Finding K Optimal Social Trust Paths for the Selection of Trustworthy Service Providers in Complex Social Networks

Guanfeng Liu
Department of Computing
Macquarie University
Sydney, Australia
guanfeng.liu@mq.edu.au

Yan Wang
Department of Computing
Macquarie University
Sydney, Australia
yan.wang@mq.edu.au

Mehmet A. Orgun
Department of Computing
Macquarie University
Sydney, Australia
mehmet.orgun@mq.edu.au

***Abstract*—In a service-oriented online social network consisting of service providers and consumers as participants, a service consumer can search trustworthy service providers via the social network between them. This requires the evaluation of the trustworthiness of a service provider along a potentially very large number of social trust paths from the service consumer to the service provider. Thus, a challenging problem is how to identify K optimal social trust paths that can yield the K most trustworthy evaluation results based on service consumers' evaluation criteria.**

In this paper, we first present a complex social network structure and a concept, Quality of Trust (QoT). We then model the K optimal social trust paths selection with multiple end-to-end QoT constraints as the Multiple Constrained K Optimal Paths (MCOP-K) selection problem, which is NP-Complete. For solving this challenging problem, based on Dijkstra's shortest path algorithm and our optimization strategies, we propose a heuristic algorithm H-OSTP-K with the time complexity of $O(m + Knlogn)$. The results of our experiments conducted on a real dataset of online social networks illustrate that H-OSTP-K outperforms existing methods in the quality of identified social trust paths.

Keywords: trust, social networks, K paths selection, service provider selection

I. Introduction

Online social networking sites have been attracting a large number of participants and are being used as the means for a variety of rich activities. For example, according to a survey on 2600 hiring managers in June 2009 by CareerBuilder[1] (a popular job hunting website), 45% of them used social networking sites to investigate potential employees. In January 2010, the ratio increased to 72%. In such an activity, trust is one of the most important factors for decision making by the participants, creating a great demand for approaches and mechanisms for evaluating the trustworthiness between two unknown participants. In service-oriented environments, social networks can be used as the means for service consumers to look for trustworthy service providers who are unknown to them prior to invoking services, with the assistance of information from other participants. For example, if a social network consists of lots of buyers and sellers, it can be used by a buyer to find the most trustworthy/reputable seller who sells the product preferred by the buyer [11].

In social network models, each node represents a participant and each link between participants corresponds to real-world interactions or online interactions between them (e.g., $A \rightarrow B$ in Fig. 1). One participant can give a trust value to another based on their past interactions. As each participant usually interacts with many other participants, multiple social trust paths may exist between two participants who have no direct links with each other, such as the trust path $A \rightarrow B \rightarrow E \rightarrow H$ and $A \rightarrow C \rightarrow F \rightarrow H$ in Fig. 1, each of which is called a *social trust path* [8, 10]. Along a social trust path linking two nonadjacent participants, such as A (termed as the *source participant*) and H (termed as the *target participant*) in Fig. 1, the source participant can evaluate the trustworthiness of the target one based on the trust information between the intermediate participants along the path. This process is called *trust propagation* [8, 10].

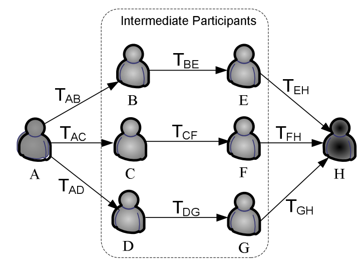

Figure 1. Social network

In large-scale social networks, there are usually over tens of thousands of social trust paths between two unknown participants [14]. Evaluating the trustworthiness of the target participant based on all these social trust paths can lead to huge computation time [2]. A challenging problem is how to select those paths yielding the most trustworthy results of trust propagation based on the source participant's trust evaluation criteria.

In the literature, there are some studies [10, 17] for addressing the path selection problem in social networks. However, in [17], the *trust information* between participants is neglected in path selection. In addition, the *social relationships* between adjacent participants (e.g., the relationship between a buyer and a seller) and the *recommendation roles* of a participant (e.g., a supervisor as a referee in his postgraduate's job application) have significant influence on trust propagation [1, 26] and can be obtained by using data mining techniques in social networks [22]. However, these factors have not been considered in the models of [10] and [17]. Furthermore, a source participant may have different purposes in evaluating the trustworthiness of a target participant, such as hiring employees, buying or introducing products. Therefore, a source participant may have different social trust path selection criteria and should be able to set certain constraints on the above factors in trust evaluation. However, such a feature is not supported by the above methods.

An optimal social trust path selection model has been

[1]http://www.careerbuilder.com/

proposed in our previous studies [19, 20], where the above three factors and constrains are considered. But all the existing methods including our previous studies focus on selecting only one social trust path between a source participant and a target participant. As illustrated in cognitive science [13], people are willing to believe what they have been told most often and by the possibility of the greatest number of different of sources. Therefore, in order to obtain a more reasonable trust evaluation result of a target participant, a source participant need to refer to multiple social trust paths from the source participant to the target one. This requires to identify K $(K \geq 2)$ optimal social trust paths, yielding the K most trustworthy trust propagation results based on the constraints specified by the source participant. Since the selection of any one of the K optimal social trust paths based on multiple constrains is the classical MCOP selection problem, which has been proved to be NP-Complete [12], the Multiple Constrained K Optimal Social Trust Paths (MCOP-K) selection is also an NP-Complete problem. But existing algorithms [5, 21, 24] for K paths selection attempt to find the K shortest paths without any end-to-end constrains, and this is not an NP-Complete problem. Thus, they can not be used for the MCOP-K selection problem.

To solve the MCOP-K selection problem in complex social networks, we first present the structure of complex social networks taking *trust information*, *social relationships* and *recommendation roles* of participants into account. We then introduce a concept, *Quality of Trust* (QoT), which is used to illustrate the ability to guarantee a certain level of trustworthiness in trust propagation along a social trust path, taking the above three factors as attributes (*see Section III and Section IV*).

In addition, since a source participant can have different social trust path selection criteria, he/she can set multiple constrains for QoT attributes in the K optimal social trust paths selection. To address the NP-Complete MCOP-K problem, based on Dijkstra's shortest path algorithm [4] and our optimization strategies, we propose a new efficient Heuristic algorithm for the K Optimal Social Trust Path selection, called H-OSTP-K (*see Section V*).

Furthermore, we have conducted extensive experiments on a real online social network dataset, the *Enron* email dataset[2]. Experimental results demonstrate that H-OSTP-K outperforms existing methods in the quality of identified social trust paths (*see Section VI*).

II. Related Work

A. Social Network Analysis

The studies of social network properties can be traced back to 1960's when the *small-world* characteristic in social networks was validated by Milgram [25], through illustrating that the average path length between two Americans was about 6 hops in an experiment of mail sending. In recent years, Mislove *et al*. [27] analyze several popular social networks including Facebook[3], MySpace[4] and Flickr[5], and validate the small-world and *power-law* (i.e. in a social network, the probability that a node has degree k is proportional to k^{-r}, $r > 1$) characteristics of online social networks by using data mining techniques.

