LCT: A Lightweight Cross-domain Trust Model for the Mobile Distributed Environment

Zhiquan Liu[1], Jianfeng Ma[1, 2], Zhongyuan Jiang[2] and Yinbin Miao[3]
[1] School of Computer Science and Technology, Xidian University
Xi'an 710071, China
[2] School of Cyber Engineering, Xidian University
Xi'an 710071, China
[3] School of Telecommunication Engineering, Xidian University
Xi'an 710071, China
[e-mail: zqliu@xidian.org.cn, jfma@mail.xidian.edu.cn, zyjiang@xidian.edu.cn, ybmiao@xidian.org.cn]
*Corresponding author: Jianfeng Ma

Received August 3, 2015; revised November 10, 2015; accepted December 11, 2015; published February 29, 2016

Abstract

In the mobile distributed environment, an entity may move across domains with great frequency. How to utilize the trust information in the previous domains and quickly establish trust relationships with others in the current domain remains a challenging issue. The classic trust models do not support cross-domain and the existing cross-domain trust models are not in a fully distributed way. This paper improves the outstanding Certified Reputation (CR) model and proposes a Lightweight Cross-domain Trust (LCT) model for the mobile distributed environment in a fully distributed way. The trust certifications, in which the trust ratings contain various trust aspects with different interest preference weights, are collected and provided by the trustees. Furthermore, three factors are comprehensively considered to ease the issue of collusion attacks and make the trust certifications more accurate. Finally, a cross-domain scenario is deployed and implemented, and the comprehensive experiments and analysis are conducted. The results demonstrate that our LCT model obviously outperforms the Bayesian Network (BN) model and the CR model in our cross-domain scenario, and significantly improves the successful interaction rates of the honest entities without increasing the risks of interacting with the malicious entities.

Keywords: Trust model, lightweight, cross-domain, mobile distributed environment, fully distributed

This work is supported by National High Technology Research and Development Program (863 Program) (No. 2015AA011704), National Natural Science Foundation of China (No. 61502375), Fundamental Research Funds for the Center Universities (No. JY10000903001) and Major Nature Science Foundation of China (No. 61370078).

http://dx.doi.org/10.3837/tiis.2016.02.026 **ISSN : 1976-7277**

1. Introduction

Nowadays, mobile devices are so ubiquitous that the number of active mobile devices has already exceeded the world's population according to GSMA (Global System for Mobile communications Assembly) Intelligence [1]. Besides, the mobile distributed environment is characterized by wireless, decentralized, dynamic, resource-limited, etc.

The trust management in the mobile distributed environment is of necessity when an entity desires to establish an acceptable trust relationship with others and avoid interacting with the malicious or selfish entities. Thus more-recent work focuses on adopting trust management as a solution for the mobile distributed environment.

Due to the distinctive characteristics of the mobile distributed environment, the trust in the mobile distributed environment has the following properties [2]:

1) ***Subjective***: Different trustors may determine different trust values towards to the same trustee due to different interaction experiences.
2) ***Asymmetric***: If Alice trusts Bob, it cannot guarantee that Bob trusts Alice to the same degree.
3) ***Partly transitive***: Given the fact that Alice trusts Bob and Bob trusts John, the conclusion that Alice trusts John to the same degree as Bob does cannot be derived.
4) ***Context-dependent***: Different contexts may result in different trust values. For instance, Alice trusts that Bob is an excellent car mechanic but not a qualified doctor.
5) ***Dynamic***: The same trustor may derive different trust values towards to the same trustee at different times due to the rapid topology changes caused by entity mobility or failure.

To take the aforementioned properties into consideration, an excellent trust model for the mobile distributed environment should satisfy the following requirements as far as possible:

1) ***Cross-domain***: In the mobile distributed environment, an entity may leave a domain and join in another one with great frequency. The trust information in the previous domains should be taken advantage of, as it contributes to quickly establishing the trust relationships with other entities in the current domain.
2) ***Fully distributed way***: To be consistent with the fully distributed characteristic of the mobile distributed environment, the trust model should be built in a fully distributed way without the super nodes or the third-party agents. Moreover, the trust information should be obtained in a lightweight manner in consideration of resource constraints. Besides, the trust relationships should be established quickly in view of the dynamic property of the trust in the mobile distributed environment.
3) ***Fine granularity***: As we know, the trust contains various aspects, such as honesty, stability and so on. To better describe trust, the trust model should contain various trust aspects with different weights, according to the personalized preferences of entities.
4) ***Robustness***: Due to the resource constraints in mobile distributed environment, selfish behaviors (e.g. refusing to provide its trust evaluations to other entities, refusing to act as relays for other entities, etc.) may occur. Moreover, malicious behaviors (e.g. providing terrible services to other entities, delaying or dropping data packages on purpose, etc.) are unavoidable because of the openness and decentralization of the mobile distributed environment. An excellent trust model should be able to detect and punish (or isolate) these misbehaving entities.
5) ***High performance***: A good trust model should be able to distinguish different kinds of entities (i.e. honest entities, general entities and malicious entities). Honest behaviors

should be stimulated while malicious ones should be punished.

Recently, a mass of trust models for the mobile distributed environment have been proposed [3-14]. In the classic trust models (as shown in **Fig. 1**), it is assumed that the previous interactions occurred between A and C, A and D, E and B, F and B in the past, where A, B, C, D, E and F represent different entities in the mobile distributed environment. When A wants to interact with a strange entity B, it needs to collect the trust recommendations from its acquaintances or physical neighbours (i.e. E and F) which have interacted with B in the previous interactions. Then A utilizes some strategies to drive the trust value of B and decides whether to interact with it or not. After the interaction, if it occurs, A and B update the trust information in the local storage. In the above process, A and B are defined as trustor and trustee in many literatures [2, 15-16], respectively. Conversely, B can also collect and evaluate the trust value of A in the same way.

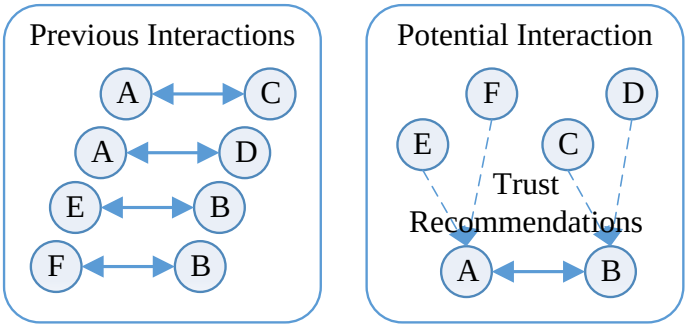


Fig. 1. Collecting the trust recommendations in the classic trust models

Obviously, collecting the trust recommendations is of great difficulty due to the topology changes caused by entity mobility or failure in the mobile distributed environment. Moreover, this process leads to lots of bandwidth and time consumption and can hardly cross domains on account of resource constraints and privacy concerns. Therefore, if an entity leaves the previous domain and joins in the current domain, the trust relationships between the entity and others in the current domain have to be rebuilt with ignoring the previous trust information. It is distinctly unreasonable.

In addition, in the existing cross-domain trust models [17-18], the super nodes or the third-party agents are supposed, which are inconsistent with the fully distributed characteristic of the mobile distributed environment. To the best of our knowledge, there exists no excellent trust model yet that can handle the cross-domain issue for the mobile distributed environment in a fully distributed way. It is just the motivation of this work.

In this paper, we improve the outstanding CR model [19] and propose a novel LCT model for the mobile distributed environment. The features and contributions of our LCT model are summarized as follows:

1) ***Our LCT model supports cross-domain***: In our LCT model, the trust certifications are collected and provided by the trustees and the trust information can be carried across domains easily. Moreover, the trust relationships can be established more quickly and reach an excellent performance in a lightweight manner.
2) ***Our LCT model is built in a fully distributed way***: In the existing cross-domain schemes, the super nodes or the third-party agents are assumed, which are inconsistent with the fully distributed characteristic of the mobile distributed environment. However, both the super nodes and the third-party agents are not needed in our LCT model as it is built in a fully distributed way.

