Int. J. Communications, Network and System Sciences, 2009, 2, 764-774
doi:10.4236/ijcns.2009.28089 Published Online November 2009 (http://www.SciRP.org/journal/ijcns/).

A Reputation-Based Multi-Agent Model for Network Resource Selection

Junfeng TIAN, Juan LI, Lidan YANG
Network Technology Institute, Hebei University, Baoding, China
Email: *tjf@hbu.edu.cn, {antylj, mousekidcn*1984}@163.*com*
Received July 7, 2009; *revised August* 12, 2009; *accepted September* 27, 2009

Abstract

Because of the anonymity and openness of on-line transactions and the richness of network resources, the problems of the credibility of the on-line trading and the exact selection of network resources have become acute. For this reason, a reputation-based multi-agent model for network resource selection (RMNRS) is presented. The model divides the network into numbers of trust domains. Each domain has one domain-agent and several entity-agents. The model prevents the inconsistency of information that is maintained by different agents through the periodically communication between the agents. The model enables the consumers to receive responses from agents significantly quicker than that of traditional models, because the global reputation values of service providers and consumers are evaluated and updated dynamically after each transaction. And the model allocates two global reputation values to each entity and takes the recognition value that how much the service provider knows the service into account. In order to make users choose the best matching services and give users with trusted services, the model also takes the similarity between services into account and uses the similarity degree to amend the integration reputation value with harmonic-mean. Finally, the effectiveness and feasibility of this model is illustrated by the experiment.

Keywords: Trust, Reputation, Trust-Domain, Multi-Agent, Similarity

1. Introduction

The World Wide Web has evolved at an extreme rate due to its capacity to provide an endless amount of resources to the public users. Hence, the user finds him lost in a pool of information, without knowing how to select resources and which resources are credible [1]. A wide range of mechanisms such as contracts and commercial laws as well as face to face meetings help reduce the likelihood of risks to the consumers in the traditional businesses. However, When doing online trading, users often have little or no prior knowledge of their potential business partner(s) and the absence of these mechanisms and face-to-face encounter make them can not check goods before paying and can not distinguish cheat from honest effectively. As a result, there has always produced a trust deficit in e-commerce [2].

Thus, the trust has become an integral part of traditional transaction and e-commerce transaction. Recommender system which is based on trust have proven to be an important method to effectively find those resources that users are interested in from endless resources in the network, by providing users with more proactive and personalized information services. And the recommender system based on trust collaborative filtering is to recommend trusted and satisfying information services to users [3,4].

The exiting recommender systems based on trust filtered recommendation information just through authenticating users' identity or removing users with low trust value [5]. They didn't consider the following four problems: 1) the role of transaction behavior for users in e-commerce has two types: buyer and seller, so we can't only use one trust value or reputation value to measure the users' trust level with different transaction behavior. Because the malicious users may use honest buy behavior to cover up the dishonest sell behavior, or vice versa [6]; 2) if recommendations are from different recommenders, we should treat them differently not only because of the recommenders' various trust levels but also because the recommenders have different knowledge to their recommendation [7]; 3) in order to make users quickly find resources, the time from making a request to receiving the response should short as much as possible

[8]; 4) after each transaction, both the participators not only should update their trust values or reputation values but also should share their trust information of transactions which can increase the spread of trust information and raise the performance of network.

2. Related Works

In Peer-to-Peer (P2P) e-commerce transaction, users exchange information or transact with others through direct communication, but those users don't know each other before and they also don't trust each other. And the openness of P2P system makes the users can't avoid others' malicious behaviors. Once users transact with one malicious node, they may incur substantial losses [9]. In order to solve the problem of security for the network service, M. Blaze proposed the concept of Trust Management firstly in 1996 [10], and its basic thought was to admit the imperfection of security information in open system. It proposed that making safety decision for system needed additional security information. Nowadays the researches on trust are mainly classified into two types, identity trust and behavior trust. The identity trust which is based on code, authentication protocol or digital signature technology checks entity's authenticity and makes the decision whether authorize the entity to access. But the behavior trust pays more attention to the trusted problem in broader meaning. According to the past behavior experiences, it updates the trust relationship between users dynamically and timely. International research indicates [11] that the network security is developing toward the direction of credible network. The future network security is the credible network with increasing credible behavior, which is a new consensus that is agreed by the network security research areas in recent years. The research on whether the users' behavior is credible not only increases the security of network through decreasing or avoiding transaction with malicious users but also improves the success rate of transactions and decreases the extra spending caused by monitor or precaution which are caused by distrust. Thereby the overall performance of the network is improved.

There have been lots of researches about behavior trust at home and abroad. Based on the transitivity of trust, the model named EigenREP was based on global reputation in P2P environment [12]. But its drawback is its astringency, high communication costs and relative global reputation, thus it can not evaluate whether the node is credible just through the value of global reputation. The model proposed in paper [8] presents that each grid domain is associated with multiple brokers and each broker with multiple entities. It eases the network traffic at the broker sites and makes the service providers' (SPs) response to the consumers' request significantly quicker, but it might lead to the information inconsistency maintained by different brokers and solving the problem has its costs. A trust model [13] based on behaviors was proposed to achieve the resource sharing and cooperation among different domains in grid environment, which dynamically reflects the entity's subject characteristic. But its limitations are that it doesn't update the trust value, so it cannot reflect the dynamics of trust computing. Traditional resource selection method [14–16] always selects service providers with the highest trust value. They don't consider whether the selected services are the services that consumer expected, that is to say whether the selected services and the expected services are accordant. Paper [17] proposed a similarity measurement about ontology-based semantic web services and paper [18] proposed a method of similarity search for web services. Both of them measure the similarity between the services with ontology and find the expected services to consumers. But the weakness is the high complexity of algorithm. Paper [3–5] proposed the idea of trust filtering in recommender systems, which considered that the recommenders should have similar tastes and preferences, should be trustworthy in the sense that they had a history of making reliable recommendations and should have different trust degree in entity trust and content trust. The weakness is that they don't solve the sparseness of similarity well when they find similar users. Paper [7] proposed a role-based recommendation and trust evaluation model which firstly takes the role of recommender into account. But it didn't present how to organize and storage a rational role hierarchy. A novel distributed trust model is propose in [6], which iteratively calculates for each node a global seller reputation value and a global buyer reputation value based on transaction history, and whether a node is credible or not can be identified from them. But it doesn't provide the computation of some coefficients.

