Scientific Research and Essays Vol. 6(18), pp. 3845-3857, 1 September, 2011
Available online at http://www.academicjournals.org/SRE
DOI: 10.5897/SRE11.437
ISSN 1992-2248 ©2011 Academic Journals

Full Length Research Paper

P2PETrust: A novel distributed transaction history vector-based trust appraisal model for peer-to-peer e-commerce networks

Zhenhua Tan[1]*, Wei Cheng[1], Yi Ma[1], Zhiliang Zhu[1] and Guiran Chang[2]

[1]Software College of Northeastern University, Shenyang City, Liaoning Province, 110819, PR China.
[2]Computing Center of Northeastern University, Shenyang City, Liaoning Province, 110819, PR China.

Accepted 27 June, 2011

Peer-to-peer (P2P) E-commerce systems are more and more popular in recent years with opportunities and threats both existing. A lot of malicious nodes threaten the system's security via cheating, selfish and attacking or else. Trust mechanisms can help estimating the trustworthiness. This paper presents a distributed trust model named P2PETrust (peer-to-peer E-commerce trust model) based on social networks' principles. A transaction history vector with multi dimensions is designed to describe the history before the distributed storage structure. In order to compute the trust, three sub trust models are defined in P2PETrust, which are local model, global model and correlation model. The local trust model is the base trust mechanism to compute history between two nodes after defining the nonlinear time factor, transaction amount factor, frequency factor and success rate for the history while the global one is based on local model to calculate a node's whole transaction histories with some global factors defined. On the basis of these jobs, a correlation trust model is presented by calculating nodes' correlative similarity multiplied with global trust degree. We present a simulation environment to validate the proposed trust model and report the set of initial experiments, showing the feasibility and benefit of our approach.

Key words: Peer-to-peer E-commerce, trust model, P2P security, social network, network security.

INTRODUCTION

Peer-to-Peer (P2P) overlay network (Touch, 2001), which is self-organized and distributed, can make full use of the edge system entities in internet-scale to construct large-scale cooperating and resource sharing environments. The P2P network can be classified into unstructured P2P and structured P2P in terms of the topology, and the structured P2P network is the third generation P2P network and more popularly used, while the unstructured P2P is the first and second generation.

With fast development in recent years, P2P electronic commerce (E-commerce) communities is applied widely in many application areas like file sharing, search engine, distributed storage and distributed on-line shop. Because of the openness, reciprocity, random and spontaneous joining of the nodes and lack of centralized system management in the structured distributed P2P system, some urgent problems regarding the availability and security of P2P network remain to be solved, such as malicious attacking, team malicious cheating, intellectual property rights, selfish and routing attacking in P2P (Zhang and Helvik, 2010; Mekouar et al., 2006; Zhang et al., 2005). Trust management has been emerging as an essential complementary to security mechanisms of P2P systems. A recent study (Esaki, 2010) reported clearly that P2P technology is a most important component of the next generation internet, and valid and effective trust mechanisms are important safeguard to P2P systems.

A well-defined trust system can help users to select

*Corresponding author. E-mail: tanzh@mail.neu.edu.cn

Abbbreviations: P2PETrust, Peer-to-peer e-commerce trust model; **E-commerce,** electronic commerce; **THVector,** transaction history vector; **TDL,** TimeDBList.

safer nodes for transactions, encouraging interactions between honest nodes and punishing or excluding malicious nodes. Nowadays, there has been a recent burst of interest on the topic of trust, due in part to the importance it plays in e-commerce applications. Traditional trust model for E-commerce system is mainly based on trusted third-party such as central certification system, not suitable for P2P E-commerce system (Dou et al., 2004), for the P2P system is quiet distributed system. In a P2P E-commerce system, we should consider more aspects, including distributed data structure, distributed storage and distributed computing.

Based on some social network principles, this paper presents a new distributed trust model named P2PETrust (peer-to-peer E-commerce trust model). Similar to other trust models, we begin the study based on nodes' histories. The model has the following innovative features:

It designs a transaction history vector and its distributed storage structure for P2P trust computing. This distributed method do not need additional topology but conforming to specific P2P topology which uses the P2PTrust.

Trust value can be calculated by a multi-dimensional factors, including time factor, transaction amount factor, and frequency factor and success rate, both in local and global environments. Especially, the transaction amount is based on value-amount which means the real value of the transaction amount calculated by the success rate, and the time factor is not simple linear to express the time's importance.

It presents three sub trust models. The basic one is local trust which has fast trust convergence, and global trust which is to resist single malicious behaviors is based on local trust. Correlation trust model is the product of global trust and nodes' history similarity, which can resist team malicious attacks.

In the remainder of the paper, we introduce some related works in the next section. Section 3 describes the transaction history vector for P2PETrust. Section 4 describes the local trust mechanism of P2PETrust while section 5 designs the computing method for global trust degree and section 6 describes the correlation trust model of P2PETrust. The simulations and results follow in section 7, with conclusions afterwards in section 8.

RELATED WORKS

Many researchers dedicated themselves to the P2P trust model. But what is trust? In social networks, trust has several connotations, the typical definition of trust follows the general intuition about trust and contains such elements as: (1) the willingness of one party (trustor) to be vulnerable to the actions of another party (trustee); (2) reasonable expectation (confidence) of the trustor that the trustee will behave in a way beneficial to the trustor; (3) risk of harm to the trustor if the trustee will not behave accordingly; (4) and the absence of trustor's enforcement or control over actions performed by the trustee (McKnight and Chervany, 1996). In 1990s, Marsh (1994) uses the definition by Gambetta (2000), which is commonly accepted in the literature: "…trust, (or symmetrically, distrust) is a particular level of the subjective probability with which an agent will perform a particular action, both before he can monitor such action (or independently of his capacity to monitor it) and in a context in which it affects his own action." Almost all of the related works on trust is based on the above definitions.

Trust is very important to a distributed P2P system. Many researchers do contributions to the P2P trustworthy issues (Aberer and Despotovic, 2001; Kamvar et al., 2003; Dou et al., 2004; Xiong and Liu, 2004; Jøsang et al., 2006; Jøsang and Bhuiyan, 2008; Wang and Wu, 2010; Jiang and Li, 2007; Li and Dai, 2010) in recent ten years.

Aberer and Despotovic (2001) propose a complaint – only trust management method for a distributed P2P system, due to the lack of incentives for submitting feedbacks. The complaint-only trust metric works in very limited cases and is over-sensitive to the skewed distribution of the community and to several misbehaviors of the system. Although this mechanism has some limitations, it is the very early trust model for P2P E-commerce.

Kamvar et al. (2003) present the EigenTrust reputation system to compute a unique global trust in very distributed way. Such a global model does not need an administration center, but it is difficult to guarantee a fast and secure convergence when computing the global trust which also inspires this works.

Dou et al. (2004) improve the EigenTrust in computing convergence and model security. However, there remain efficiency problems and its security mechanism is only from punishment and certification.

Xiong and Liu (2004) proposed a PeerTrust model with three basic trust parameters and two adaptive factors in computing trust of peers, and then define a general trust metric to combine them.

