Applied Mathematics & Information Sciences
An International Journal

TPS_DR: A Universal Dimension Reducing Algorithm for Optimal Trust Path Selection in Complex Sensor Network

Guangquan Xu[1,2], *Xiuming Tian*[1], *Xiaochun Cao*[1], *Xiaohong Li*[1] *and Zhiyong Feng*[1]

[1]School of Computer Science and Technology, Tianjin University, Tianjin 300072, China
[2]Shanghai Key Laboratory of Integrate Administration Technologies for Information Security, Shanghai Jiao Tong University, Shanghai 200240, China

Received: 9 Sep. 2012, Revised: 2 Dec. 2012, Accepted: 16 Dec. 2012
Published online: 1 Feb. 2013

Abstract: Energy efficiency, one of the key factors for sensor network, is a challenging work. It is agreed that sink routing is the main influencing factor for energy efficiency and selecting an optimal routing path becomes critical. It is hard for source participant (sink nodes, namely trustor) to select a trustworthy target participant (data collecting sensor nodes, namely trustee), especially when the scale of sensor network becomes increasingly larger. An optimal trust path selection problem (TPS problem) is a Multi-Constrained Optimal Path (MCOP) problem, which is proved to be a NP-Complete problem. In this paper, we first introduce a concept of complex trust network to model the relationships among network nodes in a complex sensor network. In this model, we represent the trust network based on high dimensional vector and matrix. Then, we propose an algorithm TPS_DR (Trust Path Selection based on Dimensionality Reduction) to simplify the trust network through reducing the dimensionality, i.e. cluster some similar nodes into a supernode. Our example demonstrates the usage and advantage of our model and algorithm.

Keywords: Dimensional Trust Vector, Complex Trust Network, Trust Path Selection, Multi-Constrained Optimal Path (MCOP) Selection Problem

1. Introduction

With the scale of sensor network becoming larger and larger, the trust relationships between nodes are accordingly becoming more complex and difficult to understand. As a result, the sink nodes can hardly find out the optimal routing path. However there can be over tens of thousands of trust paths between sink node (so-called trustor) and its target node (data collecting sensor node, trustee) in large-scale networks [1].Then the problem is how to find out an optimal trust path to get the trustworthy target. Considering that trust is a complex concept depending on lots of complex aspects it may concerns, such optimal Trust Path Selection problem (TPS problem) is typically a Multi-Constrained Optimal Path (MCOP) selection problem, which is NP-Complete[2].

Some arts made some efforts in routing scheme of communication network[3] After identifying that TPS problem is a MCOP selection problem and thus NP-Complete, Korkmaz and Krunz[2] introduced a corresponding solution, a heuristic algorithm (H_MCOP), which was an attempt to minimize both the nonlinear cost function (for the feasibility part) and the primary cost function (for the optimality part). However, such an algorithm requires the state information to be accurate, not necessarily true in the real world. Li et al.[4] advanced a composite service selection algorithm based on trust evaluation of Bayesian inference, and a Monte Carlo method (QoS constrained) based trust-oriented composite service selection algorithm was proposed. The optimal strategies in their later work[5] were advanced to improve the efficiency of the algorithm. Liu et al.[5] considered the above mentioned aspects thoroughly and proposed an efficient approximation algorithm MONTE_K. This algorithm was based on their new complex social network structure and a new concept of Quality of Trust (QoT), which illustrated the ability to guarantee a certain level of trustworthiness in trust evaluation. They considered three QoT attributes, which

* Corresponding author e-mail: xuguangquan@tju.edu.cn

however are not sufficient or accurate in describing such a complex concept as trust. Furthermore, they have not considered the correlation between the QoT attributes, and the utility function is too simple to embrace all QoT attributes. In the work[6], the path with the maximal trust value was selected as the most trustworthy social trust path. Unfortunately, some important aspects between the adjacent trust nodes and the recommendation roles of a participant have a significant influence on trust propagation[7,8]. But these factors have not been considered in existing social TPS solutions.

