

## Application of Fuzzy Comprehensive Evaluation Method in Trust Quantification

**Shunan Ma**

*College of Computer Science and Technology, Beijing University of Technology  
Beijing 100124, China  
E-mail: mashunan@emails.bjut.edu.cn*

**Jingsha He**

*School of Software Engineering, Beijing University of Technology  
Beijing 100124, China  
E-mail: jhe@bjut.edu.cn*

**Xunbo Shuai**

*Research Institute of Petroleum Exploration & Development, PetroChina  
Beijing 100083, China  
E-mail: shuaixb69@petrochina.com.cn*

### Abstract

Trust can play an important role for the sharing of resources and information in open network environments. Trust quantification is thus an important issue in dynamic trust management. By considering the fuzziness and uncertainty of trust, in this paper, we propose a fuzzy comprehensive evaluation method to quantify trust along with a trust quantification algorithm. Simulation results show that the trust quantification algorithm that we propose can effectively quantify trust and the quantified value of an entity's trust is consistent with the behavior of the entity.

*Keywords:* trust, credit, reputation, quantification, fuzzy comprehensive evaluation

### 1. Introduction

In open network environments, since there is no central authority to monitor and punish misbehaving entities, malicious nodes can easily cause degradation of service<sup>1</sup>. Trust can play an important role in open network environments to help ensure quality of service. Moreover, trust can be a useful tool for reducing the complexity of making decisions in addition to providing security<sup>2</sup>. Blaze was the first to use "trust management" for solving network security problems<sup>3</sup>. Trust can be used to help entities distinguish between good partners and bad ones. Thus, trust has received a great deal of attention in recent

years and has been used as a mechanism to ensure network security. As part of trust management, the issue of trust computation has also received a lot of attention.

There has been some research work on trust computation. Blaze et al. and Liu et al. proposed some trust computation and evaluation methods<sup>4,5</sup>. Azzedin et al. applied trust to manage grid resources in which the trust value of an entity is calculated by combining direct and recommended trust values<sup>6</sup>. In EigenTrust<sup>7</sup>, local trust values from all peers are collected to calculate the global trust value of a given peer. In addition, a binary rating function is adopted in which positive one means "satisfactory" and zero or negative one means

“unsatisfactory”. Duma et al. proposed dynamic trust metrics for peer-to-peer system in which recommendation trust and penalty factor are used for trust calculation<sup>8</sup>.

Wang et al. proposed several trust metrics for trust evaluation in a decentralized environment<sup>9</sup> in which a trust value is a probabilistic value between [0, 1]. Yuan et al. proposed a trust value calculation function based on uncertainty reasoning theory<sup>10</sup>. Christian et al. suggested a method for trust calculation based on the contexts and contents of the semantic Web<sup>11</sup>.

Sun et al. proposed a distributed trust management scheme in which trust computation is based on observation and recommendation using mathematical formulas and trust values are updated periodically based on current behavior<sup>12</sup>. Gutscher proposed a new computational model for trust that integrates authentication verification into the process of trust evaluation<sup>13</sup>. Liang et al. proposed to change the weights of direct and recommendation trust value as a means of changing the role of them in trust evaluation<sup>14</sup>. However, we notice that the fuzziness nature of trust has not been considered in any of the above work.

Trust quantification is very important for dynamic trust management. It is difficult to express trust using traditional mathematical theory since trust can be subjective and uncertain. Fuzzy theory can thus be used to handle uncertainty effectively and has been widely used in many fields such as natural science, social science and management<sup>15-20</sup>.

Fuzzy comprehensive evaluation method uses fuzzy mathematics to make scientific evaluation by considering the various factors that could influence a certain thing. Thus, fuzzy comprehensive evaluation is an efficient method to evaluate things that are affected by various factors and has been widely applied in evaluation in recent years<sup>21-25</sup>. In this paper, we propose a trust quantification algorithm based on fuzzy comprehensive evaluation by considering the fuzziness and uncertainty nature of trust.

The rest of this paper is organized as follows. In the next section, we present several important concepts on trust. In Section 3, we apply fuzzy comprehensive evaluation method to quantify trust and propose a trust quantification algorithm. In Section 4, we describe an implementation of the trust quantification algorithm. In Section 5, we present some experiment results and the

analysis. Finally, in Section 6, we conclude this paper in which we also discuss the direction of our future work.

## 2. The Concepts

Trust can meet the needs of dynamism and openness of network environments. To reflect the dynamism of trust, we first describe some concepts.