3) ***Our LCT model is fine-grained***: To better describe trust, the trust ratings in the trust certifications contain various trust aspects with different interest preference weights in our LCT model, and they are donated by the linguistic variables. Furthermore, we adopt the fuzzy simple additive weighting system [20] to handle the inherent uncertainty of the human languages and derive the trust values.
4) ***Our LCT model is of robustness***: In our LCT model, three factors, namely the number of the trust certifications, the time decay of the trust certifications and the similarity between the trustors and the certifiers, are comprehensively considered to ease the issue of collusion attacks and make the trust certifications more accurate.
5) ***Our LCT model is of high performance***: To demonstrate the performance of our LCT model, we deploy and implement a cross-domain scenario, and conduct comprehensive experiments and analysis in this work. The results indicate that our LCT model is superior to the BN model [4] and the CR model, and significantly improves the successful interaction rates of the honest entities without increasing the risks of interacting with the malicious entities.

The remainder of this paper is organized as follows. Section 2 consists of a quick look at some related work and their limitations. Section 3 shows our trust model and trust evaluation method. Next, the experiments and analysis are presented in Section 4 and Section 5 concludes this paper.

2. Related Work

In recent years, a number of trust models for the mobile distributed environment have been proposed. We review some ones in terms of the theories and tools used in them.

Many researchers utilize Bayesian Network to evaluate the trust in the mobile distributed environment due to its suitable characteristics for causal reasoning [3-6]. Wang *et al.* [3] presented a Bayesian Network based trust model for the file sharing P2P (Peer-to-Peer) application. This model can derive the trust not only in a specific aspect, but also in a combination of various aspects. Dubey *et al.* [4] improved Wang's trust model by taking the time window into account. This model can detect malicious entities earlier than Wang's scheme. Wei *et al.* [5] proposed a trust model based on Bayesian Network for MANETs (Mobile Ad Hoc Networks). They mainly concentrate on easing the treats from the malicious attackers and a more reasonable trust score can be derived. Che *et al.* [6] presented a lightweight trust model based on both Bayesian Network and Entropy Theory for WSNs (Wireless Sensor Network). In their model, the weights of different trust aspects are derived from the Entropy Theory, instead of the experts artificially.

There also exist lots of trust models based on friendship, and the trust recommendations from friends (e.g. acquaintances or physical neighbours) are the primary trust source in these schemes [7-10]. Liu *et al.* [7] presented a new complex social network structure and a MQCTT (Multiple Quality Constrained Trust Transitivity) trust model. This model conforms to the principles of social psychology and can obtain more accurate trust scores than the previous schemes. Shabut *et al.* [8] proposed a friendship-based trust model for MANETs to secure route. This model combines both direct interactive and indirect friendship-based trust information to derive the trust scores and also considers the decay of friendship degree over time. Dhurandher *et al.* [9] presented a FACES (Friend-based Ad hoc routing using Challenges to Establish Security) algorithm for secure routing in MANETs. The nodes do not need to listen to the traffic through their neighbours in this model, so the overhead of the network can be reduced significantly. Chang *et al.* [10] presented a lightweight trustworthy

service discovery scheme for service-oriented MSNs in proximity. This model can reduce the transaction costs and is equally credible as the classic schemes.

Different from the above approaches, some researchers utilized other theories and tools to establish the trust model for the mobile distributed environment. Wei *et al.* [11] proposed a unified trust model based on uncertain reasoning to enhance the security of MANETs. It contains two trust model components: the direct trust component and the indirect trust component. The former is derived by the Bayesian Inference and the latter is obtained from the Dempster-Shafer theory. Deepa *et al.* [12] presented a directory-based trust model for service discovery. This model takes advantage of the Dezert-Smarandache theory to deal with the fusion of several trust evidences. Wang *et al.* [13] balanced trust value and end-to-end delay and designed a TQR (Trust-based QoS Routing) algorithm. It can prevent malicious attacks and improve the security performance of MANETs to some extent. Cao *et al.* [14] presented a PSTM (Proxy-based Security-feedback Trust Model) for MP2P (Mobile P2P). It can reduce the malicious and selfish behaviours and improve the successful interaction rates of the honest nodes.

Although these aforementioned trust models provide some brilliant ideas, there is no consideration of the cross-domain issue in these models, which may limit the applications of these approaches. Han *et al.* [17] proposed a TPCommuTrust (Topological Potential weighted Community-based recommendation Trust) model, in which the trust information can be carried across communities. However, this model has two obvious drawbacks: a) The super nodes are assumed, which are inconsistent with the fully distributed characteristic of the mobile distributed environment. b) The cross-community nodes are supposed. This is also clearly unreasonable as different domains are probably disjoint and have no common nodes. Tian *et al.* [18] presented a novel MTC (Multi Trust Chain) model for cross-domain interactions. In this model, different domains can be disjoint, but a specialized entity (i.e. a third-party agent) is needed. This is also inconsistent with the fully distributed characteristic of the mobile distributed environment. Huynh *et al.* [19] proposed a CR model for the multi-agent systems, in which the trust information is collected by the trustees, instead of the trustors. Nevertheless, this model has four limitations as follows: a) The similarity weight is only derived from the distance of rating values, so if the trustor has no previous interaction with the trustee, the similarity weight is set to a default low value due to the absence of the rating value. b) The trust value is merely denoted as a number without the consideration of various trust aspects. c) This model does not take the number of the trust certifications as a weight. d) There is no consideration for cross-domain scenario in this model.

Table 1. Intuitive comparisons between our LCT model and other trust models

Trust Models	Fully Distributed Way	Cross-domain
BN [4]	√	×
CR [19]	√	×
MQCTT [7]	√	×
FACES [9]	√	×
TQR [13]	√	×
PSTM [14]	×	×
TPCommuTrust [17]	×	√
MTC [18]	×	√
LCT	√	√

To the best of our knowledge, there exists no outstanding trust model yet that can deal with the cross-domain issue for the mobile distributed environment in a fully distributed way. To tackle this problem, we propose a novel LCT model and the intuitive comparisons with other trust models are shown in **Table 1** (in which "√" and "×" denote support and non-support, respectively).

3. Our Trust Model and Evaluation Method

Before introducing our trust model and evaluation method, we first consider all the possible relationships of two different domains as shown in **Fig. 2**: coincident, intersecting, tangent, containing, being contained, and disjoint. In the first five cases, the classic trust models can deal with the cross-domain issue as there exist common entities in two different domains and the trust recommendations can be collected from these common entities. Thus this paper focuses on the last case (i.e. two different domains are disjoint and have no common entities), which cannot be handled by the classic trust models.

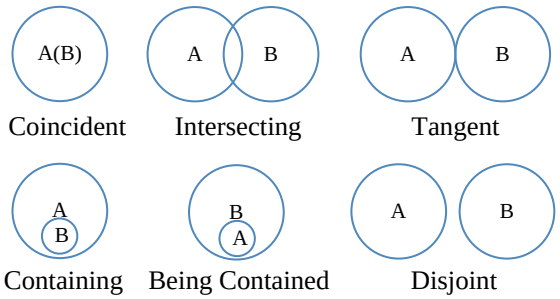


Fig. 2. All the possible relationships of two different domains

Next, we demonstrate the cross-domain scenario in the mobile distributed environment as shown in **Fig. 3**. Previously, Bob was in Domain 1, he had some interactions with other entities in Domain 1 and accumulated a certain trust level. But he left from Domain 1 for some reasons at some point and now he is in Domain 2. The other entities in Domain 2 can hardly collect the trust evaluations from the entities which have interacted with Bob in Domain 1 in the classic trust models, as they belong to different domains. So the trust relationships between Bob and the other entities in Domain 2 have to be rebuilt with ignoring the previous trust information of Bob in Domain 1. It is distinctly unreasonable. How to utilize the trust information in the previous domains and quickly establish trust relationships with other entities in the current domain is the key focus of this paper.

