Measurement of Trust Transitivity in Trustworthy Networks

Yixiang Chen, Tian-Ming Bu, Min Zhang, Hong Zhu
Shanghai Key Laboratory of Trustworthy Computing, East China Normal University, Shanghai 200062, China
Email: {yxchen, tmbu, mzhang, hzhu}@sei.ecu.edu.cn

Abstract— **In this paper, we abstract the trust network as a weighted digraph. A path from node A to node B represents a transitive trust relationship. Parallel paths between a source and a target are associated with parallel trusts respectively. We introduce two measurements for computing the derived trust degree from a source to a target: Max-min trust degree and Max-mean trust degree. The Max operator formalizes the choice among parallel paths. The min and mean operators compute the transitive trust degree along a path. We focus on the analysis of the complexity of computing both kinds of trust degrees. We show that measuring the max-min trust degree is polynomial, however, measuring the max-mean one is NP-hard. Then we propose a matrix-based method to compute the max-mean trust degree, which can be done polynomially, but may produce non-simple paths. Finally, we give a simple example of a trust reputation network to illustrate the matrix-based method.**

Index Terms— **trustworthy networks, trust transitivity, measurement of transitive trustworthiness, max-min trust degree, max-mean trust degree, NP-hardness**

I. Introduction

Since all kinds of networks are of global internet working and ubiquitous connectivity, the study of their trustworthiness becomes more and more important in recent years. One always wants to know whether the others are trustworthy or not in the same network. Therefore, it becomes one interesting topic to compute ones' trustworthy degree according to the provided information.

In this paper, a trustworthy network can be regarded as a weighted digraph $G = (V, E, \tau)$ where V is the set of nodes representing agents, E is the set of edges or arrows which shows the direct trusts and τ is the weight function from E to the interval $[0, 1]$ and the value $\tau(u, v)$, for an arrow $< u, v >$, is the (direct) trustworthy degree of node u over node v. We say that u directly trusts v with the trust degree $\tau(u, v)$. A special case is that $\tau(u, v) = 0$. For this case, we say that there is no direct trustworthy relationship, or the trustworthy degree has been forgotten in a trustworthy network. Below is a simple example of a trustworthy network which shows the relationship among six agents.

A preliminary version of this work was presented at the 3th Workshop on Web Personalization, Reputation and Recommender Systems (WPRRS'09) and the 4th International Conference on Rough Set and Knowledge Technology (RSKT2009).

This work is supported by the grant of *Natural Science Foundation of China* (No.90718013), the *Research and Development Project of High-Technology* (No: 2007AA01Z189) and the *Shanghai Leading Academic Discipline Project* (B412).

Tian-Ming Bu is the corresponding author.

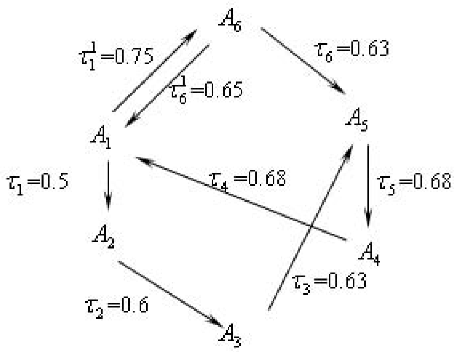

Fig.1. A simple example of trustworthy network

We notice that although u does not directly trust v, it might indirectly trusts v through other nodes. In this simple example, for instance, we can find a path connecting node A_1 and A_3 through node A_2. Thus, a path from node u to node v in a trust network induces a transitive trust of u over v. Its trustworthy degree completely depends on weights of all edges along that path.

A. Related Work and Motivation

Some methods for measuring the transitive degree have been proposed (see [1], [9], [12]).

As documented in [9], USDoD proposes the trusted computer systems evaluation criteria (TCSEC) in 1985 and two years later, Burrows et al propose the logic called as BAN-logic to represent trust. In 1993, Yahalom, Klein and Beth develop a formalism of trust relations between entities involved in authentication protocols ([20]). This trust relation is extended to the case of open networks by Beth, Borcherding and Klein in 1994 [1]. This model is called as BBK-scheme. Simmons and Meadows in [17] propose a model, called as SM-model, for studying the consequences of additional trust in shared control schemes. Jøsang in 1996 [8] introduces two types of trust: passionate and rational. He defines in a passionate entity trust as the belief that it will behave without malicious intent and exemplifies BBK-scheme and SM-model. In a rational entity, however, he defines trust as the belief that it will resist attacks from malicious agents and exemplifies BAN-logic and TCSEC.