**Trust quantification:** a way of expressing the trust of an entity on another with a numerical value.

**Credit:** the evaluation result of the owner of an object on a subject based on behavior information after certain interactions between the subject and the object have completed, which is denoted as  $C$ .

**Reputation:** the degree of trust of an object on a subject based on the credit values of multiple times, which is denoted as  $R$ .

Credit and reputation are the dynamic ways of expressing trust. The relationship among trust, credit and reputation is illustrated in Fig. 1.

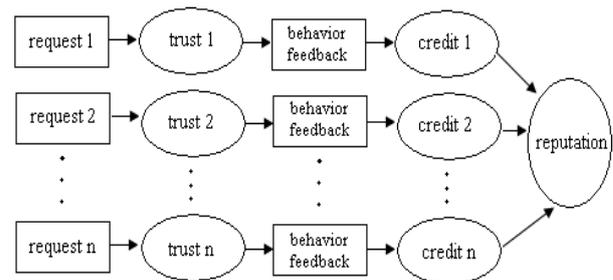


Fig. 1. Relationship among trust, credit and reputation

### 2.1. Computation of Credit and Reputation

We use the following two formulas to compute credit and reputation:

$$C = S \times T \tag{1}$$

$$R = \sum_{i=1}^n W_i \times C_i \quad \text{where} \quad \sum_{i=1}^n W_i = 1 \tag{2}$$

In Formula (1),  $T$  is a trust value and  $S$  is the satisfaction degree of the owner of an object on a subject in an interaction, and  $S \in [0,1]$ .

In Formula (2),  $n$  is the number of credits,  $C_i$  is the credit value at the  $i$ th time and  $W_i$  is the weight of  $C_i$  for computing reputation.

Every object keeps an evaluation table in which to record the subjects that have interacted with it. The

evaluation table mainly includes: subject's ID, access time, trust value, feedback satisfactory degree, credit, reputation. When a subject issues a request to access an object, the object will search its evaluation table using subject's ID. If a record is found, a new access evaluation record for the current access is inserted on top of the most recent one for the same subject. If no record is found for the subject, it means that this is the first time that this subject makes access to the object. Then, a new evaluation record is added into the evaluation table for this subject. Only the object can write and modify its own evaluation table whereas the table can be read by all other objects.

The main features of the evaluation table are as follows:

(1) It records the behavior of every subject that has interacted with the object and manages subjects' trust, credit and reputation values.

(2) It stores subjects' past trust and credit information that are used to compute subjects' reputation values. The reputation computation method proposed in this paper uses the credit values in the evaluation table.

(3) It can be shared with all other objects. When another object tries to quantify the trust of a subject, the evaluation table can provide the data for reference.

(4) It stores evaluation records in the order of access times with the most recent one being placed in the first entry of the table.

## 2.2. Weight Determination

Suppose that a subject has accessed an object  $n$  times. The corresponding credit values of the subject are thus  $C_1, C_2, \dots, C_n$ , respectively, and the reputation value is

$R = \sum_{i=1}^n W_i \times C_i$ , in which  $W_i$  is the weight of the  $i$ th credit

value and  $W_i \in [0,1]$ . Based on the attenuation characteristic of trust, for any two given credit values  $C_i$  and  $C_j$ , if  $i < j$ , then  $W_i > W_j$ .

Assume that  $R_{n+k}$  is the subject's new reputation, where  $k=1, 2, \dots, n$ . We select a set of weights  $W_1, W_2, \dots, W_n$  that satisfy the following formula:

$$F(W_1, W_2, \dots, W_n) = \min \left( \sum_{j=k+1}^{n+k} W_{j-k} \times C_{j-k} - R_{n+k} \right) \quad (3)$$

In Formula (3), it requires that  $n$  credit weights be derived according to  $n$  history credit samples and used along with the credit values to compute reputation. If the

calculated reputation value is close to the stored reputation value, then these credit weights are feasible.