Jøsang et al. (2006) proposed a method for simplifying a complex network so that it can be expressed in a series of parallel network and then be computationally analyzed. This solution may lead to loss of trust information. An edge splitting method is proposed in their further works (Jøsang and Bhuiyan, 2008) to address this problem. But this method is valid only on a simple trust network. It may not be valid on a complex trust network.

Wang and Wu (2010) proposed a multi-dimensional evidence-based trust management system with multi-trusted paths (MeTrust for short) to conduct trust computation on any arbitrarily complex trusted graph. The trust computation in MeTrust is conducted at three tiers, namely, the node tier, the path tier, and the graph

tier. It is an excellent trust model. But it does not provide distributed storage structure for P2P system.

Jiang et al. (2007) presented a novel reputation-based trust mechanism for P2P e-commerce systems. In this mechanism, a peer has two kinds of reputations, namely local reputations and global reputations. To compute the local and global reputations precisely and to obtain stronger resistibility to attacks as well, many comprehensive factors in computing trust value are introduced in the mechanism. Anyway, this model is a comprehensive mechanism. However, its time factor is only linear to express the time's importance and there is no clear method to resist team malicious behaviors.

Tan et al. (2009), presented a global trust model with correlation factor based on communication history. However, this mechanism has very simple history vector. Generally speaking, trust models above can be classified into two modes, one is local information based and one is global trust information based. The local trust of a peer relative to another peer is calculated in terms of the reference peer rating of the transaction between the two peers, whereas the global trust is computed based on all peers' rating of the transaction between them. Li and Dai (2010) have done a good job in concluding the trust models.

TRANSACTION HISTORY VECTOR

Topology of network could be described as a graph $G=(V,E)$ while V means vertices and E means the edges between vertices, such as topologies of computer network, social network and so on. A P2P overlay network belongs to a kind of G also; usually, we make an additional definition for it, that is $G_{p2p}=(V,E,d)$, while the d means the participated degree of each network vertex. Of course, a P2P E-commerce system is one kind of P2P overlay networks. In our opinion, the participated degree in P2P E-commerce network could be considered as a trust vector and such network could be defined as $G_{E-p2p}=(V,E,Trust)$. In order to define a relatively reasonable trust model for
P2P E-commerce networks, we refer to some basic principles of social networks.

In social networks, we could judge a person is trustworthy or not by his social activities history. A person is more trustworthy when he gets more social activities especially successful business. And higher activity frequencies, better appraisals and more recent activities help more positive trustworthy to a person.

According to these factual principles from social networks, we define a social-like trust model for P2P E-commerce network named P2PETrust (Peer-to-Peer E-business Trust model) in this paper. First but most important, we need construct a transaction history vector named THVector (Transaction History Vector) for P2PETrust. In THVector, six factors are considered, including transaction time stamp, transaction frequency, successful transaction times, failed transaction times, transaction currency amount and transaction appraisal. The THVector can be formalized and defined as $THVector=<t,\varphi,s,f,\omega,\phi>$, that, t is the time stamp for the transaction, φ stands for frequency, s stands for successful times while f stands for failed ones, ω on behalf of the transaction amount and the ϕ means the user appraisals. This section will give more informative definition for the THVector and design the storage structure.

Basic definition

In the following definition, the symbol *i* or *j* stands for the node's identification in the P2P E-commerce system, and this symbol would be an integer number such as 1, 2, 3, and unique in one particular P2P business system.

Definition 1

a) Using t_m^{ij} to represent the time stamp of the m-th transaction activity between nodes i and j, especially, t_{now}^{ij} stands for the latest time stamp.

b) Using t_m^i to represent the time stamp of the m-th transaction of transaction history of node i. especially, t_0^i represents the initialized time of node i and t_{now}^i stands for the latest one.

Definition 2

Using $\omega_m^{ij}=\omega(i,j,t_m^{ij})$ to represent the transaction amount between nodes i and j at the time t_m^{ij}.

Definition 3

a) Using I_{all} to stand for all of the nodes in P2P E-commerce network.

b) Using $history(i,j)=\begin{cases}true\\false\end{cases}zzzzz$ to stand for whether transaction occurs between nodes i and node j, and "true" means i have communicated with j (also means there exists history between i and j).

c) Using $I_i=\{j \mid j\in I_{all}, history(i,j)=true\}$ to stand for the set of nodes who have ever transacted with node i.

d) Using $I_{ij} = I_i \bigcap I_j$ to stand for the set of nodes who have ever both transacted with node i and node j.

Definition 4

Using S_{ij} to stand for the successful transaction times between nodes i and j while F_{ij} stands for the failed one between the two.

Definition 5

a) Using $\lambda_{ij}^m = \left(s_{ij} / (s_{ij} + f_{ij})\right)_{[t_0^i \sim t_m^i]} \in [0,1]$ to represent the successful transaction rate between i and j from t_0^i to t_m^i.

b) Using $\lambda_i^m = \left(\sum_{c \in I_i} s_{ci} \Big/ \sum_{c \in I_i} (s_{ci} + f_{ci}) \right)_{[t_0^i \sim t_m^i]} \in [0,1]$ to represent the successful transaction rate of node i from t_0^i to t_m^i.

Obviously, the λ_{ij}^m is the success rate between i and j while λ_i^m means the total success rate of node i.

Definition 6

Using $\phi_m^{ij} = \phi(i, j, t_m^{ij}) \in [-1,1]$ to stand for the appraisal feedback for the m-th transaction between i and j.

Definition 7

$$\varphi_m^{ij} = \varphi(i, t_m^{ij}) = \frac{(s_{ij} + f_{ij})}{(t_m^{ij} - t_0^i)}$$

(a) Using to stand for the transaction frequency between nodes i and j up to the time t_m^{ij}. (b) Using $\varphi_m^i = \varphi(i, t_m^i) = \frac{1}{(t_m^i - t_0^i)} \sum_{j \in I_i} (s_{ij} + f_{ij})$ to stand for the total transaction frequency of node i up to t_m^i.

Distributed storage structure for THvector

The P2P E-commerce system is a typical distributed system. So the transaction history of such a system should be distributed corresponding to the system topology. In P2PETrust, the THVector is a distributed vector and each transaction history data is stored in the corresponding node. All of the node's history vector could be combined into a whole transaction history vector.

Definition 8

Using $TimeDBList(i, j) = \left\{ < m, \omega_m^{ij}, \phi_m^{ij}, t_m^{ij} > | \, m \in [0, \max] \right\}$ (m is a natural number) to store the time-related transaction data which node i trades with node j (m is the transaction ID, and max means the maximum transaction number for nodes i and j). Such data list also be called time database list (*TimeDBList, TDL* for short).

As we can see, The $TimeDBList(i, j)$ is a set of quaternary vector $< m, \omega_m^{ij}, \phi_m^{ij}, t_m^{ij} >$, and recorded the transaction history for node i to node j, including ID, transaction amount, appraisal and time stamp. We store the $TimeDBList(i, j)$ in node i so that the transaction provider (node j) could not corrupt the history data. Figure 1 is a demonstration for $TimeDBList(i, j)$.