[2]http://www.cs.cmu.edu/enron/

[3]http://www.facebook.com

[4]http://www.myspace.com

[5]http://www.flickr.com

B. Trust in Online Social Networks

Several trust management methods have been proposed in the field of online social networks. Golback *et al*. [8] propose a trust inference mechanism for trust relation establishment between a source participant and the target one based on averaging trust values along the social trust paths. In addition, Guha *et al*. [9] propose a trust propagation model, where the number of hops in trust propagation is considered in calculating the propagated trust values between a source participant and the target one. Furthermore, in the model of [11], a buyer performs several random walks with a fixed number of hops along a path from this buyer in the social network to find the ratings of the ending participant to a seller. The degree of confidence of the seller is calculated based on the number of random walk hops, ratings and the number of random walk paths.

The above trust propagation strategies are only based on trust ratings given by participants. As pointed out in social science theories [1, 26], *social relationships* (e.g., the relationship between a buyer and a seller, or the one between an employer and an employee) and *recommendation roles* (e.g., the supervisor as a referee in a job application) [30] have significant influence on participants' decision making. However, the existing models discussed above have not considered these factors.

C. Social Trust Path Selection

In the literature, there are only a few studies addressing the path selection problem, which might be used for the social trust path selection. *SmallBlue* [17] is an online social network constructed for IBM staff. In this system, between a source participant and a target participant, up to 16 social paths with no more than 6 hops are selected and the shortest one is taken as the optimal path without taking trust between participants into consideration. Hang *et al*. [10] further take the trust between participants into consideration in path selection. In their model, the path with the the maximum of propagated trust values is selected as the optimal one. In these methods, some significant influence factors including *recommendation roles* and *social relationships* between participants are not taken into account in path selection. In [19, 20], we have proposed a multiple constrained optimal social trust path selection model, where the impact factors and constraints are considered. However, all existing methods including our previous model focus on selecting only one social trust path between two participants. To obtain a realistic trust evaluation result of a target participant, a source participant needs to refer to K optimal social trust paths, yielding the K most trustworthy trust propagation results based on the source participant's trust path selection criteria, which is still an NP-Complete problem [12]. Therefore, in this paper, we propose a heuristic algorithm, H-OSTP-K to solve this challenging NP-Complete problem.

III. Complex Social Networks

The complex social network structure depicted in Fig. 2, comprises the attributes of three impact factors. They are *trust*, *social intimacy degree* and *role impact factor* [19, 20], which influence the trustworthiness of trust propagation and hence the decision making of a source participant.

1) Trust: In social networks, *trust* is the belief of one participant in another, based on their interactions, in the extent to which the future action to be performed by the latter will lead to an expected outcome. Let $T_{AB} \in [0,1]$ denote the trust value that participant A assigns to participant B. If $T_{AB} = 0$, it indicates that A completely distrusts B while $T_{AB} = 1$

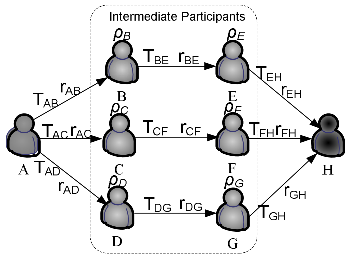

Figure 2. Complex social network

indicates A completely believes B's future action can lead to the expected outcome.

2) Social Intimacy Degree: As illustrated in social psychology [26], a participant can trust the participants with whom he/she has more intimate social relationships more than those with whom he/she has less intimate social relationships. Let $r_{AB} \in [0, 1]$ denote the *Social Intimacy Degree* (SID) between participant A and participant B in social networks. $r_{AB} = 0$ indicates that A and B have the least intimate social relationship while $r_{AB} = 1$ indicates they have the most intimate social relationship.

3) Role Impact Factor: Rich activities of participants in social networks can be categorized into different domains (e.g., hiring employees or product sale) based on their characteristics. As illustrated in social psychology [1], in a certain domain of interest, recommendations from a domain expert are more credible than that from a beginner. Let $\rho_A \in [0, 1]$ denote the *Role Impact Factor* (RIF), illustrating the impact of participant A's recommendation role on trust propagation in a certain domain. $\rho_A = 1$ indicates that A is a domain expert while $\rho_A = 0$ indicates that A has no knowledge in the domain.

Though it is difficult to build up social relationships and comprehensive role hierarchies in all domains, it is feasible to build them up in a particular application. For example, in the work by Mccallum *et al.* [22], through mining the subjects and contents of emails in *Enron* Corporation[2], the social relationship between each email sender and receiver can be discovered and their roles can be known. Then the corresponding SID and RIF values can be calculated based on probabilistic models. In addition, in academic social networks formed by large databases of Computer Science literature (e.g, DBLP or ACM Digital Library), the social relationships between two scholars (e.g., coauthors, supervisor and his/her students) and the role of scholars (e.g., professor in the field of data mining) can be mined from publications or their homepages. The SID and RIF values can be calculated by applying the PageRank model [28].

IV. Quality of Trust and QoT attributes aggregation

In this section, we first present a general concept called Quality of Trust (QoT) and then propose a novel K optimal social trust paths selection model with end-to-end *Quality of Trust* (QoT) constraints.

A. Quality of Trust (QoT)

In Service-Oriented Computing (SOC), QoS consists of a set of attributes, used to illustrate the ability of services to guarantee a certain level of performance [7]. Similar to the QoS, we present a new concept, *Quality of Trust* [18].

Definition 1: *Quality of Trust (QoT) is the ability to guarantee a certain level of trustworthiness in trust propagation along a social trust path, taking trust (T), social intimacy degree (r), and role impact factor (ρ), as attributes.*

In service invocations, users can set multiple end-to-end constraints for the attributes of QoS to satisfy their requirements (e.g., cost, delay and availability) of services. Different requirements have different constraints (e.g., total cost<\$20, delay<5s and availability>70%). In our model, to satisfy different trust evaluation criteria, a source participant can specify different multiple end-to-end constraints of QoT attributes (i.e., T, r and ρ) for social trust path selection in different domains. $Q^{\mu}_{v_s,v_t}$ ($\mu \in \{T,\ r,\ \rho\}$) denotes the end-to-end QoT constraint for the QoT attribute μ between v_s and v_t. Throughout this paper, v_s denotes the source participant and v_t denotes the target participant in a social network between them.

B. QoT Attribute Aggregation

To specify the end-to-end QoT constrains, we need to know the aggregated value of each QoT attribute in a certain social trust path.