Genetic algorithm, first introduced by John Holland<sup>26</sup> in the early 1970s, is a powerful stochastic algorithm based on the principles of natural selection and natural genetics. Genetic Algorithm is one of optimization algorithms, which is invented to mimic some of the processes observed in natural evolution. It has been applied in machine learning and optimization problems. In this paper, we use the genetic algorithm<sup>27</sup> to get a set of credit weights in which the fitness function and the constraint are shown in formula (4) below:

$$\begin{cases} F = \min \left( \sum_{i=1}^n W_i \times C_i - R_n \right) \\ s.t. \quad \sum_{i=1}^n W_i = 1 \end{cases} \quad (4)$$

## 3. The Trust Quantification Algorithm

### 3.1. Fuzzy Comprehensive Evaluation

Fuzzy comprehensive evaluation, which provides a high level of confidence in decision-making based on fuzzy logic, is a branch of artificial intelligence. It classifies or distinguishes things by means of analyzing fuzzy information as much as possible. By considering the various factors that influence a certain thing, fuzzy comprehensive evaluation method uses fuzzy mathematical methods and makes a scientific evaluation of its merits and shortcomings<sup>21</sup>.

Fuzzy comprehensive evaluation uses some concepts in fuzzy mathematics to evaluate actual problems which are comprehensive and complex. Fuzzy comprehensive evaluation is one application of fuzzy mathematical method. The basic procedure is as follows: (1) identifying the factors that can be used to judge the target set and evaluation set; (2) determining the weights of the factors and their membership degree vector, respectively, and obtaining a fuzzy evaluation matrix; (3) operating the fuzzy evaluation matrix of factors and the fuzzy weight vector to normalize the result. The final result is the fuzzy evaluation result. Although this method uses fuzzy mathematics theory, the modeling process is simple, easy to understand and effective to judge complex problems. It can be applied to many fields<sup>28</sup>. In

this paper, we use fuzzy comprehensive evaluation method to quantify trust.

Following is the procedure for completing the fuzzy comprehensive evaluation:

(1) Establishing expert set F and evaluation set M

$$F = \{1, 2, \dots, n\} .$$

$M = \{e_1, e_2, \dots, e_m\}$  , where  $e_j(j=1,2,\dots,m)$  denotes the  $j$ th evaluation grade.

(2) Establishing fuzzy evaluation matrix R

After giving single factor judgment evaluation on factor  $t_i$  and determining its grade of membership (which is expressed as  $r_{ij}$ ) to judgment grade  $e_j$ , we can get the result of a set of single judgment factor  $r_i = (r_{i1}, r_{i2}, r_{i3})$ . Then the set of total judgment factors (in the amount of  $m$ ) will create a judgment matrix R.

$$R = \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1m} \\ r_{21} & r_{22} & \dots & r_{2m} \\ \dots & \dots & \dots & \dots \\ r_{n1} & r_{n2} & \dots & r_{nm} \end{bmatrix}$$

in which  $r_{ij}$  denotes subsection of the  $i$ th expert to  $j$ th grade.

In this paper, subsection can be expressed as:

$$r_{ij} = \mu(x_i) = \begin{cases} 1 & 0 < x \leq a_1 \\ \frac{x_i - a_1}{a_2 - a_1} & a_1 < x \leq a_2 \\ 0 & x > a_2 \end{cases} \quad (5)$$

in which  $x$  is the most recent credit value of an expert on a subject.

(3) Determining the weight set

In the evaluation calculation,  $E = (e_1, e_2, \dots, e_n)$  denotes the weight coefficient of all factors on the significance of comprehensive evaluation. The weight of an expert  $i$  is then

$$E_i = \frac{C_i / R_i}{\sum_{j=1}^n (C_j / R_j)} \quad (6)$$

(4) Fuzzy comprehensive evaluation

The results of fuzzy comprehensive evaluation are as follows:

$$\begin{aligned} B &= E \cdot R \\ &= (e_1, e_2, \dots, e_n) \cdot \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1m} \\ r_{21} & r_{22} & \dots & r_{2m} \\ \dots & \dots & \dots & \dots \\ r_{n1} & r_{n2} & \dots & r_{nm} \end{bmatrix} \\ &= (b_1, b_2, \dots, b_m) \end{aligned} \quad (7)$$

Based on the principle of maximum subsection, we select the interval to which maximum subsection scale corresponds.

### 3.2. Trust Quantification Procedure

We now use the fuzzy comprehensive evaluation method to quantify trust in which the expert set  $F = \{1, 2, \dots, n\}$  and the evaluation set  $M = \{\text{distrust, distrust but not certain, trust but not certain, trust}\}$ . For each access request, we select an object node and its neighboring nodes that have interacted with the subject as the experts. The trust evaluation table is shown in Table 1.