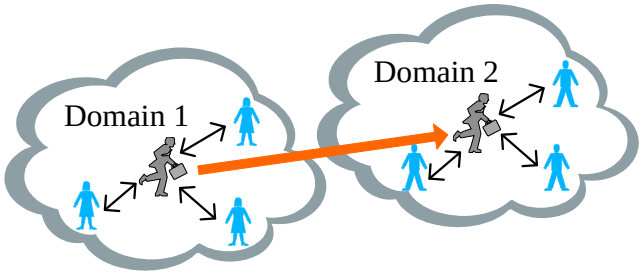


Fig. 3. The cross-domain scenario

3.1 Our Trust Model

In order to deal with the cross-domain issue, we propose a novel LCT model (as show in **Fig. 4**). For comparison, our model takes the same assumption for the previous interactions as the classic trust models shown in **Fig. 1**. When A wants to interact with a strange entity B, it first requests B to provide its own trust certifications, which are generated with digital signature information and sent to B by its previous interaction partners (i.e. E and F), and then stored and updated by B. After receiving these trust certifications from B, A can verify their authenticity through the digital signature technology when necessary. Then A can derive the trust value of B and decide whether to interact with B or not. After the interaction, if it occurs, A and B generate the trust certifications and send them to each other. In the above process, A and B are defined as trustor and trustee, respectively, and E and F are defined as certifiers. Conversely, B can also request A to provide its own trust certifications and then evaluate the trust value of A in the same way.

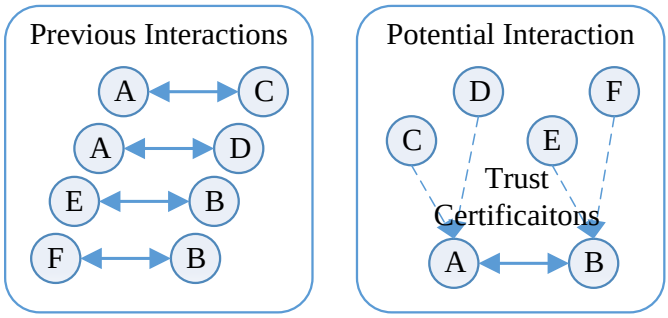


Fig. 4. Providing the trust certifications in our LCT model

In the actual interactions, the certifier (i.e. E or F) may refuse to provide its trust certification for the purpose of saving energy (as in the case of classic trust models). However, this problem is much smaller than that in the classic trust models. In our LCT model, each trust certification is provided for only once, while in the classic trust models each trust evaluation is collected for many times (The quantitative comparison is shown in Subsection 4.4), so the certifier is more likely to provide its trust certification in our LCT model than in the classic trust models. Meanwhile, if an entity behaves well, most of its interaction partners would like to provide their trust certifications to it, even though a few selfish ones will not do that. Thus it still can obtain sufficient trust certifications to prove its own trust. Furthermore, providing the trust certification can be taken as a part of the standard trust evaluation procedure and the certifier is forced to provide its trust certification [19].

As the trust certifications for an entity are stored and provided by itself, these information can be carried across domains easily. Besides, in our LCT model the trust certification contains digital signature information and any change to the trust certification can be easily detected [19], so the trustee cannot modify the trust certification even though it can obtain the trust rating values contained in the trust certification. Furthermore, the difficulty of validating certain trust certification is much smaller than that of collecting trust evaluations, and the validation is merely utilized when necessary. For example, we can optionally examine them for improving the efficiency of our approach.

Aiming at building a lightweight trust model, in this paper we mainly consider the case that both interaction partners are in close proximity to each other (i.e. they can directly interact with each other) and do not need to depend upon the trust propagation, which requires a significant amount of resource consumption. After an interaction, the certifier can directly

send its trust certification to the trustee as it is in close proximity to the trustee at that time.

3.2 The Representation of the Trust Certifications

In our scheme, the trust certification generated by the certifier i for the trustee j is denoted as equation 1.

$$\boldsymbol{Tc}(i, j) = (Id(i), Id(j), \boldsymbol{Rt}(i, j), \boldsymbol{Wg}(i), Ts(i, j), Ds(i, j)) \tag{1}$$

Where $Id(i)$ and $Id(j)$ denote the ids of the certifier i and the trustee j, respectively, and $\boldsymbol{Rt}(i, j)$ is represented as equation 2.

$$\boldsymbol{Rt}(i, j) = (Rt(i, j, 1), Rt(i, j, 2), \ldots, Rt(i, j, n)) \tag{2}$$

Where $Rt(i, j, m)$ $(1 \le m \le n)$ denotes the rating value of the m-th trust aspect and its value is represented in the form of linguistic variables (e.g. "Good", "Fair" and "Poor"), which can be handled by the fuzzy simple additive weighting system [20]. $\boldsymbol{Wg}(i)$ is represented as equation 3.

$$\boldsymbol{Wg}(i) = (Wg(i, 1), Wg(i, 2), \ldots, Wg(i, n)) \tag{3}$$

Where $Wg(i, m)$ $(1 \le m \le n)$ denotes the interest preference level of corresponding trust aspect and its value is also represented in the form of linguistic variables (e.g. "High", "Medium" and "Low"), similar to $Rt(i, j, m)$ $(1 \le m \le n)$. $Ts(i, j)$ denotes the timestamp when the trust certification is generated and $Ds(i, j)$ denotes the digital signature information.

In order to facilitate the trust calculations, we adopt the mapping from linguistic variables to fuzzy ratings and crisp ratings for $Rt(i, j, m)$ as illustrated in **Table 2** [20]. It should be noted that the crisp rating $Rc(i, j, m)$ in the last column is the signed distance of corresponding fuzzy rating $Rf(i, j, m)$. For a fuzzy rating $Rf(i, j, m) = (a, b, c, d)$, the value of $Rc(i, j, m)$ can be gained from equation 4.

$$Rc(i, j, m) = d(Rf(i, j, m)) = (a + b + c + d) / 4 \tag{4}$$

Table 2. Mapping from linguistic variables to fuzzy ratings and crisp ratings

Linguistic Variables (Rt)	Fuzzy Ratings (Rf)	Crisp Ratings (Rc)
Very poor (VP)	(0, 0, 0, 20)	5
Between very poor and poor (BVPP)	(0, 0, 20, 40)	15
Poor (P)	(0, 20, 20, 40)	20
Between poor and fair (BPF)	(0, 20, 50, 70)	35
Fair (F)	(30, 50, 50, 70)	50
Between fair and good (BFG)	(30, 50, 80, 100)	65
Good (G)	(60, 80, 80, 100)	80
Between good and very good (BGVG)	(60, 80, 100, 100)	85
Very good (VG)	(80, 100, 100, 100)	95

Similarly, $Wg(i, m)$ can be converted into the fuzzy weight $Wf(i, m)$ as illustrated in **Table 3** [20], and the corresponding crisp weight $Wc(i, m)$ is computed as shown in equation 5.

$$Wc(i, m) = \frac{d(Wf(i, m))}{\sum_{k=1}^{n} d(Wf(i, k))} \tag{5}$$

Table 3. Mapping from linguistic variables to fuzzy weights

Linguistic Variables (Wg)	Fuzzy Weights (Wf)
Very low (VL)	(0, 0, 0, 3)
Low (L)	(0, 3, 3, 5)
Medium (M)	(2, 5, 5, 8)
High (H)	(5, 7, 7, 10)
Very high (VH)	(7, 10, 10, 10)

Furthermore, the fuzzy trust score $Sf(i, j)$ of $\boldsymbol{Tc}(i, j)$ can be computed as shown in equation 6, where the operation $\otimes$ is similar to the standard way of matrix multiplication. Then the final trust score $Sc(i, j)$ of $\boldsymbol{Tc}(i, j)$ is obtained from equation 7.

$$Sf(i, j) = (Rf(i, j, 1), Rf(i, j, 2), ..., Rf(i, j, n)) \otimes \begin{pmatrix} Wc(i, 1) \\ Wc(i, 2) \\ ... \\ Wc(i, n) \end{pmatrix} \tag{6}$$

$$Sc(i, j) = d(Sf(i, j)) \tag{7}$$

Due to the mapping in **Table 2**, the derived rating values on all the service aspects range from 0 to 100. Besides, the derived preference weights on all the service aspects (in equation 5) fall in the range of [0, 1]. Therefore, we can easily find that $Sc(i, j)$ is in the range of [0, 100].