1) Trust Aggregation: The trust values between a source participant and the target participant in a social path can be aggregated based on trust transitivity (i.e., if A trusts B and B trusts C, then A trusts C to some extent) [8]. Since trust is discounted with the increase of the number of transitivity hops [3], in our model, we adopt the strategy proposed in [16, 29]; if there are n participants $a_1, ..., a_n$ in order in a social trust path (denoted as $p(a_1, ..., a_n)$), the aggregated trust value is calculated as in Eq. (1).

$$T_{p(a_1,...,a_n)} = \prod_{a_i, a_{i+1} \in p(a_1,...,a_n)} T_{a_i\ a_{i+1}} \tag{1}$$

This aggregated trust value will be combined with the social intimacy degree and the role impact factor in the following context to select K optimal social trust paths.

2) Social Intimacy Degree Aggregation: Firstly, social intimacy between participants decays with the increasing number of hops between them in a social trust path [15]. In addition, the intimacy degree decays fast when it is approaching one. In contrast, the intimacy degree decays slowly when it is approaching zero [26]. Namely, the decay speed of the social intimacy degree is non-linear in social networks. The aggregated r value in path $p(a_1, ..., a_n)$ can be calculated by Eq.(2) whose function image is a *hyperbolic curve*, fitting the characteristic of social intimacy attenuation.

$$r_{p(a_1,...,a_n)} = \prod_{a_i, a_{i+1} \in p(a_1,...,a_n)} r_{a_i\ a_{i+1}} \tag{2}$$

3) Role Impact Factor Aggregation: As illustrated in social psychology [23], a social role (e.g., a professor in the field of data mining) is the position of an individual in a given society. Therefore in the same society, the role impact factor of an agent *does not decay* with the increase of transitivity hops. Thus, the aggregated ρ value of path $p_{(a_1,...a_n)}$ can be calculated by Eq. (3), the characteristic of a social role.

$$\rho_{p(a_1,...,a_n)} = \frac{\sum_{i=2}^{n-1} \rho_{a_i}}{n-2} \tag{3}$$

C. Utility Function

In our model, we define the utility (denoted as $\mathcal{F}$) as the measurement of the trustworthiness of social trust paths. The utility function takes the QoT attributes T, r and ρ as arguments in Eq. (4)

$$\mathcal{F}_{p(a_1,...,a_n)} = \omega_T * T_{p(a_1,...,a_n)} + \omega_r * r_{p(a_1,...,a_n)} + \omega_\rho * \rho_{p(a_1,...,a_n)} \tag{4}$$

where ω_T, ω_r and ω_ρ are the weights of T, r and ρ respectively; $0 < \omega_T, \omega_r, \omega_\rho < 1$ and $\omega_T + \omega_r + \omega_\rho = 1$.

In MCOP-K selection, a *feasible solution* is the path, where the aggregated QoT attributes of that path can satisfy multiple end-to-end QoT constraints. The goal of K *optimal social trust path* selection is to select K social trust paths which are feasible and can yield the K best utilities with the weights specified by the source participant [12].

V. K Optimal Social Trust Paths Selection

In this section, we first analyze some existing algorithms for K shortest paths selection and then propose an efficient heuristic algorithm H-OSTP-K for the NP-Complete MCOP-K selection in complex social networks.

A. Existing Algorithms

K shortest paths selection has been used in many applications, such as power transmission route selection, automatic translation between natural languages, and biological sequence alignment [5]. In the literature, several algorithms have been proposed to solve the K shortest paths selection problem, including (1) algorithms to find K *general shortest path* (paths allowing loops), and (2) algorithms to find K *simple shortest paths* (paths without loops) [5]. As a social trust path may contain loops [8], we introduce some existing algorithms for finding K general shortest paths as follows.

The algorithms for finding K general shortest paths can be classified into two categories. They are (1) K general paths selection based on Dijkstra's shortest algorithm [4], and (2) K general paths selection based on A^* algorithm.

In *Category 1*, Fox [6] proposes a K paths selection algorithm, where each intermediate node $v_k,\ (v_k \neq v_s)$ records up to K minimal path lengths from v_s to v_k. At each step, up to K nodes are selected from a priority queue as the expansion nodes based on the maximal path length record at the nodes. If a node is selected, the algorithm counts the number of times it has been visited. If all the nodes have been visited K times, the K shortest paths from v_s to each node of the sub-network are selected. Miaou [24] proposes a similar algorithm by using a binary heap to store the priority queue, which improves the efficiency of K path selection. The time complexity of this type of algorithm is $O(m + Knlogn)$. Throughout this paper, K $(K \geq 2)$ stands for the number of selected paths, m for the number of links, and n for the number of nodes.

In *Category 2*, Yen proposes a classic K general shortest paths selection algorithm based on the A^* algorithm [31]. This algorithm first computes the shortest path from v_s to v_t. Then it regards each node of the newly discovered shortest path as a *deviation node*. For each deviation node, this algorithm executes a single-source shortest path algorithm from the deviation node to v_t, forming a candidate *deviation path*. The next shortest path is chosen from all the candidates deviation paths with the minimal path length. This process continues until K different shortest paths are finally determined. In addition, Martins [21] improves the runtime performance of Yen's algorithm by ordering the deviation node based on deviation paths' length. Furthermore, Eppsten [5] proposes a well-known K general shortest paths selection algorithm. This algorithm builds a shortest path tree rooted at the target node first, then selects certain links outside the shortest path tree, forming the paths to be discovered. The time complexity of Eppsten's algorithm reaches $O(m + nlogn + K)$, which is also the lowest bound of the K general paths selection problem.

The above algorithms address the K general shortest path selection problem well. However, they are all deterministic and thus can not be used to solve the NP-Complete MCOP-K selection problem [2].

B. Our Proposed H-OSTP-K

In this section, we propose a novel heuristic algorithm H-OSTP-K, for the K optimal social trust path selection with end-to-end QoT constraints in complex social networks. In H-OSTP-K, we first adopt the *Backward_K-Search* procedure from v_t to v_s to (1) investigate whether there exists a feasible solution in the sub-network, (2) indicate the number of feasible solutions when this number is less than K $(K \geq 2)$, and (3) record the aggregated QoT attributes (i.e., T, r and ρ) of the identified K paths from v_t to each intermediate node v_k. If there exists at least one feasible solution, we then adopt the *Forward_K-Search* procedure to search the network from v_s to v_t to deliver the near-optimal solutions (*see Algorithm 1*).