Table 1. Trust evaluation table

Expert	Weight	Credit	Reputation	Evaluation level			
				trust	trust but not certain	distrust but not certain	distrust
Object	$E_1$	$C_1$	$R_1$				
Neighbor 1	$E_2$	$C_2$	$R_2$				
...	...	...	...				
Neighbor n-1	$E_n$	$C_n$	$R_n$				

In this paper, we divide the evaluation set into four levels: {distrust, distrust but not certain, trust but not certain, trust} which corresponds to the intervals of  $\{[0,0.25],[0.25,0.5],[0.5,0.75],[0.75,1]\}$  respectively.

The trust quantification algorithm consists of the following steps:

Step 1: Select n-1 neighboring nodes that have interacted with the subject and get the most recent credit and reputation values from these nodes for the subject. Denote the credit and reputation values as  $Credit_i$  and  $Reputation_i$ , respectively. If an object has not had any

interaction with the subject, both values are 0, namely,  $Credit_i=Reputation_i=0$ .

Step 2: Compute the subjection of credit to trust interval using formula (5). Then establish evaluation matrix R.

Step 3: Compute the weight of every expert using formula (6).

Step 4: Carry out fuzzy comprehensive evaluation that generates the result  $B = (b_1, b_2, \dots, b_m)$ .

Step 5: Select the interval  $[c, d]$  that corresponds to  $\max(b_j)$  based on the principle of maximum subjection.

Step 6: The trust quantification value is thus  $Trust = c + (d - c) \times b_j$ .

Trust quantification algorithm flow is shown in Fig. 2.

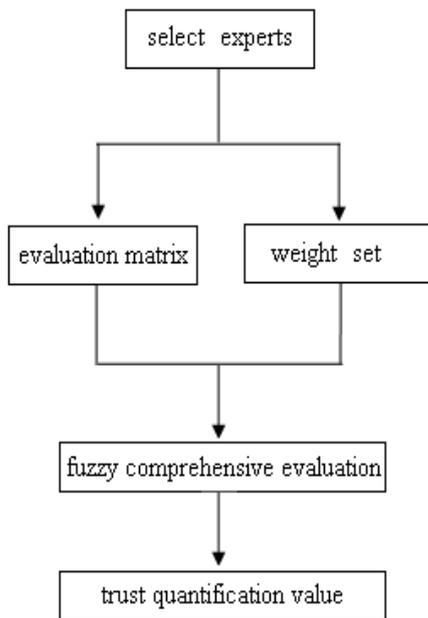


Fig. 2. Trust quantification algorithm flow

### 3.3. Algorithm Analysis

Following is our convergence analysis of the trust quantification algorithm.

Since  $\forall E_i \in E$  and from formula (6),  $E_i = \frac{C_i / R_i}{\sum_{j=1}^n (C_j / R_j)}$ , therefore,  $E_i \in [0,1]$ . Since  $\forall r_{kl} \in R$

and from the algorithm,  $r_{kl} = \mu(x_k)$  and  $T_l = (a_1, a_2]$ , therefore,  $r_{kl} \in T_l$ . We can thus see from formula (5) that  $r_{kl} \in [0,1]$ . Since  $B = E \bullet R$ ,  $\forall b_m \in B$ ,

$$b_m = [e_1, e_2, \dots, e_n] \bullet [r_{1m}, r_{2m}, \dots, r_{nm}]^T, \quad \text{namely,}$$

$$b_m = \sum_{p=0}^n (e_p r_{pm}).$$

Let  $L = \min(r_{1m}, r_{2m}, \dots, r_{nm})$  and  $H = \max(r_{1m}, r_{2m}, \dots, r_{nm})$  where  $L \in [0,1]$  and  $H \in [0,1]$ . Then,

$$\sum_{p=0}^n (e_p L) \leq b_m \leq \sum_{p=0}^n (e_p H), \quad \text{namely, } b_m \in [L, H].$$

We can conclude that  $b_m \in [0,1]$ . Thus, the trust quantification algorithm converges.

Following is time complexity analysis of the trust quantification algorithm. According to algorithm process, the efficiency of the algorithm is determined by expert set number  $n$  and evaluation set number  $m$ . We can see from formula (6), the complexity of expert weight is  $o(n)$ . The complexity of evaluation matrix is  $o(m \times n)$ . According to formula (7), the time complexity of the algorithm is  $o(m \times n^2)$ .

### 4. Implementation of the Algorithm

First of all, selection of experts is very important in our trust quantification method.

In the evaluation, we maintain two data structures: one is evaluation table, which is described in section 2.1, the other is access record table. An access record table is created for every subject to record the information of objects accessed by the subject. This table includes: object's ID, access time and the address of an object's evaluation table. Every subject has an access record table but has neither "write" nor "modify" permissions to its access record table. The objects that interact with the subject can "read", "write" and "modify" their related records in the subject's access table.