Definition 9

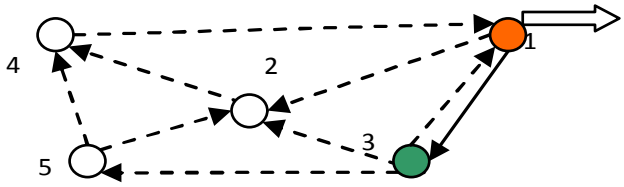
Using *mdh-list(i)* to store the transaction history of node I, and $mdh\text{-}list(i) = \{ < j, s_{ij}, f_{ij}, TimeDBList(i, j) > | \, j \in I_i \}$. The *mdh-list(i)* is a set of $< j, S_{ij}, F_{ij}, TimeDBList(i, j) >$, each records the history between nodes i and j. Each node maintains such a list. Figure 2 shows an example.

P2PETRUST: LOCAL TRUST DEGREE COMPUTING

The local trust degree is the trust expectation of one node to another node according to the transaction history data between the two nodes. From the perspective of social networks, higher transaction frequency, more transaction amount and better appraisals will help the trust value between nodes. Meantime, elder transaction history should give lighter impact on the trust computing. Therefore, we discuss three factors for local trust firstly, including time factor, transaction amount factor and frequency factor.

Time factor

The more recent appraisals impact the trust deeper, that is: $0 \le tf_1 < tf_2 < \ldots < tf_{now} \le 1$ (*tf* means time factor). In order to strengthen the importance of the current time, we appoint $tf_{now} = 1$ in this paper. Then, we use a right triangle Δ_{now} to calculate the time factor tf_m (m is the sequence number), just like Figure 3.

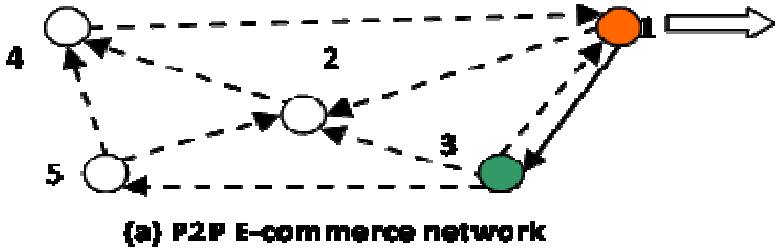


(a) P2P E-commerce network

m	ω_m^{13}	ϕ_m^{13}	t_m^{13}
1	30	0.7	t_1^{13}
2	22	0.5	t_2^{13}
3	800	0.5	t_3^{13}
4	0	-0.6	t_4^{13}
5	60	0.8	t_5^{13}

(b) *TimeDBList* (1, 3)

Figure 1. In this demonstration, about 4 nodes traded with node 1, and Figure 1 shows the *TimeDBList(1,3)*. As shown, 5 transaction activities happened from node 1 to 3 with different trade amount and appraisals. Of course, if node 1 selects node 2 for transaction, the data list *TimeDBList(1,2)* will be stored on node 1 too.



(a) P2P E-commerce network

J	S_{1J}	F_{1J}	TDL(I,J)
3	4	1	TDL(1,3)
2	2	0	TDL(1,2)

(b) mdh-list (1)

Figure 2. This is a demonstration of *mdh-list*. As shown, the node 1 has ever traded with node {2, 3}, 5 times with node 3 and twice with node 2. Also, the success rate between node 1 and 3 is 80% while it is 100% between node 1 and 2. The related data lists were linked separately.

As a result, the time factor tf_m is the area of Δ_m (the shadow right triangle in Figure 3). Obviously, $S\Delta_m < S\Delta_{now}$.

Definition 10

Using tf_m^{ij} to represent the time factor of the m-th transaction of node i to node j.

$$tf_m^{ij} = s(\Delta_m) = \frac{1}{2} \bullet (t_m^{ij} - t_0^i) \bullet \left(\frac{2 \bullet (t_m^{ij} - t_0^i)}{\delta(t)^2}\right) = \left(t_m^{ij} - t_0^i \Big/ t_{now}^{ij} - t_0^i\right)^2 \quad (1)$$

Obviously, $0 \le tf_1^{ij} < tf_2^{ij} < \ldots < tf_{now}^{ij} = 1$.

Transaction amount factor

In real social network, people think that larger transaction amount means better trust. At the same time, the transaction time is also very important to the amount factor. Suppose we have the same total trade amount in three transaction situation. Under the first situation, the biggest amount happened at the very early time; under the second condition, the biggest one happened in the middle time; and the biggest amount happened at the very current time in the third mode. Figure 4 shows these three kinds.

To be mentioned, we think the value of a transaction is relevant to the successful transaction rate (λ_{ij}^m) up to now. For instance, the transaction amount = \$100, but the node's λ_{ij}^m is only 0.1, we think the value of this transaction is only 100 * 0.1 = \$10. This policy can stimulate node's better service and better quality so as to improve the success rate.

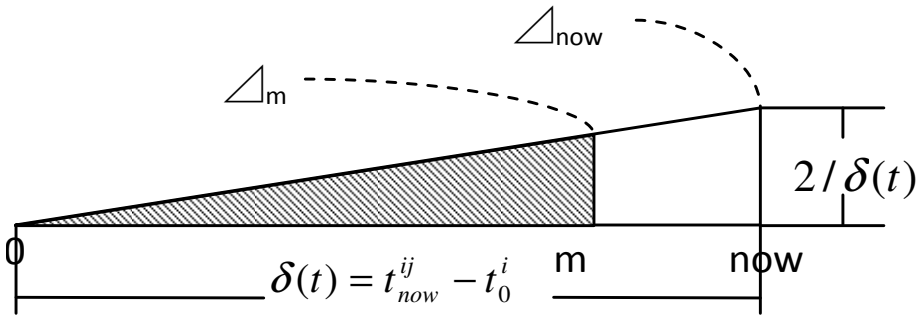


Figure 3. Appoint the area of this right triangle is a unit 1, and the under leg is the time interval $\delta(t) = t^{ij}_{now} - t^{i}_{0}$. So that, the length of the vertical leg is $2/\delta(t)$. Then we can computing the time factor by calculate the area of ⊿m

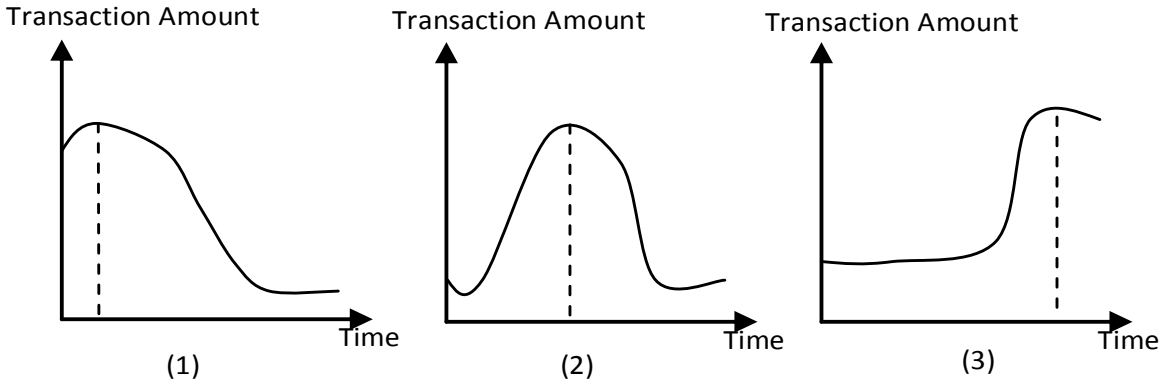


Figure 4. It shows three kinds of transaction with biggest amount. We think the third one is most effective to trust because; it is the most recent one.