In MOCP-K selection, if a path satisfies multiple QoT constraints, it means that each aggregated QoT attribute of that path should be larger than the corresponding QoT constraint. Based on this observation, we propose an objective function in Eq. (5) to investigate whether the aggregated QoT attributes of a path can satisfy the QoT constraints. From Eq. (5), we can see that $\delta(p) \leq 1$, *if and only if* each aggregated QoT attribute of a social trust path satisfies the corresponding QoT constraint. Otherwise $\delta(p) > 1$.

$$\delta(p) \triangleq max\{(\frac{1-T_p}{1-Q^T_{v_s,v_t}}),(\frac{1-r_p}{1-Q^r_{v_s,v_t}}),(\frac{1-\rho_p}{1-Q^\rho_{v_s,v_t}})\} \quad (5)$$

Backward_K-Search: Assume there exist at least K social trust paths in the sub-network. In the backward search from v_t to v_s, H-OSTP-K identifies K social trust paths from v_t to v_s (denoted as $p^{B_1}_{v_s \to v_t}$ to $p^{B_K}_{v_s \to v_t}$) with the K minimal δ based on Dijkstra's shortest path algorithm [4]. In the searching process, at v_k, the aggregated QoT attributes of K paths from v_t to v_k with the K minimal δ are recorded. According to the results in our previous work [20], the *Backward_K-Search* procedure can investigate whether there exists a feasible solution in the sub-network. In addition, according to *Theorem 1* given below, this procedure can also indicate the number of feasible solutions when there exist less than K feasible solutions in the sub-network (*see Algorithm 2*).

Theorem 1: In the *Backward_K-Search* procedure, the process of identifying K paths with the K minimal δ can indicate the number of feasible solutions when there exist less than K feasible solutions in a sub-network.

Proof: Let $p^{B_1}_{v_s \to v_t}, ..., p^{B_K}_{v_s \to v_t}$ be the K paths identified by the *Backward_Search* procedure from v_t to v_s with the K minimal δ value, and $\mathcal{S}$ is the number of feasible solutions in the subnetwork between v_s and v_t. In the identified K paths from v_s to v_t, if there exists G $(0 < G < K)$ paths (denoted as $p^{B_1}_{v_s \to v_t}, ..., p^{B_G}_{v_s \to v_t}$), where $\delta(p^{B_1}_{v_s \to v_t}) \leq 1, ..., \delta(p^{B_G}_{v_s \to v_t}) \leq 1$, then based the theorems in [20], there exist at least G feasible solutions in the sub-network between v_s and v_t (i.e., $\mathcal{S} \geq G$). In addition, the *Backward_Search* procedure can always identify K paths with K minimal δ value [24]. Therefore, there exist no more than G feasible solutions in the sub-network between v_s to v_t (i.e., $\mathcal{S} \leq G$). Then $\mathcal{S} = G$. □

Without loss of generality, we assume there are at least K social trust paths in the sub-network, though not all of them are feasible solutions. The *Backward_K-Search* can always identify K paths with the K minimal δ. In all the identified K paths, if $\delta_{min} > 1$, it indicates there is no feasible solution in the

Algorithm 1: H-OSTP-K

Data: M, $Q^{T}_{v_s,v_t}$, $Q^{r}_{v_s,v_t}$, $Q^{\rho}_{v_s,v_t}$, v_s, v_t, K
/ M is an adjacency matrix that represents the sub-network between v_s and v_t */*
Result: $\mathcal{F}(p^{F_1}_{v_s\to v_t})\ldots\mathcal{F}(p^{F_G}_{v_s\to v_t})$
begin
  $p_s=\emptyset$, $p_t=\emptyset$
  Backward_K-Search (M, $Q^{T}_{v_s,v_t}$, $Q^{r}_{v_s,v_t}$, $Q^{\rho}_{v_s,v_t}$, v_s, v_t, K)
  if *Min* $\delta(p^{B_1}_{v_s\to v_t})\ldots\delta(p^{B_K}_{v_s\to v_t})>1$ **then**
    return no feasible solution
  else
    return G,$BAQoT(v).T$, $BAQoT(v).r$, $BAQoT(v).\rho$
    / G is the number of feasible solution identified by the Backward_K-Search procedure, and BAQoT records the aggregated QoT attributes in the backward search. */*
    Forward_K-Search (M, $BAQoT(v).T$, $BAQoT(v).r$, $BAQoT(v).\rho$, $Q^{T}_{v_s,v_t}$, $Q^{r}_{v_s,v_t}$, $Q^{\rho}_{v_s,v_t}$, v_s, v_t, G)
    return $\mathcal{F}(p^{F_1}_{v_s\to v_t}),\ldots,\mathcal{F}(p^{F_G}_{v_s\to v_t})$
end

sub-network. If $\delta_{min}\leq 1$, it indicates there exists at least one feasible solution. In addition, if there exist G $(0<G<K)$ paths, where the δ values of these paths are no more than one, it means there are G feasible solutions in the sub-network.

Forward_K-Search: Assume there exist at least K ($K\geq 2$) feasible solutions in the sub-network. In the *Forward_K-Search* procedure, the aggregated QoT attribute values recorded at each v_k is adopted to identify whether there exist futher K paths $p^{F_1}_{v_s\to v_t},\ldots,p^{F_K}_{v_s\to v_t}$, each of which is better than the corresponding path of $p^{B_1}_{v_s\to v_t},\ldots,p^{B_K}_{v_s\to v_t}$ (i.e.,$\mathcal{F}(p^{F_1}_{v_s\to v_t})>\mathcal{F}(p^{B_1}_{v_s\to v_t}),\ldots,\mathcal{F}(p^{F_K}_{v_s\to v_t})>\mathcal{F}(p^{B_K}_{v_s\to v_t})$ (*see Algorithm 3*).

In this procedure, H-OSTP-K first searches the path with the K maximal $\mathcal{F}$ value from v_s. Assume node $v_m\in\{neighboring\ nodes\ of\ v_s\}$ is selected based on Dijkstra's shortest path algorithm in the i^{th} path ($i\in[1,K]$). H-OSTP-K calculates the aggregated QoT attribute values of the path from v_s to v_m (denoted as path $p^{F_i}_{v_s\to v_m}$). Then K *foreseen paths* from v_s to v_t via v_m (denoted as $fp^{F_i+B_\sigma}_{v_s\to v_m\to v_t}=p^{F_i}_{v_s\to v_m}+p^{B_\sigma}_{v_m\to v_t}$ $(\sigma\in[1,K])$) are formed. Depending on whether $fp^{F_i+B_\sigma}_{v_s\to v_m\to v_t}$ is feasible, H-OSTP-K adopts the following searching strategies.

Situation 1: If each aggregated QoT attribute of one of the foreseen paths from v_s to v_t via v_m, (i.e., $fp^{F_i+B_\sigma}_{v_s\to v_m\to v_t}$ $(\sigma\in[1,K])$ satisfies the corresponding end-to-end QoT constraint, then v_m is put into the priority queue for the next search step.

Situation 2: If all the foreseen paths $fp^{F_i+B_\sigma}_{v_s\to v_m\to v_t}$ $(\sigma\in[1,K])$ are infeasible, v_m is not put into the priority queue. Subsequently, H-OSTP-K performs the *Forward_K-Search* procedure to search the path from v_s in the sub-network without taking the link $v_s\to v_m$ into consideration.

Theorem 2: If v_t is selected from the priority queue, then a social trust path from v_s to v_t is identified (denoted as p_t). If any of the K optimal social trust paths has not been identified, p_t is one of the K optimal social trust paths.