After all, trust cannot be described or evaluated by considering only several aspects. Furthermore, scientists from multi-discipline have different viewpoints about the concept of trust and the influence factors of trust[9]. Practically, the limited number of influence factors considered in the existing algorithms[2,5,11–15] can not deliver enough performance. The probable solution is to consider as many influence factors as possible to get the approximate solution.

In this paper, considering the complexity of trust and its propagation[10], we will first introduce a concept of complex trust network which contains the complex trust information, trust relationships and other important aspects of trust and reputation. We then propose a model to represent the trust network based on high dimensional vector and matrix, finally the TPS problem is modeled as Multi-Constrained Optimal Path (MCOP) selection problem, which is a NP-Complete problem[2]. Therefore, we propose an approximate algorithm TPS_DR, Trust Path Selection based on Dimensionality Reduction, to simplify the trust network through reducing the dimensionality of the trust vectors so as to cluster some similar nodes into a supernode. Our example demonstrates the usage and advantage of our model.

2. Complex Trust Network

As for selecting the optimal trust path for sink node, the critical problem is to construct a feasible trust network, which contains the needed trust information while contains the least unrelated information. In existing trust network, nearly most researchers have considered only several aspects about trust to simply the process of trust path selection. However, as we have argued in the previous section, considering the reasonability and accuracy, it is inappropriate to consider only partial influence factors when we solve such a problem as TPS.

In this paper, we will not consider the specific trust attributes but try to build up a prototype which contains enough attributes as demanded in various circumstances. That is to say, we aim at construct such a trust network to model a complex sensor network. Each trust node contains enough attributes and they are interconnected by some trust relationships. For example, in Fig. 1, source and target participants, named trustor and trustee, contain tens of thousands of attributes, while the trust relationships are represented by the arrows between the trust nodes. It should be noted that here we just give out the prototype or template of the trust network, without considering the specific attributes and relationships. In one case, if we regard trustor as an employer who is seeking a potential employee in the large scale social network, trustor can evaluate the trustworthiness of trustee along some trust path selected by some strategies. In another case, we can regard trustor as one of some other participants, such as a traveler who is seeking a trustworthy travel route in the complex travel network. In a word, our trust network can represent a lot of real applications, without the requirement of considering the specific attributes for each node or the relationships among nodes. In this way, we can construct a complex trust network to describe the complex and uncertain relationships between nodes in the online social network. For example, in Fig. 2, trustor S1 tries to select a trust path to get trustee T1. It is known that, for a trust network with n nodes, there are at most N paths:

$$N = 1 + C_{n-2}^{1} + 2! \times C_{n-2}^{2} + 3! \times C_{n-2}^{3} + \cdots + (n-3)! \times C_{n-2}^{n-3} + (n-2)! \times C_{n-2}^{n-2} \geq 2^{n-2} \quad (1)$$

If n=100, then $N \geq 2^{98}$, which is a quite large number that we can hardly examine all solution space quickly. Therefore, to solve such a solution explosion problem, we must reduce the solution space to an acceptable size.

2.1. Trust Vector Representation

As we mentioned above, trust is a complex concept influenced by dozens of aspects. General solutions tried to find out the key influence factors, however it is not enough and also inconsistent with the facts in most cases. In this paper, we will not care about what on earth the influence factors are, and the only thing we want to do is describing each trust node with a high dimensional vector A:

$$A = [X_1, X_2 \ldots, X_n]^T \quad (2)$$

Where $[.]^T$ denotes the transpose of the vector, $X_1, \ldots, X_n$ are the values of all complete trust attributes involved. In this way, we can construct a matrix N to represent the complex trust network:

$$N = \begin{pmatrix} T_{11} & \cdots & T_{1n} \\ \vdots & \ddots & \vdots \\ T_{m1} & \cdots & T_{mn} \end{pmatrix} \quad (3)$$

Here T_{ij} is the trust relationship vector between trust nodes i and j, if $i \neq j$, T_{ij} stands for the trust degree that trustor i transfer to trustee j, otherwise T_{ij} is the self-confidence of trust node i. Please note that the difference between two concepts of trust vector of a node (A) in Equation (2) and trust relationship between nodes (T_{ij}) in Equation (3).