Definition 11

Using v^{ij}_{m} to represent the value of the m-th transaction between nodes i and j, call value-amount. And:

$$v^{ij}_{m} = \omega^{ij}_{m} \bullet tf^{ij}_{m} \bullet \lambda^{m}_{ij} \tag{2}$$

Definition 12

Using ωf^{ij}_{m} to represent the transaction amount factor, ranged between [0, 1]. And:

$$\omega f^{ij}_{m} = v^{ij}_{m} \Big/ \sum_{m\in[1,now]} v^{ij}_{m} \tag{3}$$

Transaction frequency factor

In real social network, higher transaction frequency has greater positive effect on the trust.

Definition 13

Using φf^{ij}_{m} to represent the transaction frequency between nodes i and j. And:

$$\varphi f^{ij}_{m} = \varphi^{ij}_{m} \Big/ \sum_{m\in[1,now]} \varphi^{ij}_{m} \tag{4}$$

Local trust degree and its storage

Based on the above analysis, the local trust degree can be defined easily.

Definition 14

Using TL_{ij} to express the local trust degree of node j

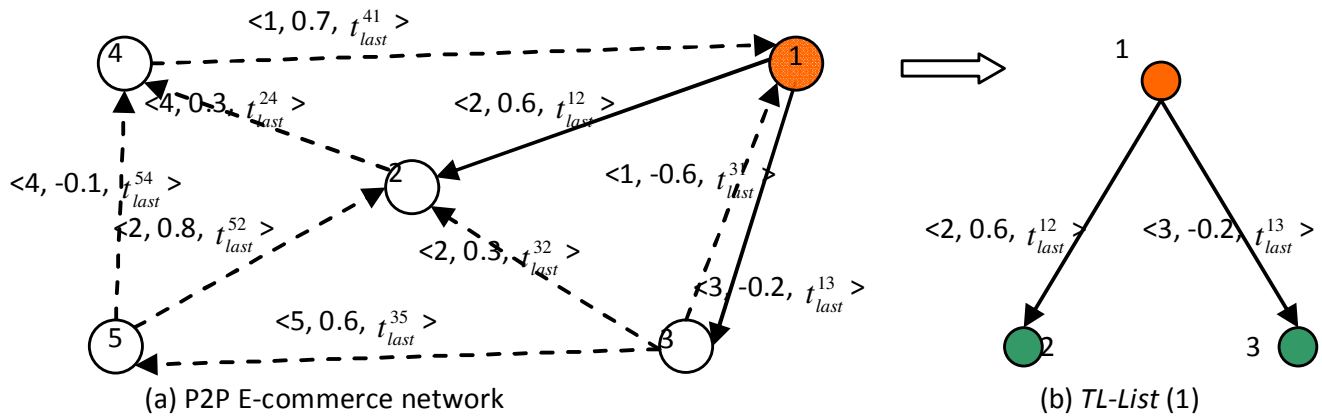


Figure 5. Demonstration of *TL-list(1)*. In the above TL-list network, *TL-list(1)*={<2, 0.6, t_{last}^{12} > , <3, -0.2, t_{last}^{13} >}, it means that in node 1 perspective, node 2 local trust is 0.6, and node 3 local trust is -0.2. Surely, the $TL-list(i)$ will update after node i trades with node j in a new transaction.

which is computed by node i. In other words, from the perspective of node I, the TL_{ij} is the trust expectation of node j. It is composed of appraisals, time stamp, amount and frequency, ranged between [-1, 1]. Assumed the TL_{ij} = 0 at the initialized time and max = s_{ij}+f_{ij}. And:

$$TL_{ij} = \begin{cases} 0\ ,\ time = t_0^i \\ \sum_{m\in[1,\max]} \left(tf_m^{ij} \bullet \omega f_m^{ij} \bullet \varphi f_m^{ij} \bullet \phi_m^{ij} \right) \Big/ \max\ ,\ else \end{cases} \quad (5)$$

However, in the above definition, the TL_{ij} is a product by multiply five decimal fraction ranged between [0, 1] or [-1, 1]. As a result, the TL_{ij} modulus would be too small to be analyzed. In order to facilitate the computing and result analysis, we revise the definition of local trust degree by extracting the TL_{ij} five times but remain the computing order and logic. The revised formula is:

$$TL_{ij} = \begin{cases} 0\ ,\ time = t_0^i \\ \sqrt[5]{\sum_{m\in[1,\max]} \left(tf_m^{ij} \bullet \omega f_m^{ij} \bullet \varphi f_m^{ij} \bullet \phi_m^{ij} \right) \Big/ \max}\ ,\ time = others \end{cases} \quad (6)$$

Now, the distributed storage for local trust degree should be discussed here.

Definition 15

Using *TL-list(i)* to store the local trust degree TL_{ij}. And: $TL-list(i) = \{< j, TL_{ij}, t_{last}^{ij} > | \ j \in I_i\}$.

The *TL-list(i)* is a set of $< j, TL_{ij}, t_{last}^{ij} >$ and stored on node i. In P2PETrust, the TL-list could store all of the local trust information according to the specified P2P routing algorithm. Each node is responsible for its corresponding local trust information, and then all of the TL-list combines into a TL-list network. Figure 5 demonstrates the TL-list.

P2PETRUST: GLOBAL TRUST DEGREE

Local trust has some limitations on evaluating nodes' trust because of the computed history only comes from the related two nodes (nodes i and j) and could not avoid single malicious node's cheating or data attacking. For example, node may revise its history data or give false appraisal to a transaction.

Global trust factors

For the purpose of computing global trust degree, we need to consider impact factors as same as local trust, including global success rate, global frequency and global time factor. The former two factors have been defined in definitions 5(b) and 7(b), thus, the focus is on discussing global time factor.

Like the local time factor, we also use the right triangle (in time factor) to compute the global time factor.

