Mobile Internet Mobile Agent System Dynamic Trust Model for Cloud Computing

Weijin Jiang[1,2,3(✉)], Yirong Jiang[4], Yang Wang[1], Jiahui Chen[1], Yuhui Xu[1], and Li'na Tan[1]

[1] Institute of Big Data and Internet Innovation, Mobile E-business Collaborative Innovation Center of Hunan Province, Hunan University of Commerce, Changsha 410205, China
jlwxjh@163.com, 18508488203@163.com, 810663304@qq.com, 363168449@qq.com, 18785024@qq.com

[2] Key Laboratory of Hunan Province for New Retail Virtual Reality Technology, Hunan University of Commerce, Changsha 410205, China

[3] School of Computer Science and Technology, Wuhan University of Technology, Wuhan 430073, China

[4] Tonghua Normal University, Tonghua 134002, China
307553803@qq.com

Abstract. Aiming at the security and trust management of mobile agent system in cloud computing environment, based on the simple public key infrastructure (SPKI) trust mechanism, a mobile agent system objective trust peer management model (MAOTM) is proposed to solve the problem of identity authentication, operation authorization and access control in the mobile agent system; On this basis, the Human Trust Mechanism (HTM) is used to study the subjective trust formation, trust propagation and trust evolution law, and the subjective trust dynamic management algorithm (MASTM) is proposed. Based on the interaction experience between the mobile agent and the execution host and the third-party recommendation information to collect the basic trust data, the public trust host selection algorithm is given. The isolated malicious host algorithm and the integrated trust degree calculation algorithm realize the function of selecting the trusted cluster and isolating the malicious host, so as to enhance the security interaction between the mobile agent and the host. The simulations of the proposed algorithms are verified and proved to be feasible and effective.

Keywords: Cloud computing · Mobile agent system · Subjective trust · Objective trust · Dynamic trust management · Mobile internet

1 Introduction

Mobile Agent technology is an emerging technology that is multi-disciplinary and at the forefront of international research. It is also a product of the combination of Agent and Internet in the field of artificial intelligence [1, 2]. In the cloud computing environment, the problem of resource allocation of trust security domain in mobile agent system is studied [3, 4]. It can track international cutting-edge technology, enrich the theoretical system of cloud computing and mobile agent model, and enhance the

Y. Tang et al. (Eds.): HCC 2018, LNCS 11354, pp. 169–180, 2019.
https://doi.org/10.1007/978-3-030-15127-0_17

security performance of agent application system. It can also promote the further development of China’s distributed application technology and information security technology, and promote the application of mobile agent technology in various fields in the cloud computing environment. Therefore, it is of great theoretical and practical significance to study the security trust problem in the design of mobile agent system in cloud computing environment.

2 Related Work

Currently, the trust management model is generally divided into two categories. The first category is the objective rational model. It uses a rational and accurate method to express and deal with complicated trust relationships, and has objective and static management characteristics. The second category is the subjective empirical model, which considers trust to be a subjective judgment of a particular level of specific characteristics or behavior of the object. The method of fuzzy sets is used to study direct trust management [5]. Because trust is somewhat uncertain, fuzzy logic is used to deal with the subjectivity and uncertainty of subject characteristics and behavior cognition, and make decisions based on the determined trust strategy to provide effective support for judging trust status. Trust and the issue of non-trust communication. Guba-oriented e-commerce system presents a complete set of belief propagation algorithm, part of the expression of trust between individuals for predicting the level of trust between any two objects. The Simple Public Key Infrastructure (SPKI) method was proposed by Carl Ellison and Bill Frantz in 1996 [6], which was standardized by the IETF in 1999. The SPKI standard now in use is a mixed version of SPKII/SDSI.

The complexity of trust verification is reduced from the design principle. Compared with the existing PKI-based trust management model, the trust certificate is designed based on SPKI+RBAC, which reduces the complexity of trust verification. It can not only meet the requirements of trust transfer and verification of mobile agent system, but also control the attenuation of trust caused by excessive trust chain [7–9].

