have a strong capability to detect and resist malicious entities' actions. In the first set of experiments, we mainly evaluated the benefits of our approach compared to other approaches in the accuracy of the trust decision-making mechanism. In this subsection, we firstly introduce two formulas for forecast error to reflect system's capability of decision-making accuracy, which are defined as the difference between actual quantity and the forecast error, it is shown as follows[21−22].

1) Mean Absolute Deviation (MAD):

$$\mathrm{MAD} = \frac{\sum |e_{t_{\mathrm{TS}}}|}{t_{\mathrm{TS}}} \tag{21}$$

where $e_{t_{\mathrm{TS}}}$ is the forecast error for period t_{TS}, $e_{t_{\mathrm{TS}}} = A_{t_{\mathrm{TS}}} - F_{t_{\mathrm{TS}}}$, $A_{t_{\mathrm{TS}}}$ is actual demand for period t_{TS}, and t_{TS} is the total number of periods of evaluation.

2) Mean Absolute Percentage Error (MAPE):

$$\mathrm{MAPE} = \frac{1}{t_{\mathrm{TS}}} \sum \left|\frac{e_{t_{\mathrm{TS}}}}{A_{t_{\mathrm{TS}}}}\right| (\times 100\%). \tag{22}$$

As the same in (21), $e_{t_{\mathrm{TS}}}$ is the forecast error for period t_{TS}, $e_{t_{\mathrm{TS}}} = A_{t_{\mathrm{TS}}} - F_{t_{\mathrm{TS}}}$, $A_{t_{\mathrm{TS}}}$ is actual demand for period t_{TS}, and t_{TS} is number of periods of evaluation. The values of MAD and MAPE are indicators of bias in the forecasts. They are checked to determine if they are within the acceptable control limits. The values are better if they are closer to zero. In order to compute the values of MAD and MAPE, we must know the value of $e_{t_{\mathrm{TS}}}$. To calculate the value of $e_{t_{\mathrm{TS}}}$, we must first know the real value of $A_{t_{\mathrm{TS}}}$, but the real value of $A_{t_{\mathrm{TS}}}$ is usually very difficult to calculate. In the simulation, we used standard deviation through repeated determination, and obtained the arithmetic average as a real value of $A_{t_{\mathrm{TS}}}$.

Table 4. Results under Parameter λ

λ	MAD	MAPE (%)
0.5	0.248 080	18.35
0.6	0.172 460	17.69
0.7	0.130 945	14.72
0.8	0.125 950	12.06
0.9	0.196 256	19.25
1.0	0.256 923	20.64

We firstly observe the experimental results under different situation parameters λ in Algorithm 1 and different values H in (1). The experimental environment is a more stable community environment, in which only a small number of entities in the systems are malicious nodes (MFR + EFR + CFR = 20%), 80% of SP always provide stable service (DSP = 20%), and 80% of the entities are not free to join or leave the system (DPP = 20%). Table 4 is the simulated results for λ and Table 5 is the simulated results for H. From the experimental results in Table 4 it can be seen that under a relatively stable community environment, when the value of situation parameter λ is 0.8, the values of MAD and MAPE of the trust model are the optimum. Therefore, in the later of the experiments, we set $\lambda = 0.8$ as the basic value of the situation parameter.

In Definition 3.1, we introduce a threshold H, when $h < H$, the trust model need to compute FTD. If $h \geqslant H$, it indicates that current direct evidences are enough to judge peer's trust degree, and the CAT model does not need to compute FDT. In the simulation, H is defined with possible values 2∼12. From the experimental results in Table 5, we can see when the value of the parameter H is 2, the values of MAD and MAPE of the trust model are the lowest. When the value of parameter H is higher than 6, the values of MAD and MAPE of the trust model change in a flat trend. At the same time, when the value of parameter H is higher than 6, the values of MAD and MAPE are very low. Averagely, the value of MAD is 0.1225 and the value of MAPE is 12.12. As mentioned above, the key to an efficient P2P trust system is the feedback aggregating mechanism. However, in a practical network, there are many dishonest feedback peers and the feedback mechanism inevitably need to consume substantial resources. Based on human cognitive psychology, in Definition 1, we use HEW mechanism as far as possible to reduce the number of computing FDT, which can partly overcome the defects brought by feedback. From the simulation results in Table 5, the higher the values of H the better the performance from the perspective of accuracy. But in the P2P system with sparse transactions, setting a lower H may exclude the help of feedback rating. So, the setting of H needs the help of the actual network situation. In a P2P system with sparse transactions, it can be given a smaller value. On the contrary, it should be given a bigger value. In our simulation, we set the P2P network as a busy transaction system. According to the results of Table 5, we set $H = 6$ as the basic value for this parameter.