When a subject issues an access request to an object, the object will assign to the subject corresponding permissions after trust quantification. Before the subject accesses the object, the object would search the subject's access record table to find a previous access record. If such a record is found, the object will update it. Otherwise, it is the first access by the subject and the object would add its ID and its evaluation table address into this subject's access record table. Then, the subject makes access to the object. Finally, the subject's trust, credit and reputation information is added into the object's evaluation table.

Fig. 3 shows the relationship between subject D’s access record table and objects’ evaluation tables.

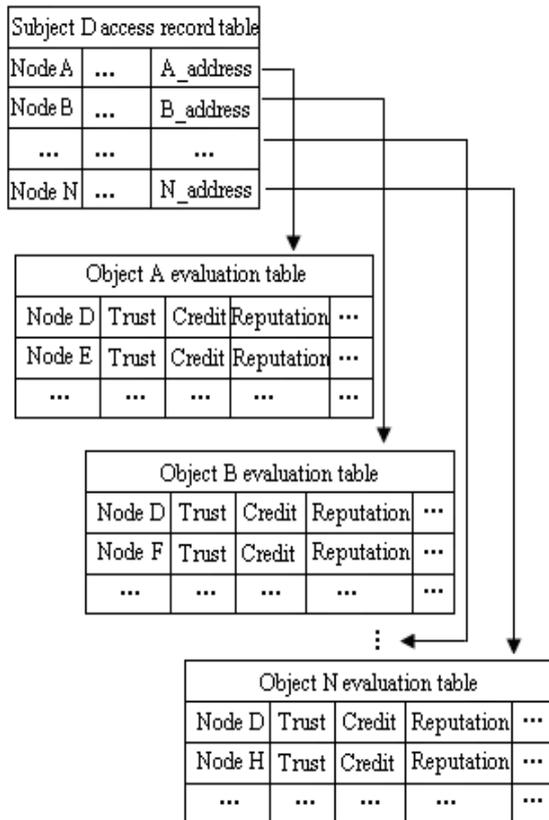


Fig. 3. Relationship between a subject’s access record table and objects’ evaluation tables

Suppose that we use the fuzzy comprehensive evaluation method to quantify trust in which the number of experts is  $n$  ( $n \geq 1$ ) and the number of records in a subject’s access record table is  $m$ , namely, the subject has accessed  $m$  objects. An object would use a subject’s access record table to select the experts with the following three scenarios:

- (1) If  $m=0$ , it means that the subject has not accessed any objects. The subject’s trust quantification value is initialized to 0.5.
- (2) If  $0 < m < n$ , then  $n = m + 1$ . All the objects are selected as the experts. To lower the risk of malicious nodes, create a “virtual expert” with credit and reputation values between 0.5 and 1, respectively.
- (3) If the subject accessed the object in the past, the object is selected as an expert. If  $n \leq m < 2n$ , randomly select  $n$  objects as the experts. If  $m \geq 2n$ , based on the access time, determine the  $2n$  objects that the subject

most recently accessed and then randomly select  $n$  objects from them as experts.

The procedure for an object to use the access record table and the evaluation table is shown in Fig. 4.

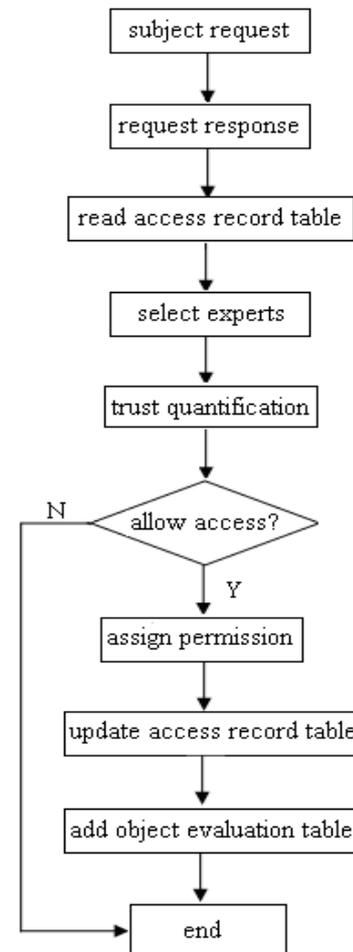


Fig. 4. Procedure for an object to use the access record table and the object evaluation table

## 5. An Example and Experiment Analysis

### 5.1. The Example

In the example, we divide the evaluation set into four levels: {distrust, distrust but not certain, trust but not certain, trust} that correspond to the intervals of  $\{[0,0.25],[0.25,0.5],[0.5,0.75],[0.75,1]\}$ , respectively.