3 Dynamic Management Method of Subjective Trust in Mobile Agent System

3.1 Mobile Agent System Subjective Trust Dynamic Management Model

Subjective Trust Dynamic Management Model

The composition of the mobile agent system supervisor trust model is shown in Fig. 1 [9, 10]. Bottom-up: The host is in an open and dynamic network environment. The mobile agent platform (MAP) is located on the host, providing an Execution Environment (EE) for the mobile agent. The MAP is not necessarily unique on a host, but more than one. The subjective trust management model is located in the MAP and consists of three trust components: They are the trust formation component, the trust propagation component, and the trust evolution component. These management mechanisms provide a trust interaction context for the interaction entity, and monitor the interaction process and behavior of the entity. The trust formation component mainly implements the collection and calculation of trust data, the trust propagation component mainly implements the protocol exchange of trust data, and the trust evolution component mainly realizes the update of the trust data.

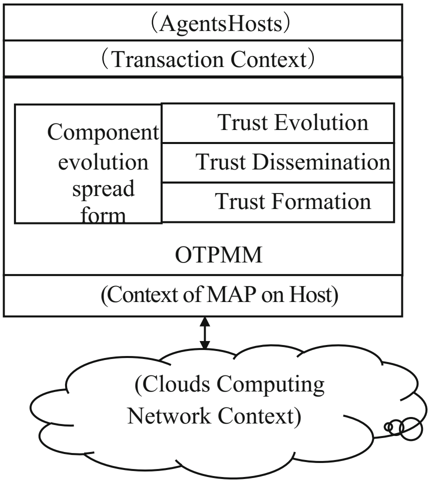

Fig. 1. Subjective trust dynamic management model

Basic Trust Data Collection

1. Recommended trust basic data collection

If the host H_a does not have sufficient understanding of the host H_x, envisaging the host H_a want to interact with the host H_x in order to evaluate the degree of trusted host H_x, and the host H_a queries a group of hosts $\{H_1, H_2, \ldots, H_K\}$ for trust information about host H_x. The recommendation information between the entities defined here is a set of "basic data" about the interaction behavior rather than the comprehensive calculation result of the degree of trust. The purpose of this is to eliminate the accumulation of errors generated during the comprehensive calculation of the degree of trust, and the impact of the trust preferences of the host $\{H_1, H_2, \ldots, H_K\}$. Under normal circumstances, the number of interactions between the host H_i and entity H_x is n, the number of successes n_1, the number of failures n_2, $D_{ki} = n_1/(n_1 + n_2)$, $(\mathrm{n} = n_1 + n_2)$.

2. Direct trust data collection

Assume that the host H_a has to interact n times with the host H_x directly, successes n_1, failures n_2, $(\mathrm{n} = n_1 + n_2)$. $D_{ax} = n_1/(n_1 + n_2)$ to reflect the host H_a and host H_x interactive behavior is good or bad, which is defined as a direct trust underlying data.

3.2 Recommended Trust Computation Algorithm

We divide the continuous system runtime into equally spaced inspection periods, each of which is called a "time frame" and is represented by τ(τ = 1, 2, …, n). Then the interactive behavior of the interactive host is transformed into the quantitative calculation of trust degree. The Gaussian probability distribution theory is used to improve the average algorithm, and a more optimized algorithm is given. The algorithm is as follows.

Algorithm 1. Recommended trust computation algorithm.

Initialization: let the basic data received by host H_x about host H_a be: $\{D_1, D_2, \ldots, D_k\}$, where: $D_i = n_1/(n_1 + n_2)$, (0 ≤ D ≤ 1). n_1 is the number of positive interaction results about host H_x collected from M_i during the inspection period, and n_2 is the number of negative interactions.