Firstly, it needs to rank time stamps (t_{last}^{ji}) of the local trust history which is from I_i to node i, become a local trust appraisals list $\rho(i) = \{< j, TL_{ji}\ ,\ t_j^i > | \ j \in I_i\}$ where t_j^i is the t_{last}^{ji} from I_i. Assumed the global time factor of the

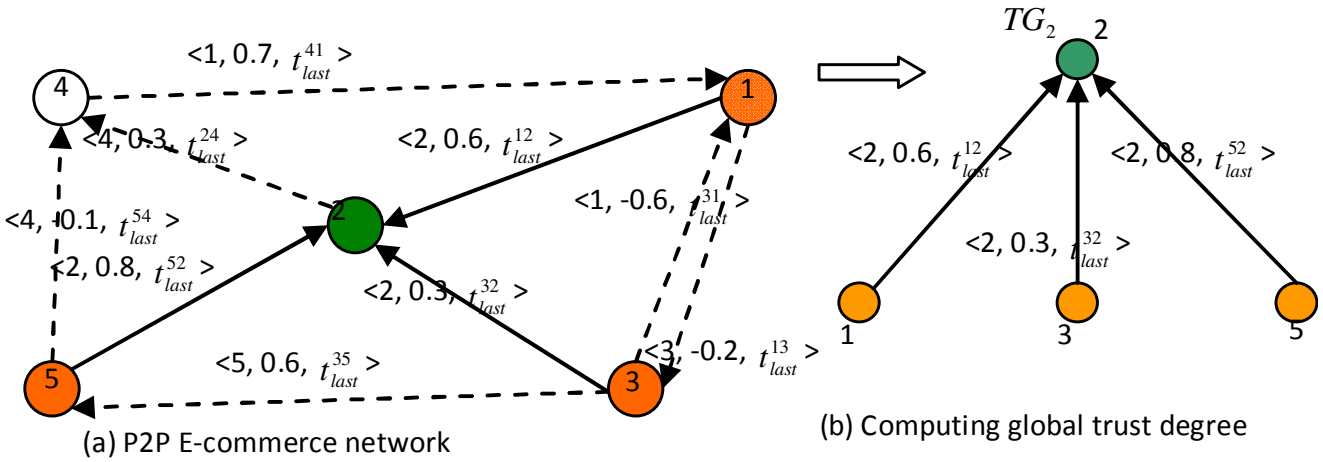


Figure 6. Distributed computing the global trust degree is based on local trust model. In order to compute the global trust degree of node 2, we collect local trust degrees of node 2 from nodes {1, 3, 5}, that is *TL(1,2), TL(3,2) and TL(5,2)*. Then, calculate global trust degree of node 2.

latest time stamp in $\rho(i)$ (maximum number index) is 1, that is $gf^i_{\max} = 1$ (gf is shortcut for global time factor); we can compute the entire global time factor according to the method in time factor.

Definition 17

Using gf^i_j to stand for the global time factor, ranged between [0, 1], and:

$$gf^i_j = \left(t^i_j - t^i_0 \Big/ t^i_{\max} - t^i_0 \right)^2 \tag{7}$$

Global trust degree

Definition 16

Using TG_i to express the global trust degree of node i, which is integrated by all of the local trust degrees from I_i set (defined in definition 4). Figure 6 shows the distributed computing process of the global trust.

As we can see from Figure 6, we should design policy to calculate the global trust degree based on the local trust degrees. This policy should mean that the computing order should not affect the computing result. Iteration computing method and average value method could meet the policy's demand, and we use the latter in this paper.

Via average value method, the global trust degree is multiplied by three decimal fractions which are global success rate factor λ_i^{now}, global frequency factor φ^i_{now} and global time factor gf^i_j. Thus, to facilitate the computing and analysis, we amplify the result by extracting the result three times. Formula for global trust degree (ranged between [-1,1]) is:

$$TG_i = \sqrt[3]{\left(\sum_{c \in I_i} \left(gf^i_c \bullet TL_{ci} \right) \Big/ Count(I_i) \right) \bullet \lambda_i^{now} \bullet \varphi^i_{now}} \tag{8}$$

In the above formula, $Count(I_i)$ is the quantity of the nodes in I_i.

CORRELATION TRUST DEGREE

Unlike the local trust degree, the global trust degree can avoid malicious behaviors from single node. However, it can not defend attacks from team malicious nodes that are cooperated with each other as a team. For example, the global trust degree could not recognize malicious team nodes' high appraisals to each other. Thus, a correlation trust degree is created.

Similarity factor

The correlation of appraisals by two different nodes is used to describe the associated extent of the specified two nodes via computing the history among these two nodes and their common third-party nodes. This situation is similar to social networks in judging a strange person. For example, A did not know B, but both share common friends {C, D}, then A can judge the correlation with B via his friends {C, D}. If the communicated history between

A-{C, D} is similar to the history between B-{C, D}, then we say A and B have very similar correlation.

There are many methods to calculate the similarity between two items, such as cosine similarity, correlation similarity and adjusted cosine similarity (Deng et al., 2003; Benesty et al., 2008; Sarwar et al., 2001).

Cosine similarity is a measure of similarity between two vectors of n dimensions by finding the cosine of the angle between them, often used to compare documents in text mining. In addition, it is used to measure cohesion within clusters in the field of Data Mining. Given two vectors of attributes, A and B, the cosine similarity, θ, is represented using a dot product and magnitude as;

$$Sim(A,B)=\cos(\theta)=\frac{A\bullet B}{\|A\|\bullet\|B\|} \tag{9}$$

Pearson's correlation coefficient is the most typical correlation similarity computing method, usually used in text data mining and information retrieval. The formula is;

$$sim(i,j)=\frac{\sum_{c\in I_{ij}}(R_{ic}-\overline{R_i})(R_{jc}-\overline{R_j})}{\sqrt{\sum_{c\in I_{ij}}(R_{ic}-\overline{R_i})^2}\sqrt{\sum_{c\in I_{ij}}(R_{jc}-\overline{R_j})^2}} \tag{10}$$

Where, R_{ic} is the appraisal of node i, and $\overline{R_i}$ is the average appraisal of node i to I_{ij}. And R_{jc}, $\overline{R_j}$ have the similar meaning as i. Higher absolute value of sim(i, j) means high similarity. In P2PETrust, we calculate the similarity by TL-list. First but most important, converting the TL-list into a local trust degree matrix $R(n\times n)$.

Definition 18

Using $R(n\times n)$ to stand for the local trust degree matrix converted from *TL-list*. Each element r_{ij} in $R(n\times n)$ is;

$$r_{ij}=\begin{cases}TL_{ij}, when\ history(i,j)=true;\\ \varepsilon, when\ history(i,j)=false;\\ 1, when\ i=j.\end{cases} \tag{11}$$

Value $\varepsilon>0$ is a very small non-zero decimal. For example, 0.00001 or else, it is used to facilitate the non-history nodes' computing. And, nodes give themselves a full trust with 1. Then,

$$R=(r_{ij})_{n\times n}=\begin{bmatrix}r_{11} & r_{12} & \cdots & r_{1n}\\ r_{21} & r_{22} & \cdots & r_{2n}\\ \cdots & \cdots & \cdots & \cdots\\ r_{n1} & r_{n2} & \cdots & r_{nn}\end{bmatrix}$$

According to formula (10), $\overline{R_i}$ means the average appraisal of node i made to I_{ij}, thus, $\overline{R_i}=\frac{\sum_{c\in I_{ij}}TL_{ic}}{n}$. Then we can get the formula for calculating similarity of two nodes as;

$$sim(i,j)=\left|\frac{\sum_{c\in I_{ij}}(TL_{ic}-\overline{R_i})(TL_{jc}-\overline{R_j})}{\sqrt{\sum_{c\in I_{ij}}(TL_{ic}-\overline{R_i})^2}\sqrt{\sum_{c\in I_{ij}}(TL_{jc}-\overline{R_j})^2}}\right| \tag{12}$$