Proof: Let $p^{F*}_{v_s\to v_t}$ denote the path from v_s to v_t that is selected from the priority queue at the J^{th} step. Let $p^{F_1}_{v_s\to v_t},\ldots,p^{F_K}_{v_s\to v_t}$ denote the K optimal social trust paths from v_s to v_t identified by the *Forward_K-Search* procedure. If $p^{F*}_{v_s\to v_t}\notin\{p^{F_1}_{v_s\to v_t},\ldots,p^{F_K}_{v_s\to v_t}\}$, then $\mathcal{F}(p^{F*}_{v_s\to v_t})$ is less than any of $\{\mathcal{F}(p^{F_1}_{v_s\to v_t}),\ldots,\mathcal{F}(p^{F_K}_{v_s\to v_t})\}$. At the J^{th} step, in addition to v_t, there are $K-1$ nodes selected from the priority queue. Thus, at least one node in paths $\{p^{F_1}_{v_s\to v_t},\ldots,p^{F_K}_{v_s\to v_t}\}$ is not selected at the J^{th} step. Then $\mathcal{F}(p^{F*}_{v_s\to v_t})$ is greater than one of $\{\mathcal{F}(p^{F_1}_{v_s\to v_t}),\ldots,\mathcal{F}(p^{F_K}_{v_s\to v_t})\}$, which contradicts that $\mathcal{F}(p^{F*}_{v_s\to v_t})$ is less than any of $\{\mathcal{F}(p^{F_1}_{v_s\to v_t}),\ldots,\mathcal{F}(p^{F_K}_{v_s\to v_t})\}$.

Algorithm 2: Backward_K-Search

Data: M, $Q^{T}_{v_s,v_t}$, $Q^{r}_{v_s,v_t}$, $Q^{\rho}_{v_s,v_t}$, v_s, v_t, K
Result: $BAQoT(v).T$, $BAQoT(v).r$, $BAQoT(v).\rho$
begin
  set $v_x.\delta=\infty$ $(v_x\neq v_t)$, $v_t.\delta=0$, $S_x=\emptyset$, $v_x.bvisit=0$, $G=0$
  add v_t into S_x
  while $S_x\neq\emptyset$ **do**
    $S_{TopK}=K\ min(v^*_a.\delta)$ $(v^*_a\in S_x)$
    / S_x is the priority queue in the backward search, and S_{TopK} is a set that contains the K minimal δ values. */*
    for *each* $v_a\in S_{TopK}$ **do**
      if $v_a==v_s$ *and* $v_a.\delta\leq 1$ **then**
        $G=G+1$
      for *each* $v_b\in adj[v_a]$ **do**
        / $adj[v_a]$ are all neighboring nodes of v_a */*
        $p_b=v_b$ to v_t via v_a
        if $v_b\notin S_x$ **then**
          put v_b into S_x
        else if $\delta(p_b)<max(v_b.\delta)$ **then**
          update $BAQoT(v_b).T$, $BAQoT(v_b).r$, $BAQoT(v_b).\rho$
          put v_b into S_x
      $v_a.bvisit=v_a.bvisit+1$
      / the visited times of v_a plus one */*
      if $v_a.bstatus==K$ **then**
        remove v_a from S_x
  return G, $BAQoT(v).T$, $BAQoT(v).r$, $BAQoT(v).\rho$
end

Algorithm 3: Forward_K-Search

Data: M, $BAQoT(v).T$, $BAQoT(v).r$, $BAQoT(v).\rho$, $Q^{T}_{v_s,v_t}$, $Q^{r}_{v_s,v_t}$, $Q^{\rho}_{v_s,v_t}$, v_s, v_t, G
Result: $\mathcal{F}(p^{F_1}_{v_s\to v_t}),\ldots,\mathcal{F}(p^{F_G}_{v_s\to v_t})$
begin
  set $\mathcal{F}'=1/\mathcal{F}$, $v_y.\mathcal{F}'=\infty$ $(v_y\neq v_s)$, $v_s.\mathcal{F}'=0$, $S_y=\emptyset$, $v_y.fvisit=0$
  add v_s into S_y
  $J=G$
  / J is the number of unidentified paths from v_s to v_t. */*
  while $S_y\neq\emptyset$ **do**
    $S_{TopJ}(\mathcal{F}')=K\ min(v^*_i.\mathcal{F}')$ $(v^*_i\in S_y)$
    / S_y is the priority queue in the forward search, and S_{TopJ} is a set that contains the J minimal $\mathcal{F}'$ values. */*
    for *each* $v_i\in S_{TopJ}(\mathcal{F}')$ **do**
      if $v_i==v_t$ *and* $v_a.\delta\leq 1$ **then**
        $J=J-1$
        / Only $J-1$ paths need to be identified in the following search. */*
      for *each* $v_j\in adj[v_i]$ **do**
        / $adj[v_i]$ are all neighboring nodes of v_i */*
        $p_j=v_s$ to v_j via v_i
        if $\exists fp^{F_i+B_j}_{v_s\to v_j\to v_t}$ $(i,j\in[1,G])$ *is feasible* **then**
          if $v_j\notin S_y$ **then**
            put v_j into S_y
          else if $\mathcal{F}'(p_j)<Max(v_j.\mathcal{F}')$ **then**
            update $FAQoT(v_b).T$, $FAQoT(v_b).r$, $FAQoT(v_b).\rho$
            / FAQoT records the aggregated QoT attributes in the forward search. */*
            put v_j into S_y
      $v_i.fvisit=v_i.fvisit+1$
      / the visited times of v_i plus one */*
      if $v_i.fvisit==K*$ **then**
        remove v_i from S_x
  return $\mathcal{F}(p^{F_1}_{v_s\to v_t}),\ldots,\mathcal{F}(p^{F_G}_{v_s\to v_t})$
end

Therefore, *Theorem 2* is correct. □

Based on *Theorem 1* and *Theorem 2*, we propose two optimization strategies to improve the efficiency of the Forward K-Search procedure.

Optimization Strategy 1: The *Forward_K-Search* procedure is to identify up to K optimal social trust paths which are feasible. if there exist G $(0 < G < K)$ feasible solutions identified by the *Backward_K-Search* procedure based on *Theorem 1* in a sub-network, the *Forward_K-Search* procedure does not need to search K paths but G paths from v_s to v_t.

Optimization Strategy 2: If v_t has been selected J $(1 \leq J < K)$ times from the priority queue, in the following process, H-OSTP-K only needs to search $K - J$ optimal social trust paths from v_s to v_t.

Then, if there exist l $(1 \leq l \leq K)$ feasible solutions, the *Forward_K-Search* procedure can identify them all, and they are the l optimal social trust paths. Otherwise, this procedure can identify K feasible solutions which are not worse than those identified by the *Backward_K-Search* procedure. Namely, *Theorem 1* and *Theorem 2* can guarantee the effectiveness of our algorithm.