In 1997, Jøsang [9] analyses these four formal models, BBK-sckeme, SM-model, BAN-logic and TCSEC with

doi:10.4304/jetwi.2.4.319-325

the purpose of determining their strong and weak sides. He gives an example to show the weak side of BBK-scheme for formalism of transitive trust: " *If you tell me that you trust NN by 100% and I only trust you by 1% to recommend me somebody, then I also trust NN by 100%*" by using the BBK model for computing the derived trust $V_1 \odot V_2$ by the following formula,

$$V_1 \odot V_2 = 1 - (1 - V_2)^{V_1}$$

where A trusts B by V_1 and B trusts C by V_2.

Almost ten years later, Jøsang, Hayward and Pope describe a method for trust network analysis using subjective logic (TN-SL) in [12]. They think that the trust network is consists of transitive trust relationships between people, organisations and software agents connected through a medium for communications and interactions. They formalize these transitive trust relationships through the trust paths linking the parties together. The subjective logic introduced by Jøsang in [8] is a logic for uncertain probabilities using elements from the Dampster-Shafer theory. In subjective logic, a quadruple $\omega_B^A = (b_B^A, d_B^A, u_B^A, a_B^A)$ is used to express the direct trust of agent A over agent B. Here b, d and u represent belief, disbelief and uncertainty respectively, where $a, b, u \in [0, 1]$ and $a + b + u = 1$. The parameter $a \in [0, 1]$ is the base rate. A's indirect functional trust in C can then be derived. The derived trust is defined as $\omega_C^{A:B} = (b_C^{A:B}, d_C^{A:B}, u_C^{A:B}, a_C^{A:B})$, which is the discounting operator in [12], by

$$\begin{cases} b_C^{A:B} &= b_B^A b_C^B \\ d_C^{A:B} &= b_B^A d_C^B \\ u_C^{A:B} &= d_B^A + u_B^A + b_B^A u_C^B \\ a_C^{A:B} &= a_C^B. \end{cases}$$

The feature of the subjective logic model is that the trust is considered as the whole of belief, disbelief and others. Jøsang et al points out that the effect of discounting in a transitive path is to increase the uncertainty, i.e., to reduce the confidence in the expectation value. Since the belief's transition subjects to the rule of products, we think that this transition rule makes the belief degree fall sharply.

If A trusts B by $(0.8, 0.1, 0.1)$ and B trusts C by $(0.7, 0.2, 0.1)$, then A will trust C by $(0.56, 0.16, 0.28)$. If we add further into our example that C trusts D by $(0.8, 0.2, 0)$, then A will trust D by $(0.452, 0.112, 0.336)$. In fact, we believe intuitively that A will trust D by the belief degree around 0.7. We illustrate this intuition with a security network. We assume that a link between node A and D consists of three parts: A to B, B to C and C to D. People often think that this link is secure if and only if each part from A to D, e.g., part $A \to B$, part $B \to C$ and part $C \to D$, are secure. In other words, the path security degree should be the same level as the one each part is.

This consideration leads us to propose a criteria for transitive trust, which is called *controllability*.

Definition 1.1: A measurement method for transitive trust is said to be *controllable*, if for any trustworthy network $G = (V, E, \tau)$, and for any pair of nodes u and v, whenever there exists a path p from u to v and weights of all edges along the path p are greater than a value α, the transitive trust degree of u over v along that path through this measurement method is also greater than that value α.

B. Our Contributions

The contribution of this paper is to propose two controllable measurement methods, Max-min measurement and Max-mean measurement for transitive trust. The Max operator will choose the maximal one from parallel paths. The min operator will define the minimal weight of all edges along that path as the transitive trust degree. In the mean operator, the arithmetical average of all weights along a path denotes the transitive trust degree along this path. In this paper, we show that measuring the max-min trust degree is polynomial, however, measuring the max-mean one is NP-hard. Then we propose a matrix-based method to compute the max-mean trust degree, which can be done polynomially, but may produce non-simple paths.

The rest of the paper is organized as follows. In Section 2, we introduce the max-min measurement and show that it can be computed polynomially. In Section 3, we introduce the max-mean measurement and prove that computing the max-mean trust degree is NP-hard. Section 4 proposes the Max-Sum matrix operator to compute the max-mean trustworthy degrees of paths. Section 5 give a simple example to illustrate the Max-Sum matrix operator. This example is a trustworthy network with reputation of agents and experience between agents. This example shows that the Max-Sum matrix operator might produce non-simple paths. The conclusions and the future prospects are followed in section 6.