Suppose that the number of experts is  $n=5$  which are the object and four other objects that have interacted with the subject. Denote the objects as: *Object*, *Object*<sub>1</sub>, *Object*<sub>2</sub>, *Object*<sub>3</sub> and *Object*<sub>4</sub>. The most recent credit and

reputation values from these objects for the subject are shown in Table 2.

Table 2. The most recent credit and reputation values

	Object	Object <sub>1</sub>	Object <sub>2</sub>	Object <sub>3</sub>	Object <sub>4</sub>
Credit	0.85	0.74	0.69	0.83	0.75
Reputation	0.91	0.81	0.74	0.87	0.70

Using formula (5), we compute the fuzzy relation matrix R.

$$R = \begin{bmatrix} 0 & 0 & 0 & 0.4 \\ 0 & 0 & 0.96 & 1 \\ 0 & 0 & 0.76 & 1 \\ 0 & 0 & 0 & 0.32 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

Using formula (6), we compute the weight fuzzy relation matrix E.

$$E = [0.19 \ 0.19 \ 0.19 \ 0.20 \ 0.23]$$

The results of the fuzzy comprehensive evaluation are as follows:

$$B = E \bullet R = [0.19 \ 0.19 \ 0.19 \ 0.20 \ 0.23] \bullet \begin{bmatrix} 0 & 0 & 0 & 0.4 \\ 0 & 0 & 0.96 & 1 \\ 0 & 0 & 0.76 & 1 \\ 0 & 0 & 0 & 0.32 \\ 0 & 0 & 1 & 1 \end{bmatrix} = [0 \ 0 \ 0.56 \ 0.73]$$

Based on the principle of maximum subjection, the subject's trust level is trust, and the trust quantification value is  $T = 0.75 + (1 - 0.75) \times 0.73 = 0.93$ .

### 5.2. Experiment Analysis

In order to verify the validity of the trust quantification algorithm, we have performed some simulation developed in Java in which we assume that there are two subjects and ten objects.

Suppose there are two subjects, one is honest and the other one is hostile. In order to show dynamism of the trust quantification algorithm and the algorithm's fast response to the change of trust, the trust quantification value for the honest subject is randomly initialized to a value between 0.5 and 1 and that for the hostile subject is randomly initialized to a value between 0.9 and 1. After each interaction, the satisfaction degree of the object to

the honest subject is  $S \in [0.85, 1]$  and that of the object to the hostile subject is  $S' \in [0, 0.4]$ .

We use an experiment to determine the number of experts. The experiment randomly generates ten trust values for the honest subject and ten trust values for the hostile one. Monte Carlo Method<sup>29,30</sup> is used to evaluate the number of experts. The experiment executes ten times. Then we average out the twenty Monte Carlo evaluated values and get the result for the number of experts which is four. Consequently, we select the most recent four credit values to compute reputation.

Based on the sequence of access time, we use genetic algorithm to get the credit weights for the computation of reputation.

Fitness function of genetic algorithm can be constructed based on formula (4) with binary encoding scheme. The uniform crossover probability  $p_c$  of genetic algorithm is 0.45, the point mutation probability  $p_m$  of genetic algorithm is 0.1, and the genetic alternation is 100. We randomly search for twenty generated sequences and then average them out. Based on the sequence of access time, the credit weights in reputation computation are:  $W_1=0.482$ ,  $W_2=0.246$ ,  $W_3=0.143$ ,  $W_4=0.129$ .

The number of experts that participate in subject evaluation is four. After each access, satisfaction degree is randomly created in the satisfaction degree interval. The two subjects randomly access the object nodes 30 times. The experiment executes 100 times from which we randomly extract one for illustration here. The changes of the trust quantification value for the honest and the hostile subjects are shown in Fig. 5 and Fig. 6, respectively.

