4-2013

Finding the optimal social trust path for the selection of trustworthy service providers in complex social networks

Guanfeng LIU
Macquarie University

Yan WANG
Macquarie University

Mehmet A. ORGUN
Macquarie University

Ee Peng LIM
Singapore Management University, eplim@smu.edu.sg

DOI: https://doi.org/10.1109/TSC.2011.58

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

Guanfeng Liu, *Student Member*, *IEEE*, Yan Wang, *Senior Member*, *IEEE*, Mehmet A. Orgun, *Senior Member*, *IEEE*, and Ee-Peng Lim, *Senior Member*, *IEEE*

Abstract—Online Social networks have provided the infrastructure for a number of emerging applications in recent years, e.g., for the recommendation of service providers or the recommendation of files as services. In these applications, trust is one of the most important factors in decision making by a service consumer, requiring the evaluation of the trustworthiness of a service provider along the social trust paths from a service consumer to the service provider. However, there are usually many social trust paths between two participants who are unknown to one another. In addition, some social information, such as social relationships between participants and the recommendation roles of participants, has significant influence on trust evaluation but has been neglected in existing studies of online social networks. Furthermore, it is a challenging problem to search the optimal social trust path that can yield the most trustworthy evaluation result and satisfy a service consumer's trust evaluation criteria based on social information. In this paper, we first present a novel complex social network structure incorporating trust, social relationships and recommendation roles, and introduce a new concept, Quality of Trust (QoT), containing the above social information as attributes. We then model the optimal social trust path selection problem with multiple end-to-end QoT constraints as a Multiconstrained Optimal Path (MCOP) selection problem, which is shown to be NP-Complete. To deal with this challenging problem, we propose a novel Multiple Foreseen Path-Based Heuristic algorithm MFPB-HOSTP for the Optimal Social Trust Path selection, where multiple backward local social trust paths (BLPs) are identified and concatenated with one Forward Local Path (FLP), forming multiple foreseen paths. Our strategy could not only help avoid failed feasibility estimation in path selection in certain cases, but also increase the chances of delivering a near-optimal solution with high quality. The results of our experiments conducted on a real data set of online social networks illustrate that MFPB-HOSTP algorithm can efficiently identify the social trust paths with better quality than our previously proposed H_OSTP algorithm that outperforms prior algorithms for the MCOP selection problem.

Index Terms—Trust, social networks, trust path selection, service selection

1 Introduction

Online social networking sites have become very popular, attracting a large number of participants and are being used as a means for a variety of rich activities. For example, according to a survey on 2,600 hiring managers in 2008 by CareerBuilder[1] (a popular job hunting website), 22 percent of those managers used social networking sites to investigate potential employees. In June 2009, the ratio increased to 45 percent. In addition, Microsoft has developed a dynamic CRM (Customer Relationship Management) system,[2] which allows business professionals to analyze customers' conversations on social networking sites, and as a consequence, provides real-time status updates about their products and services accordingly. In the above situations, trust is one of the most important factors for participants' decision making, requiring approaches and mechanisms for evaluating the trustworthiness between participants who are unknown to each other.

In service-oriented environments, social networks can be used as a 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, at FilmTrust,[3] which is a social networking site for movie recommendations, a participant can evaluate the trustworthiness of a recommender via the social network between them. As another 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 [20].

In social networks, each node represents a participant and each link between participants corresponds to the real-world interactions or online interactions between them (e.g., $A \rightarrow B$ and $A \rightarrow C$ in Fig. 1). One participant can give a trust value to another based on the direct interactions between them. For example, a trust rating can be given by a participant to another based on the quality of the movies recommended by the latter at FilmTrust. As each participant usually interacts with many other participants,

1. http://www.careerbuilder.com/.
2. http://crm.dynamics.com/.