3.3 Three Factors of the Trust Certifications

Due to the unique characteristic of our trust model, the trustees may just provide the favorable trust certifications to their potential interaction partners, and even collude with other entities to improve their own trust values. To ease the issue of collusion attacks and make the trust certifications more accurate, we comprehensively consider three factors, namely the number of the trust certifications, the time decay of the trust certifications and the similarity between the trustors and the certifiers.

A) The Number of the Trust Certifications

In order to balance the overhead of storage and bandwidth with the robustness against collusion attacks, the trustee j stores $n(j)$ $(n(j) \le No)$ trust certifications, which come from $n(j)$ different certifiers and are the most favourable for itself in our trust evaluation method. No is a system threshold, and it is set such that there are at most $\lfloor (No-1)/2 \rfloor$ certifiers colluding to provide false trust certifications. The weight $Wn(j)$ of $n(j)$ is denoted as a piecewise function as shown in equation 8.

$$Wn(j) = \begin{cases} 0 & , if \ \ n(j) < No \\ 1 & , otherwise \end{cases} \tag{8}$$

If $n(j)$ is less than No, the trust certifications are considered as inauthentic, so $Wn(j)$ is set to 0. Otherwise, the trust certifications are regarded as authentic, so $Wn(j)$ is set to 1.

B) The Time Decay of the Trust Certifications

Next, we consider the time decay weight $Wt(i, j)$ for $\boldsymbol{Tc}(i, j)$, as the relatively recent trust certification is more credible than the less recent one, and the outdated trust certification may be incredible at all due to the high mobility and topology changes in the mobile distributed

environment. So $Wt(i, j)$ is represented as a piecewise function of $Ts(i, j)$ as shown in equation 9.

$$Wt(i, j) = \begin{cases} 0 & , if \ \ Tn - Ts(i, j) > Tw \\ e^{-\frac{Tn - Ts(i, j)}{\alpha}} & , otherwise \end{cases} \tag{9}$$

Where Tn is the current timestamp and Tw is a time window. α is a time unit which controls the speed of time decay. If the time difference between Tn and $Ts(i, j)$ exceeds Tw, $\boldsymbol{Tc}(i, j)$ is regarded as unreliable, so $Wt(i, j)$ is set to 0. Otherwise, $Wt(i, j)$ is represented as an exponential decay function of $Ts(i, j)$ [21].

C) The Similarity between the Trustors and the Certifiers

Except for $Wn(j)$ and $Wt(i, j)$, the similarity weight $Ws(i, j, k)$ should also be considered, as a fact that the trust certification from an entity which has similar preferences with itself is more convincing than that from an entity which has nothing in common with itself. In the view of the trustor k, there is nothing available but $\boldsymbol{Tc}(i, j)$ regarding to the certifier i, as they may be in different domains. Thus the similarity is mainly derived from $\boldsymbol{Tc}(i, j)$. Two cases are as follows:

Case 1: If the trustor k has no previous interaction with the trustee j, then the trustor k does not have $\boldsymbol{Tc}(k, j)$, but it can determine $Wg(k)$, so $Ws(i, j, k)$ can be computed based on the weighted Euclidean distance between $Wc(k)$ and $Wc(i)$ as shown in equation 10 and 11 [22].

$$Ws(i, j, k) = 1 - Dw(i, k) \tag{10}$$

$$Dw(i, k) = \sqrt{\frac{\sum_{m=1}^{n} (Wc(k, m) - Wc(i, m))^2 * Wc(k, m)}{\sum_{m=1}^{n} Wc(k, m)}} \tag{11}$$

Case 2: If the trustor k has previous interactions with the trustee j, then the trustor k has $\boldsymbol{Tc}(k, j)$, so $Ws(i, j, k)$ can be gained from the weighted Euclidean distance of both interest preference levers and rating vales according to equation 11, 12 and 13 [22], where the operation $\odot$ is similar to the standard way of vector subtraction.

$$Ws(i, j, k) = 1 - (Dw(i, k) + Dr(i, j, k)) / 2 \tag{12}$$

$$Dr(i, j, k) = \frac{1}{100} * \sqrt{\frac{\sum_{m=1}^{n} (d(Rf(k, j, m) \odot Rf(i, j, m)))^2 * Wc(k, m)}{\sum_{m=1}^{n} Wc(k, m)}} \tag{13}$$

3.4 The Procedure of the Trust Evaluation

In this subsection, we introduce the procedure of the trust evaluation as shown in **Fig. 5**. The procedure mainly includes four steps as follows:

A) Provide a Trust Certification

At the end of the previous interaction, the certifier i generates $\boldsymbol{Tc}(i, j)$ and sends it to the trustee j. After receiving $\boldsymbol{Tc}(i, j)$, the trustee j updates its local storage for preferably certifying its own trust in the future. The $n(j)$ most favourable trust certifications are selected based on the weighted rating value $Rw(i, j)$, which is calculated according to equation 14.

$$Rw(i, j) = Sc(i, j) * Wt(i, j) \tag{14}$$

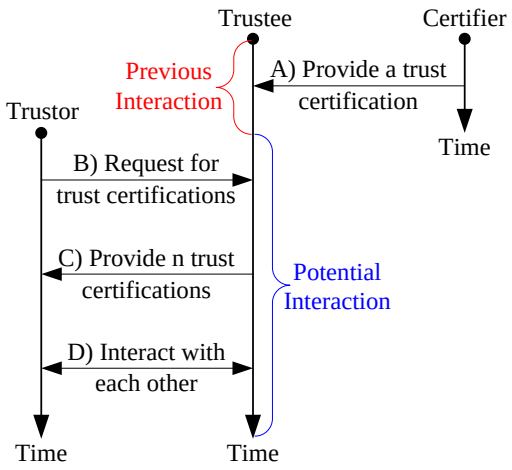


Fig. 5. The procedure of the trust evaluation

B) Request for Trust Certifications

In the potential interaction, if the trustor k (may be in a different domain with the certifier i) wants to interact with the trustee j, it sends a request for trust certifications to the trustee j.

C) Provide n Trust Certifications

When the trustee j receives the request from a potential interaction partner (i.e. the trustor k), it sends its own $n(j)$ trust certifications to the trustor k. Then the trustor k determines their weights and derives the trust value of the trustee j. In concrete terms, the total weight $Wa(i,j,k)$ of $\boldsymbol{Tc}(i,j)$ can be computed as shown in equation 15, and then the weighted trust value $Ra(i,j,k)$ of $\boldsymbol{Tc}(i,j)$ can be gained from equation 16. Finally, the total trust value $Rx(j,k)$ of the trustee j from the view of the trustor k can be calculated as shown in equation 17.

$$Wa(i,j,k) = Wn(j) * Wt(i,j) * Ws(i,j,k) \tag{15}$$

$$Ra(i,j,k) = Sc(i,j) * Wa(i,j,k) \tag{16}$$

$$Rx(j,k) = \frac{\sum_{i=1}^{n(j)} Ra(i,j,k)}{n(j)} \tag{17}$$

Due to the normalization processing in equation 8 ~ equation 13, three factor weights, namely $Wn(j)$, $Wt(i,j)$ and $Ws(i,j,k)$, all fall in the range of [0, 1], thus we can find that $Rx(j,k)$ ranges from 0 to 100.

D) Interact with Each Other

If $Rx(j,k)$ reaches the trust threshold $Ro(k)$ of the trustor k, then the trustor k trusts the trustee j and agrees to interact with it, otherwise the trustor k considers that the trustee j is not credible enough and refuses to interact with it or requests it to provide more favourable trust certifications. After the interaction, if it occurs, the trustor k also provides a trust certification $\boldsymbol{Tc}(k,j)$ to the trustee j as the certifier i does in the first step of our procedure. In other words, the trustor k acts as a certifier from the point of later potential interactions.