Step 1: The data on the host recommended averaging and variance are calculated as follows:

$$\bar{D} = \frac{1}{k}\sum_{i=1}^{k} D_i, S^2 = \frac{1}{k}\sum_{i=1}^{k} (D_i - \bar{D})^2 \tag{1}$$

Step 2: Order: $\mu = \overline{D}$, $\sigma^2 = S^2$, according to a Gaussian distribution theory, that $K(\mu, \sigma^2)$ use as the characteristic parameters for a random variable T, T obtained probability density functions $p(x)$, which (μ, σ^2) are called expectation and variance of the Gaussian distribution. When μ = 0, $\sigma^2 = 1$, the time, T is called the standard normal distribution.

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} (\sigma > 0), (-\infty < \mathrm{x} < +\infty) \tag{2}$$

Step 3: the possibility of random variable T in (−∞, v), (v, +∞) appearing in the range can be obtained. Wherein, P (≤ V) T indicates the possibility of appearing in the range of v or less, P (> v) T indicates the possibility of occurring within the range of greater than v.

$$P(\leq v) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\frac{v-\mu}{\sigma}} e^{-\frac{x^2}{2}} dx, \ \mathrm{P}(> \mathrm{v}) = \frac{1}{\sigma\sqrt{2\pi}} \int_{\frac{v-\mu}{\sigma}}^{\infty} e^{-\frac{x^2}{2}} dx$$

For a given interval value (v_1, v_2), T appears in the specified range possibilities:

$$\mathrm{P}(v_1, v_2) = \frac{1}{\sigma\sqrt{2\pi}} \int_{\frac{v-\mu}{\sigma}}^{\frac{1-\mu}{\sigma}} e^{-\frac{x^2}{2}} dx, \quad (v_1 < v_2) \tag{3}$$

Step 4: Then the variable T in the specified range (v, 1), the possibility [0, 1] appear respectively:

$$\mathrm{P}(\mathrm{v}, 1) = \frac{1}{\sigma\sqrt{2\pi}} \int_{\frac{v-\mu}{\sigma}}^{\frac{1-\mu}{\sigma}} e^{-\frac{x^2}{2}} dx, \mathrm{P}(0, 1) = \frac{1}{\sigma\sqrt{2\pi}} \int_{\frac{0-\mu}{\sigma}}^{\frac{1-\mu}{\sigma}} e^{-\frac{x^2}{2}} dx \tag{4}$$

Step 5: Then calculated variables (v, 1) within the scope of the likelihood ratio in the range [0, 1] and T:

$$P_{ax}(v) = \frac{P(v, 1)}{P(0, 1)} = \int_{\frac{v-\mu}{\sigma}}^{\frac{1-\mu}{\sigma}} e^{-\frac{x^2}{2}} dx \Big/ \int_{\frac{v-\mu}{\sigma}}^{\frac{1-\mu}{\sigma}} e^{-\frac{x^2}{2}} dx \tag{5}$$

Step 6: The host H_a recommended level of trust on the host H_x (0 < v <1) defined as the ratio (v) is $P_{ax}(\mathrm{v})\,(0 < \mathrm{v} < 1)$. Referred to as $T_{x-rec}(v)$, $T_{x-rec}(v) = P_{ax}(\mathrm{v})$ $(0 < \mathrm{v} < 1)$, where v is the calculated threshold value.

3.3 Comprehensive Calculation of Trust

Considering the direct trust level T_{x-dir} of H_a to H_x, and the recommended trust level T_{x-rec} of H_x collected by H_a, define a variable that comprehensively measures trust, called "Trust Degree". "Trust Degree" is used to quantitatively represent the trustworthiness of interactive hosts in a mobile agent system, which means the possibility that host H_a interacts with another host H_x in a given environment to obtain a positive result (trust: $T_x = 1$). T_x is a comprehensive measure of trust that is used to quantitatively evaluate the current level of trustworthiness of an entity in a mobile agent system and to predict the degree of trust in the next interaction with that entity. The following is a "trust" comprehensive calculation process.