II. MAX-MINIMUM MEASUREMENT

In this section we will introduce a minimum operator, to compute the transitive trustworthy degree along a path. In a minimum operator, we take the minimum weight of all edges on a path from u to v as the transitive trust degree of u over v along that path. This degree is called transitive *minimum* degree.

Definition 2.1 (Transitive minimum degree): For a path $P = u_1 u_2 \cdots u_n$ in a trustworthy network (V, E, τ), the *transitive minimal trustworthy degree, or transitive min degree shortly*, of u_1 over u_n along the path P, denoted by ε_{u_1,P,u_n}, is computed as follows:

$$\varepsilon_{u_1,P,u_n} = \min\{\tau(u_i, u_{i+1}) \mid i = 1, \ldots, n-1\}.$$

For parallel paths which have the same source and target, we take the maximum of transitive min degrees of all these paths. The *maximal min degree* of u over v, denoted by $\Delta(u, v)$, is the maximum of transitive min trustworthy degrees of all paths from u to v, i.e.,

$$\Delta(u, v) = \max\{\varepsilon_{u,P,v} \mid \; P \text{ is any path from } u \text{ to } v\}.$$

Clearly, the following proposition is hold.

Proposition 2.2: The Max-minimum measurement is controllable.

Here we give a polynomial-time algorithm for computing $\Delta(u, v)$ of any nodes u and v.

Algorithm 1 Computing the max-min degree of s over t. Array P records the path.

$\delta[s] = 1$;
$P[s] = s$;
for each vertex i in $V - \{s\}$ **do**
  $\delta[i] = 0$;
$X = \emptyset$, $Y = V$;
while Y is not empty **do**
  find the vertex u in Y with the maximum δ;
  $X = X + u$, $Y = Y - u$;
  for each edge (u, i) where i is in Y **do**
    if $(\delta[i] < \min\{\delta[u], \tau(u, i)\})$ **then**
      $\delta[i] = \min\{\delta[u], \tau(u, i)\}$;
      $P[i] = u$;
Print t;
repeat ▷ Print the path in a reversed order.
  Print $P[t]$;
  $t = P[t]$;
until $t \neq s$

Theorem 2.3: In Algorithm 1, $\delta[t]$ equals $\Delta(s, t)$, i.e., the value of the max-min path from s to t.

Proof: Suppose the vertices moved from Y to X are ordered by $u_1, u_2, \ldots, u_n$. We prove the correctness by induction on such order.

Since $u_1 = s$, the algorithm is obviously correct for u_1. Suppose the value of the first k vertices entering X is correct. For the k+1th vertex z, suppose the path based on the algorithm is $P = s, u_{1'}, u_{2'}, \ldots, u_{x'}, z$. If the correct path should be $P' = s, v_1, v_2, \ldots, v_y, z$, let v_p denote the last vertex appears in P'. If the algorithm is incorrect, namely the length of P' is larger than the length of P, then $\delta[z] = \min\{\delta[u_{x'}], \tau(u_{x'}, z)\} < \min\{\delta[v_y], \tau(v_y, z)\} \leq \min\{\delta[v_p], \tau(v_p, v_{p+1})\} = \delta[v_{p+1}]$. So in the $k + 1$th round, instead of z, v_{p+1} should be added to X, which contradicts our assumption.

Clearly, the complexity is $O(V^2 + E)$. ∎

Since the weights of edges represent trustworthy degrees between agents, the greater the weights are, the more an agent trusts another. Therefore, we can run the algorithm firstly and ignore those transitive trust degrees which are lower than the given threshold. However, there is an alternative method. In the method, we first get the subgraph G' from the trust graph G by deleting the edges whose weights are lower than the given threshold. Then the algorithm is run on the subgraph G'. The following proposition guarantees that the two methods return the same results.

Proposition 2.4: Given the trust graph G and the threshold, the two methods mentioned above generate the same results.

Proof: If the transitive trust degree graph between nodes u and v based on the original is lower than the threshold, then for all the parallel paths between u and v, there exists an edge whose weight is lower than the threshold according to the max-min algorithm. So if we delete these edges firstly, the transitive trust degree graph between u and v is 0.

On the other hand, if the transitive trust degree graph between u and v is higher than the threshold, then there exists a path that all the edges on the path are higher than the threshold. So even if we delete the edges whose weights are lower than the threshold first, the path above is still saved and the transitive trust degree doesn't change. ∎

Under the discovery, in order to decrease the number of edges and increase the algorithm's efficiency, before computing the algorithm, we may delete the edges whose weights are lower than threshold and keep the other pairs before computing the transitive trust degree. When the new induced graph is sparse enough, we can get the more efficient algorithm of time complexity $(V + E)\log V$ by using binary heap to implement the priority queue.