The sim(i, j) ranged between [0, 1], and the similarity increases when the value sim(i,j) increased. Nevertheless, it would be error when the denominator in formula (11) is zero. We could not get the similarity at that time. Moreover, it would be trouble when nodes in I_{ij} (which are the common third-party nodes) are so very rare that can not calculate the similarity. To solve these problems, we adjust the similarity formula and assume that sim(i, j) equals to zero when $Count(I_{ij})<\tau$ (τ is a settable threshold value by user). The adjusted formula is:

$$sim(i,j)=\begin{cases}0, when\ \sqrt{\sum_{c\in I_{ij}}(TL_{ic}-\overline{R_i})^2}\sqrt{\sum_{c\in I_{ij}}(TL_{jc}-\overline{R_j})^2}=0\ ,\ or\ Count(I_{ij})<\tau;\\ \left|\frac{\sum_{c\in I_{ij}}(TL_{ic}-\overline{R_i})(TL_{jc}-\overline{R_j})}{\sqrt{\sum_{c\in I_{ij}}(TL_{ic}-\overline{R_i})^2}\sqrt{\sum_{c\in I_{ij}}(TL_{jc}-\overline{R_j})^2}}\right|, else.\end{cases} \tag{13}$$

Correlation trust degree

Correlation trust degree considers the histories similarity between nodes based on the global trust degree.

Definition 19

Using $CorTG_{ij}$ to stand for the correlation trust degree of node j which is evaluated by node i. And:

$$CorTG_{ij}=\begin{cases}\min(0, TG_j) & (sim(i,j)=0)\\ sim(i,j)\bullet TG_j & (sim(i,j)>0)\end{cases} \tag{14}$$

Due to the similarity in $CorTG_{ij}$, the correlation trust

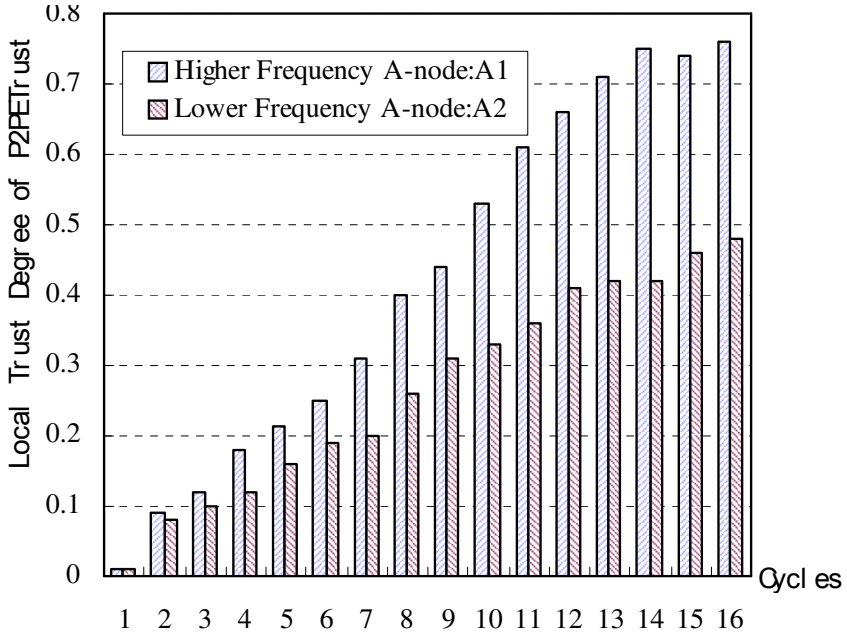


Figure 7. This simulation is aiming to test the rightness of local trust degree. By the results obviously, A1 has more local trust degree than A2, since A1 has higher frequency. The results confirm the definition of local trust, and prove that the local trust computing method of P2PETrust is valid and right.

degree is more objective than global trust degree. As can be imagined, the similarity between normal nodes and team malicious nodes would be very small for the common transaction histories would be less.

SIMULATIONS AND ANALYSIS

In order to verify the rightness of P2PETrust, we design a simulation platform by C# programming language, and simulate the P2P environment with multi processes and threads. Three kinds of nodes are designed.

(1) Class A. It describes the normal and 'good' nodes that provide correct appraisals and good service in P2P E-commerce system.
(2) Class B. It describes single malicious node in P2P E-commerce system, providing false service and making false appraisals. But this kind of node does not work coordination with other malicious nodes.
(3) Class C. It describes team malicious nodes in P2P E-commerce system, providing dishonest service and giving incorrect appraisals to A or B nodes. At the same time, these C nodes overstate appraisals to each other.

Simulation 1: Local trust degree of P2PETrust

The purpose of this simulation is to verify the validity of local trust factors, such as time stamp, transaction amount, frequency and appraisals. To facilitate the simulation, all of the tested nodes are generated by Class A.

Mark one node as 'A1' and another as 'A2', and compute the local trust degree both of A1 and A2. In the course of simulation, A1 is always doing transaction in high frequency and A2 is relatively lower. Meanwhile, A1 and A2 have the same transaction amount. Figure 7 shows the statistics result after cycles simulations.

Simulation 2: Global trust degree of P2PETrust

Here we simulate the transaction activities of a real P2P E-commerce system. Firstly, generate 200 A-nodes, 50 B-nodes and 100 C-nodes. Then put 2000 trusted file abstracts (with name, key words size and virtual price) on A-nodes, and put 1000 fake file abstracts on C-nodes, and put 500 fake file abstracts on B-nodes. Mark one of the A-nodes as 'A1', one of the B-nodes as 'B1' and one of the C-nodes as 'C1'. Initiating the simulation, all of the nodes transacts in a random P2P environment according to a random commerce demand lists. Observe the trends of global trust degrees of nodes A1, B1, and C1, shown in Figure 8.

Simulation 3: Correlation trust degree of P2PETrust

This simulation has same environment as the above

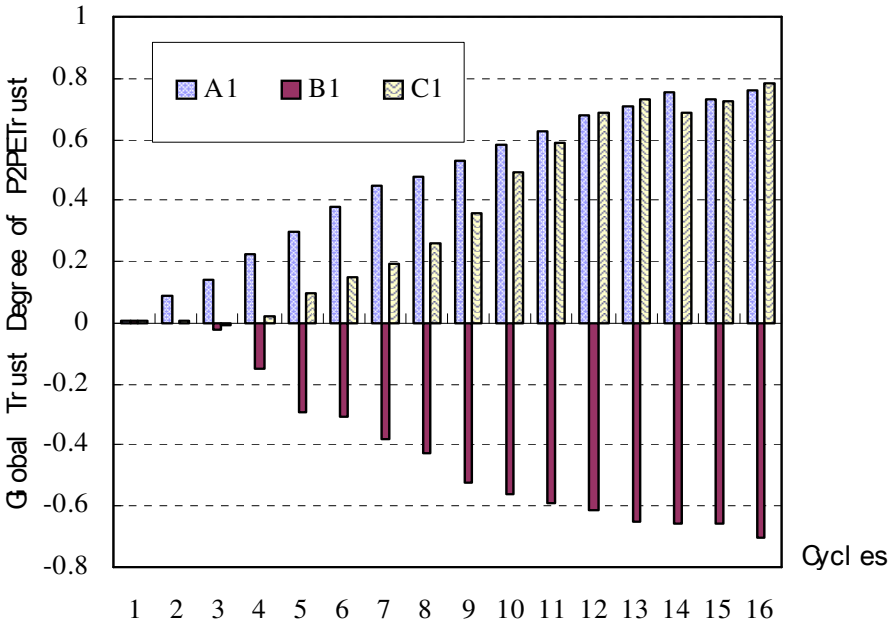


Figure 8. This simulation is to test the global trust degree in P2PETrust. From the results of three specific nodes A1, B1 and C1, As shown, the global trust degree of B1 decreased quickly while that of A1 and C1 both increased. It is owing to the fact that, B1 is 'single' while A1 (normal nodes) and C1 (team malicious node) has their own team. However, this result implies the global trust degree in P2PETrust could not resist team malicious behaviors although it can recognize the single malicious nodes.