• G. Liu, Y. Wang, and M.A. Orgun are with the Department of Computing, Macquarie University, North Ryde, NSW 2109, Australia.
E-mail: {guanfeng.liu, yan.wang, mehmet.orgun}@mq.edu.au.
• E.-P. Lim is with the School of Information Systems, Singapore Management University, 80 Stamford Road, Singapore 178902.
E-mail: eplim@smu.edu.sg.

Manuscript received 24 June 2011; revised 10 Oct. 2011; accepted 15 Nov. 2011; published online 15 Dec. 2011.
For information on obtaining reprints of this article, please send e-mail to: tsc@computer.org, and reference IEEECS Log Number TSCSI-2011-06-0062.
Digital Object Identifier no. 10.1109/TSC.2011.58.

3. http://trust.mindswap.org/filmtrust/.

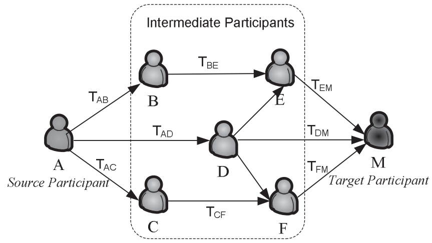

Fig. 1. A social network.

multiple trust paths may exist between two given participants who have no direct links with each other. For example, in Fig. 1, A and M are indirectly linked by two paths, $A \rightarrow B \rightarrow E \rightarrow M$ and $A \rightarrow D \rightarrow M$. If a trust path links two nonadjacent participants (i.e., there is no direct link between them), the source participant can evaluate the trustworthiness of the target one based on the trust information found in the path. This process is called *trust propagation* and the path with trust information linking the source participant and the target one is called a *social trust path* [14], [18]. For example, in Fig. 1, if A is a buyer and M is a seller, A can evaluate the trustworthiness of M using the social trust paths from A to M. We refer to A as the *source participant* and M as the *target participant*.

In large-scale social networks, there could be tens of thousands of social trust paths between a source participant and the target one [22]. Evaluating the trustworthiness of the target participant based on all these social trust paths is very time consuming. Alternatively, we can search the *optimal* path yielding the most trustworthy trust propagation result from multiple paths. We call this the optimal social trust path selection problem which is known to be a challenging research problem [29].

In the literature, Lin et al. [26] propose an optimal social path selection method, where all links are assigned the same weight and the shortest path between the source participant and the target one is selected as the optimal one. This method neglects *trust information* between participants. In another work [18], the path with the maximal propagated trust value is selected as the most trustworthy social trust path. However, *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 student's job application) have significant influence on trust propagation [1], [35] and can be discovered by using data mining techniques [32]. However, these factors have not been considered in other existing trust propagation and social trust path selection methods. In addition, a source participant may have different purposes in evaluating the trustworthiness of the target participant, such as hiring employees, or introducing products. Therefore, a source participant may have different social trust path selection criteria (e.g., with more focus on the recommendation roles of participants in employment and/or with more focus on the social relationships between participants in making friends) and should be able to set certain constraints on the above factors in trust propagation. This can help the source participant select the optimal social trust path that yields the most trustworthy trust propagation result. However, such a capability is not supported by existing methods [18], [26].

To address the above issues, in our previous work [29],[4] we have proposed a social trust path selection method where the above impact factors and participants constraints are considered. In addition, we proposed a heuristic algorithm H_OSTP for optimal social trust path selection and demonstrated that H_OSTP outperformed the most promising algorithm for the path selection problem in both the quality of the selected path and the efficiency. However, this work still has some disadvantages. In some cases, H_OSTP cannot deliver a near optimal solution with a high utility. The advantages and disadvantages of this algorithm are analyzed in detail in Section 5.2.