This minimum operator magnifies the importance of the minimum weight, since those weights greater than the minimum weight do not contribute to the transitive minimum degree any more. For example, if A trusts B by 0.8 and B trusts C by 0.1, then A will trust C by 0.1 according to the min operator. 0.8 contributes nothing to the trust of A over C. So, this measurement is conservative. If we are optimistic then we can choose the maximum rather than minimum of the weights of all edges along a path as the transitive trust degree. For the same example, we can get that A trusts C by 0.8 if we are optimistic.

In the next section we will introduce another comparatively mild algorithm, in which all weights will contribute to the transitive trust.

III. MAX-MEAN MEASUREMENT

In this section we will introduce the arithmetic average operator and analyze its complexity.

Mean operator: Mean operator means that the trust degree of a path is calculated with the geometric or arithmetic mean of those weights of all edges along that path. That is, if the length of a path is m and the weight of each arrow on this path is $\alpha_i (1 \leq i \leq n)$, then the trust degree of that path, according to the mean operator, is either the geometrical mean $(\alpha_1 \times \alpha_2 \times \cdots \times \alpha_n)^{\frac{1}{n}}$ or the arithmetic mean $\frac{1}{n}(\alpha_1 + \alpha_2 + \cdots + \alpha_n)$. These two operators can be converted into each other through the exponential function e^x and the logarithm function $\ln x$. So we only consider the arithmetical mean algorithm in this paper.

Definition 3.1 (Transitive mean degree): For a path $P = u_1 u_2 \cdots u_n$ in a trustworthy network (V, E, τ), the *transitive mean trustworthy degree, or transitive mean degree shortly* of u_1 over u_n along the path P, denoted by ε_{u_1,P,u_n}, is computed as follows:

$$\varepsilon_{u_1,P,u_n} = \frac{1}{n-1}\sum_{i=1}^{n-1}\tau(u_i, u_{i+1}).$$

For parallel paths which have the same source and the same target, we take the maximum of transitive mean degrees of all these paths. The *maximal mean degree* of u over v, by $E(u, v)$, is the maximum of transitive mean trustworthy degrees of all paths from u to v, i.e.,

$$E(u, v) = \max\{\varepsilon_{u,P,v} \mid P \text{ is any path from } u \text{ to } v\}.$$

Noting that due to the fact that cycles are of no use in computing transitive trust degrees, here we only consider the simple paths in trust transitivity.

Similarly, we can define the *minimal transitive mean degree* by replacing max with min in the above formula, which is equal to $\min\{\varepsilon_{u,P,v} \mid P \text{ is any path from } u \text{ to } v\}$.

Clearly, the following is also true.

Proposition 3.2: The Max-mean algorithm has the controllability.

Given a weighted digraph $G = (V, E, \tau)$, we want to find the maximum transitive mean simple paths of a graph. This problem is called the maximum mean path problem. The dual of this problem is the minimum mean path problem.

The maximal mean paths problem seems similar to the maximum mean cycle problem proposed by Karp in 1978 [15]. It has many applications in rate analysis of embedded systems, in discrete-event system, and of course, in graph theory (e.g., [4]–[6]). In fact the two problems are quite different essentially. The maximum mean cycle problem has polynomial algorithms, yet the maximum mean path problem is NP-hard, as showed below.

Proposition 3.3: The max-mean path problem is as hard as the min-mean path problem.

Proof: The reduction from maximum mean path to minimum mean path is as follows.

Given a weighted graph $G = (V, E, \tau)$ a source vertex B, we get a new weighted digraph $G' = (V, E, \tau')$ as following:

Let Maximumweight $\leftarrow 0$
For $(u, v) \in E$
If $\tau(u, v) > Maximumweight$ **then** $Maximumweight \leftarrow \tau(u, v)$
For $(u, v) \in E$
$\tau'(u, v) = Maximumweight - \tau(u, v)$

Then the maximal mean path of G is exactly the minimum mean path of G'.

Similarly, we can get the reverse-side reduction.

From above, we claim that minimum mean path and maximum mean path are equivalently hard. ■

Theorem 3.4: The min-mean path problem and the max-mean path problem are NP-hard optimization problems.

Proof: By the previous proposition, we only need to show that the minimum mean path problem is NP-hard, by reducing the Hamilton path problem, which is known to be NP-Complete, to it.