To solve the problems of existing distributed trust model, the paper proposes a novel reputation-based multi-agent model for network resource selection (RMNRS). With nodes' identity and their recognition to services, the model computes for each node a global buyer reputation and a global seller reputation. And the model estimates whether one node is credible or not through the final Trust-Value which is the harmonic-mean of Trust-Value and similarity degree between request services and provided services. The model's characters are listed bellow.

1) Our model is based on trust domain which adopts multi-level trust management mechanism to manage agents belonging to different levels. The periodically communication between agents prevents the information's inconsistency between the agents.

2) Our model computes the similarity between request service and provided service by ontology. We want to find the most similar services to the requestor. Combined with the computing of Trust-Value, the services that we

supplied will be not only trusty but also matching to the request.

3) Our model takes the recognition value that how much the service provider knows the service into account, which makes the provided services more similar to the request.

4) Our model doesn't use one trust value to determine whether a node is credible or not. It keeps global buyer reputation value and global seller reputation value for each node, thus it can reflect node's different trust level with different transaction behavior.

5) After each transaction, the participators can share mutual trust information under certain condition, namely trust propagation.

6) Our model provides the computation of coefficient which is the weight when integrating two global reputation values.

This paper is organized as follows. Section 3 presents the related definitions, algorithms and fundamental principle of the model. Section 4 presents the simulation to validate our model's effectiveness and feasibility. And finally Section 5 concludes our work.

3. The Related Definitions, Algorithms and Fundamental Principle of the RMNRS Model

3.1. Related Definitions

Definition 1 (Trust): Trust is the subjective probability expectation of trustor to trustee's specific behavior which is relied on experiences and continuous to modify its value as the change of trustee's behavior. Paper [7] proposed that trust is a complex subject relating to an entity's belief in honesty, trustfulness, competence and reliability of another entity. Paper [19] proposed that trust is to believe others, which is established on their own knowledge and experiences and is a subjective behavior between entities. Trust is different from the belief that person believes in object things, which is a subjective judgment. The trust itself is not a fact or proof, it is acknowledge of observed fact. According to the different achieving trust way when entities interact with each other, Beth [20] divided the trust into direct trust and recommendation trust. To trust an entity directly means to believe in its capabilities with respect to the given trust class. Recommendation trust expresses the belief in the capability of an entity to decide whether another entity is reliable in the given trust class and in its honesty when recommending third entities.

Definition 2 (Trust-Domain): According to the Web-Based activities and related application, we divide the virtue network into numbers of self-government domains and define the self-government domain as Trust-Domain.

Definition 3 (Domain-Entity): Domain-Entity is the node or object who has some resources in network. It can be a user, service or resource. The interaction between entities has two types: the interaction of intra-domain and inter-domain.

Definition 4 (Transaction): One transaction is one interactive behavior which happens between two nodes when they need mutual services in the P2P network, such as one business dealing in e-commerce, one file download and so on. The buyer is the one who requests the services and the seller is the one who provides the requested services.

Definition 5 (Reputation): Reputation is the expectation of one entity's future behavior based on the observation of the entity's past behavior or evaluation information in transactions [9]. There are two types, local reputation and global reputation. The local reputation is defined as the expectation of a node's future behavior based on the past evaluation information which is provided by one of its buyers. The global reputation is defined as the expectation of a node's future behavior based on the past evaluation information which is provided by those nodes who had transacted with node j ago.

In this paper, each node has four types of reputation value: local buyer reputation, local seller reputation, global buyer reputation and global seller reputation.

Definition 6 (Trust-Value): The Trust-Value of Trust-Domain is defined as the mean of the Trust-Value of all entity-agents. The Trust-Value of entity-agent is the mean of the global reputation value of all the entities managed by the entity-agent. The Trust-Value of one entity is the weighted mean of global buyer reputation value and global seller reputation value of the entity.

Definition 7 (Identity): The Identity is not the role of entities in transaction. It refers to the identity symbol of one entity's recognition degree to one service in certain service domain, such as the social positions, social titles or certificates.