During the same study period, the direct trust level T_{x-dir} and the recommended trust strength T_{x-rec} are weighted and summed, where ρ is the confidence coefficient.

$$T_x = \rho\, T_{x-dir} + (1 - \rho) T_{x-rec} (0 < \rho \leq 1) \tag{6}$$

The algorithm implements two functions: first, if entity H_a repeatedly interacts with entity H_x and finds that H_x continues to maintain good behavior (affirmative event), H_x's trust degree Tx will continue to grow, tending to a maximum of 1, and if entity H_x has malicious behavior, its trust will drop rapidly; second, if there is a large change in the degree of trust between time frames n-1 and n, this large amount of change H_x^n will have a large effect, and vice versa. $\sigma(\tau)$ known as the coefficient update control the update speed of trust.

$$T_x^{n+1} = T_x^{n-1} + \sigma(\tau)(T_x^n - T_x^{n-1}) \tag{7}$$

4 Simulation Experiment Analysis

The key properties and parameter selections of Eqs. (1) to (7) given in the above dynamic trust metrics and evaluation methods are verified by a series of simulation experiments [12–15].

4.1 Experimental Conditions

Experiment 1 examines the algorithm of Eq. (1), and the result is shown in Fig. 2. Take host H_a to investigate multiple interactions with H_x as an example to illustrate [11]: The more bad interactions of host H_x, the higher the value converted to the underlying trust data D_{ax}. If H_x always maintains good behavior, when the interaction time reaches $\tau = 50$ frames, the value of D_{ax} is almost equal to 1; conversely, the more bad interactions of host H_x, the lower the value converted to the base trust data D_{ax}. If the bad behavior persists, the value of D_{ax} is almost equal to 0 when the interaction time reaches $\tau = 50$ frames. If the host's H_x interaction behavior is mixed, the value of D_{ax} fluctuates around 0.5. Therefore, D_{ax} in Eq. (1) can correctly reflect the degree of behavior of H_a in the series of interactions with host H_x. When K does not understand H_x, H_a considers $D_{ax} = 0.5$.

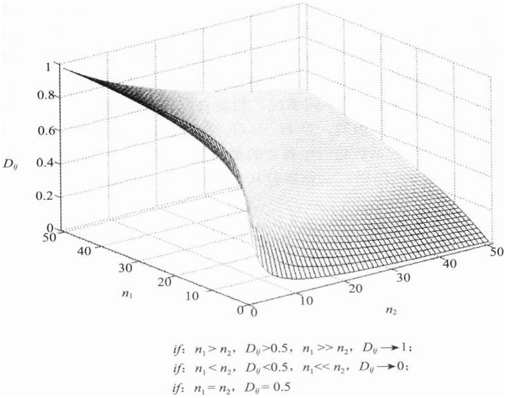

Fig. 2. Verify Dij(n1, n2), (t = 1, 2, …,50)

Experiment 2 examines the algorithm of Eq. (5), and the result is shown in Fig. 3. Let the trust demand threshold be T_o, and let $v = T_o$, you can see the change of the direct (or recommended) trust degree T_x of the host H_x when calculating the base value v (depending on the collected data is direct or recommend trust data). It can be seen that the higher the T_o value, the higher the expectation requirement for the host satisfying $T_x > T_o$, and the fewer the number of hosts satisfying the condition. For example, when the calculation base value v = trust threshold $T_o = 0.8$, it can be seen from Fig. 3 that only the candidate host whose mathematical expectation value of the host interaction behavior $\mu > 0.6$ can be included in the trusted interaction object.

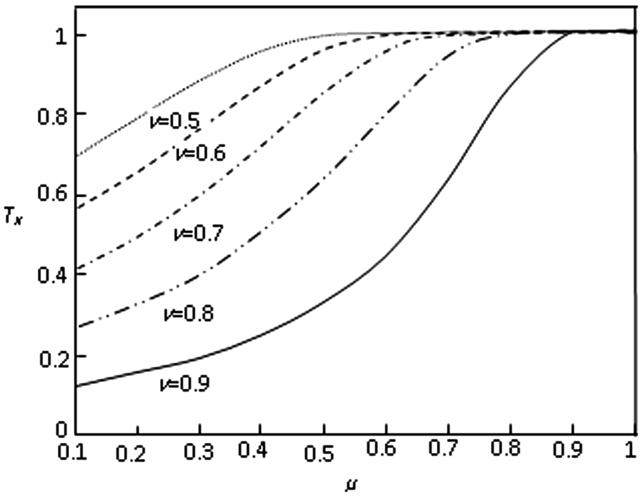

Fig. 3. The relationship between the degree of Hx trust and the mathematical expectation of its behavior at different v values