Given an instance of Hamilton path problem, i.e., a graph G with n vertices , we set all edges of G to 1 and add one additional vertex w and a directed edge $v \to w$ with the weight 2 to G, where v is an vertex of G. Then, we get a new weighted digraph G' with $n + 1$ vertices. Given a vertex u in G, we claim that there is a Hamilton Path P from u to v in the original graph G if and only if $P' = u \to \cdots \to v \to w$ is the minimum mean path in the new graph G' with minimum path mean $\frac{n+1}{n}$. ■

IV. MATRIX-BASED COMPUTING METHOD FOR MAX-MEAN MEASUREMENT

Due to the NP-hardness of computing the Max-mean trust degree, we introduce a matrix-based method to compute it. The method ,however, may produce non-simple paths.

Given a trustworthy network $G = (V, E, \tau)$ with n vertices, we use the matrix $A = (a_{uv})_n$ to represent relations between vertices with weights in such a way that

$$a_{uv} = \begin{cases} \tau(u, v), & \text{if } < u, v > \in E \\ 0, & \text{otherwise.} \end{cases}$$

we assume that $\tau(u, v) \neq 0$ for each edge (u, v) in this weighted digraph, i.e., $a_{uv} \neq 0$ if u directly trusts v and that $a_{uv} = 0$ if u does not directly trust v. We assume that $a_{uu} = 0$ for each node u of G.

Based on this matrix A, we define a series of matrices $S^k = (s^k_{uv})_n$ for max-mean degree inductively as $S^1 = A$ and $S^k = A \uplus S^{k-1}$ in which, for any $k \geq 2$,

$$s^k_{uv} = \begin{cases} 0, & u = v \\ \max\{a_{ur} \oplus s^{k-1}_{rv} \mid 1 \leq r \leq n\}, & \text{otherwise} \end{cases}$$

where

$$a \oplus b = \begin{cases} 0, & \min\{a, b\} = 0 \\ a + b, & \text{otherwise} \end{cases}$$

The matrix S^k is computed through sum $\oplus$ and maximum, called Max-Sum operator, similar to the ordinary sum-product operator of matrices.

By applying mathematical induction on k, we can easily check that $s^k_{uv} = 0$ implies that there is no paths of length k from u to v. In Max-Sum operators, we always set $s^k_{uu} = 0$ so that self-cycles will not participate in the next computation step, since we have stated that we are interested in simple paths. But, the above procedure still can not remove all cycles in the result (see the next section).

For S^k, the elements can be greater than 1. We introduce a matrix $T^k = S^k/k$, i.e.,

$$t^k_{uv} = s^k_{uv}/k.$$

Thus, t^k_{uv} falls down into the interval $[0, 1]$.

Let $M_{mean} = T^1 \vee T^2 \vee \cdots \vee T^{n-1}$. Then matrix M_{mean} is the matrix of maximal mean transitive trustworthy degrees.

V. SIMULATION

In the paper[1], Chen, Zhang and Zhu introduce model the trustworthy networks as double-weighted digraphs, which is different to the one presented here. In their model, there are two key notions: reputation and direct trust. The reputation describes the node's social evaluation, and the direct trust describes the experience of one node over another. Here, we keep the trustworthy networks as one-weighted digraphs, while we call double-weighted digraphs as trust-reputation networks.

A trust-reputation network consists of four components V, E, γ and ε. V denotes the set of vertices. These vertices can be agents or entities etc.. E is a subset of $V \times V$ representing arrows or edges. These arrows show the direction of trust. γ is a mapping from V to the interval $[0, 1]$. The value $\gamma(v)$ represents the reputation (or social) degree of vertex v, which is usually defined by other vertices within this network. ε is a mapping from the set E to the interval $[0, 1]$. If $e = (u, v)$ is an arrow then $\varepsilon(e)$ defines the experience (or direct) degree of u over v. This degree is often defined by u'self according to its experience with v.

Both the security network and recommending network can be modeled as such trust reputation networks. Fig. 2 is a simple example.

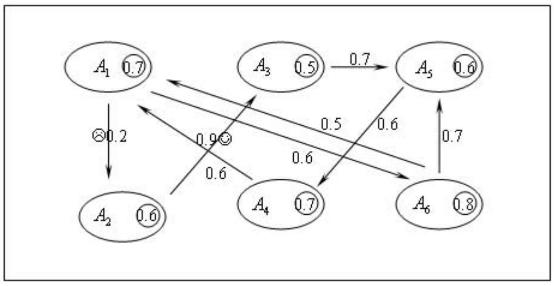

Fig.2. A simple trust reputation network.