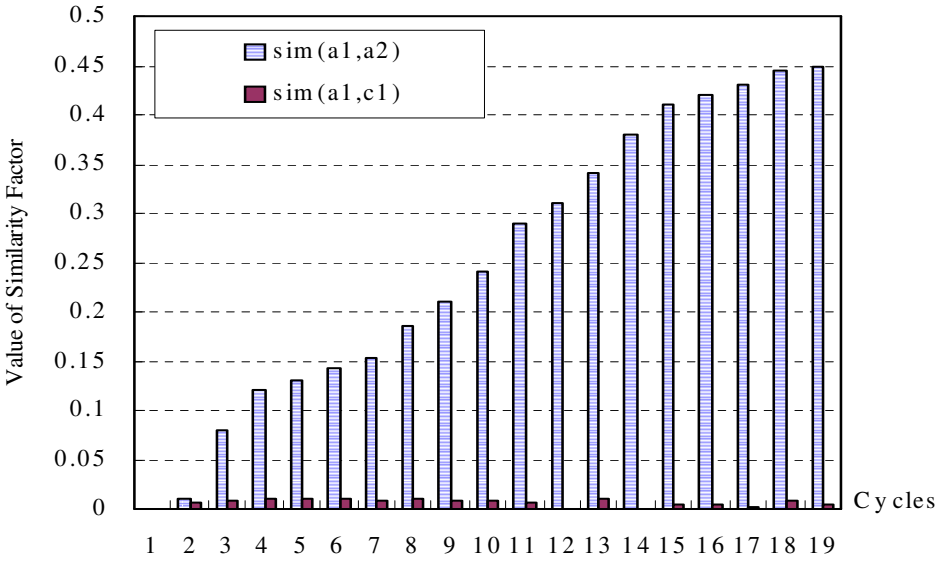


Figure 9. This is the simulation of similarity factor for correlation trust model of P2PETrust. As we can see from the chart, with the time going on, the similarity between A1-A2 was increasing rapidly while that of A1-C1 always stayed in a low level. It proved that the similarity factor is reasonable.

simulation 2. In that environment, mark another node of class A as 'A2'. Initiate the simulation again to watch correlation trust degree between nodes A1 and A2, and also correlation trust degree between nodes A1 and C1.

Firstly, observe the similarities of A1 – A2 and A1 – C1. Figure 9 shows the result. Meanwhile, observe the correlation trust degree of A1 – A2 and that of A1 – C1. Figure 10 shows the results. As shown, the CorTG(A1,

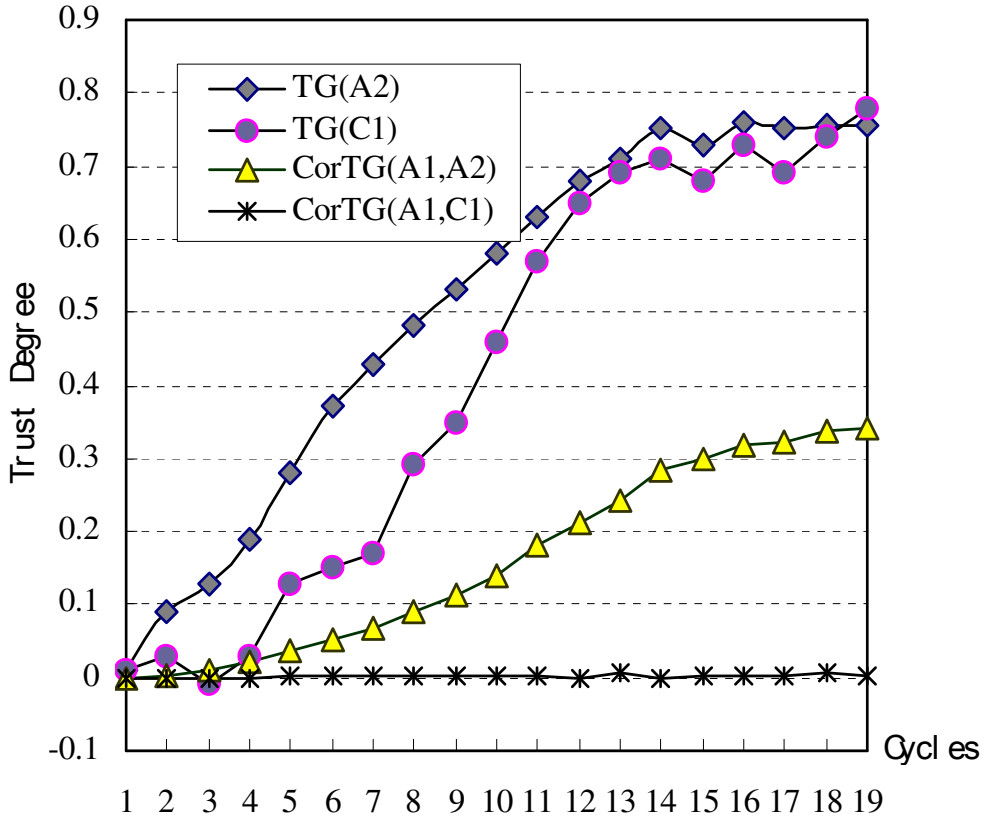


Figure 10. This is the simulation to test the correlation trust degree. In the results, A2 and C1 both had high global trust degree (TG(A2) and TG(C1)) because of the impact from team. However, the correlation trust degree of the two were quiet different because of the similarity factor.

A2) is increasing always but CorTG(A1,C1) is very low to zero even. Results showed that the correlation trust degree is objective, and can resist malicious behaviors from team malicious nodes.

Simulation 4: Comparing P2PETrust with other trust model

In this simulation, we compare the correlation trust degree model in P2PETrust model with EigenTrust (Kamvar et al., 2003), Jiang (Jiang and Li, 2007), NBRTrust (Tan et al., 2009) and random model.

Assumed *TotalCount* is the total transaction times and BC_Count is the total transaction times with B-nodes or C-nodes which are malicious nodes. Then, the download-resisting performance from malicious nodes could be described as $dr = \frac{BC\_Count}{TotalCount}$. For example, a node from A-nodes has 100 transactions, and 20 transactions with B-nodes or C-nodes, then the download-resisting performance is *dr* = 20%. We do statistics of the download-resisting performance for these trust models. Figure 11 shows the results.