Table 5. Results under Parameter H

	MAD	MAPE (%)
$H = 2$	0.251 200	20.132 50
$H = 4$	0.163 120	16.012 10
$H = 6$	0.132 145	12.232 50
$H = 8$	0.127 110	12.230 10
$H = 10$	0.122 620	12.012 25
$H = 12$	0.122 610	12.011 21

Fig.5 shows the experimental results of MAPE and MAD under a relatively stable community environment. In simulation, the total percentage of malicious FR (MFR + EFR + CFR) is set to 20%; the total percentage of malicious SP (MSP) also is 20%, which reflects the community is a relative good community with less malicious entities. From Fig.5(a), we can see that, within 2000 time-steps, all of the three models have relatively stable performance with a value of MAD arranging from 0.123 to 0.17, which means that all of the three models play well when facing the little number of malicious entities. Under a real network environment, most of entities are good entities. So from a practical point of view, all of the three models can meet the practical demand. Moreover, it is observed that our trust mechanism slightly outperforms the other two mechanisms. Fig.5(b) shows the value of MAPE being computed by the three mechanisms under the same community as Fig.5(a). From the figure, we can see that the MAPE obtained by the CAT is the best among all the three models. Averagely, it is 2% lower than aPET model, and 4% lower than PT model. Through comprehensive comparison between Fig.5(a) and Fig.5(b), we find that the CAT has a more stable and accurate performance than the other two models.

Fig.6 shows the experimental results of MAPE and MAD under a malicious community environment. In the simulation, the total percentage of malicious FR (MFR + EFR + CFR) is set to 50%; the total percentage of malicious SP (MSP) is 80%, which reflects the community is a terrible community with 50% of HFR and 80% of BSP.

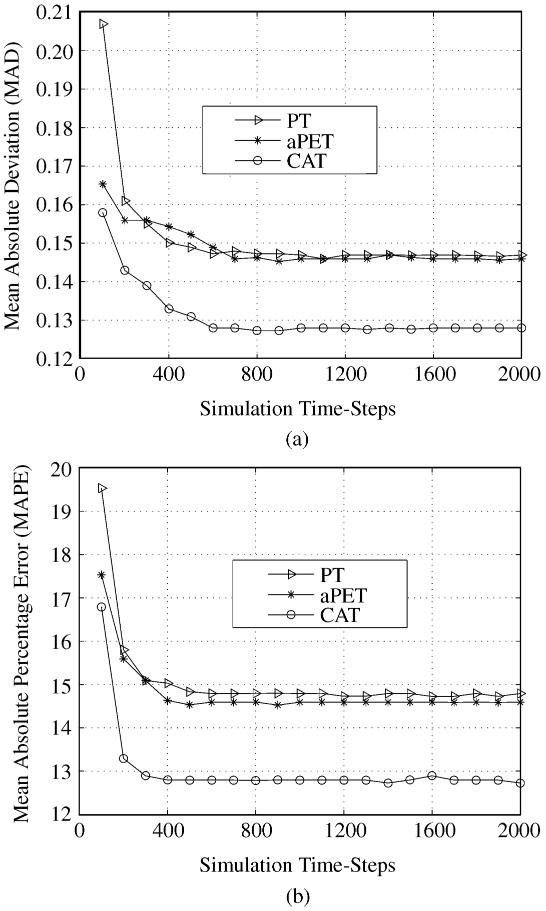


Fig.5. Accuracy evaluation under a relatively stable community environment. (a) Comparison of MAD. (b) Comparison of MAPE.