In Fig.2, there are six agents $(A_1, A_2, \cdots A_6)$ in the business network. A_1 has experiences with A_2 and A_6, A_2 has experiences with A_3, and so on. Every agent puts his reputation public (for example, A_1's reputation is 0.7, that means his service gets 70% of satisfaction). In addition, each agent gives his experience degree for the past trades over the others (For example, A_1 thinks the service of A_2 with 20% of satisfaction). Reputation and experiences become the basis of a new trade. For example, when A_1 wants to trade with A_6, he will do an overall consideration of the reputation of A_6 and his experience with A_6 rather than only his personal experience with A_6. If A_1 wants to do a business with A_5, then he could evaluate A_5 firstly by $A_1 \to A_2 \to A_3 \to A_5$ and $A_1 \to A_6 \to A_5$.

For the convenience of calculation, actually we can induce a trust reputation network into an edge-weight digraph $G(V, E, \tau)$ by assigning edges combinations of the reputation values of nodes and the trust values of edges. For example, for every edge $e = (u, v)$ we define the product $\tau(e) = \gamma(v) \times \varepsilon(e)$ and the linear combination $\tau(e) = \alpha_1 \times \gamma(v) + \alpha_2 \times \varepsilon(e)$ with the coefficients (α_1, α_2) such that $\alpha_1 + \alpha_2 = 1$. We can adjust the values of the coefficients α_1 and α_2 to show which one is important for this combination. For example, if we take $\alpha_1 = 3/4$ and $\alpha_2 = 1/4$ then we express the idea that the reputation is more important than the experience. In addition, $\alpha_1 = \alpha_2 = 1/2$ represents the reputation and the trust are equally important. Actually, Fig. 1 is the induced trustworthy digraph-$(3/4, 1/4)$ from Fig. 2.

Based on figure 1, the min operator tells us that the transitive trust degree of A_1 over A_5 along the path $A_1A_6A_5$ is 0.63, whilst one along the path $A_1A_2A_3A5$ is 0.5. According to the Max-min measurement method, A_1's transitive trust degree over A_5 is 0.63.

In the following, we will illustrate how to compute the max-mean trust degrees of Fig. 1 by the matrix based method.

A. Max-Sum Matrix operator

According to Max-Sum matrix operator and Figure 1, we have

$$S^1 = \begin{pmatrix} 0 & 0.50 & 0 & 0 & 0 & 0.75 \\ 0 & 0 & 0.60 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.63 & 0 \\ 0.68 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.68 & 0 & 0 \\ 0.65 & 0 & 0 & 0 & 0.63 & 0 \end{pmatrix}$$

$$S^2 = \begin{pmatrix} 0 & 0 & 1.10 & 0 & 1.38 & 0 \\ 0 & 0 & 0 & 0 & 1.23 & 0 \\ 0 & 0 & 0 & 1.31 & 0 & 0 \\ 0 & 1.18 & 0 & 0 & 0 & 1.43 \\ 1.36 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1.15 & 0 & 1.31 & 0 & 0 \end{pmatrix}$$

$$S^3 = \begin{pmatrix} 0 & 1.90 & 0 & 2.06 & 1.73 & 0 \\ 0 & 0 & 0 & 1.91 & 0 & 0 \\ 1.99 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1.78 & 0 & 2.06 & 0 \\ 0 & 1.86 & 0 & 0 & 0 & 2.11 \\ 1.99 & 0 & 1.75 & 0 & 2.03 & 0 \end{pmatrix}$$

$$S^4 = \begin{pmatrix} 0 & 0 & 2.50 & 2.41 & 2.78 & 0 \\ 2.59 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2.49 & 0 & 0 & 0 & 2.74 \\ 0 & 2.58 & 0 & 0 & 2.41 & 0 \\ 0 & 0 & 2.46 & 0 & 0 & 0 \\ 0 & \frac{2.55}{2} & 0 & 2.71 & 2.38 & 0 \end{pmatrix}$$

$$S^5 = \begin{pmatrix} 0 & 3.30 & 0 & 3.46 & 3.13 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3.34 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3.18 & 0 & 3.46 & 0 \\ 0 & 3.26 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{3.15}{2} & 3.06 & 3.43 & 0 \end{pmatrix}$$

[1] Yixiang Chen, Min Zhang and Hong Zhu. A Model of Trustworthy Networks, Manuscript, East China Normal University, August, 2007.