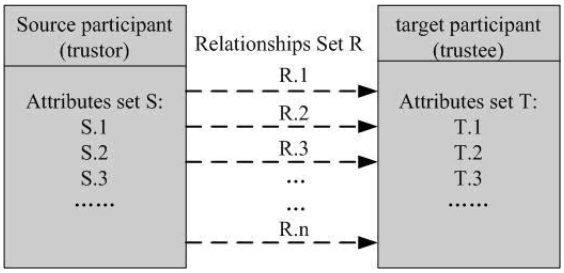


Figure 1 The structure of our trust network.

2.2. The Similarity of Trust Nodes

In order to simplify the complex trust network, we need to cluster the similar trust nodes into a collective (or supernode). The first question remains: how do we assemble the similar trust nodes into a supernode? The method we here use is through judging the similarity degree of trust nodes. If the similarity is high, then such nodes can be clustered into a collective, such is the super-node. Then how to compute the degree of similarity of each trust node? In this paper, we judge the similarity by computing the dot product of two nodes vectors, if the value is close to one, then the degree of similarity of such two nodes is high; otherwise the degree is low. Please note that all trust vectors should be normalized before they are computed by dot product. For example, there are two trust nodes represented by the corresponding vectors A and B:

$$A = [X_1, X_2 \ldots, X_n]$$
$$B = [Y_1, Y_2 \ldots, Y_n]$$

, then normalize them by:

$$A' = \frac{[X_1, X_2 \ldots, X_n]}{|A|} = \frac{[X_1, X_2 \ldots, X_n]}{\sqrt{X_1^2 + X_2^2 + \cdots + X_n^2}} \quad (4)$$

$$B' = \frac{[Y_1, Y_2 \ldots, Y_n]}{\sqrt{Y_1^2 + Y_2^2 + \cdots + Y_n^2}} \quad (5)$$

So the dot product (similarity degree S_{AB}) is computed as follows:

$$S_{AB} = A' \bullet B' = \frac{X_1Y_1 + X_2Y_2 + \cdots + X_nY_n}{\sqrt{\left(\sqrt{X_1^2 + X_2^2 + \cdots + X_n^2}\right)\left(\sqrt{Y_1^2 + Y_2^2 + \cdots + Y_n^2}\right)}} \quad (6)$$

Definition 1 (similar node) *If the dot product of two trust vectors is close to 1, then the two trust nodes (A and B) are called similar nodes, one is the similar node of the other, that is: $A \cong B$, here $\cong$ is symbol of similarity.*

Of course, the threshold value is preferred by different participants. If two trust nodes are similar, then they have also similar or congenial values of trust. In the process of trust path selection, if there is no equivalent node to replace some selected node, then similar node can be replaced approximately.

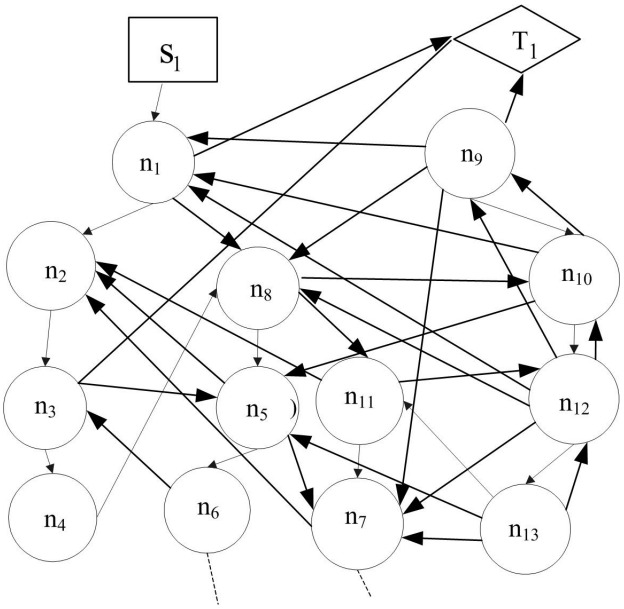


Figure 2 An example of complex trust network

Definition 2 (the matrix of similarity) *For the complex trust network, we use the matrix of similarity S_N to describe its similarities between all nodes, that is:*