Experiment 3 verifies the result is shown in Fig. 4. The main consideration here is that if the host H_a is a new member, it wants to interact with H_x. Since there is no direct empirical data, the initial value of the confidence factor is 0, and the trust information must first be collected from the third-party entity. As the interaction experience increases, the confidence factor gradually increases. (a, b) is a pair of constants, a is the maximum confidence factor, and b is used to control the growth rate of the confidence factor according to different situations. The experimental results show that when (a, b) takes different values, when the number of interactions between H_a and H_x increases gradually, the confidence coefficient ρ gradually increases smoothly, and the confidence coefficient approaches the maximum value. In the experiment, three sets of data were selected for comparison. When the time frame $\tau = 40$, the confidence coefficient almost reached their respective maximum values.

Experiment 4 verifies the results are shown in Fig. 5. The amount of change between the two time frames is $0 < \Delta T_{ax} < 1$, and the parameter w controls the evolution speed of T_{ax}. The experimental results show that when the coefficient w = 1, $\sigma(\tau)$ min = 0, the evolution rate is the slowest, and $\sigma(\tau)$ max = 0.46 has the fastest evolution speed. At this time, the maximum influence of ΔT_{ij} on the T_{ax} value in the next time frame is 0.46. As can be seen from Fig. 9, when the parameter is selected w = 2, the trust will get faster the speed of evolution, but the stable region of the algorithm $\Delta T_{\max}$ is around 0.77. When choosing w = 1.5, $\Delta T_{\max}$ is around 1. If compared with w = 2, Tax has a slower rate of evolution; if compared with w = 1, T_{ax} has a faster rate of evolution.

The experimental results obtained according to the trust degree synthesis algorithm (14) are shown in Fig. 6. It can be seen that the algorithm has a slow rising fast falling feature. If host H_x continues to maintain good interaction behavior for multiple time frames, host H_x can gradually gain high trust; after obtaining high trust, H_x suddenly