From Fig.6(a), we find that, in a malicious environment, our CAT model can get the best robust service capability, which has a good MAD, averagely to 0.125. However, compared to CAT, the certain increase of the value of MAD for aPET and PT, and especially for PT, increase its MAD to 0.185. Fig.6(b) shows a similar result as Fig.6(a), which CAT's MAPE is better than the other two models under a malicious community environment. From this set of experiments, we can find that CAT has a more robust capability of decision-making accuracy under both relatively stable community environment and malicious community environment.

4.2 Dynamic Adaptation Capability

Dynamic adaptation capability is also called robustness of trust models; it indicates system's providing stable service capability under many kinds of complex and dynamic network environment. One of the most challenging issues in open environments is to handle dynamic behaviors, which attracts a lot of attentions of researches. In our second experiment, we will demonstrate how our trust model enhances the service capability and reduces the number of false services in a highly dynamic changing distributed network.

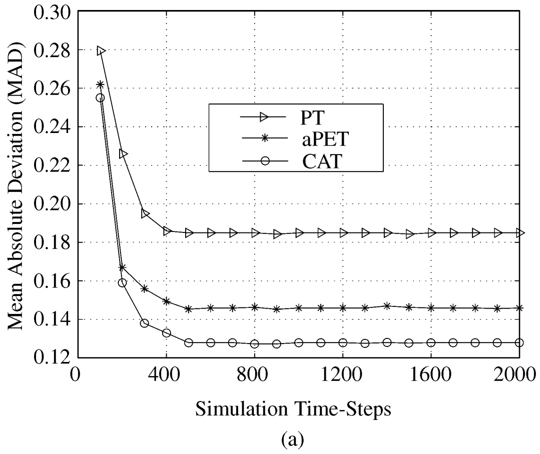


Fig.6. Accuracy evaluation under a malicious community environment. (a) Comparison of MAD. (b) Comparison of MAPE.

Using Successful Service Percent (SSP) to evaluate the performance of dynamic adaptation capability of the proposed trust model. Suppose that $G_{t_{\rm TS}}$ is the total number of good service detected by trust model in a certain period $t_{\rm TS}$, $S_{t_{\rm TS}}$ is the total number of service at time-step point $t_{\rm TS}$, and then SSP is defined as:

$$\mathrm{SSP} = \frac{\sum_{t=1}^{t_{\rm TS}} G_{t_{\rm TS}}}{\sum_{t=1}^{t_{\rm TS}} S_{t_{\rm TS}}} \times 100\%. \tag{23}$$

In general, the dynamic of the open network is caused by three reasons: 1) entities' dynamic, all entities in the network can randomly join or leave the networks; 2) SP's dynamic, SP can dynamically change their identities between good services and bad services; 3) service dynamic, there are more service requests in a busy distributed system than in an idle system. High successful service percent reflects system having a good dynamic adaptation capability. Referring to [28], in our simulation, we use three parameters to reflect the dynamic P2P system: 1) Service requesting frequency (SRF $\in [0,1]$). For each entity, after a random time, entity sends out a service requesting to an SP. The bigger the SRF value is, the more frequent the service requesting is sent by entities, where illuminate system is a busy system. 2) Service dynamic factor (SDF $\in [0,1]$). After a random time, SP oscillates providing good and bad services. 3) Dynamic peers' percentage (DPP $\in [0,1]$). It illuminates there are DPP $\times$ N peers which are instable; they are free to leave or join the system at any moment.

High successful service percent reflects system has a good dynamic adaptation capability. In the experiments, we configured the raters percent as HFR = 80%, MFR = 10%, EFR = 5% and CFR = 5% according to a practical P2P system. In a practical P2P system, a majority of entities are honest (HFR = 80%), only a small part of entities are malicious (MFR% + EFR% + CFR% = 20%) according to two conditions: (a) idle