$$S_N = \begin{pmatrix} S_{11} & \ldots & S_{1n} \\ \vdots & \ddots & \vdots \\ S_{m1} & \cdots & S_{mn} \end{pmatrix} \quad (7)$$

Here S_{ij} is the similarity of trust nodes i and j ($i \neq j$), whose value is decided by Equation (6). Obviously, $S_{ij} = 1$ if condition $i = j$ is met, this is a matter of course. Then Equation (7) is modified as:

$$S_N = \begin{pmatrix} 1 & \ldots & S_{1n} \\ \vdots & \ddots & \vdots \\ S_{m1} & \cdots & S_{mn} \end{pmatrix} \quad (8)$$

Furthermore, considering it is such an undirected relationship as similarity, we believe that:

$$S_{ij} = S_{ji} \quad (9)$$

Therefore, if S_N is a square matrix ($m = n$), it is a symmetric matrix, then we can only consider the left or right lower triangular equivalent form.

2.3. The Equivalence of Trust Node

Besides the relationship of similarity, there is another relationship of equivalence. These two concepts are

related tightly, similar nodes can be equivalent if some pre-conditions are met, and on the contrary equivalent nodes are surely similar.

Definition 3 (equivalent node) *If the degree of similarity of two trust nodes (A and B) is high enough, then such two trust nodes can be regarded as equivalent nodes, one is the equivalent node of the other, that is: $A \equiv B$, here $\equiv$ is symbol of equivalence.*

Some pessimistic trustors can sometimes believe that two nodes are equivalent only if the value of similarity is higher than 0.9, while the optimistic ones may assume the threshold value is 0.6. That is to say, the threshold value is vague and different between various trust decision-makers (trustors). We define the equivalence of two trust nodes as such: if two nodes are equivalent, then the value of equivalence is one, otherwise zero, that is:

$$E_{AB} = \begin{cases} 1, & A \equiv B \\ 0, & otherwise \end{cases} \tag{10}$$

Definition 4 (the matrix of equivalence) *we use the matrix of equivalence E_N to describe its similarities between all nodes,*

$$E_N = \begin{pmatrix} E_{11} & \dots & E_{1n} \\ \vdots & \ddots & \vdots \\ E_{m1} & \cdots & E_{mn} \end{pmatrix} \tag{11}$$

Here E_{ij} is the equivalence of trust nodes i and j, whose value is decided by Equation (10). Obviously, $E_{ij} = 1$ ($i = j$), and $E_{ij} = E_{ji}$. Then Equation (11) is modified as:

$$E_N = \begin{pmatrix} 1 & \dots & E_{1n} \\ \vdots & \ddots & \vdots \\ E_{m1} & \cdots & E_{mn} \end{pmatrix} \tag{12}$$

Similarly, if E_N is a square matrix ($m = n$), it is a symmetric matrix, then we can only consider the left or right lower triangular equivalent form.

Generally, if the matrix of similarity is given, then we can derive the matrix of equivalence by fuzzy inferring methods, surely the results can be influenced greatly by selecting different membership functions.