Since H-OSTP-K adopts Dijkstra's shortest path algorithm based K general social trust paths selection method twice, it has the same time complexity of $O(m + Knlogn)$ as that of the algorithms in *Category 1*.

VI. Experiments

A. Experiment Settings

The *Enron* email dataset[2] has been proved to possess the small-world and power-law characteristics of social networks and thus it has been widely used in the studies of social networks [19, 20, 22, 32]. In addition, as we explained in *Section III*, the social intimacy degree between participants and the role impact factor of participants can be calculated through mining the subjects and contents of emails in the *Enron* email dataset [22]. Therefore, in contrast to other real social network datasets (e.g., Epinions[6] and FilmTrust[7]), the *Enron* email dataset fits our proposed complex social network structure better. Thus, to validate our proposed algorithm, we select the *Enron* email dataset[2] with 87,474 nodes (participants) and 30,0511 links (formed by sending and receiving emails) as the dataset for our experiments.

Firstly, in order to study the performance of our proposed heuristic algorithm in sub-networks of different scales and structures, we first randomly select 100 pairs of source and target participants from the *Enron* email dataset[2]. We then extract the corresponding 100 sub-networks between them by using the exhaustive search method. Among them, the maximal length of a social trust path varies from 4 to 7 hops following the *small-world* characteristic. These sub-networks are grouped by the number of hops. In each group they are ordered by the number of nodes in them. In the simplest case, the sub-network has 33 nodes and 56 links (4 hops), while in the most complex case, the sub-network has 1695 nodes and 11175 links (7 hops).

Secondly, as we have analyzed in *Section V-A*, existing K general shortest paths selection algorithms are all deterministic algorithms, and can not be used for solving the NP-Complete MCOP-K problem [2]. Therefore, to study the performance of our heuristic H-OSTP-K, we first compare the maximal utility of the identified K social trust paths with that of our previously proposed H_OSTP, which so far outperforms the other existing algorithms for the NP-Complete Multiple Constrained Optimal social trust Path (MCOP) selection problem in complex social networks [20]. In addition, since existing methods are not suitable for the NP-Complete MCOP-K selection problem, in order to study the efficiency of our proposed optimization strategies, we compare the execution time of H-OSTP-K with that of the modified version of H-OSTP-K without our proposed optimization strategies (denoted as H-WOP-K) (*see Section VI-B*).

[6]http://epinions.com/

[7]http://trust.mindswap.org/FilmTrust/

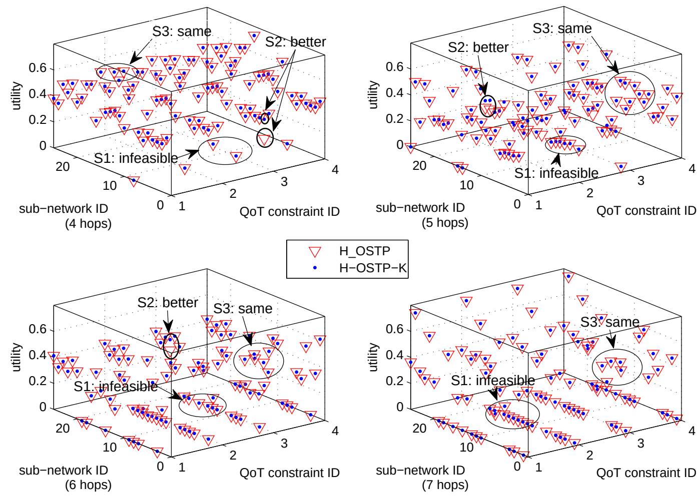

Figure 3. The path utilities of sub-networks with each group of hops

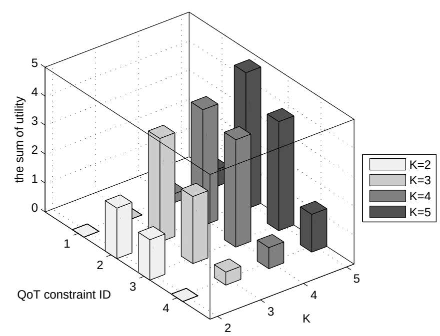

Figure 4. The sum of path utilities with different K values

Finally, to investigate the performance of H-OSTP-K in social trust path selection with different QoT constraints, four groups of QoT constraints are set and listed in Table I. In addition, the three QoT attributes are given the same weights in the utility function. Furthermore, since the detailed mining method of QoT attributes values are out of scope of this paper, these QoT attributes values are randomly generated by using $rand()$ in *Matlab*.

Each of H-OSTP-K, H-WOP-K and H_OSTP is implemented using Matlab R2008a running on an IBM ThinkPad SL500 laptop with an Intel Core 2 Duo T5870 2.00GHz CPU, 3GB RAM, Windows XP SP3 operating system and MySql 5.1.35 database. The results are plotted in Fig. 3 to Fig. 8, where the execution time of each of H-OSTP-K and H-WOP-K is averaged based on 3 independent runs.

B. Experiment Results

Comparison of Path Utility: Fig. 3 plots the path utilities of the identified social trust path by H_OSTP and the maximal path utility of the identified K optimal social trust paths by H-OSTP-K in sub-networks in 4 groups. From these figures, we can observe that in some sub-networks (i.e., 32 out of 100 sub-networks), if there is no feasible solution, both H-OSTP-K and H_OSTP can investigate the infeasibility (e.g., S1 in Fig. 3), and thus the path utilities in these sub-networks are zero. This is because that H_OSTP also computes δ_{min} in the backward search from v_t to v_s based on Dijkstra's shortest path algorithm