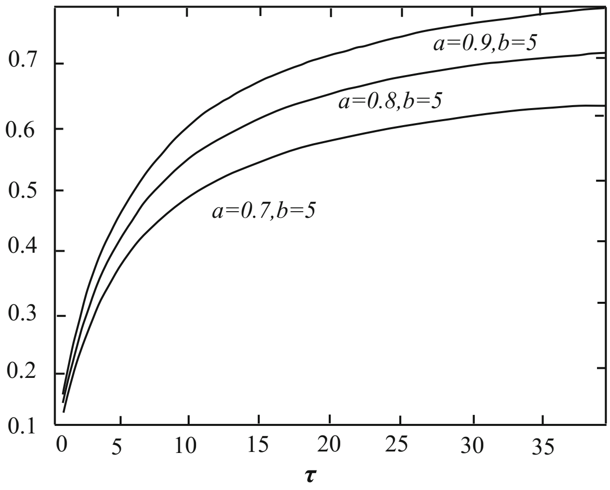

Fig. 4. ρ (τ) changes in the trend

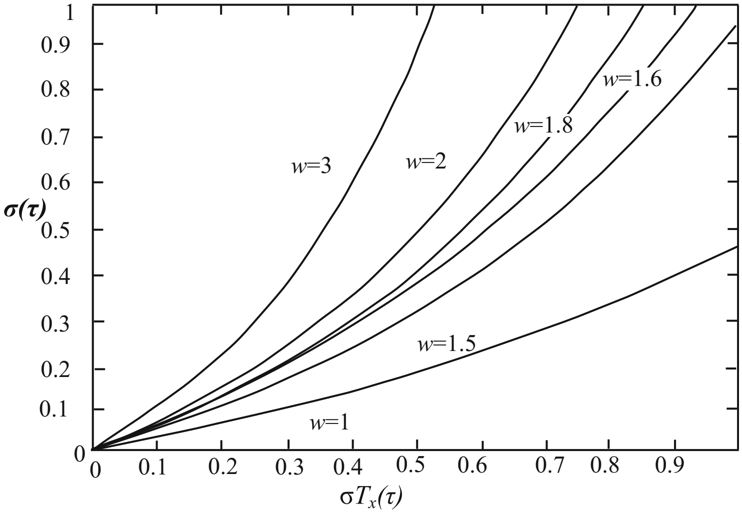

Fig. 5. Change trend of σ (τ) at different w

implements malicious behavior in the interaction to obtain illegal benefits, his trust will decline rapidly, be seen by H_a and spread through trust, so that the trusted group is isolated. If H_x wants to restore the higher trust he once had, it needs to make long-term efforts to maintain good interactions in order to restore his “trust”. This main feature of the trust degree comprehensive calculation algorithm effectively suppresses the malicious behavior of the host.

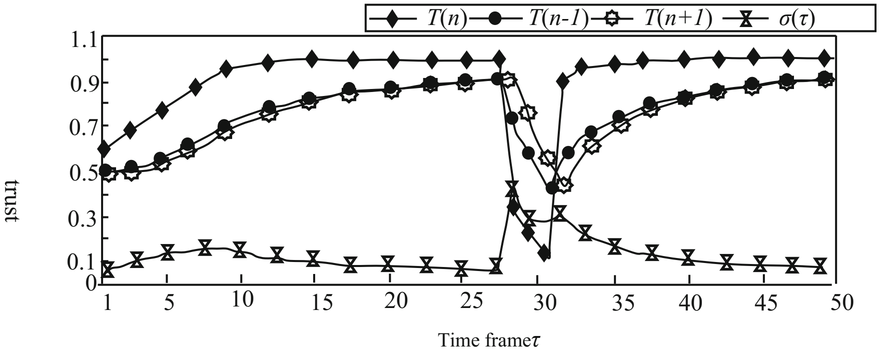

Fig. 6. Changes in trust trends (w = 1.5)

Given an algorithm for host H_a to isolate the host M_k (k = 1, 2, …) with bad or malicious recommendation behavior based on direct empirical data on host H_x. The simulation experiment results are illustrated as follows in Figs. 7, 8 and 9: it can be seen from Fig. 7 that the recommendation data of the recommender M_3 to the host H_x is the best with the direct empirical data of the host H_a to H_x. Depending on the algorithm used, H_a can conclude that: M_3 is more believable, and M_1 and M_2 may be suspected of malicious recommendations.

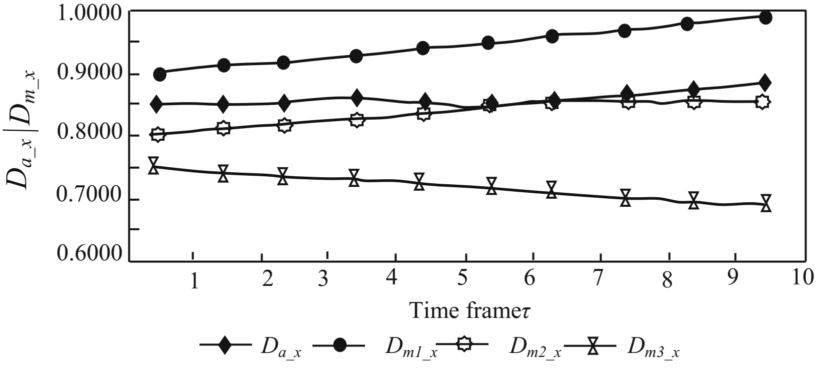

Fig. 7. Comparison of direct empirical data with recommended data

The effect of the difference between the direct empirical data and the recommended data on T_{m_k} is shown in Fig. 9.