3. An Example

For another example as Fig. 3 shows, there are two trust nodes A and B denoted by (for simplicity, we assume there are only ten kinds of influence factors) in the trust network shown as Fig. 3.

$$A = [0.6, 0.4, 0.7, 0.7, 0.2, 0.8, 0.9, 0.6, 0.5, 0.7]$$

$$B = [0.3, 0.4, 0.6, 0.5, 0.9, 0.4, 0.7, 0.8, 0.2, 0.8]$$

First we can get $|A||B| \approx 2.02 * 1.91 = 3.8582$, the dot product is:

$$S_{AB} = A' \bullet B' = \frac{3.38}{3.8582} \approx 0.8761$$

Similarly, if there are four trust nodes in a trust network and the other values of similarity are respectively given as Fig. 3, then the matrix of similarity S_N is:

$$S_N = \begin{bmatrix} S_{AA} & S_{AB} & S_{AC} & S_{AD} \\ S_{BA} & S_{BB} & S_{BC} & S_{BD} \\ S_{CA} & S_{CB} & S_{CC} & S_{CD} \\ S_{DA} & S_{DB} & S_{DC} & S_{DD} \end{bmatrix} = \begin{bmatrix} 1 & 0.8761 & S_{AC} & S_{AD} \\ S_{BA} & 1 & S_{BC} & S_{BD} \\ S_{CA} & S_{CB} & 1 & S_{CD} \\ S_{DA} & S_{DB} & S_{DC} & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0.8761 & 0.8873 & 0.7896 \\ 0.8761 & 1 & 0.2453 & 0.6543 \\ 0.8873 & 0.2453 & 1 & 0.9235 \\ 0.7896 & 0.6543 & 0.9235 & 1 \end{bmatrix}$$

If we set the threshold value of equivalence is 0.8, namely, our fuzzy inferring rule is: if the value of similarity is equal to or higher than 0.8, then such two nodes are equivalent. Thus, we can get the matrix of equivalence E_N as follows:

$$E_N = \begin{bmatrix} E_{AA} & E_{AB} & E_{AC} & E_{AD} \\ E_{BA} & E_{BB} & E_{BC} & E_{BD} \\ E_{CA} & E_{CB} & E_{CC} & E_{CD} \\ E_{DA} & E_{DB} & E_{DC} & E_{DD} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

Obviously, we can see that all nodes are equivalent, and then such a trust network can be looked as a super-node.

Considering the fact that there is different personal preference between various trustors in different circumstances, it depends on personal preference in a specific case that whether such two nodes be clustered into a collective (super-node) or not.

4. TPS_DR: Optimal Trust Path Selection Algorithm

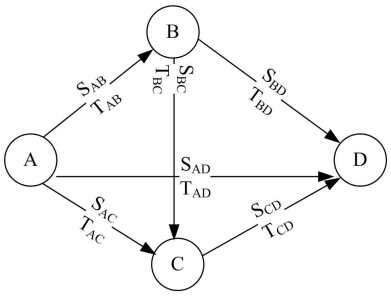
We design our algorithm on condition that the following proposition is true, which is about the judgment of equivalent node.

Proposition 1 (judgment of equivalence) If two nodes are equivalent, then their previous nodes and next nodes are equivalent respectively, and vice versa. Note that similarity is directed while equivalence is undirected, and we adopt trust node-pair graph method to judge the equivalence just like using state-pair graph in timed circuit.

Our algorithm studies from the related principle in digital logics, which is used to delete the extra states in states graph. The algorithm can be described as follows.

Step 1 Define related variables.

Related variables include: vector representation of trust relationship-T_{ij}, similarity degree/matrix (S_{ij}/S_N),



S_{AB}=0.8761 T_{AB}=0.6788
S_{AC}=0.8873 T_{AC}=0.9044
S_{AD}=0.7896 T_{AD}=0.8874
S_{BC}=0.2453 T_{BC}=0.3422
S_{BD}=0.6543 T_{BD}=0.7762
S_{CD}=0.9235 T_{CD}=0.9880

Figure 3 an example of trust network with four nodes

equivalence degree/matrix (E_{ij}/E_N]), and threshold value of similarity (ToS), Complex Trust Network representation (CTN).