It should be added that there exists no trust certification for the newcomers, so their trust values derived from the aforementioned evaluation method are 0. To ensure that they have certain opportunities to interact with other entities, their trust values are set to a default low value To. Meanwhile, the malicious trustees may also act as newcomers and refuse to provide the trust certifications as they are unfavourable, thus their trust values are also equal to To.

4. Experiments and Analysis

In order to demonstrate the performance of our LCT model, the comprehensive experiments and analysis are presented in this section. We first deploy and implement a cross-domain scenario, and then we validate the average trust value variations and the average successful interaction rates of three kinds of different entities when they move across domains in Experiment 1. Furthermore, we compare the performance of our LCT model with that of the BN model and the CR model in Experiment 2. Next, we analyse and verify the robustness against the collusion attacks as well as the resource consumption of our LCT model in Experiment 3. Finally, Experiment 4 shows the performance of our LCT model in the more realistic scenarios with comparing to the other outstanding trust models.

4.1 Experiment Settings

In our experiments, we employ the standard evaluation indexes (i.e. trust value variation, successful interaction rate, robustness against the collusion attacks and resource consumption) and prevalent experiment methods, which are widely adopted in related work [3, 4, 6, 9, 12, 14, 17, 18, 19], to comprehensively measure the performance of our LCT model through comparing to the other outstanding trust models. Nevertheless, to the best of our knowledge, there is no existing application or open source dataset yet for the fully distributed cross-domain scenario. To facilitate the experiments, we first deploy and implement the following cross-domain scenario: There are 6 mutually disjoint domains (Domain 1 to Domain 6) and 50 trustors in each domain. The trust thresholds of these trustors are randomly generated. The investigated entities (i.e. the trustees) move across domains in the sequence of *Domain 1→Domain 2→Domain 3→Domain 4→Domain 5→Domain 6*. When a trustee (e.g. j) is in a domain, it takes an interaction testing with every trustor in this domain. After each interaction testing, the timestamp adds 1. If the derived trust value of the trustee j reaches the trust threshold of a trustor (e.g. k), this interaction testing is regarded as successful, and after that, the trustor k provides the trustee j with a trust certification, in which the rating values are based on the behaviour of the trustee j and the interest preference levels are randomly generated. Three kinds of different entities, namely 10 honest entities (which only provide excellent services), 10 general entities (which randomly provide good or terrible services) and 10 malicious entities (which merely provide terrible services), are separately investigated and intercompared. The parameters in our experiments are set as **Table 4**.

Table 4. The values of the parameters in our experiments

Parameters	Symbols	Values
The number of trust aspects	n	3
The number threshold of Tc	No	20
The time window	Tw	100
The time unit	α	40
The default trust value	To	10

4.2 Experiment 1

In this experiment, we mainly validate the average trust value variations of three kinds of different entities when they move across domains. Furthermore, we also compute the average successful interaction rates for three kinds of different entities in every domain. The experiment is repeated 100 times for every entity, and the average results are shown in **Fig. 6** and **Fig. 7**.

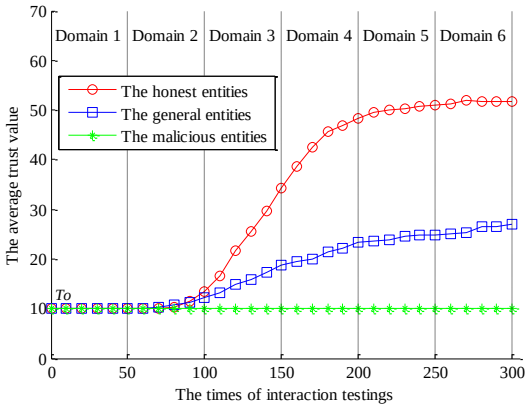


Fig. 6. The average trust value variations of three kinds of different entities with the times of interaction testings

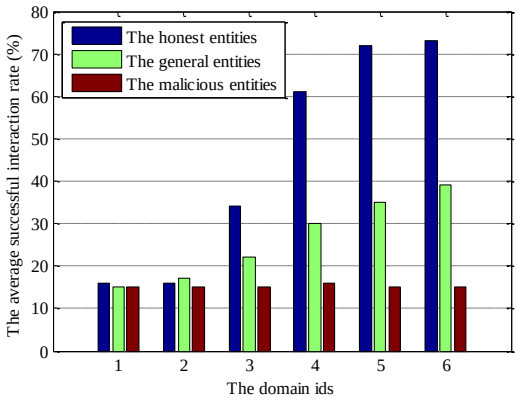


Fig. 7. The average successful interaction rates of three kinds of different entities in every domain

In the beginning, three kinds of different entities have the same initial trust value *To* (i.e. 10) as they all have no sufficient trust certifications to certify their own trust. With the increase of the interaction times, the average trust value of the honest entities rises rapidly (from 10 to 52.1) as they can provide very favourable trust certifications which contain high rating values, and their average successful interaction rate also increases quickly correspondingly (from 15% to 73%). Nevertheless, the average trust value of the malicious entities keeps unchanged as *To*, since they cannot provide advantageous trust certifications, and their average successful interaction rate also remains about the same (about 15%). In addition, the average trust value variation of the general entities (from 10 to 27.1) falls in between that of the honest entities and the malicious entities, and so does their average successful interaction rate (from 15% to 39%).

As we know, interacting with the honest entities brings benefits and interacting with the malicious entities means risks. So the average successful interaction rate of the honest entities

is the higher, the better, and that of the malicious entities is the lower, the better. Therefore, the experiment results indicate that our LCT model can significantly improve the successful interaction rates of the honest entities without increasing the risks of interacting with the malicious entities.

4.3 Experiment 2

In this experiment, we compare the performance of our LCT model with that of the BN model and the CR model. As we know, the BN model is an outstanding classic trust model, in which the trust information is collected by the trustors from their physical neighbours. This model takes various trust aspects and the time window into account, similar to our LCT model. The CR model provides a novel "Certified Reputation" idea that the trust information is collected by the trustees, instead of the trustors, similar to our LCT model. Thus we choose these two trust models for comparison. Moreover, we deploy and necessarily modify the BN model and the CR model in our cross-domain scenario. As we know, the ranges of trust values in the BN model and the CR model are [0, 1] and [-1, 1], respectively. They are different from that in our LCT model, therefore they are all scaled up to [0, 100] for comparison. The experiment is repeated 100 times for every entity and every trust model (including the case without any trust model), and the average trust value variations of three kinds of different entities in every trust model in Domain 6 are shown in **Fig. 8**. Furthermore, we also compare the average trust value variations of the honest entities and the general entities in every trust model in Domain 6 (The figures in the other domains are omitted due to space limitation.) as shown in **Fig. 9**, respectively. As the average trust value of the malicious entities in every trust model remains the same as To, the figure is omitted. In addition, we also compute the average successful interaction rates for three kinds of different entities in every trust model in Domain 6, and the outputs are shown in **Fig. 10**.

We first analyse the average trust value variation and the average successful interaction rate of the honest entities in every trust model in Domain 6 as shown in **Fig. 9** (a) and **Fig. 10** (the left part). In the case without any trust model, the average trust value keeps unchanged as To and the average successful interaction rate is also very low (15%) due to lack of a trust mechanism to feedback their honest behaviours and improve their trust values. In the BN model, their initial average trust value is To, as the BN model does not support cross-domain. With the increase of interaction times, their average trust value rises rapidly (from 10 to 39.5) due to their good behaviours in the interaction testings, and their average successful interaction rate is also relatively high (39%). In the CR model and our LCT model, the trust information can be carried across domains due to the "Self-Certified" characteristic, so they both have accumulated certain trust levers in the previous domains (Domain 1 to Domain 5) and dynamically maintain relatively high trust values in Domain 6. But due to the limitations of the CR model mentioned in Section 2, the average trust value in the CR model (21.2) is obviously lower than that in our LCT model (50.5), and is even lower than that in the BN model (30.8). Correspondingly, the average successful interaction rate in the CR model is relatively low (31%) and that in our LCT model is significantly high (72%).