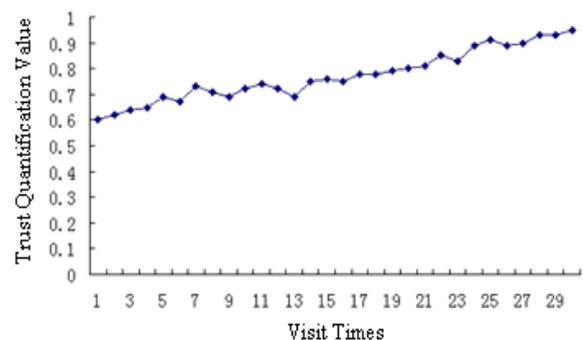


Fig. 5. Change of the trust quantification value for the honest subject

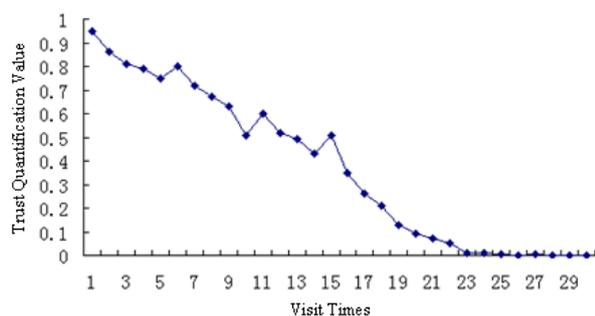


Fig. 6. Change of the trust quantification value for the hostile subject

We can see from the figures that, since the honest subject has good behavior, its trust quantification value is gradually increasing and, as the hostile subject's behavior is malicious, its trust quantification value drops very quickly. In Fig. 6, there are some rebounds in the decline process for the hostile subject's trust quantification values. The explanation is that since some experts have not interacted with the hostile subject recently, their evaluation of the hostile subject remains the same as earlier.

## 6. Conclusion

In this paper, by considering the fuzziness and the uncertainty nature of trust, we proposed to apply the fuzzy comprehensive evaluation method to quantify trust and presented a trust quantification algorithm. We also performed some analysis and experiment to show that the trust quantification algorithm can effectively quantify trust and that hostile behavior will get punished severely with respect to trust while good behavior will get awarded with increasingly higher levels of trust. In the future, we will further refine our algorithm and apply trust quantification to solving security problems in open systems.

## Acknowledgement

This work is supported by Science Foundation of Beijing Education Commission (Grant No. KM201010005027).

## References

1. Z. Zhang, X. Wang and Y. Wang, A P2P Global Trust Model based on Recommendation, in *Proc. 4th International Conference on Machine Learning and Cybernetics*, (Guangzhou, China, 2005), pp. 18-21.
2. X. Ni and J. Luo, A Trust Aware Access Control in Service Oriented Grid Environment, in *Proc. 6th International Conference on Grid and Cooperative Computing*, (Urumqi, China, 2007), pp. 417-422.
3. M. Blaze, J. Feigenbaum and J. Lacy, Decentralized Trust Management, in *Proc. Symposium on Security and Privacy*, (Oakland, CA, 1996), pp. 164-173.
4. M. Blaze, J. Ioannidis and A. D. Keromytis, Experience with the KeyNote Trust Management System: Applications and Future Directions, in *Proc. 1st International Conference on Trust Management*, (Grete, Greece, 2003), pp. 284-300.
5. S. Liu and X. Liu, A New Method of Elevation of Confidence Level of Large-Scale Perplexing Simulation System, *Journal of System Simulation*, 13(5) (2001), pp. 666-669.
6. F. Azzedin and M. Maheswaran, A Trust Brokering System and Its Application to Resource Management in Public-Resource Grids, in *Proc. 18th International Parallel and Distributed Processing Symposium*, (Santa Fe, New Mexico, 2004), pp. 289-298.
7. S. D. Kamvar, M. T. Schlosser and H. Garcia-Molina, The Eigentrust Algorithm for Reputation Management in P2P Networks, in *Proc. 12th International World Wide Web Conference*, (Budapest, Hungary, 2003), pp. 640-651.
8. C. Duma and N. Shahmehri, Dynamic Trust Metrics for Peer-to-Peer System, in *Proc. 16th International Workshop on Database and Expert Systems Applications*, (Copenhagen, Denmark, 2005), pp. 776-781.
9. Y. Wang and V. Varadharajan, Interaction Trust Evaluation in Decentralized Environments, in *Proc. 5th International Conference on Electronic Commerce and Web Technologies*, (Zaragoza, Spain, 2004), pp. 144-153.
10. L. Yuan, G. Zeng, L. Jiang and C. Jiang, Dynamic Level Scheduling Based on Trust Model in Grid Computing, *Chinese Journal of Computers*, 29(7) (2006), pp. 1217-1224.
11. B. Christian and O. Radoslaw, Using Context and Content based Trust Policies on the Semantic Web, in *Proc. 13th International World Wide Web Conference*, (New York, 2004), pp. 228-239.
12. T. Sun, K. Mieso and A. Denko, Distributed Trust Management Scheme in the Pervasive Computing Environment, in *Proc. 2007 Canadian Conference on Electrical and Computer Engineering*, (Vancouver, BC, 2007), pp. 1219-1222.
13. A. Gutscher, A Trust Model for an Open, Decentralized Reputation System, in *Proc. Joint iTrust and PST Conferences on Privacy, Trust Management and Security*, (New Brunswick, Canada, 2007), pp. 285-300.
14. Z. Liang and W. Shi, Analysis of Ratings on Trust Inference in Open Environments, *Performance Evaluation*, 65(2) (2008), pp. 99-128.
15. M. Jablonowski, Implications of Fuzziness for the Practical Management of High-Stakes Risks, *International Journal of Computational Intelligence Systems*, 3(1) (2010), pp. 1-7.
16. X. Guo, Q. Yang, Risk Assessment Model for Project Investment Based on Fuzzy Theory, in *Proc. 2010*