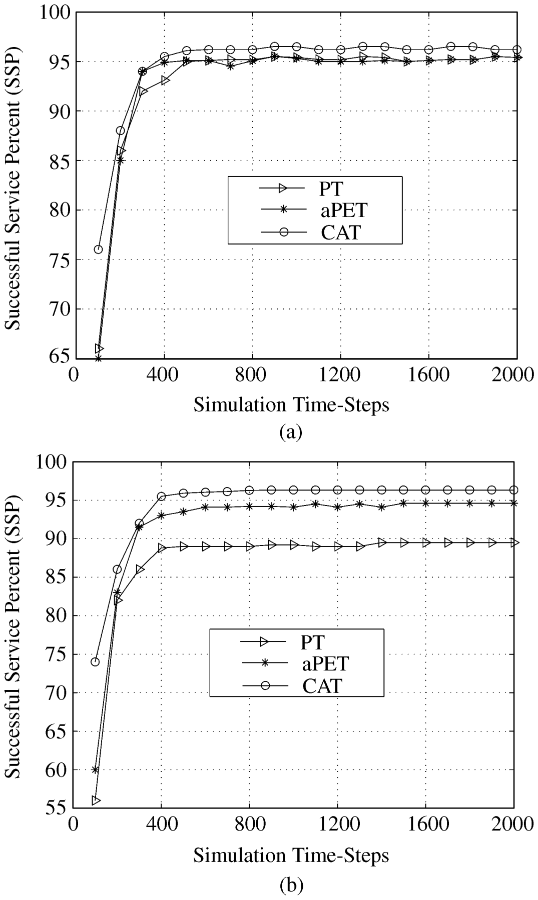


Fig.7. Comparison of SSP. (a) In an idle and stable community. (b) In a highly busy and dynamic community.

and stable environment, where SRF = 0.2, SDF = 0.2, DPP = 0.2; (b) highly busy and dynamic environment, where SRF = 0.8, SDF = 0.8, DPP = 0.8.

First, let us look at the case in an idle and stable environment. From Fig.7(a), we can see that the three mechanisms have a correspondingly robustness in providing good services, which all of their values of SSP are beyond 90%, which reflects that all of the three models play well in an idle and stable community. From Fig.7(b), we find that, in a busy and highly dynamic environment, the CAT model gets the best dynamic adaptation capability, which has about 95% of SSP. However, comparing to CAT, the significant increase of the percentage of dynamic entities incurs the significant decrease of the adaptation capability for aPET and PT. And especially for PT, it drops its SSP down to 90%. From this set of experiments, we find that the dynamic adaptation capability of all three algorithms have a notable decrease, but compared to the other two algorithms, the CAT model gets the higher dynamic adaptation capability under a highly busy and dynamic environment.

4 Conclusion

Trust is one of the most fuzzy, dynamic and complex concepts in both social and business relationships. The difficulty in measuring trust and forecasting trust value in large-scale P2P network environments leads to many questions. These questions include issues such as how to measure the willingness and capability of individuals in the trust dynamic nature and how to assign a concrete level of trust to peers, especially, how to provide robust service capacity when system scale is very large. There are many of state-of-the-art P2P trust models proposed. But many of these studies paid little attention to the adaptability, dynamics and scalability of P2P trust systems for large-scale P2P networks.

In this paper, we regard adaptability of trust as the first requirement for large-scale and dynamic P2P networks. The main contributions include:

1) An adaptive trusted decision-making method based on HEW is proposed, which not only can reduce the risk and improve system efficiency, but also can solve trust measuring and forecasting problem when the direct evidences are insufficient.

2) Direct trust computing method based on IOWA operator and feedback trust converging mechanism based on DTT are set up, which makes our model a higher practicability and a better scalability than previous studies.

3) Two new parameters, confidence factor and feedback factor, are introduced to adjust the weight of direct trust and feedback trust adaptively, which overcomes the shortage of traditional methods, in which the weights are set up by subjective approach.

Our CAT model has the capacities of fast aggregating speed, good dynamic adaptability against malicious peers, and high scalability for large-scale P2P networks. This paper provides both theoretical foundations and experimental results to validate the trust mechanism, which extends significantly from our preliminary results reported in [2, 20, 27, 31–33]. However, there are still lots of open issues. As a next step, we will be looking for ways to make the approach more robust against malicious behaviors, such as collusion among peers. We are also interested in combining trust management with intrusion detection to address concerns of sudden and malicious attacks. Implementing and evaluating our CAT model on various P2P systems, such as distributed file sharing and P2P grid computing, is another direction for future research. Particularly, we will develop a new mechanism in which the value of H can be adaptively configured according to the environment context, and it will be a good supplement of the adaptiveness for the proposed model.