Next, we analyse the average trust value variation and the average successful interaction rate of the malicious entities in every trust model in Domain 6 as shown in **Fig. 10** (the right part). In the case without any trust model, their average trust value keeps unchanged as To due to lack of a trust mechanism to feedback their behaviours. In the BN model, their average trust value will not exceed To because of their malicious behaviours, and will not be less than To in fact due to the "Re-entry" strategy [23], so their average trust value also remains the same as To. In the CR model and our LCT model, the malicious entities cannot provide favourable

trust certifications to prove their trust, thus their average trust value also keeps unchanged as To. Correspondingly, their average successful interaction rate in every trust model is the same (15%).

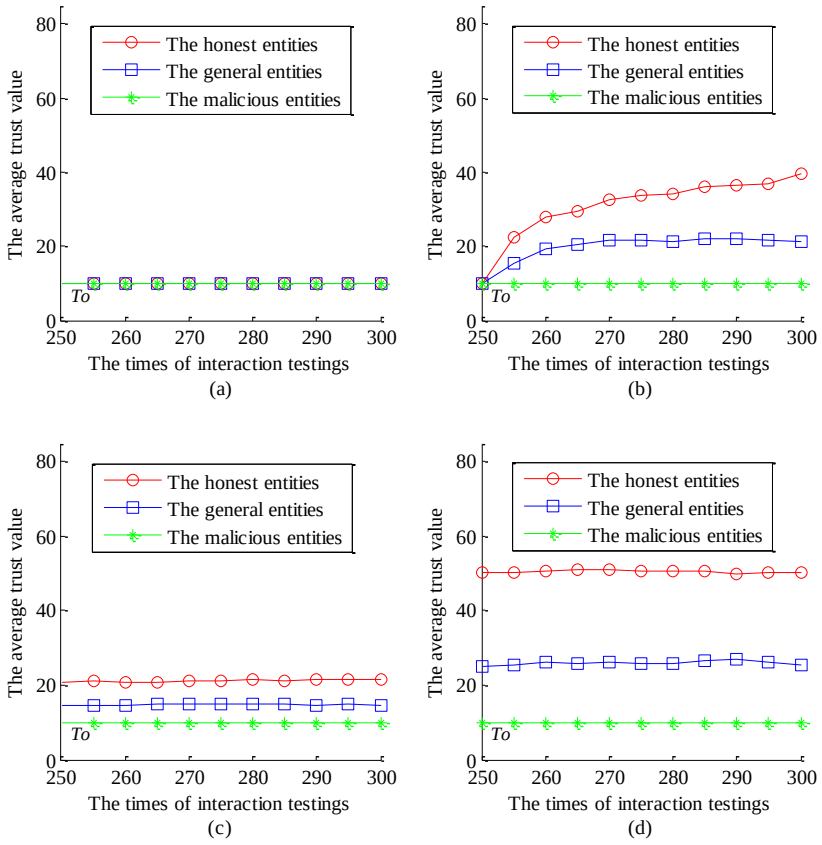


Fig. 8. The average trust value variations of three kinds of different entities with the times of interaction testings in every trust model in Domain 6 (a) Without any trust model (b) In the BN model (c) In the CR model (d) In our LCT model

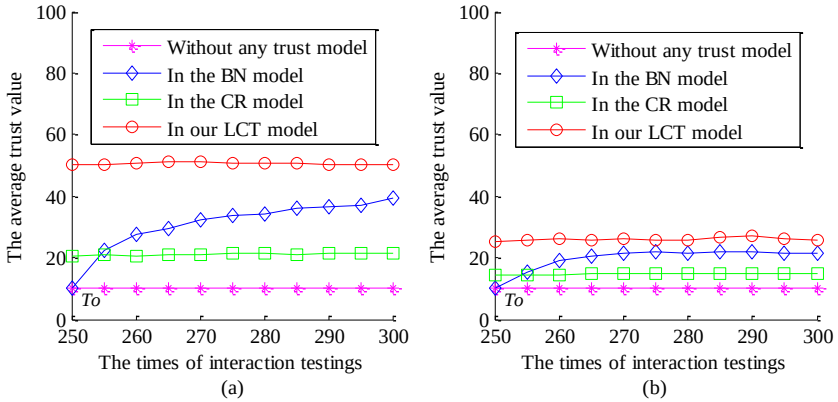


Fig. 9. The average trust value variations of the honest entities and the general entities with the times of interaction testings in Domain 6 (a) The honest entities (b) The general entities

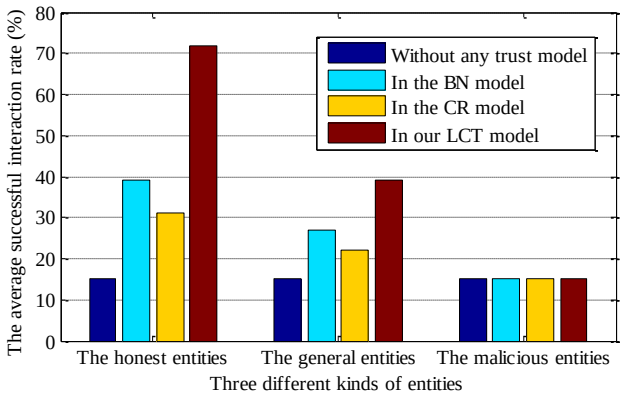


Fig. 10. The average successful interaction rates of three kinds of different entities in every trust model in Domain 6

In addition, the average trust value of the general entities in every trust model (10, 19.7, 14.7 and 26, respectively) falls in between that of the honest entities and the malicious entities as shown in **Fig. 9** (b), and so does their average successful interaction rate (15%, 27%, 22% and 39%, respectively) as shown in **Fig. 10** (the middle part).

Through above analysis, we can discover that our LCT model limits the risks of interacting with the malicious entities as well as the other trust models do (15%), but it significantly increases the successful interaction rates of the honest entities (by 380%, 85% and 132%, respectively) when comparing to the other three trust models. Thus our LCT model obviously overmatches the other trust models in our cross-domain scenario.

4.4 Experiment 3

In the previous two experiments, we mainly consider the cases without collusion attacks. While in this experiment, we focus on analysing and verifying the collusion-resistance ability of our LCT model, comparing to the CR model. It should be noted that the comparison with the BN model is omitted as there is no consideration of collusion attacks in the BN model. In addition, we also analyse and compare the resource consumption in the three trust models.

A) Collusion-resistance

In the classic trust models, the collusive entities may provide positive trust evaluations to improve the trust values of their companions as well as provide unfavourable trust evaluations to slander their competitors. While due to the "Self-Certified" feature in the CR model and our LCT model, the trustees will not provide adverse trust certifications, thus the collusive entities could only launch the former attack (i.e. providing profitable trust certifications to elevate the trust values of their companions). Therefore, in this part we merely need to consider the case that the malicious entities collude with others to improve their own trust values. In the CR model, the number of trust certifications is not taken as a weight, so the malicious entities are able to merely provide the collusive part. As a result, the trust values of the malicious entities increase rapidly once there are other entities colluding with them. While in our LCT model, we take the number of trust certifications as an important weight. Moreover, a suitable threshold No can be set according to the actual demands such that the maximum number of collusive entities is no more than $\lfloor (No-1)/2 \rfloor$. Therefore, the trust values of the malicious entities grow slowly with the number of collusive entities.

Furthermore, we conduct the comprehensive experiment to verify the above analysis. In concrete terms, we calculate the trust values of the malicious entities in both the CR model and our LCT model when there are 0 ~ 10 (i.e. 0% ~ 50% of No) entities colluding with them, respectively. The experiment is repeated 100 times and the average results are shown in **Fig. 11** (the solid part). To facilitate comparisons, we also draw two baselines (the dashed part) according to the average trust values (21.2 and 50.5, respectively) of the honest entities in the CR model and our LCT model.