- International Symposium on Computer, Communication, Control and Automation*, (Taiwan, 2010), pp. 400-402.
17. A. Pahlavani, A New Fuzzy MADM Approach and its Application to Project Selection Problem, *International Journal of Computational Intelligence Systems*, 3(1) (2010), pp. 103-114.
  18. G. Wei, X. Zhao, R. Lin, Some Induced Aggregating Operators with Fuzzy Number Intuitionistic Fuzzy Information and their Applications to Group Decision Making, *International Journal of Computational Intelligence Systems*, 3(1) (2010), pp. 84-95.
  19. X. Tang, Q. Guo, P. Wu, H. Song, A Combination Of Fuzzy Theory And Genetic-Neural Network Algorithm, in *Proc. Ninth International Symposium on Distributed Computing and Applications to Business, Engineering and Science*, (Hong Kong, 2010), pp. 639-642.
  20. E. Zio, P. Baraldi, I. C. Popescu, From Fuzzy Clustering to a Fuzzy Rule-based Fault Classification Model, *International Journal of Computational Intelligence Systems*, 1(1) (2008), pp. 60-76.
  21. J. F. Zhang and B. R. Deng, *Application of Fuzzy Mathematics*, (Geological Publishing House, Beijing, China, 1991).
  22. X. Li and Y. Gu, Fuzzy Comprehensive Evaluation Model of Customer Satisfaction Degree, in *Proc. International Conference on E-Business and E-Government*, (Guangzhou, China, 2010), pp. 3111-3114.
  23. Y. Chen and Y. Wang, Fuzzy Comprehensive Evaluation of Product Innovation Design, in *Proc. Second International Conference on Computer Modeling and Simulation*, (Sanya, China, 2010), pp. 407-411.
  24. X. He, Z. Zhu, Y. Zhou, G. Lu and Q. Liu, University Teaching Quality Evaluation Using Fuzzy Comprehensive Evaluation Approach, in *Proc. Second International Workshop on Education Technology and Computer Science*, (Wuhan, China, 2010), pp. 616-619.
  25. X. Wang, L. Li and L. Tian, Research on Fuzzy Comprehensive Evaluation of Enterprise Websites, in *Proc. 2010 IEEE International Conference on Information Theory and Information Security*, (Beijing, China, 2010), pp. 937-940.
  26. J. H. Holland, *Adaptation in Natural and Artificial Systems*, (University of Michigan Press, Ann Arbor, Michigan, 1975).
  27. J. C. Potts, D. G. Terri and B. Y. Surya, The Development and Evolution of an Improved Genetic Algorithm based on Migration and Artificial Selection, *IEEE Trans. Systems, Man and Cybernetics*, 24(1) (1994), pp. 73-86.
  28. D. Du, Q. Pang and Y. Wu, *Modern Comprehensive Evaluation Methods and Case Selected*, (Tsinghua University Press, Beijing, China, 2008).
  29. F. He and C. Zhang, Application and Calculation Cases of Monte Carlo Method, *Journal of North China Electric Power University*, 32(3) (2005), pp. 110-112.
  30. X. Wang, F. Miao, Y. Xiong, D. Zhou and L. Yue, The Monte Carlo Algorithm to Evaluate the Cover Quality of Object Points in Wireless Sensor Network, *ACTA ELECTRONICA Sinica*, 34(12A) (2006), pp. 2485-2487.