As we can see, the random model had the worst performance while P2PETrust had the best convergence. EigenTrust and Jiang had good convergences also but they could not resist team malicious nodes. This result proved the P2PETrust's rightness and is effective.

CONCLUSION

In order to compute the trust degree in P2P E-commerce system, this paper presents a new distributed trust model named P2PETrust. The P2PETrust includes three sub models which are local trust, global trust and correlation trust models. The local trust degree is related to local transaction appraisals, time stamps, transaction amount and frequencies between two specific nodes. Local trust model has fast trust convergence in very normal condition but could not recognize malicious behaviors. Global trust degree is based on local trust model, and it combines global factors such as global time stamps, frequencies and success rates. Global trust degree can resist single malicious nodes' behaviors efficiently but does not work on team malicious nodes' environments. Correlation trust model imports similarity factor based on global trust model and it can compute the similarity of

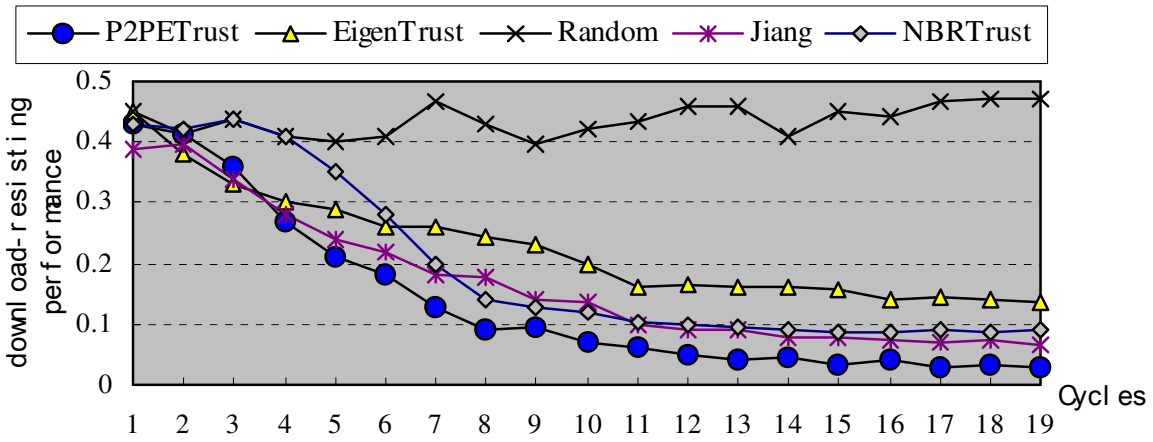


Figure 11. This is the simulation for download-resisting performances by comparing some typical trust models.

nodes histories data to avoid team malicious nodes' behaviors. Simulations proved the rightness, and can resist single and team malicious nodes.

However, it will be a long time to study the trust model for a distributed system. There are many problems waiting to be improved and solved. For examples, how to improve the distributed communication algorithms for P2P Trust model? How to use more trust principals of social networks for P2P Trust model? Can we compute the trust degree by transaction content in distributed networks? These problems will conduct our future work.

ACKNOWLEDGEMENT

This work is supported by the National Natural Science Foundation of China under Grant No. 61070162, No. 71071028, No. 60802023 and No. 70931001; the Specialized Research Fund for the Doctoral Program of Higher Education under Grant No. 20100042110025 and No. 20070145017; the Fundamental Research Funds for the Central Universities under Grant No. N090504003 and No. N090504006.

REFERENCES

Aberer K, Despotovic Z (2001). Managing trust in a peer-to-peer information system. In proceedings of the 10th International Conference on Information and Knowledge Management, ACM Press, Atlanta, GA, United States, pp. 1-7.

Benesty J, Chen JD, Huang YT (2008). On the importance of the Pearson correlation coefficient in noise reduction. IEEE Transactions on Audio, Speech and Language Processing,16(4): 757-765.

Deng AL, Zhu YY, Shi BL (2003). A Collaborative Filtering Recommendation Algorithm Based on Item Rating Prediction. J. Software., 14(9): 1621-1628.

Dou W, Wang HM, Jia Y (2004) . A recommendation-based peer-2-peer trust model. J. Software., 15(4): 571-583.

Esaki H (2010). A consideration on R&D direction for future Internet architecture. Int. J. Commun. Syst., 23(6-7): 694-707.

Gambetta D (2000). 'Can we trust Trust?' in Gambetta, Diego (ed.) Trust: Making and Breaking Cooperative Relations, electronic edition, Department of Sociology, University of Oxford, Basil Blackwell.

Jiang SX, Li JZ (2007). A Reputation-Based Trust Mechanism for P2P E-Commerce Systems. J. Software.,18(10): 2551-2563.

Jøsang A, Bhuiyan T (2008). Optimal trust network analysis with subjective logic. In proceedings of SECURWARE, pp. 179-184.

Jøsang A, Hayward R, Pope S (2006). Trust network analysis with subjective logic. In proceedings of ACSC, pp. 85-94.

Kamvar SD, Schlosser MT, GarciaMolina H (2003). The EigenTrust algorithm for reputation management in P2P networks. In proceedings of ACM WWW, pp. 640-651.

Li YJ, Dai YF (2010). Research on Trust Mechanism for Peer-to-Peer Network. Chinese J. Comput., 33(3): 390-405.

Marsh S (1994). Formalising Trust as a computational concept. Ph.D. Thesis, Department of Computing Science and Mathematics, University of Stirling.

McKnight DH, Chervany NL (1996). The Meanings of Trust. In: University of Minnesota Available:http://www.misrc.umn.edu/wpaper/wp96-04.htm

Mekouar L, Iraqi Y (2006). Boutaba R.: Peer-to-peer's most wanted: malicious peers. Comput. Netw., 50(4): 545-562.

Sarwar B, Karypis G, Konstan J, Riedl J (2001). Item-Based collaborative filtering recommendation algorithms. In proceedings of the 10th International World Wide Web Conference, pp. 285-295.

Tan ZH, Wang H, Cheng W, Chang GR (2009). A Distributed Trust Model for P2P Overlay Networks Based on Correlativity of Communication History. J. Northeastern University (Natural Science), 30(9): 1245-1248.

Touch J (2001). Overlay networks. Computer Networks, 36(2):115-116.

Wang G, Wu J (2010). Multi-dimensional evidence-based trust management with multi-trusted paths. Future Generation Computer Systems, doi:10.1016/j.future.2010.04.015, in press.

Xiong L, Liu L (2004). PeerTrust: supporting reputation-based trust for Peer-to-Peer electronic communities. IEEE Transactions on Knowledge and Data Engineering,16(7). 843-857.

Zhang P, Helvik BE (2010). Modeling QoS in P2P File-Sharing with Benign and Malicious Peers by Stochastic Activity Networks. In proceedings of the 7th IEEE Consumer Communications and Networking Conference, IEEE Computer Society Press, Las Vegas, NV, United states, pp. 461-465.

Zhang Q, Sun Y, Liu Z, Zhang X, Wen XZ (2005). Design of a distributed P2P-based grid content management architecture. In proceedings of the 3rd Annual Communication Networks and Services Research Conference.IEEE Computer Society Press, Halifa, NS, Canada, pp. 339-344.