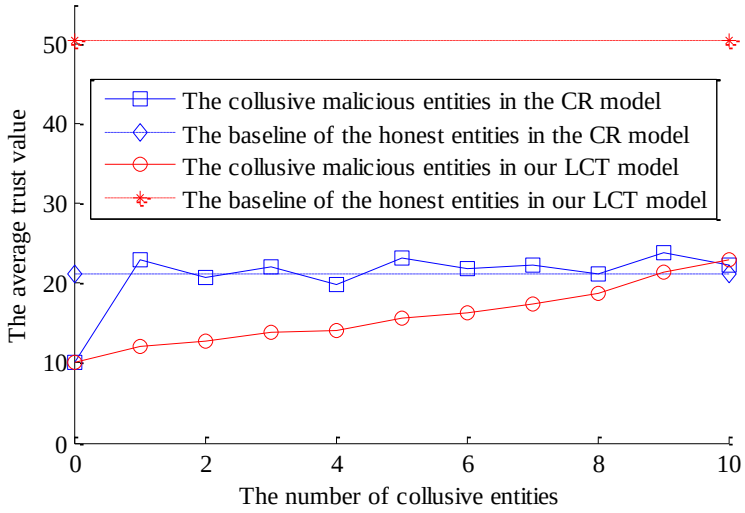


Fig. 11. The average trust value variations of the mailicous entities with the number of collusive entities in the CR model and our LCT model

In the CR model, the average trust value of the collusive malicious entities is very close to that of the honest entities. That is to say, the malicious entities and the honest entities cannot be effectively distinguished when there are collusion attacks in the CR model. While in our LCT model, the difference between the average trust value of the collusive malicious entities and that of the honest entities is large enough even in the extreme case (i.e. the number of collusive entities reaches 50% of No), so the collusive malicious entities and the honest entities can be easily divided. The experiment results are consistent with the above analysis, and show that our LCT model eases the collusion attacks better than the CR model does.

B) Resource Consumption

As we know, interaction round is an important indicator of resource consumption (e.g. time and bandwidth) for collecting/providing trust information. So, next we analysis the interaction rounds of each domain for collecting/providing trust information in the three trust models, respectively. In the BN model, every trustor needs to send a request to its 49 neighbourhood entities (which are in the same domain with the trustor) and then receives 49 responses from these neighbours, respectively (i.e. 49 rounds for each trustor). So the total interaction rounds in each domain can be computed as shown in equation 18.

$$IR("BN") = 49*50 = 2450 \tag{18}$$

In the CR model and our LCT model, every trustor needs to send a request to the trustee and receive a response from the trustee before the interaction, and then send a trust certification to the trustee after the interaction (i.e. 1.5 rounds for each trustor). Thus the total interaction rounds in each domain can be calculated as shown in equation 19.

$$IR("CR") = IR("LCT") = 1.5*50 = 75 \tag{19}$$

The intuitive comparison of interaction rounds in the three trust models is shown in **Table 5**. Through the analysis, we can find that our LCT model significantly decreases the interaction rounds (by 96.9%) for collecting/providing trust information when comparing to the BN model, as well as the CR model does. As a result, the trust relationships can be rebuilt quickly in a lightweight manner when an entity moves across domains in our LCT model.

Table 5. Intuitive comparison of interaction rounds of each domain for collecting/providing trust information in the three trust models

Trust Models	Interaction Rounds
BN [4]	2450
CR [19]	75
LCT	75

4.5 Experiment 4

In this experiment, we mainly illustrate the performance of our LCT model in the more realistic scenarios through comparing to the BN model and the CR model, and the three kinds of different entities are comprehensively considered. Different from the settings in Subsection 4.1, we assume that three kinds of different entities have the same function (e.g. providing the same service) and their total number is 100. They move across six different domains together, and each trustor evaluates their trust values and selects a trustworthy trustee to interact with for only once according to their trust values (i.e. the probability of certain trustee being selected is proportional to its trust value). We vary the proportions of the three kinds of different entities and then calculate their successful interaction rates in each case in Domain 6 (The data in the other domains is omitted due to space limitation.), respectively. This experiment is repeated 100 times for each case and the average results are shown in **Table 6**.

Table 6. The average successful interaction rates of three kinds of different entities in every case in Domain 6

Cases		Without	BN [4]	CR [19]	LCT	LCT vs. BN	LCT vs. CR
Case 1	H (30)	4.7%	28.8%	17.9%	53.6%	↑ 86.1%	↑ 199.4%
	G (60)	9.4%	16.5%	11.9%	18.9%	↑ 14.5%	↑ 58.8%
	M (10)	1.5%	0.7%	0.9%	0.4%	↓ 42.9%	↓ 55.6%
Case 2	H (30)	4.6%	36.3%	21.3%	64.1%	↑ 76.6%	↑ 200.9%
	G (10)	1.5%	4.2%	2.8%	4.9%	↑ 16.7%	↑ 75.0%
	M (60)	9.4%	5.6%	6.8%	3.9%	↓ 30.4%	↓ 42.6%
Case 3	H (10)	1.5%	24.4%	13.9%	48.1%	↑ 97.1 %	↑ 246.0%
	G (30)	4.6%	15.3%	9.4%	19.8%	↑ 29.4%	↑ 110.6%
	M (60)	9.4%	6.3%	7.5%	5.0%	↓ 20.6%	↓ 33.3%
Case 4	H (33)	5.2%	32.3%	19.9%	59.1%	↑ 83.0%	↑ 197.0%
	G (33)	5.1%	11.0%	7.4%	12.1%	↑ 10.0%	↑ 63.5%
	M (34)	5.3%	2.7%	3.5%	1.8%	↓ 33.3%	↓ 48.6%
Case 5	H (10)	1.5%	28.0%	15.6%	54.9%	↑ 96.1%	↑ 252.0%
	G (10)	1.5%	8.3%	4.3%	10.1%	↑ 21.7%	↑ 134.9%
	M (80)	12.5%	9.6%	10.8%	8.0%	↓ 16.7%	↓ 25.9%

Note: "Without" is short for "Without any trust model"; "H", "G" and "M" represent the honest entities, the general entities and the malicious entities, respectively, and the numbers behind them denote their proportions; "↑" and "↓" represent improvement and reduction, respectively.

In concrete terms, we consider five cases (i.e. Case 1 ~ Case 5) in this experiment and the proportions in these cases are different. When there is no trust model, the average successful

interaction rates of the three kinds of different entities in all the five cases are approximately proportional to their numbers. While when there is certain trust model (i.e. the BN model, the CR model or our LCT model), the average successful interaction rates of the honest entities and the general entities increase and that of the malicious entities decreases (as the proportion of their trust values decreases with the increasing trust values of the other kinds of entities) in all the five cases. Besides, among the three trust models, our LCT model has better performance than the BN model and the CR model. Take the extreme case (i.e. Case 5) for example. Although the proportion of the honest entities is very small (10%), their average successful interaction rate in our LCT model is quite high (54.9%), and it is greatly higher than that in the BN model (28.0%) and the CR model (15.6%). Analogously, the average successful interaction rate of the general entities in our LCT model (10.1%) is relatively higher than that in the BN model (8.3%) and the CR model (4.3%). Moreover, although the proportion of the malicious entities is rather large (80%), their average successful interaction rate in our LCT model is quite low (8.0%) and it is relatively lower than that in the BN model (9.6%) and the CR model (10.8%). In the other four cases, we can also get the similar conclusions.

Through above analysis, we can easily find that our LCT model greatly increases the average successful interaction rate of the honest entities and improves that of the general entities to some extent when comparing to the BN model and the CR model. Besides, our LCT model is also slightly better than the other two trust models in reducing the risks of interacting with the malicious entities. Thus our LCT model significantly outperforms the BN model and the CR model in our cross-domain scenarios.