In this paper, we aim to solve the optimal social trust path selection problem in a social network, which contains complex social relationships and recommendation roles. Our contributions in this paper are summarized as follows:

1. We first present the structure of complex social networks and a novel concept, *Quality of Trust* (QoT).[5] We then model the multiple QoT constrained optimal social trust path selection problem as a Multiconstrained Optimal Path (MCOP) selection problem, which is proved to be NP-Complete in [21] (*see Section 4*).
2. Based on our previously proposed heuristic algorithm H_OSTP [29], we propose a novel Multiple Foreseen Path-Based Heuristic algorithm, MFPB-HOSTP, where multiple Backward Local Paths (BLPs, rather than only one path in H_OSTP) are identified in the backward search from a target participant to the source participants. These BLPs will be used in the forward search from the source to the target, forming multiple foreseen paths, in order to avoid a failed feasibility estimation of a foreseen path. Our novel search strategies can help deliver better solutions than H_OSTP (*see Sections 5 and 6*).
3. We have conducted extensive experiments on a real online social network data set, *Enron* e-mail corpus,[6] which is formed by sending and receiving e-mails between participants. Experimental results have demonstrated the good performance of our proposed algorithm MFPB-HOSTP (*see Section 7*).

The paper is organized as follows: Section 2 introduces related work. Section 3 presents the complex online social network structure which incorporates social relationships and recommendation roles. Section 4 presents a novel social trust path selection model. While Section 5 analyzes existing social trust path selection algorithms, Section 6 proposes a novel heuristic algorithm, MFPB-HOSTP. Section 7 presents the experimental results and analysis. Finally, Section 8 concludes this paper with a summary and discussion of future work.

2 Related Work

2.1 Social Network Analysis

The studies of social network properties can be traced back to 1960s when the *small-world* characteristic in social

4. This paper won the Best Paper Award of IEEE SCC 2010.
5. The complex social network structure and the QoT concept have been presented in our previous work published at IEEE SCC 2010 [29].
6. http://www.cs.cmu.edu/enron/.

networks was validated by Milgram [34], through illustrating that the average path length between two Americans was about 6 hops in an experiment of mail sending. In addition, the influences of small-world characteristic on human interactions was further analyzed by Pool and Kochen [37] in the 1970s. In recent years, as online social networks have been gaining more popularity, sociologists and computer scientists have started to investigate their characteristics. In [36], Mislove et al. analyzed several popular social networks including Facebook,[7] MySpace,[8] and Flickr,[9] and validated the *small-world* and *power-law* characteristics (i.e., in a social network, the probability that a node has degree k is proportional to k^{-r}, $r > 1$) of online social networks using data mining techniques. Also using data mining techniques, Mccallum et al. [32] discovered the social roles (e.g., *a chief financial officer* or *in-house lawyer*) and social relationships (e.g., *partnership in a funding application*) in an e-mail-based online social network of *Enron* Corporation. Guo et al. [17] further analyzed the influence of social interactions between buyers on the purchase decisions made by a buyer in buying products in online shopping websites.

2.2 Social Trust Evaluation in Online Social Networks

Trust is a critical factor in the decision making of participants in online social networks [23]. In this field, several trust management methods have been proposed.

In the studies of trust propagation, Golback and Hendler [14] proposed a trust inference mechanism for establishing the trust relation between a source participant and the target one based on averaging trust values along the social trust paths. They further adopted this model into an online social network of film recommendations to indicate the reputation of films. Guha et al. [16] proposed 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. In [30], a trust antecedent framework is used to determine trust relevant feature categories, namely 1) trustee ability, 2) trustee benevolence, and 3) trustee integrity to derive features for predict the trust level between two users.

In the studies of trust-oriented recommendation systems, Walter et al. [39] proposed a recommendation system in a social network. In their model, a participant can give a trust value to a recommender based on the recommendation behavior of participants. This trust value is visible and regarded as a reference for other participants to select recommendations. Jamali and Ester [20] proposed a random walk model in a social network consisting of sellers and buyers. In their model, 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 given by the ending participant to a seller who sells products preferred by the buyer. The degree of confidence on the seller is calculated based on the number of random walk paths, hops, and ratings of the seller in each path.

7. http://www.facebook.com.
8. http://www.myspace.com.
9. http://www.flickr.com.

The above trust management strategies are solely based on trust ratings given by participants. As pointed out in social science theories [1], [35], *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) both have significant influence on participants' decision making.