Step 2 Initialize and assign for equivalence matrix.

a) Set a threshold value of similarity, ToS, any two nodes with greater value than which are regarded as equivalence, otherwise is inequivalent.

b) If two nodes are equivalent, E_{ij}=1; otherwise E_{ij}=0.

Step 3 Judgment for equivalent nodes.

Here we exercise trust node-pair graph method. Considering the scale is often too large, we will divide the CTN into a fit number of sub-networks. The detailed flows are:

a) Construct part trust node-pair from the similarity matrix, since the scale is large enough;

b) For each given node-pair till all node-pairs are contained, considering the equivalence for all previous nodes and next nodes, if yes, then we can make sure they are equivalent, otherwise turn to d);

c) Identify the equivalent node-pairs, based on which we modify the equivalence matrix E_N and renew it in the related context.

d) Checking whether there are new equivalent node-pair, if yes, turn to b);

e) If no, stop.

Step 4 Reducing method for CTN.

For all equivalent nodes, all but one should be deleted. The previous and next nodes for the deleted nodes can be forwarded to their equivalent node still alive in CTN. In this way, we have finished simplifying the CTN, which is especially meaningful for large scale CTN, hence contributes much for our TPS problem.

Step 5 Trust Path Selection process.

This process is out of our discussion scope in this stage, we will consider it in more detail in the near future.

a) Use k nearest neighbor algorithm, k-NN.

b) Compute the absolute value of trust vector T_{ij}, called utility function, which is different from that of others in that we will not consider the detailed designation of this function but just give a universal vector representation.

c) Search the path forwardly, till we have found the target node-trustee, or when we have arrived at the seventh search.

This process is about trust path selection, which is our future goal after giving the simplified trust network. Our algorithm is based on the small world theory, that is, there is a maximum step-number between the source and the target, the number is six. Therefore we will stop our search process at the seventh step.

5. Conclusions and future work

In this paper, we addressed the TPS problem for sink node in complex sensor network, which is a Multi-Constrained Optimal Path (MCOP) selection problem, hence a NP-complete problem. Comparing with other work, our work will not consider the detailed trust computing method but regard the trust relationship of each sensor node as a trust vector, which makes it universal in trust metrics. To solve TPS problem, our work aims at reducing the scale of the complex sensor network, which is transferred to the treatment with its complex trust network. As for the properties of software systems, trustworthiness is critical [16], so our near future tasks is to reduce the scale of trust network to get the complete solution for TPS problem.

Our more future work includes adopting fuzzy reasoning to map the similarity matrix into equivalence matrix, applying various utility functions to verify the advantage of our algorithm. Even more importantly, more examples should be taken to exemplify the usage of our algorithm and experiments should be made to compare the other work.

6. Acknowledgements

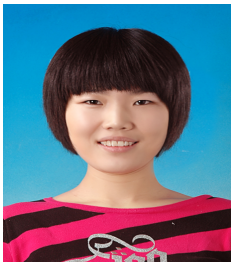
The authors would like to thank the editors and anonymous referees for their suggestions and the remarkable improvements they brought to this paper. This paper has been supported by the National Natural Science Foundation of China (No. 61003080, 61070202), the Opening Project of Shanghai Key Laboratory of Integrate Administration Technologies for Information Security, Tianjin Research Program of Application Foundation and Advanced Technology (No.10JCZDJC15700) and the program of the 985 project of Tianjin University.

References

[1] J. Kunegis, A. Lommatzsch, and C. Bauckhang, The Slashdot zoo: mining a social network with negative edges. In WWW-09 (2009), 741-750.