References

[1] Chang E, Thomson P, Dillon T *et al.* The fuzzy and dynamic nature of trust. In *Proc. the 2nd International Conference on Trust, Privacy and Security in Digital Business* (*Trustbus'05*) *in conjunction with DEXA*, Copenhagen, Denmark, August 22–26, 2005, pp.161–174.

[2] Li X Y, Gui X L. Research on dynamic trust model in large-scale distributed environment. *Journal of Software*, 2007, 18(6): 1510–1521.

[3] Li H Z, Singhal M. Trust management in distributed systems. *IEEE Computer*, 2007, 40(2): 45–53.

[4] Ji M, Orgun M A. Trust management and trust theory revision. *IEEE Transactions on Systems, Man and Cybernetics*, 2006, 36(3): 451–460.

[5] Sloman M. Trust-management in Internet and pervasive systems. *IEEE Intelligent Systems*, 2004, 19(5): 77–79.

[6] Xiong L, Liu L. Peer-Trust: Supporting reputation-based trust in peer-to-peer communities. *IEEE Transactions on Data and Knowledge Engineering*, 2004, 16(7): 843–857.

[7] Liang Z Q, Shi W S. Enforcing cooperative resource sharing in un-trusted peer-to-peer environments. *Journal of Mobile Networks and Applications-Springer*, 2005, 10(6): 771–783.

[8] Liang Z Q, Shi W S. PET: A personalized trust model with reputation and risk evaluation for P2P resource sharing. In *Proc. the 38th Hawaii International Conference on System Sciences*, Los Alamitos, USA, Jan. 3–6, 2005, pp.201–210.

[9] Zhou R F, Hwang K. Power-Trust: A robust and scalable reputation system for trusted Peer-to-Peer computing. *IEEE Trans. Parallel and Distributed Systems*, 2007, 18(4): 460–473.

[10] Song S, Hwang K, Zhou R F *et al.* Trusted P2P transactions with fuzzy reputation aggregation. *IEEE Internet Computing Magazine*, Special Issue on Security for P2P and Ad Hoc Networks, Nov/Dec 2005, 9(6): 24–34.

[11] Tran T, Cohen R. Modeling reputation in agent based marketplaces to improve the performance of buying agents. In *Proc. the Ninth International Conference on User Modeling*

(*UM-03*), Johnstown, PA, USA, June 22–26, 2003, pp.273–282.

[12] Cornelli F, Damiani E, Vimercati S *et al.* Choosing reputable servants in a P2P network. In *Proc. the Eleventh International World Wide Web Conference*, Hawaii, USA, Nov. 18–22, 2002, pp.376–386.

[13] Damiani E, Vimercati S, Paraboschi S *et al.* Managing and sharing servants' reputations in P2P systems. *IEEE Trans. Knowledge and Data Engineering*, 2003, 15(4): 840–854.

[14] Damiani E, Vimercati S, Paraboschi S *et al.* A reputation-based approach for choosing reliable resources in peer-to-peer networks. In *Proc. the Ninth ACM Conference on Computer and Communications Security*, Washington DC, USA, 2002, pp.207–216.

[15] Kamvar S D, Schlosser M T, Garcia M. The Eigen-Trust algorithm for reputation management in P2P networks. In *Proc. the Twelfth International World Wide Web Conference*, Budapest, Hungary, May 20–24, 2003, pp.640–651.

[16] Richardson M, Agrawal R, and Domingos P. Trust management for the semantic Web. In *Proc. the Second International Semantic Web Conference*, Sardinia, Italy, Oct. 20–23, 2003, pp.351–368.

[17] Guha R. Propagation of trust and distrust. In *Proc. the World Wide Web Conf.* (*WWW2004*), ACM Press, New York, USA, May 19–21, 2004, pp.403–412.