3.2. Trust Domain and Agent

In social network, everyone or group has his or her own interesting and joins in trusted communication circle, they have high credibility on the people or group in the same circle [21]. While in virtue network, because of the disparate resources, the sharing, cooperation and high performance of resources has become difficult. The wide connectivity of network requires to establish public and effective security mechanism between different nodes and peoples and to implement consistent security strategy. It also requires doing specific security management according to the application of multi-network. In this paper, we import the agent mechanism to abstract the

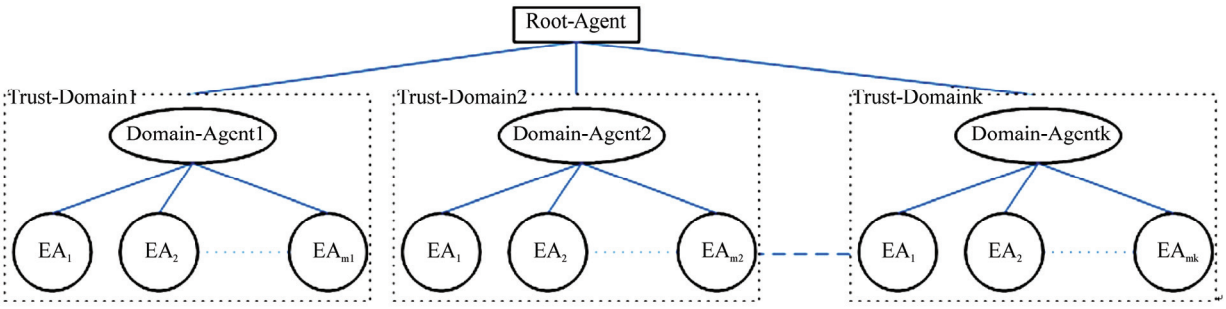

Figure 1. Trust-Domain and agent management architecture.

network, as shown in Figure 1 which implies the architecture of the model. We use this mechanism to manage the entities' trust computing and trust relationship between entities. Our model is not only manageable but also permits the managed object to cooperate independently, which is in accordance with the network computing mode and development tend [23].

We assume that there is an absolute trusted agent: Root-Agent (RA) who takes charge of every agent in each domain. Every domain has a Domain-Agent (DA) and numbers of Entity-Agents (EA)

1) The Root-Agent is the manager of global trust relationship of the system. It is an absolute trusted root. It manages and collects all the Domain-Agents' information and maintains the global trust relationship. The information that the RA maintains is the ID of DA, the Trust-Value of domain and semantic base.

2) The Domain-Agent is the manager of part trust relationship of the system and the trust relationship of domain. If there are entities request to join in the domain, the DA has the right that permits them to join in or judges which EA they belong to and it passes the information of entities to RA, so that RA can update the semantic base timely. When one EA's Trust-Value is less than threshold, the DA has the authority to retake the authority and awards the authority to other entities. According to the Trust-Value of the entity who wants to be the EA, DA makes the decision that whether it can be or not. The entity has the status to be an EA only when the Trust-Value of one entity is over the threshold. All the entities that an EA maintains provide the same or similar services. The DA collects EAs' information periodically (the period is decided by the size of system or the request of interaction) that it maintains and broadcasts the information as guide Trust-Value to each EA. The information that the DA maintains is the ID of EA and the direct transaction table (the domain-ID of entity, the entity-agent-ID of the entity, global buyer reputation value, global seller reputation value, service type, time, cost and the satisfaction degree of transaction).

3) The entities that maintained by Entity-Agent is organized as Binary Sorting Tree which has such characters: if the left sub-tree is not null then all the Trust-Value of entities in left sub-tree will be smaller than that of root; if the right sub-tree is not null then all the Trust-Value of entities in right sub-tree will be bigger than that of root; both the sub-tree are Binary Sorting Trees. We can add, delete, update and lookup needed information from the Binary Sorting Tree. The information that maintained by EA is illustrated as follows.

a) The storage structure of entities in Binary Sorting Tree.

```
Typedef struct BiTree {
DataType degree; // the Trust-Value of entity;
struct BiTree   * lchild,* rchild;   // the pointer of the left sub-tree and right sub-tree;
} BiTnode,*Bitree;
```

The binary tree is sorted by the trust degree of entities. The EA checks the Trust-Value of entities periodically and updates its position in the tree. If the Trust-Value of one entity is smaller than threshold, the EA can remove it from the tree. When one entity wants to join in, the EA puts it in the feat position.

b) The storage structure of entities in Binary Sorting Tree of entity's identity.

```
Typedef struct RoTree {
DataType Re-degree;        // the recognition degree of entity's identity;
struct BiTree   * lchild,* rchild;
}RoTnode,* Rotree;
```

The Binary Sorting Tree of entity's identity is established based on the recognition level of one entity to the services that it provides. The tree nodes are ordered by the recognition degree. The operation that we can do to the tree is similar to that of a). The reason that why the EA maintains the role tree is that it helps to select suitable entity.

c) Direct transaction table: the domain-ID of transaction entity, the entity-agent-ID of the entity, global buyer reputation value, global seller reputation value, service type, time, cost and the satisfaction degree of transaction.

The communication between entities is through Trusted Communication Agent Interface (TCAI) [22], which makes the communication between entities more

reliable. The agents pass information on to each other periodically. Once one entity's information was updated, it can push [23] its updated information to the related agents, which can avoid the information disaccording to others.

Our model is based on domain and has three levels, so we divide the trust relationship between entities into two types, the trust relationship of intra-domain and inter-domain.

1) The trust relationship of inter-domain: The evaluation objects of trust are based on domain. The trusted level of a domain is evaluated through the behavior that the entities showed in the transaction. Thus the trust level of the domain reflects all the entities' trust level in the domain.

2) The trust relationship of intra-domain: The trust of intra-domain is mainly the interaction between entities; the evaluation objects of trust are the entities in the domain. After the initialization, the trust level will be updated according to the behavior of entities in the following transaction.

3.3. The Fundamental Principles of RMNRS

This section mainly shows the basic principles of this model and the specific algorithms are listed in Subsection 3.4-3.9.