[2] T. Korkmaz, and M. Krunz, Multi-constrained optimal path selection. In: INFOCOM-01 (2001), 834-843.
[3] Ahyoung Lee, Ilkyeun Ra, A Queuing Network Model Based on Ad Hoc Routing Networks for Multimedia Communications, APPLIED MATHEMATICS & INFORMATION SCIENCES, vol. **6**, si: 1, JAN 2012, pp. 271-283.
[4] Lei Li, Yan Wang, and Ee-Peng Lim, Trust-Oriented Composite Service Selection with QoS Constraints, Journal of Universal Computer Science (2010), vol. **16**, no. 13, 1720-1744.
[5] Guanfeng Liu, Yan Wang and Mehmet Orgun, Optimal Social Trust Path Selection in Complex Social Networks, the Twenty-Fourth AAAI Conference on Artificial Intelligence, Atlanta, Georgia, USA (2010), July 11-15, pp. 1391-1398.
[6] C. Hang, Y. Wang, and M. Singh, Operators for propagating trust and their evaluation in social networks, In: AAMAS-09 (2009), 1025-1032.
[7] P. S. Adler, Market, hierarchy, and trust: The knowledge economy and the future of capitalism, Organization Science (2001), **12(12)**: 215-234.
[8] R. Miller, D. Perlman, and S. Brehm, Intimate Relationships, McGraw-Hill College, 4th edition (2007).
[9] T3 Trust Theory. http://t3.istc.cnr.it/trustwiki/index.php/T3_Trust _Theory (2011-5-12).
[10] J. Golbeck, and J. Hendler, Inferring trust relationships in web-based social networks, ACM Transactions on Internet Technology (2006), **6(4)**: 497-529.
[11] Annika Hinze and Qiu Quan, Trust- and Location-Based Recommendations for Tourism, On the Move to Meaningful Internet Systems: OTM 2009, Lecture Notes in Computer Science (2009), Volume 5870/2009, 414-422, DOI: 10.1007/978-3-642-05148-7_31.
[12] Wei Guo, Ren-Zuo Xu, Bin Liu, Research on Subjective Trust Routing Algorithm for Mobile Ad Hoc Networks, 2010 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM), Chengdu, 23-25 Sept. (2010), pp. 1-6.
[13] Fernando Kuipers, Piet Van Mieghem, Turgay Korkmaz,Marwan Krunz, An Overview of Constraint-Based Path Selection Algorithms for QoS Routing, IEEE Communications Magazine, December (2002), pp. 50-55.
[14] Cuirong Wang, Xiaozong Yang, and Yuan Gao, A Routing Protocol Based on Trust for MANETs, In: H. Zhuge and G.C. Fox (Eds.): GCC 2005, LNCS 3795 (2005), pp. 959-964.
[15] T. Yu, Y. Zhang, and K.J. Lin, Efficient Algorithms for Web Services Selection with End-to-End QoS Constraints, ACM Transactions on the Web, Vol. **1**, No. 1, Article 6 (2007).
[16] Yanguo Michael Liu, Issa Traore, Properties for Security Measures of Software Products, APPLIED MATHEMATICS & INFORMATION SCIENCES, vol. **1**, no. 2, MAY 2007, pp. 129-156.

Guangquan Xu is a Ph.D. and associate professor at the School of Computer Science and Technology, Tianjin University, China. He received Ph.D. degree from Tianjin University in March 2008. He is a member of the CCF and ACM. His research interests include verified software, trusted computing, trust and reputation.



Xiuming Tian is a master student in computer science and technology at the School of Computer Science and Technology, Tianjin University, whose research interests include security and privacy, social network.

Xiaochun Cao is currently a professor of Computer Science at the School of Computer Science and Technology, Tianjin University. He received the B.E. and M.E. degrees both in computer science from Beihang University (BUAA), Beijing, China, and the Ph.D. degree in computer science from the University of Central Florida, Orlando, USA. His research interests include information security, computer vision.

Xiaohong Li is a professor of the School of Computer Science and Technology, Tianjin University, China. She received Ph.D. degree from the School of Computer Science and Technology, Tianjin University. Her research interests lie in trusted software, and information security.

Zhiyong Feng is a professor and associate dean of the School of Computer Science and Technology, Tianjin University, China. He received Ph.D. degree from the School of Mechanical Engineering, Tianjin University. His research interests lie in computer integrated manufacturing system (CIMS), knowledge engineering, distributed AI, pervasive computing and information security.