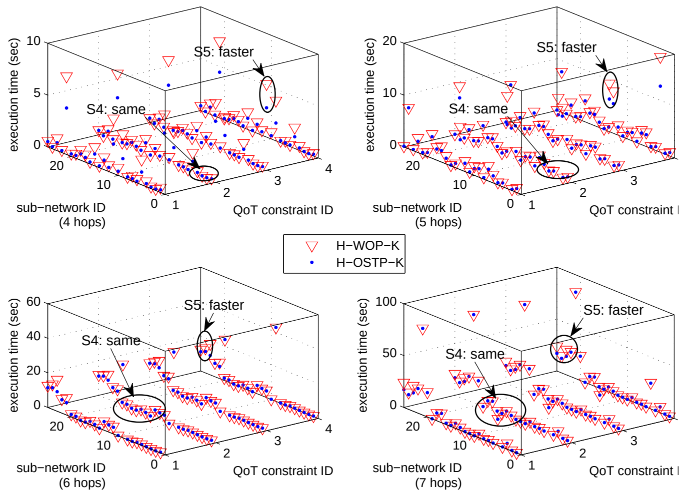

Figure 5. The execution time of $K = 2$

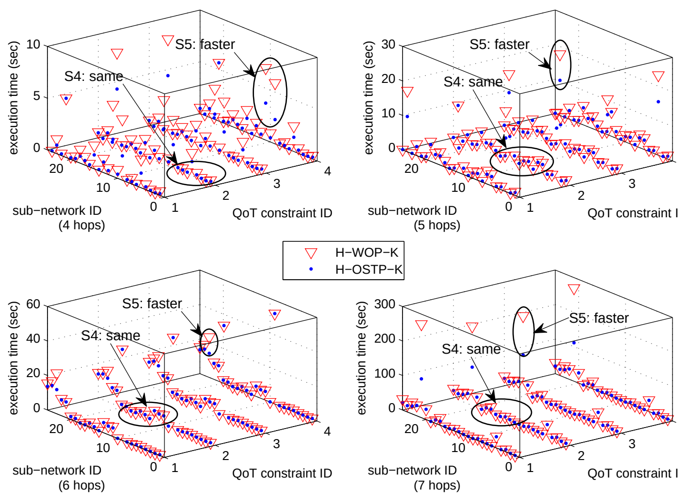

Figure 6. The execution time of $K = 3$

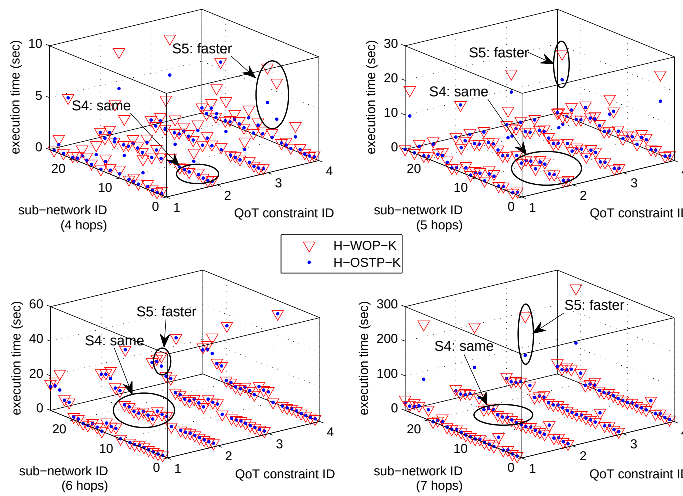

Figure 7. The execution time of $K = 4$

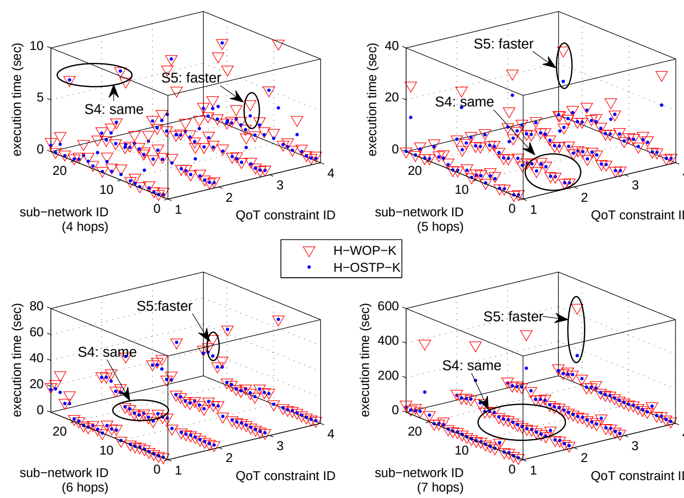

Figure 8. The execution time of $K = 5$

[20]. Therefore, based on the theorems in [20], both of them can always investigate whether there is a feasible solution existing in a sub-network.

In addition, from Fig. 3, we can see that in some cases (i.e., 49 out of 100 sub-networks), H-OSTP-K can deliver the same path utilities with those of H_OSTP (e.g., S3 in Fig. 3). This is because that firstly, in a sub-network, if the path with the maximal path utility in the K paths identified by H-OSTP-K and the path identified by H_OSTP are selected based on the same foreseen path formed at each of the intermediate nodes of these paths, according to the searching strategies in [20], the two paths are the same. Secondly in a sub-network, if there exists only one feasible solution, both H-OSTP-K and H_OSTP can identify it, and thus they deliver the same path utility.

Furthermore, from Fig. 3, we can also see that H-OSTP-K can deliver better social trust paths than H_OSTP (e.g., S2 in Fig. 3) in some cases (i.e., 19 out of 100 sub-networks). In addition, as depicted in Fig. 4, given the same constraint ID, the larger the K value, the greater the sum of the utility of these sub-networks. Table II lists the sum of utilities computed by H-OSTP-K and H_OSTP in these sub-networks, where we can see that the sum of utilities computed by our proposed heuristic algorithm is 49.66% higher than that of H_OSTP in 4 hops sub-networks, 20.24% higher in 5 hops, 13.39% higher in 6 hops. On average, the path utility computed by H-OSTP-K is 20.29% higher than that of H_OSTP. This is because that in H-OSTP-K, it is to form K foreseen paths rather than only one foreseen path in H_OSTP, and thus H-OSTP-K have more chances to deliver a better social trust path.

Comparison of Execution Time: Since H-WOP-K has the same functionality as H-OSTP-K, they both deliver the same path utilities of K paths in a sub-network. Therefore, we only compare the difference in their execution time, and the experiment results are plotted in Fig. 5 to Fig. 8.

From Fig. 5 to Fig. 8, we can observe that the execution time of H-OSTP-K is the same as that of H-WOP-K in some sub-networks (e.g., S4 in Fig. 5 to Fig. 8). This is because if there is no feasible solution in a sub-network, H-OSTP-K only performs the *Backward_K-Search* procedure which has the same search strategy with H-WOP-K. Therefore, they have the same execution time.

In addition, from these figures, we can also observe that the execution time of H-OSTP-K is less than that of H-WOP-K in other sub-networks (e.g., S5 in Fig. 5 to Fig. 8). The total execution time of each of H-OSTP-K and H-WOP-K in each group of hops is listed in Table III, where we can see that

Table I
THE SETTING OF QOT CONSTRAINTS

Constraint ID	$Q^{T}_{v_s,v_t}$	$Q^{r}_{v_s,v_t}$	$Q^{\rho}_{v_s,v_t}$
#1	0.05	0.05	0.05
#2	0.1	0.05	0.05
#3	0.05	0.1	0.05
#4	0.05	0.05	0.1

Table II
THE COMPARISON OF PATH UTILITY

Algorithms	The sum of utility				
	4 hops	5 hops	6 hops	7 hops	total
H-OSTP-K	2.5461	17.9369	8.0839	0	28.5669
H_OSTP	1.7012	14.9174	7.1295	0	23.7481
difference	49.66% higher	20.24% higher	13.39% higher	0	20.29% higher

the total execution time of our proposed heuristic algorithm is only 41.86% of that of H-WOP-K in 4 hops sub-networks, 70.60% in 5 hops, 89.51% in 6 hops and 51.03% in 7 hops. On average, H-OSTP-K is 37.22% faster than H-WOP-K. From the above results, we can see that H-OSTP-K is much more efficient than H-WOP-K. The reasons are twofold. Firstly based on *Theorem 1*, if there exist G $(0 < G < K)$ feasible solutions, then H-OSTP-K searches only G optimal social trust paths from v_s to v_t, significantly saving execution time (see details in *Optimization Strategy 1*). Secondly, based on *Theorem 2*, assuming there exist K $(K \geq 2)$ feasible solutions and v_t has been selected J $(0 < J < K)$ times from the priority queue, then in the following searching steps, H-OSTP-K searches only $K - J$ optimal social trust paths from v_s to v_t, and thus saves execution time (see details in *Optimization Strategy 2*).