[18] Buchegger S, Le B. A robust reputation system for P2P and mobile ad-hoc networks. In *Proc. the 2nd Workshop Economics of Peer-to-Peer Systems*, Boston, USA, June 4–5, 2004, pp.119–123.

[19] Chang J S, Wang H M, Yin G. A time-frame based trust model for P2P systems. In *Proc. the 9th International Conference on Information Security and Cryptology*, Busan, Korea, Nov. 30–Dec. 1, 2006, pp.155–165.

[20] Li X Y, Gui X L. Novel scalable aggregation algorithm of feedback trust information. *Journal of Xi'an Jiaotong University*, 2007, 41(8): 879–883.

[21] Chase R B, Jacobs F R, Aquilano N J. Operations Management. Chase R B (eds.), New York: Mcgraw-Hill, April 2005.

[22] Song Q, Chissom B S. Forecasting enrollment with fuzzy time. *Fuzzy Sets and Systems*, 1993, 54(1): 1–9.

[23] Yager R R. On ordered weighted averaging aggregation operators in multi-criteria decision making. *IEEE Transactions on Systems, Man, and Cybernetics*, 1988, 18(1): 183–190.

[24] Mitchell H B, Estrakh D D. A modified OWA operator and its use in Lossless DPCM image compression. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 1997, (5): 429–436.

[25] Yager R R. Induced aggregation operators. *Fuzzy Sets and Systems*, 2003, 137(1): 59–69.

[26] Kacprzyk J, Zadrozny S. A general collective choice rule in group decision making under fuzzy preferences and fuzzy majority: An OWA operator based approach fuzzy systems. In *Proc. the 2002 IEEE International Conference on Fuzzy*, Honolulu, USA, May 2002, pp.1280–1285.

[27] Li X Y, Gui X L. Engineering trusted P2P system with fast reputation aggregating mechanism. In *Proc. the IEEE International Conference on Robotics and Biomimetics*, Sanya, China, Dec. 15–18, 2007, pp.2007–2012.

[28] Liang Z Q, Shi W S. Analysis of recommendations on trust inference in open environment. *Journal of Performance Evaluation*, 2008, 65(2): 99-128.

[29] Tisue S. NetLogo. 2008, http://ccl.northwestern.edu/netlogo/.

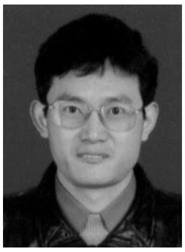
[30] Liang Z Q, Shi W S. TRECON: A framework for enforcing trusted ISP peering. In *Proc. the 15th IEEE International Conference on Computer Communications and Networks* (*ICCCN 2006*), Arlington, USA, Oct. 9–11, 2006, pp.383–389.

[31] Li X Y, Gui X L. SDT: A scalable dynamic trust model with multiple decision factors. *Dynamics of Continuous Discrete and Impulsive Systems Series B: Applications & Algorithms.* 2007, 14(2): 23–27.

[32] Li X Y, Gui X L. Research on adaptive prediction model of dynamic trust relationship in open distributed systems. *Journal of Computational Information Systems*, 2008, 4(4): 1427–1434.

[33] Li X Y, Gui X L. Trust quantitative model with multiple decision factors in trusted network. *Chinese Journal of Computers*, 2009, 32(3): 405–416.

Xiao-Yong Li is a Ph.D. candidate in Xi'an Jiaotong University in China. As the first author, he has published more than thirty journal papers. In 2009, he is awarded outstanding graduates in Shaanxi Province. His current research interests mainly include networks computing and trusted system.



Xiao-Lin Gui is a professor and a Ph.D. supervisor in Xi'an Jiaotong University. He has published more than eighty papers and obtained five patents and three software copyrights. In 2006, he is awarded New Century Excellent Talents in University (NCET). Now he is in charge of a project of the National High-Tech Research and Development 863 Program and a project of the National Nature Science Foundation of China. Currently, he leads the Trusted Computing Technology Research Center (TCT Lab) at Xi'an Jiaotong University. His research interests include networks computing and trusted system.