It can be seen from Fig. 9 that according to equations in the course of the investigation, the recommendation data of the recommender 3 to the host H_x and the direct empirical data of the host H_a to H_x are the best. Host H_a's direct trust to him gradually increases, and the degree of improvement depends on the degree of consistency. The higher the consistency, the faster the increase; the recommendation data of

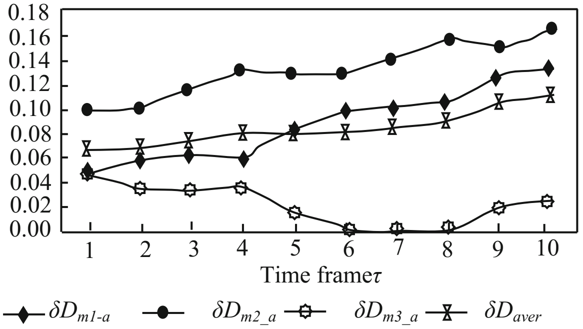

Fig. 8. Differences between direct empirical data and recommended data

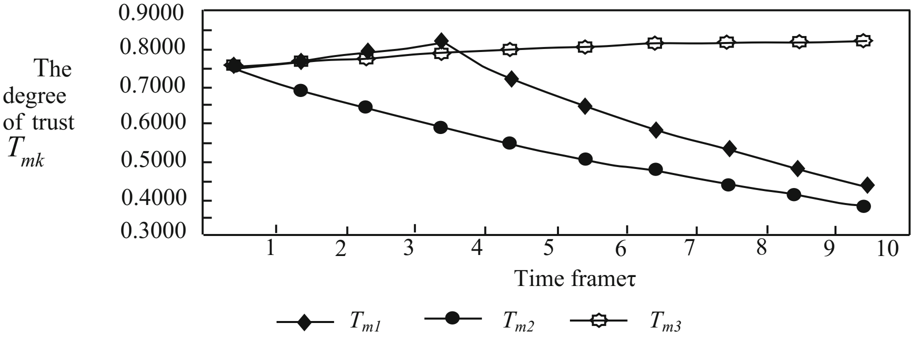

Fig. 9. The effect of the difference between the direct empirical data and the recommended data on T_{m_k}

recommenders M_1 and M_2 is inconsistent with the direct empirical data of H_a. Therefore, the functions of the equations can effectively isolate the malicious recommendation host.

Therefore, the following conclusions can be drawn: the simulation results verify the correctness of the "public letter host selection algorithm", "isolated malicious recommender algorithm" and "trust level comprehensive calculation algorithm" given in this paper [16]. It can be used to evaluate the subjective trust status of the host to be interacted in the mobile agent system, and predict the trustworthiness of the host to be interacted in the next time frame. The series of algorithms given can stimulate the trusted host and isolate the malicious host, which has the function of "punishing evil and promoting good". It can effectively manage the subjective trust dynamic management of the mobile agent system.

5 Conclusion

Divide the trust problem in the mobile agent system into objective trust and subjective trust to divide and conquer. Based on the analysis of the use of SPKI related certificates, this paper studies the problem of dynamic management of subjective trust in mobile agent system under the objective trust management framework of SPKI-based mobile agent system. The trust requirement of the entity (host or mobile agent) in the mobile agent system is analyzed, and a subjective trust dynamic management model consisting mainly of three trust components is proposed. Among them, the trust forms a component and completes the collection of trust data; the trust propagation component completes the communication of trust and the exchange of trust data; trusts the evolution component and completes the update of the trust data. The quantitative representation method of trust in the mobile agent system is given. Based on the basic ideas of description and metric trust proposed in the Josang network trust management model, two basic concepts of Evidence Space and Opinion Space are introduced in the mobile agent system. In the fact space of the mobile agent system, the "good or bad result of entity interaction behavior" is transformed into the "level of credibility of the entity" in the concept space. Using Gaussian Probability Distribution Theory, the method of changing the degree of trust of the host in the mobile agent system is given. "Trust" is used to indicate the degree of trust that Host H_a considers Host H_x within a specified time frame in the Mobile Agent system to evaluate and predict the next secure interaction with Host H_x. A subjective trust dynamic management algorithm is proposed. Finally, through a set of simulation experiments, the feasibility of the proposed algorithm to measure the degree of host trust in the mobile agent system is verified, and the effectiveness of the trust group to improve the security of the interaction in the mobile agent system is verified.