Step 1: The consumer sends the request which contains the service type and the mini-trust threshold to the agents.

Step 2: The agents select those nodes with higher global reputation value and calculate their integrated Trust-Value.

Step 3: The agents firstly calculate the similarity degree between buyer and Candidates, and then the agents calculate the harmonic-mean of similarity degree and integrated Trust-Value.

Step 4: If there are entities whose final Trust-Value (harmonic-mean) is bigger than the mini-trust threshold, go to Step 5, conversely go to Step 11.

Step 5: If the number of candidates is over one, go to Step 6, on the other hand, go to Step 7.

Step 6: Under the same credible condition, we select the nodes with the highest global buyer reputation.

Step 7: The buyer transacts with the selected entity and both parties give each other the satisfaction degree to this transaction.

Step 8: Both their entity-agents and domain-agents will update their reputation value and transaction table after transaction.

Step 9: According to the satisfaction degree, if both parties want to share their trust information, go to Step 10, on the other hand, go to Step 13.

Step 10: Both parties share their trust information of transaction.

Step 11: The agent ask the buyer whether it wants to modify the mini-trust threshold, so that it can find the suitable entity to transact with. If the user wants to decrease the threshold, go to Step 1, and on the other hand go to Step 12.

Step 12: The transaction is failure.

Step 13: The transaction is success.

3.4. The Initialization of Reputation

When one entity didn't have any interactions with other entities ago, should we trust it? In this paper we show several methods. According to specific environment you can select suitable method.

1) You can set the initial global buyer reputation value and global seller reputation value to be $\omega_1 \in [0,1]$ and $\omega_2 \in [0,1]$ separately.

2) According to the security information (such as the information of identity [7]) that the user provides, you can convert the information into the initial reputation value of entity through the function, where R is the set of role information and D is the domain that R belongs to.

After the initialization, the global reputation value will be updated according to the behavior of entities in the following transaction. To simplify the experiment, we set both the initial global buyer and seller reputation value of entity to be 0.5.

3.5. The Computation of Local Reputation Value

According to the definition of the local reputation, the value of local reputation is calculated in the light of the historical transaction evaluation that the buyer sent to the seller. After each transaction, both the participators feed back the degree of satisfaction to each other. And the user also should give out a threshold, so if the degree of satisfaction is higher than the threshold, we think that this transaction is satisfactory, on the other hand, we think it is unsatisfactory. We use the percentage of the number of satisfactory transactions in all transactions to represent the local reputation. And we let Lb_{ij} represent the local buyer reputation, which mean that the local reputation given by node i as the buyer to the node j as the seller and let Ls_{ij} represent the local seller reputation, which mean that the local reputation given by node i as the seller to node j as the buyer. The formula is illustrated as follows.

$$Lb_{ij} = \frac{N_{bg}(i,j)}{N_{bg}(i,j) + N_{bb}(i,j) \times N_{punish}} \quad (1)$$

$$Ls_{ij} = \frac{N_{sg}(i,j)}{N_{sg}(i,j) + N_{sb}(i,j) \times N_{punish}}. \quad (2)$$

where $N_{bg}(i, j)$, $N_{bb}(i, j)$, represent the number of satisfactory transaction and the number of unsatisfactory transaction between buyer i and seller j. $N_{sg}(i,j)$, $N_{sb}(i,j)$ represent the number of satisfactory transaction and the number of unsatisfactory transaction between seller i and buyer j. N_{punish} is the coefficient which is the punishment to malicious transaction. Especially when the cost of the transaction is very high, the punishment coefficient should be bigger. So the formula is

$$N_{punish} = ((\alpha+1)\times \sum_{k=1}^{N_{bb}(i,j)} C_{mk}(i,j) - \alpha) \Big/ \sum_{k=1}^{N_{bb}(i,j)} C_{mk}(i,j),$$

where $C_{mk}(i,j)$ is the cost of k^{th} unsatisfactory transaction between node i and node j. We assume all the cost of transaction is bigger than 1. On the contrary, we set it to be 1; The formula not only reflects the higher the cost is, the bigger the weight is, but also satisfies the monotonic increasing with the increasing transaction cost and its result is in the cope of [1,1+α], where α is the regulatory factor. So that the user can adjust the scope of punishment according to his needs.

On the basis of the analysis above, the properties of the local reputation formula are shown as follows.

1) The model evaluates whether the node is credible or not directly. The more the number of satisfactory transaction is, the closer to 1.0 the reputation value is. The more the number of unsatisfactory transaction is, the closer to 0.0 the reputation value is.

2) The introduction of the punishment coefficient $N_{punish} \geq 1$ makes the reputation value decrease quicker than rise. It embodies the punishment to the malicious nodes and especially to those nodes who make use of high cost transactions to cheat.

3.6. The Computation of Global Reputation Value

The global reputation value is the integrated evaluation of one node, which is obtained form those nodes who had transacted with the node. We let Gb_i denote the global buyer reputation, where node i is the buyer and Gs_i to be the global seller reputation, where node i is the seller. But there are some factors we should consider when we calculate the value of global reputation.

1) The global buyer (seller) reputation of node i should be the integrated evaluation of all those nodes who had transacted with node i.

2) The evaluation of different nodes with different Trust-Value should be kept separate. That is because the evaluation of the nodes with higher Trust-Value is more important than that of nodes with lower Trust-Value.

3) The more the number of transaction between both parties is, the more credible the evaluation is.

4) The global reputation is an accumulative process, so only through persistent credible transaction, the value will be higher.