5. Conclusion

In this paper, we have proposed a novel LCT model, in which the trust certifications are collected and provided by the trustees and the trust information can be carried across domains, for the mobile distributed environment in a fully distributed way. The trust ratings in the trust certifications contain various trust aspects with different interest preference weights, and they are donated by the linguistic variables, which can be handled by the fuzzy simple additive weighting system. Furthermore, we have comprehensively considered three factors to ease the issue of collusion attacks and make the trust certifications more accurate. Finally, we have deployed and implemented a cross-domain scenario, and conducted the comprehensive experiments and analysis. The results demonstrate that our LCT model greatly improves the successful interaction rates of the honest entities without increasing the risks of interacting with the malicious entities, and significantly outperforms the BN model and the CR model in our cross-domain scenario.

In the future, we aim to improve our LCT model and apply it to the realistic mobile e-commerce scenario, as the vendors may move across domains and need to prove their own trust to potential consumers for the purpose of enjoying the trust of more consumers and improving their transaction volumes.

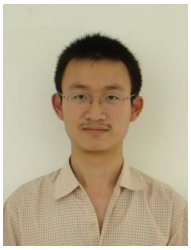
References

[1] Number Of Active Mobile Devices Surpasses World Population, 2014. Article (CrossRef Link)

[2] J. H. Cho, A. Swami and I. R. Chen, "A survey on trust management for mobile ad hoc networks," *IEEE Communications Surveys and Tutorials*, vol. 13, no. 4, pp. 562-583, November, 2011. Article (CrossRef Link)

[3] Y. Wang and J. Vassileva, "Bayesian network-based trust model," in *Proc. of IEEE Int. Conf. on Web Intelligence*, pp. 372-378, October 13-17, 2003. Article (CrossRef Link)
[4] J. Dubey and V. Tokekar, "Bayesian network based trust model with time window for pure P2P computing systems," in *Proc. of IEEE Global Conf. on Wireless Computing and Networking*, pp. 219-223, December 22-24, 2014. Article (CrossRef Link)
[5] Z. Wei, H. Tang, F. R. Yuand and P. Mason, "Trust establishment based on Bayesian networks for threat mitigation in mobile ad hoc networks," in *Proc. of IEEE Int. Conf. on Military Communications*, pp. 171-177, October 6-8, 2014. Article (CrossRef Link)
[6] S. Che, R. Feng, X. Liang and X. Wang, "A lightweight trust management based on Bayesian and Entropy for wireless sensor networks," *Security and Communication Networks*, vol. 8, no. 2, pp. 168-175, January, 2015. Article (CrossRef Link)
[7] G. Liu, Y. Wang and M. A. Orgun, "Trust transitivity in complex social networks," in *Proc. of American Association for Artificial Intelligence*, vol. 11, pp. 1222-1229, August 7-11, 2011. Article (CrossRef Link)
[8] A. M. Shabut, K. Dahal and I. Awan, "Friendship based trust model to secure routing protocols in mobile ad hoc networks," in *Proc. of IEEE Int. Conf. on Future Internet of Things and Cloud*, pp. 280-287, August 27-29, 2014. Article (CrossRef Link)
[9] S. K. Dhurandher, M. S. Obaidat, K. Verma, P. Gupta and P. Dhurandher, "Faces: friend-based ad hoc routing using challenges to establish security in MANETs systems," *IEEE Systems Journal*, vol. 5, no. 2, pp. 176-188, June, 2011. Article (CrossRef Link)
[10] C. Chang, S. Ling and S. Srirama, "Trustworthy service discovery for mobile social network in proximity," in *Proc. of IEEE Int. Conf. on Pervasive Computing and Communications*, pp. 478-483, March 24-28, 2014. Article (CrossRef Link)
[11] Z. Wei, H. Tang, F. R. Yu, M. Wang and P. Mason, "Security enhancements for mobile ad hoc networks with trust management using uncertain reasoning," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 9, pp. 4647-4658, November, 2014. Article (CrossRef Link)
[12] R. Deepa and S. Swamynathan, "A trust model for directory-based service discovery in mobile ad hoc networks," *Recent Trends in Computer Networks and Distributed Systems Security*, pp. 115-126, March, 2014. Article (CrossRef Link)
[13] B. Wang, X. Chen and W. Chang, "A light-weight trust-based QoS routing algorithm for ad hoc networks," *Pervasive and Mobile Computing*, vol. 13, pp. 164-180, August, 2014. Article (CrossRef Link)
[14] X. M. Cao, H. T. Zhu, H. Y. Shen and G. H. Chen, "Proxy-based security-feedback trust model in MP2P network," in *Proc. of Applied Mechanics and Materials*, pp. 1144-1151, February, 2014. Article (CrossRef Link)
[15] S. Hazra and S. K. Setua, "Probabilistic trust management in wireless communication system," in *Proc. of IEEE Int. Conf. on Electrical and Computer Engineering*, pp. 54-57, December 20-22, 2014. Article (CrossRef Link)
[16] W. Jiang, G. Wang and J. Wu, "Generating trusted graphs for trust evaluation in online social networks," *Future Generation Computer Systems*, vol. 31, pp. 48-58, February, 2014. Article (CrossRef Link)
[17] Q. Han, H. Wen, M. Ren, B. Wu and S. Li, "A topological potential weighted community-based recommendation trust model for P2P networks," *Peer-to-Peer Networking and Applications*, pp. 1-11, June, 2014. Article (CrossRef Link)
[18] L. Tian and W. Jiang, "A multi trust chain scheme in trusted cross-domain interaction," in *Proc. of IEEE Int. Conf. on Industrial Control and Electronics Engineering*, pp. 550-553, August 23-25, 2012. Article (CrossRef Link)
[19] T. D. Huynh, N. R. Jennings and N. R. Shadbolt, "Certified reputation: how an agent can trust a stranger," in *Proc. of the 5th ACM Int. Joint Conf. on Autonomous Agents and Multiagent Systems*, pp. 1217-1224, May 8-12, 2006. Article (CrossRef Link)
[20] S. Y. Chou, Y. H. Chang and C. Y. Shen, "A fuzzy simple additive weighting system under group decision-making for facility location selection with objective/subjective attributes," *European Journal of Operational Research*, vol. 189, no. 1, pp. 132-145, August, 2008.

Article (CrossRef Link)
[21] T. D. Huynh, N. R. Jennings and N. R. Shadbolt, "An integrated trust and reputation model for open multi-agent systems," *Autonomous Agents and Multi-Agent Systems*, vol. 13, no. 2, pp. 119-154, March, 2006. Article (CrossRef Link)
[22] H. T. Nguyen, W. Zhao and J. Yang, "A trust and reputation model based on Bayesian network for web services," in *Proc. of IEEE Int. Conf. on Web Services*, pp. 251-258, July 5-10, 2010. Article (CrossRef Link)
[23] A. Jøsang and J. Golbeck, "Challenges for robust trust and reputation systems," in *Proc. of the 5th Int. Workshop on Security and Trust Management*, September, 2009. Article (CrossRef Link)



Zhiquan Liu received his B.S. degree in School of Science from Xidian University in 2012. Currently he is a Ph.D. student of School of Computer Science and Technology in Xidian University. His research interests include trust management, service selection and social network.

Jianfeng Ma graduated from Department of Mathematics at Shaanxi Normal University in 1985, and he received his M.S. degree and Ph.D. in School of Computer Science and Technology and School of Telecommunication Engineering from Xidian University in 1988 and 1995, respectively. Currently he is a chair professor of "Cheung Kong Scholar" in Xidian University. His research mainly focuses on trust management, wireless security and system survivability.

Zhongyuan Jiang received his B.S. degree and Ph.D. from Beijing Jiaotong University in 2009 and 2013, respectively. Currently he is on the faculty board of School of Cyber Engineering in Xidian University. His research interests include complex network and urban computing.

Yinbin Miao received his B.S. degree in School of Telecommunication Engineering from Jilin University in 2011. Currently he is a Ph.D. student of School of Telecommunication Engineering in Xidian University. His research mainly focuses on network security and cryptography.