2.3 Social Trust Influence on Service Selection

As indicated in social psychology [5], [12], in the reality of our society, a person prefers the recommendation from his/her trusted friends over those from others. In addition, in the discipline of computer science, based on statistics, Bedi et al. [4] has demonstrated that, given a choice between recommendations from trusted friends and those from recommender systems, trusted friends' recommendations are more preferred in terms of quality and usefulness. Furthermore, in several recent studies, some researchers [8], [10] have investigated how and to what extent a participant's service selection behavior (e.g., installing a specific application software) impacts on his/her friends' decision making in service selection. These studies have indicated that the recommendations from trusted friends have significant influence on service or target selection, not only in the society in the real world, but also in online social networks.

Although a complete social network-based trust-oriented service recommendation system does not yet exist, it has become an important research topic in recent years. Some researchers [17], [31] have proposed several models to provide more accurate recommendations of products and/or services by taking some social context information into consideration. In these studies, social trust path selection is a critical problem. We analyze some existing studies for this problem in the following section.

2.4 Social Trust Path Selection Methods

In the literature, there are only a few works addressing the social path selection problem. *SmallBlue* [26] is an online social network constructed for IBM staff. In this system, up to 16 social paths with no more than 6 hops are selected between a source participant and a target participant and the shortest one is taken as the optimal path. However, in this method, some major factors including *trust information*, *recommendation roles* and *social relationships* between participants are not taken into account in path selection. Hang et al. [18] proposed a social trust path selection method in online social networks, where the social trust path with the highest belief (i.e., the maximum of propagated trust values) is selected as the optimal one that yields the most trustworthy result of trust propagation between a source participant and the target participant. Wang and Wu [40] aggregated trust values given to each of the recommenders (i.e., the intermediate node) in the network between a source participant and the target participant. If the aggregated trust value of a recommender is greater than the threshold specified by the source participant, the recommender is kept in the network for trust evaluation. Otherwise, the recommender (the node) is deleted from the network. In their models, although trust information is taken into consideration in trust path selection, they cannot be applied to social networks which contain social information, including social relationships and recommendation roles.

As mentioned above, a source participant can have different purposes in evaluating the trustworthiness of the target participants (e.g., employment or buying products). Therefore, the source participant can have different trust evaluation criteria in different applications, and thus they should be able to specify certain constraints of the above social impact factors for social trust path selection. But this flexibility is not supported in other existing methods.

3 Complex Social Networks

In this section, we present a complex social network structure originally proposed by us in [29]. Unlike the other existing models reported in the literature, it takes trust information, social relationships, and recommendation roles of participants into account.

3.1 Trust

In human societies, trust is a complex topic subject to a lot of factors, such as previous experience, and other people's recommendations [14]. Many different trust definitions have been proposed addressing different aspects. Alunkal et al. [2] define that "trust is the value attributed to a specific entity, including an agent, a service, or a person, based on the behaviors exhibited by the entity in the past." Golbeck and Hendler [14] define that "trust in a person is a commitment to an action based on a belief that the future action of that person will lead to a good outcome."

In the context of this paper, trust between participants in social networks can be defined as follows:

Definition 1. 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 $T_{AB} = 1$ indicates A completely believes B's future action can lead to the expected outcome.

3.2 Social Intimacy Degree

As illustrated in social psychology [3], 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. Therefore, we introduce the social intimacy degree between participants into complex social networks structure, and give its definition as follows:

Definition 2. $r_{AB} \in [0,1]$ *is the* Social Intimacy Degree *between any given participants A and B in online social networks. $r_{AB} = 0$ indicates that A and B have no social relationship while $r_{AB} = 1$ indicates they have the most intimate social relationship.*

3.3 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 [41]. 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. Therefore, we introduce the role impact factor of a participant into the complex social network structure, and give its definition as follows:

Definition 3. $\rho_A \in [0,1]$ *is the value of the* Role Impact Factor, *illustrating the impact of participant A's recommendation role on trust propagation. $\rho_A = 1$ 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 construct social relationships and comprehensive role hierarchies in all domains for the whole society, and obtain their global values, it is feasible to build them up in a specific social community.