The formulas are listed as follows.

$$Gb_i(k+1) = \frac{\sum_{j\in Vsi} Gs_j(k)\times(1-e^{-\frac{Nsg(j,i)-Nsb(j,i)}{5}})\times Ls_{ji}}{\sum_{j\in Vsi} Gs_j(k)\times(1-e^{-\frac{Nsg(j,i)-Nsb(j,i)}{5}})} \quad (3)$$

$$Gs_i(k+1) = \frac{\sum_{j\in Vbi} Gb_j(k)\times(1-e^{-\frac{Nbg(j,i)-Nbb(j,i)}{5}})\times Lb_{ji}}{\sum_{j\in Vbi} Gb_j(k)\times(1-e^{-\frac{Nbg(j,i)-Nbb(j,i)}{5}})} \quad (4)$$

where the meanings of $N_{sg}(j,i)$, $N_{sb}(j,i)$, $N_{bg}(j,i)$ and $N_{bb}(j,i)$ are the same as that of Chapter 3.5. Vb_i represents the set of nodes who had transacted with node i and they are buyer. Vs_i represents the set of nodes who had transacted with node i and they are seller.

The weighted average method not only can embody the views of all the nodes, but also keep the meaning of global reputation unchanged. The character of 1-exp(-($N_g(j,i)$-$N_b(j,i)$)/5) negative exponent increase as $N_g(j,i)$-$N_b(j,i)$ in accordance with the feature that credible transaction can improve the reputation and incredible transaction can decrease the reputation.

3.7. The Computation of Trust-Value

The Trust-Value of node j represented as T_{ij} is the critical factor for node i to determine whether to transact with node j. Because it is the integrated value of Ls_{ji} and Gs_j, its computation must consider the trust level of node i to j's local reputation and global reputation. If node i has transacted with node j, the Trust-Value of node j is the weighted sum of Ls_{ji} and Gs_j. If node i didn't have any transactions with node j before, node i can ask his entity-agent and domain-agent to recommend providers. Both the agents search those entities that had transacted with node j and ask them to recommend node j. Those recommenders give views about node j according to the history performance of node j. Then the agents feed back these information to the requestor i. To ensure the validity of recommendation, the rules of recommendation are shown as follows.

1) The recommendations have time limitation, the scope of time is [σ_1, σ_2] which can be defined as the transaction needs. That is because the previous transaction information can not exactly reflect the *SPs*' credibility of the current situation.

2) Because the recommendation is finite, the depth that the agent searches recommenders in the binary sorting tree must be smaller than h.

3) In order to reduce the likelihood of collusion, we set the number of recommenders must more than certain

threshold ϑ where $\vartheta > 0$. If the number of recommenders is less than ϑ, we can add some virtue nodes as recommender whose reputation is the same as the initial Trust-Value and the number of nodes that had transacted with each recommender is $+\infty$.

Because of the asymmetry of trust, the recommendation involves the consumer's trust to the recommender's recommendation. The higher trust level the consumer is to the recommender, the more trust the consumer is to the recommendation. If recommender has high recognition about his services, the trust level of his recommendation also can be increased. We let T_{ij} denote the Trust-Value of node *j* as opposite to node *i*. The formula is shown as follows.

$$T_{ij} = \frac{\sum_{j \in Dir} (\alpha Ls_{ji} + \beta Gs_{ji})}{\|Dir\| + \|Undir\|} + \frac{\sum_{r \in Undir} T_{ir} \times rec_r \times (\alpha Ls_{jr} + \beta Gs_{jr})}{\|Dir\| + \|Undir\|} \quad (5)$$

where *Dir* is the set of nodes who had transacted with node *i* directly and *Undir* is the set of nodes who are recommended by recommender *r* to transact with node *i*. The $\|Dir\|$ and $\|Undir\|$ represent the length of the corresponding set. $rec_r \in [0,1]$ is the recognition degree of recommender *r* to service. α and β is the weighting of local reputation and global reputation separately and $\beta=1-\alpha$. The value of α is computed as follows.

The local reputation is the evaluation of one node according to his history transaction behavior with another node. So if the number of transactions is more, the transaction cost and the evaluation of the transaction are higher, the buyer will more trust the local reputation which is achieved through his direct experiences.

$$\alpha = \sqrt{(\sum_{j=1}^{m} C_{mbig} \Big/ \sum_{i=1}^{n} C_m) \times (\sum_{j=1}^{m} Eva_{good} \Big/ \sum_{i=1}^{m} Eva)}$$

where *n* is the total number of transaction, *m* is the number of transactions with high cost and good evaluation. The user can defined a threshold to determine the value of m. C_m is the cost of transaction and C_{mbig} is the high cost of transaction. Eva is the evaluation of transaction, Eva_{good} is the good evaluation of transaction and *Eva,* $Eva_{good} \in [0,1]$.

3.8. The Harmonic-Mean of Trust-Value and Similarity Degree

The traditional recommender systems based on trust believe that the higher the global reputation of recommender is, the more credible the recommendation is. But in fact the value of global reputation is not in conformity with the importance of recommendation. For example, some malicious nodes may achieve high global reputation value through fake or some nodes collude in order to remote their Trust-Value or to destroy other competitor. Thus the recommender not only has the high trust degree of recommendation but also should make sure that the content of recommendation is credible. In this paper, in order to ensure the content of recommendation is credible, the model imports the semantic-based service matching method which calculates the similarity degree between the request service and the provided service. The service is reliable only when the provided service satisfies the user's needs. The computation of similarity adopts one simple and efficient method provided by paper [24], denoted as *WS*.