Through the above experiments conducted in sub-networks with different scales and structures, we can see that H-OSTP-K is an effective and efficient algorithm for MCOP-K selection in complex social networks.

VII. CONCLUSIONS

In this paper, we have presented a complex social network structure that takes trust information, social relationship and recommendation roles into account, reflecting the real-world situations better. For selecting the K optimal social trust paths with end-to-end QoT constraints in complex social networks, which is an NP-Complete problem, we have also proposed an efficient heuristic algorithm H-OSTP-K. The results of our experiments conducted on a real dataset of social networks demonstrate that H-OSTP-K significantly outperforms existing methods in the quality of identified social trust paths.

In our future work, we plan to develop a new trust-oriented social service search engine, which maintains a database of participants and the complex social network containing them. In this system, our proposed method will be applied, for instance, to help a buyer identify the most trustworthy seller from all sellers selling the product preferred by the buyer.

REFERENCES

[1] P. S. Adler. Market, hierarchy, and trust: The knowledge economy and the future of capitalism. *Organization Science*, 12(12):215–234, 2001.

[2] S. Baase and A. Gelder. *Computer Algorithms Introduction to Design and Analysis*. Addision Wesley, 2000.

[3] B. Christianson and W. S. Harbison. Why isn't trust transitivie? In *International Workshop on Security Protocols*, pages 171–176, 1996.

[4] E. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, pages 269–271, 1959.

[5] D. Eppstein. Finding the k shortest paths. *SIAM Journal on Computing*, 28(2):652–673, 1999.

[6] B. FoX. K-th shortest paths and applications to the probabilistic networks. *Joint National Meeting of ORSA/TIMS*, 23, 1975.

[7] L. Franken. Quality of service management: A model-based approach. *PhD Thesis, University of Twente, Netherlands*, 1996.

Table III
THE COMPARISON OF EXECUTION TIME

Algorithms	The sum of execution time (sec)				
	4 hops	5 hops	6 hops	7 hops	total
H-OSTP-K	1.4152e+003	4.1990e+003	9.4535e+003	2.1763e+004	3.6831e+004
H-WOP-K	2.2380e+003	5.4336e+003	1.0445e+004	3.2421e+004	5.0538e+004
difference	58.14% less	29.40% less	10.49% less	48.97% less	37.22% less

[8] J. Golbeck and J. Hendler. Inferring trust relationships in web-based social networks. *ACM Transactions on Internet Technology*, 6(4):497–529, 2006.

[9] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins. Propagation of trust and distrust. In *WWW'04*, pages 403–412, 2004.

[10] C. Hang, Y. Wang, and M. Singh. Operators for propagating trust and their evaluation in social networks. In *AAMAS'09*, pages 1025–1032, 2009.

[11] M. Jamali and M. Ester. Trustwalker: A random walk model for combinging trust-based and item-based recommendation. In *KDD'09*, pages 29–42, 2009.

[12] T. Korkmaz and M. Krunz. Multi-constrained optimal path selection. In *INFOCOM'01*, pages 834–843, 2001.

[13] R. F. Korte. Biases in decision making and implications for human resource development. *Advances in Developing Human Resources*, 5(4):440–457, 2003.

[14] J. Kunegis, A. Lommatzsch, and C. Bauckhang. The slashdot zoo: Mining a social network with negative edges. In *WWW'09*, pages 741–750, 2009.

[15] G. Levinger. Development and change. *Close Relationships*, pages 315–359, 1983.

[16] L. Li, Y. Wang, and E. Lim. Trust-oriented composite services selection and discovery. In *ICSOC'09*, pages 50–67, 2009.

[17] C. Lin, N. Cao, S. Liu, S. Papadimitriou, J. Sun, and X. Yan. Smallblue: Social network analysis for expertise search and collective intelligence. In *ICDE'09*, pages 1483–1486, 2009.

[18] G. Liu, Y. Wang, and M. Orgun. Quality of trust for social trust path selection in complex social networks. In *AAMAS'10*, 2010.

[19] G. Liu, Y. Wang, and M. A. Orgun. Optimal social trust path selection in complex social networks. In *AAAI'10*, pages 1397–1398, aaai.

[20] G. Liu, Y. Wang, M. A. Orgun, and E. p. Lim. A heuristic algorithm for service provider selection in complex social networks. In *SCC'10*, pages 130–137, 2010.

[21] E. Martins, M. Pascoal, and J. Santos. Deviation algorithms for ranking shortest paths. *International Journal of Foundations of Computer Science*, 10(3):247–261, 1999.

[22] A. Mccallum, X. Wang, and A. Corrada-Emmanuel. Topic and role discovery in social networks with experiments on enron and academic email. *Journal of Artificial Intelligence Research*, 30(1):249–272, 2007.

[23] K. Merton. The role set problems in sociological theory. *British Journal of Sociology*, 8(2):110–113, 1957.

[24] S. Miaou and S. Chin. Computing k-shortest path for nuclear spent fuel highway transportation. *European Journal of Operational Research*, 53:64–80, 1975.

[25] S. Milgram. The small world problem. *Psychology Today*, 1967.

[26] R. Miller, D. Perlman, and S. Brehm. *Intimate Relationships*. McGraw-Hill College, 4th edition, 2007.

[27] A. Mislove, M. Marcon, K. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. In *ACM IMC'07*, pages 29–42, 2007.

[28] J. Tang, J. Zhang, L. Yan, J. Li, L. Zhang, and Z. Su. Arnetminer: Extraction and mining of academic social networks. In *KDD'08*, pages 990–998, 2008.

[29] F. Walter, S. Battiston, and F. Schweitzer. A model of a trust-based recommendation system on a social network. *Autonomous Agent Multi-Agent System Journal*, 16(1):57–74, February 2008.

[30] Y. Wang and V. Varadharajan. Role-based recommendation and trust evaluation. In *IEEE EEE'07*, pages 278–295, 2007.

[31] J. Yen. Shortest path network problems. *Mathematical Systems in Economics*, 1975.

[32] S. Yoo, Y. Yang, F. Lin, and I. Moon. Mining social networks for personalized email prioritization. In *KDD'09*, pages 967–976, 2009.