In these matrices, the notation $\frac{\alpha}{2}$ means that there are two paths and the largest sum is α. For example, in the matrix S^4, $s^4_{6,2} = \frac{2.55}{2}$ shows that there are two paths of length 4 from node A_6 to A_2 and the maximal sum is 1.38. These paths can be recorded. In fact, these two paths are $A_6A_1A_6A_1A_2$ and $A_6A_5A_4A_1A_2$. Their weight sums are 2.49 and 2.55, respectively. Transitive mean degrees of A_6 over A_2 along these two paths are 0.623 and 0.638, respectively. The maximum mean degree is 0.638 and its corresponding path is the second one: $A_6A_5A_4A_1A_2$. The other path includes a cycle $A_6A_1A_6A_1$. This algorithm does not remove this cycle. So, this algorithm does not guarantee removing all non-simple paths. But it is polynomial.

Here is the matrices T^k defined by S^k/k before.

$$T^1 = \begin{pmatrix} 0 & 0.50 & 0 & 0 & 0 & 0.75 \\ 0 & 0 & 0.60 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.63 & 0 \\ 0.68 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.68 & 0 & 0 \\ 0.65 & 0 & 0 & 0 & 0.63 & 0 \end{pmatrix}$$

$$T^2 = \begin{pmatrix} 0 & 0 & 0.55 & 0 & 0.69 & 0 \\ 0 & 0 & 0 & 0 & 0.62 & 0 \\ 0 & 0 & 0 & 0.66 & 0 & 0 \\ 0 & 0.59 & 0 & 0 & 0 & 0.72 \\ 0.68 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.58 & 0 & 0.66 & 0 & 0 \end{pmatrix}$$

$$T^3 = \begin{pmatrix} 0 & 0.63 & 0 & 0.69 & 0.58 & 0 \\ 0 & 0 & 0 & 0.64 & 0 & 0 \\ 0.66 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.59 & 0 & 0.69 & 0 \\ 0 & 0.62 & 0 & 0 & 0 & 0.70 \\ 0.66 & 0 & 0.58 & 0 & 0.68 & 0 \end{pmatrix}$$

$$T^4 = \begin{pmatrix} 0 & 0 & 0.63 & 0.60 & 0.70 & 0 \\ 0.65 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.62 & 0 & 0 & 0 & 0.69 \\ 0 & 0.65 & 0 & 0 & 0.60 & 0 \\ 0 & 0 & 0.62 & 0 & 0 & 0 \\ 0 & 0.64 & 0 & 0.68 & 0.60 & 0 \end{pmatrix}$$

$$T^5 = \begin{pmatrix} 0 & 0.66 & 0 & 0.69 & 0.63 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.67 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.64 & 0 & 0.69 & 0 \\ 0 & 0.65 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.63 & 0.61 & 0.69 & 0 \end{pmatrix}$$

The matrix M_{mean}, i.e., the matrix of maximal mean transitive trustworthy degrees, is

$$M_{mean} = \begin{pmatrix} 0 & 0.66 & 0.63 & 0.69 & 0.70 & 0.75 \\ 0.65 & 0 & 0.60 & 0.63 & 0.62 & 0.67 \\ 0.66 & 0.62 & 0 & 0.66 & 0.63 & 0.69 \\ 0.68 & 0.65 & 0.64 & 0 & 0.69 & 0.72 \\ 0.68 & 0.65 & 0.62 & 0.68 & 0 & 0.70 \\ 0.66 & 0.64 & 0.63 & 0.68 & 0.69 & 0 \end{pmatrix}$$

The matrix tells us that this digraph in Figure 1 is connected. That means, any pair of vertices has transitively trustworthy. The maximal average transitive trust degree of A_u over A_v is the value m_{uv} of matrix M_{mean}. For instance, A_1 trusts A_4 with the maximal average transitive trust degree of 0.69 and the corresponding path is $A_1A_6A_5A_4$.

VI. Conclusions and Future Prospects

In this paper, we abstract the trustworthy network as weighted digraphs and the transitive trustworthy relationship as the paths in the weighted digraph. We propose two algorithms, Max-min algorithm and Max-mean algorithm, to measure the transitive trustworthiness. In the paper, we prove that the Max-Mean algorithm is NP-hard. Recently, Zhu, Wang and Zhou shew that it is impossible to find an approximately optimal solution which is close to the exactly optimal solution to the Max-mean path problem in polynomial time[2]. Of course, we can use some heuristics to deal with the max-mean path problem, as what we proposed in this paper. It is valuable that these measurements can be used to solve practice issues (e.g., [18], [21]).