This paper uses the similarity degree to harmonize the Trust-Value with weighting. The higher the similarity degree is, the bigger the harmonic-mean is. The harmonic-mean is the reciprocal of the arithmetic mean of the reciprocals, which is mainly used to the situation that the initial digitals is not the direct initial digitals but its frequency had been computed. The formula is shown as follows.

$$ST_{ij} = \frac{2 \times WS_{ij} \times T_{ij}}{\delta \times T_{ij} + (1-\delta) \times WS_{ij}} \quad (6)$$

where δ is the adjustment factor; WS_{ij} is the similarity degree of the request service and provided service; T_{ij} is the trust level that how much node *i* trust in node j; ST_{ij} is the harmonic-mean of similarity and Trust-Value which directly determines whether node *i* transacts with node *j* or not.

3.9. The Updates of Trust-Value and the Sharing of Trust Information

After each transaction, both parties will feed back the evaluation of this transaction to each other, namely the satisfaction degree. Both the entity-agents of two parties need to respectively update the global buyer reputation (Gb_i) of buyer and the global seller reputation (Gs_j) of seller. The updates in time ensure that the users don't need to calculate the reputation value when they send the request, so that the model can respond the user's request quickly. The formulas are listed as follows.

$$Gb_i^{k+1} = \varphi \times Gb_i^k + (1-\varphi) \times S_k(i,j) \times C_k, \quad (7)$$

$$Gs_j^{k+1} = \phi \times Gs_j^k + (1-\phi) \times S_k(j,i) \times C_k \times rec_k \quad (8)$$

where $S_k(i,j) \in [0,1]$ represents the satisfaction degree of buyer *i* to seller *j* after the k^{th} transaction. $S_k(j,i) \in [0,1]$ represents the satisfaction degree of seller *j* to buyer *i* after the k^{th} transaction. The satisfaction degree indicates the level of satisfaction achieved by the consumer on the

service provided by the *SP*. A normalized value between 0 and 1 is used, with 1.0 indicating 100% satisfaction and 0.0 indicating the lowest satisfaction. $C_k = (C_{mk}-1)/C_{mk}$ represents the weight of transaction cost in the k^{th} transaction. $rec_k \in [0,1]$ is the recognition degree of entity to the service that he provided in k^{th} transaction which is maintained in the binary sorting tree of entity's identity by his entity-agent and is registered by entity when he joined the domain. $\varphi, \phi \in [0,1]$ are the weights which are assigned according to last transaction time.

In addition to this, we also need to update the trust tables which are related to the entity and maintained by EA and DA. The update method is pushing the information to the needed agent.

Updating the reputation value in time can effectively reduce the response time that the buyer waits from his agents and it is done among free time without influencing the transaction. Thus the user can use the reputation value directly only with time decay such as linear decline, exponential decline and so on.

Both the transaction parties can set a threshold to determine whether to share trust information or not. That is to say, they share their trust information only when their Trust-Value is more than the threshold. The sharing of trust information makes the trust propagate quickly and improves the performance of network.

4. Simulations and Analysis

In this paper, the simulation is based on the PeerSim which is written in the Java language and is based on components [25]. It can support the extensibility and dynamic of the P2P network better. And it adopts the modular design and uses the configuration file to custom the modules and parameters. Thus it has high expandability. It also provides the interfaces and statistical methods to generate the network and makes the simulation and the evaluation of one algorithm more easily.

The principle of organization of the RMNRS model makes us not use the existing protocols of PeerSim directly. We are obliged to inherit one existing protocol and write our own algorithm to simulate. The protocol that we inherit is: Idle Protocol-Average Function. The control is cycle-based. In this paper we just inherit the interconnect relation between nodes and we write the algorithms about the model of EigenRep and RMNRS autonomously.

According to the character of the nodes in the network, we divide the nodes into four types: absolute trust nodes, trust nodes, critical trust nodes and distrust nodes.

Definition 1: absolute trust is the trust that is established on the basis of both parties in the partnership experience long-term transaction and major event test;

Definition 2: trust is the trust that is established on the basis of both parties in the partnership experience long-term transaction and general event test;

Definition 3: critical trust is the trust that both parties in the partnership don't have sufficient reason to trust or distrust each other;

Definition 4: distrust is the trust that after the long-term transaction, both parties in the partnership don't trust one at least.

We define the absolute trust node's value to be T where $T \geq T_{max}$, trust node's value to be T where $T_{mid} < T < T_{max}$, critical trust node's value to be T where $T_{min} \leq T \leq T_{mid}$, distrust node's value to be T where $T < T_{min}$, and $0 \leq T_{min} \leq T_{mid} \leq T_{max} \leq 1$.

4.1. Experiment 1

According to the paper, we define the network size is 1000, the simulation cycle is 50, the degree of a node is 6 and the init global reputation value is 0.5. We assume that T_{min}=0.3, T_{mid}=0.6and T_{max}=0.9. We define the percent that the four type nodes respectively are 10%, 30%, 30% and 30%. All of the nodes belong to different trust domains. Each domain has four types of nodes. Entities' Trust-Value will be changed timely along with the transaction in the domain or between domains. Along with the increase of the number of transactions the variation of the four types of entities' average Trust-Value is showed in Figure 2.