Acknowledgments. This work was supported by the National Natural Science Foundation of China (61472136; 61772196), the Hunan Provincial Focus Social Science Fund (2016ZDB006), Hunan Provincial Social Science Achievement Review Committee results appraisal identification project (Xiang social assessment 2016JD05). The authors gratefully acknowledge the financial support provided by the Key Laboratory of Hunan Province for New Retail Virtual Reality Technology (2017TP1026).

References

1. Boss, G., Malladi, P., Quan, D., et al.: Cloud computing [EB/OL]. http://download.boulder.ibm.com/ibmdl/pub/software/dw/wes/hipods/Cloud_computing_wp_final_8Oct.pdf
2. Cloud Computing Forum & Workshop [EB/OL]. http://www.nist.gov/itl/cloud.cfm. Accessed 20 Aug 2010
3. Gray, R., Cybenko, G., Kotz, D., et al.: Mobile agents and state of the art. In: Bradshaw, J. (ed.) Handbook of Agent Technology. AAAI/MIT Press, Cambridge (2012)
4. Busi, N., Padovani, L.: A distributed implementation of mobile nets as mobile agents. In: Steffen, M., Zavattaro, G. (eds.) FMOODS 2005. LNCS, vol. 3535, pp. 259–274. Springer, Heidelberg (2005). https://doi.org/10.1007/11494881_17

5. Tang, W., Chen, C.: Subjective trust management model based on fuzzy set. Ruan Jian Xue Bao/ J. Software **14**(8), 1401–1408 (2003)
6. Jiang, S., Li, J.: For P2P e-commerce system based on trust mechanism reputation. Ruan Jian Xue Bao/J. Software **18**(10), 2551–2563 (2007)
7. Chen, H., Sun, J., Liu, C., Li, H.: A lightweight secure and trusted virtual execution environment. SCIENCE CHINA Information Sciences **42**(5), 617–633 (2012)
8. Lange, D.B., Oshima, M.: Seven good reasons for mobile agents. Commun. ACM **42**(3), 88–89 (2015)
9. Claessens, J., Preneel, B., Vandewalle, J.: How can mobile agents do secure electronic transactions on untrusted hosts? a survey of the security issues and the current solutions. ACM Trans. Inter. Tech. **3**(11), 28–48 (2016)
10. Ma, H.D., Yuan, P.Y., Zhao, D.: Research progress on routing problem in mobile opportunistic networks. Ruan Jian Xue Bao/J. Software **26**(3), 600–616 (2015). (in Chinese). http://www.jos.org.cn/1000-9825/4741.htm
11. Jiang, W., Xu, Y., Zhang, L.: Research on knowledge reuse dynamic evolvement model based on multi-agent system component. Syst. Eng. Theory Pract. **33**(10), 2663–2673 (2013)
12. Jiang, W., Xu, Y., Wang, X.: Active learning of pair-wise constraints in semi-supervised clustering. J. Syst. Sci. Math. Sci. **33**(6), 708–723 (2013)
13. Jiang, W.: Research on transaction security mechanism of mobile commerce in mobile internet based on MAS. Int. J. Secur. Appl. **9**(12), 289–302 (2015)
14. Liu, X., Li, J.-B., Yang, Z.: Atask collaborative execution policy in mobile cloud computing. Chin. J. Comput. **40**(2), 364–377 (2017)
15. Hu, H., Liu, R., Hu, H.: Multi-objective optimization for task scheduling in mobile cloud computing. J. Comput. Res. Dev. **54**(9), 1909–1919 (2017)
16. Jiang, W.: Multi agent system-based dynamic trust calculation model and credit management mechanism of online trading. Intell. Autom. Soft Comput. **22**(4), 639–649 (2016)