References

[1] Tomas Beth, Malte Borcherding and Birgit Klein: Valuation of Trust in Open Networks. In proceeding of the European Symposium on Research in Computer Security(ESORICS), 1994 LNCS 875, 3-18.

[2] Yixiang Chen, Min Zhang and Hong Zhu Tianming Bu, Average Transitive Trustworthy Degrees for Trustworthy Networks,P. Wen et al. (Eds.): RSKT 2009, LNCS 5589,pp.682-689 2009.

[3] Bruce Christianson and William S. Harbison. Why Isn't Trust Transitive? Proceedings of the International Workshop on Security Protocols, Lecture Notes In Computer Science; Vol. 1189, page 171-176, Springer-Verlag London, 1996.

[4] Thomas H. Cormen, Charles E.Leiserson, Ronald L.Rivest: Introduction to Algorithms (2nd edition), MIT Press, 1999.

[5] Ali Dasdan, Rajesh K. Gupta : Faster Maximum and Minimum Mean Cycle Algorithms for System Performance Analysis. IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems, 17(10): 889-899, (1998).

[6] Dingzhu Du, Ker I. Ko: Theory of Computational Complexity. John Wiley & Sons. Inc. 2000.

[7] R. Guha, Ravi Kumer, Prabhakar Raghavan and Andrew Tomkins. Propagation of Trust and Distrust. In Proceedings of the 13th international conference on World Wide Web (International World Wide Web Conference), pagee 403-412, New York, 2004.

[8] Audun Jøsang: The right type of Trust for Distributated Systems. In C. Meadows, editors, Proc. of 1996 New Security Paradigms, page 119-131, September 17-20, 1996, Lake Arrowhead, Canifornia, USA.

[9] Audun Jøsang. Prospectives for modelling trust in information security. Lecture Notes in Computer Science Vol: 1270, page 1611-3349, Springer Berlin / Heidelberg, 1997.

[10] Audun Jøsang. A logic for uncertain probabilities. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 9(3)(2001): 279-311.

[2] Hong Zhu, Dakan Wang, Jie Zhou: On Min/Max Mean Path. regular talk on The First AAAC Annual Meeting (AAAC08), April 26-27, 2008, University of Hong Kong, Pokfulam, Hong Kong.

[11] Audun Jøsang, Elizabeth Grey and Michael Kinateder. Simplification and analysis of transitive trust networks. Web Interlligence and Agent Systems: An international journal 4 (2006) 139-161.

[12] Audun Jøsang, Ross Hayward and Simon Pope. Trust Network Analysis with Subjective Logic. ACM International Conference Proceeding Series; Vol. 171, Proceedings of the 29th Australasian Computer Science Conference - Volume 48, Hobart, Australia Pages: 85 - 94, 2006.

[13] Audun Jøsang, Roslan Ismail and Colin Boyd. A survey of trust and reputation systems for online service provision. Decision Support Systems 43 (2007) 618-644.

[14] Audun Jøsang and Simon Pope. Semantic constraints for trust transitivity. Conferences in Research and Practice in Information Technology Series; Vol. 107, Proceedings of the 2nd Asia-Pacific conference on Conceptual modelling - Volume 43, page 59-68, 2005.

[15] R.M. Karp : A characterization of the minimum cycle mean in a digraph. Discrete Mathematics 23(1978):309-311.

[16] Like Mui, Mojdeh Mohtashemi and Halberstadt. A Computational Model of Trust and Reputation. Proceedings of the 35th Hawaii International Conference on System Sciences, Big Island, HI, USA, January 2002, page 188-196.

[17] G.J. Syverson and C. Meadows. The role of trust in information integrity protocols. Journal of Computer Security 3(1)(1995): 71-84.

[18] Jaeki Song and Fatemeh "Mariam" Zahedi. Trust in Health Infomediaries. Decision Support Systems 43: 390-407, 2007.

[19] Yuan Wang, Feng Xu, and Jian Lü: Establishing Recommandation Trust Relationship for Internetwares. ACM SIGSOFT Software Engineering Notes. 2006 31(1) Page 1-5.

[20] R.Yahalom, B.Klein, Th. Beth. Trust Relationships in Secure Systems-A Distributed Authentication Perspective. Proc. 1993 IEEE Symp. on Research in Security and Privacy, page 150-164, 1993.

[21] Cai-Nicolas Ziegler and Jennifer Golbeck. Investigating interactions of trust and interest similarity. Decision Support Systems 43 (2007) 460-475.