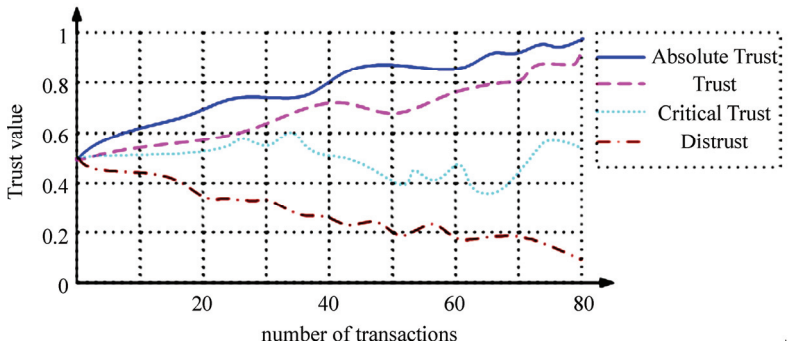

Figure 2. The variation of the average Trust-Value of four types of nodes.

As shown in Figure 2, the initial reputation value of all nodes is 0.5 and along with the increasing number of trades, the model updates the reputation value according to the Equation (5). Finally we calculate the Trust-Value of each node. According to the definitions of the four types of nodes, we can find that the result of RMNRS model is in accordance with the expected analysis. And along with the increase of the number of trades, it well reflects that the variation of the nodes' Trust-Value.

4.2. Experiment 2

The assumption of nodes is the same as the Experiment 1 in Chapter 4.1. In this simulation, we assume the degree of each node is 3, that is to say, the direct transaction table of each node maintains three nodes' trust information in the primary stages and the three nodes are selected randomly by the PeerSim protocols. When we write the program, we let the nodes' initial reputation value be random which is between 0 and 1. The experiment is designed to record the selection number from the node sending out the request to it receives the request. The network consumption is based on the number of selection step when the consumer selects the transaction partner.

In this model, we allocate two global reputations for each node. Thus the user can select the nodes with higher global seller reputation in the first stage which expands the range of selection. Then the model computes the harmonic mean of the similarity degree and the integration Trust-Value. The final Trust-Value sincerely reflects whether the provided service satisfies the users' needs, which can reduce the unnecessary selection. However the EigenRep model just follows the traditional method to select the provider and the comparison number is more than the RMNRS model. Figure 3 realistically reflects the analysis of this paper and verifies the efficiency of the RMNRS model.

4.3. Experiment 3

Whether the transaction is success or not is decided by the satisfaction degree which is fed back by user. If the satisfaction degree is larger than 0.6 then this transaction is success and vice versa. We define the success ratio of transaction to be the proportion of the success number.

In this simulation, we assume that the absolute trust nodes provide the credible services with the probability of 100%. And we also assume that the RMNRS and the EigenRep select the absolute nodes with the probability of 80%. As seen in Figure 4, when there are not malicious nodes in the environment, the success ratio of transaction is 95%. Along with the increase of malicious nodes, the transaction success ratio of EigenRep declines obviously. When the ratio of malicious nodes is 50%, the transaction success ratio of EigenRep is only about 60%. This is because there is lack of punishment to the malicious nodes in the EigenRep. So its success ratio has bigger drop. The RMNRS punishes the malicious nodes and matches the services between the request and the provided services. And it uses the similarity degree to amend the Trust-Value with the harmonic mean which ensures that the provided service is the needed service. The RMNRS avoids the transaction with the malicious nodes and improves the success ration of transaction. Under the condition of existing malicious nodes with the property of 50%, the transaction success ration of the RMNRS is about 80%. The experiment verifies the feasibility and efficiency of the RMNRS.

5. Conclusions

In this paper we present a reputation-based multi-agent model for network resource selection (RMNRS) which prevents the inconsistency of information maintained by different agents through the periodically communication between the agents. The model enables the consumer to receive the response from the agent significantly more

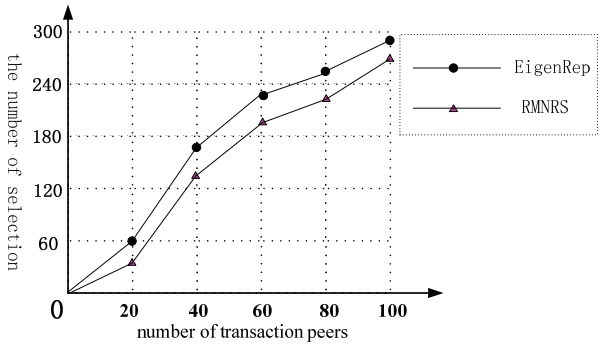

Figure 3. The comparison between RMNRS and EigenRep in the network consumption.

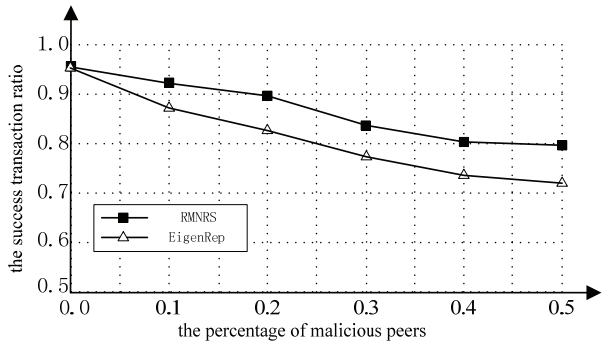

Figure 4. The comparison between RMNRS and EigenRep for success ratio of the transaction.

quickly than that of traditional models because the global reputation values of both parties are evaluated and updated dynamically after the completion of each transaction. And the model allocates two global reputation values to each entity and takes the recognition value that how much the service providers know the service into account. In order to make users choose the best matching services to their request and give users with trusted services, the model also takes the similarity between services into account and uses the similarity degree to amend the integration reputation value with the harmonic-mean. The following work is to research how to avoid the sharing of incredible information with malicious nodes and how to punish those malicious nodes.

6. Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant No. 60873203), the Natural Science Foundation of Hebei Province (Grant No. F2008000646) and the Guidance Program of the Department of Science and Technology in Hebei Province (Grant No. 072135192).

7. References

[1] H. Ibrahim, P. K. Atrey, and E. S. Abdulmotaleb, "Semantic similarity based trust computation in websites," International Multimedia Conference, New York, ACM, pp. 65–72, 2007.

[2] S. K. Chong and J. H. Abawajy, "Feedback credibility issues in trust management systems," 2007 International Conference on Multimedia and Ubiquitous Engineering: proceedings: MUE'07, Los Alamitos, Calif., IEEE Computer Society, pp. 387–391, 2007.

[3] Donovan J O and Smyth B, "Trust in recommender systems," in proceedings of the 10th international conference on Intelligent user interfaces, New York, ACM, pp. 167–174, 2005.

[4] P. Massa and P. Avesani, "Trust-aware recommender systems," in proceedings of the 2007 ACM conference on Recommender systems. New York, ACM, pp. 17–24, 2007.

[5] Y. Gil and D. Artz, "Towards content trust of web resources," in proceedings of the 15th international conference on World Wide Web, New York, ACM, pp. 565–574, 2006.

[6] D. S. Peng, C. Lin, and W. D. Liu, "A distributed trust mechanism directly evaluating reputation of nodes," Journal of Software, Vol. 19, No. 4, pp. 946–955, April 2008.

[7] Y. Wang and V. Varadharajan, "Role-based recommendation and trust evaluation," in the 9th IEEE International Conference on E-Commerce. Technology and the 4th IEEE International Conference on Enterprise Computing, E–Commerce and E–Services, Tokyo, IEEE, pp. 278–288, 2007.

[8] P. Varalakshmi, S. Thamarai Selvi and M. Pradeep, "A multi-broker trust management framework for resource selection in grid," in Communication Systems Software and Middleware, COMSWARE'07, 2nd International Conference on Bangalore, IEEE, pp. 7–12 January 2007.

[9] S. X. Jiang and J. Z. Li, "A reputation–based trust mechanism for p2p e-commerce systems," Journal of Software, Vol. 18, No. 10, pp. 2551–2563, 2007.

[10] M. Blaze, J. Feigenbaum, and J. Lacy, "Decentralized trust management," in proceedings of the 17th Symposium on Security and Privacy, CA, IEEE Computer Society Press, pp. 164–173, 1996.

[11] C. Lin, L. Q. Tian, and Y. Z. Wang, "Research on user behavior trust in trustworthy network," Journal of Computer Research and Development, Vol. 45, No. 12, pp. 2033–2043, 2008.

[12] S. D. Kamvar and M. T. Schlosser, "EigenRep: Reputation management in P2P networks," in Lawrence S, ed. Proceedings of the 12th International World Wide Web Conference Budapest, ACM Press, pp. 123−134, 2003.

[13] C. F. Wang and F. C. Sun, "Hierarchical entity self-determined trust model based on behaviors in grid environment," Computer Engineering and Applications, Vol. 43, No. 16, pp. 135–138, 2007.

[14] F. Maheswaran and M. Maheswaran, "Evolving and managing trust in grid computing systems," in proceedings of the 2002 IEEE Canadian Conference on Electrical Computer Engineering, pp. 1424–1429, 2002.

[15] F. Azzedin and M. Maheswaran, "A trust brokering system and its application to resource management in public resource grid," Parallel and distributed Computing Symposium. pp. 22, 2004.

[16] X. Li and L. Liu, "Peertrust: supporting reputation-based trust for peer-to-peer electronic communities," IEEE Transactions on Knowledge and Data Engineering, Special Issue on Peer to Peer Based Data Management, Vol. 16, No. 7, pp. 843–857, 2004.

[17] X. Dong, A. Halevy, J. Madhavan, *et al*., "Similarity search for web services," in Proceedings of the Thirtieth International Conference on Very Large Data Bases, VLDB Endowment, pp. 372–383, 2004.

[18] X. Wang, Y. H. Ding, and Y. Zhao, "Similarity measurement about ontology-based semantic web services," in Conjunction with 4th European Conference on Web Services (ECOWS'06), pp. 4–6, 2006.

[19] X. Y. Li and X. L. Gui, "Research on dynamic trust model for large scale distributed environment," Journal of Software, Vol. 18, No. 6, pp. 1510−1521, 2007.

[20] T. Beth, M. Borcherding and B. Klein, "Valuation of trust in open networks," in proceedings of the European Symposium on Research in Computer Security, Brighton UK, Springer-Verlag, pp. 3–18, 1994.

[21] X. Z. Zhang, "The research of virtual enterprise trust mechanism—the innovation of trust management in the network environment," Hunan, Hunan People's Publishing House, July 2005.

[22] Steve Hanna, Co-Chair, TNC Work Group, TCG. TNC: Open Standards for Network Access Control. https://www.trustedcomputinggroup.org.

[23] J. F. Tian, B. Xiao, X. X. Ma, *et al*, "The trust model and its analysis in TDDSS," Journal of Computer Research and Development, Vol. 44, No. 4, pp. 598–605, April 2007.

[24] X. Qing, "Semantic-based web service discovering algorithm," Master's degree, Jinan, Shandong University, pp. 17–26, 2006.

[25] PeerSim: A simulation environment for P2P protocols in java. Version 1.0.4. 2008