For example, in the work by Mccallum et al. [32], through mining the subjects and contents of e-mails in *Enron* Corporation, the social relationship between each e-mail sender and receiver can be discovered and their roles can be known. Then, the corresponding social intimacy degree and role impact factor values can be estimated based on probabilistic models. In addition, in academic social networks formed by large databases of Computer Science literature (e.g., DBLP[10] or ACM Digital Library[11]), the social relationships between two scholars (e.g., coauthors, a supervisor, and his/her students) and the role of scholars (e.g., a professor in the field of data mining) can be mined from publications or their homepages. The social intimacy degree and role impact factor values can be calculated as an example by applying the PageRank model [38]. Furthermore, in addition to mining these values, the social position of a participant can be specified directly [42]. If the participant becomes a recommender, this social position information could illustrate his/her role impact factor in the recommendation of a specific domain.

Based on the above discussion, in addition to participants and the links between them, we propose a new structure for complex social networks that models trust, social intimacy degree, and role impact factors, as depicted in Fig. 2.

4 Quality of Trust and QoT Attributes Aggregation

In this section, we first present a novel general concept Quality of Trust and then propose a novel social trust path selection model with end-to-end QoT constraints [29].

4.1 Quality of Trust

In Service-Oriented Computing (SOC), quality of service (QoS) consists of a set of attributes, used to illustrate the ability of services to guarantee a certain level of performance [13]. Similar to QoS, we present a new concept, *Quality of Trust* [27].

Definition 4. Quality of Trust *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.*

10. http://www.informatik.uni-trier.de/ley/db/.
11. http://portal.acm.org/.

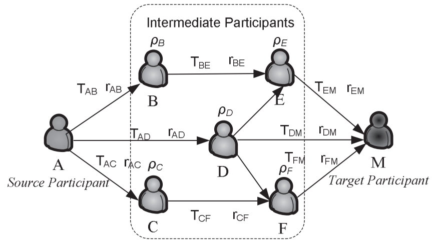

Fig. 2. Complex social network.

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 < 5 s, and availability $> 70\%$). In our model, to satisfy different trust evaluation criteria, a source participant can specify multiple end-to-end constraints for QoT attributes (i.e., T, r, and ρ) as the requirements of trust propagation in a social trust path of different domains.

Let $Q^{\mu}_{v_s,v_t}$ ($\mu \in \{T, r, \rho\}$) denote the end-to-end constraint of QoT attribute μ for the paths 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). For example, as shown in Fig. 2, to *hire employees*, A, a retailer manager specifies the end-to-end QoT constraints for the social trust paths from A to M as $Q_{AM} = \{Q^T_{A,M} > 0.3, Q^r_{A,M} > 0.3, Q^{\rho}_{A,M} > 0.8\}$, if he/she believes the social position of participants is more important in the domain of *employment*. But when looking for new customers for *selling products*, A could specify QoT constraints as $Q_{A,M} = \{Q^T_{A,M} > 0.8, Q^r_{A,M} > 0.3, Q^{\rho}_{A,M} > 0.3\}$, if he/she believes the social relationships between participants are more important in the domain of *product sale*.

4.2 QoT Attribute Aggregation

To specify end-to-end QoT constraints, we present the QoT attribute aggregation methods as follows [29].

4.2.1 Trust Aggregation

The trust values between a source participant and the target participant in a social path can be aggregated based on trust transitivity property (i.e., if A trusts B and B trusts C, then A trusts C to some extent) [14]. Since trust is discounted with the increase of transitivity hops [9], in our model, we adopt the strategy proposed in [25], [39], where if there are n participants $a_1, \ldots, a_n$ in order in a social trust path (denoted as $p(a_1, \ldots, a_n)$), the aggregated trust value is calculated as in (1). This strategy has been widely used in the literature as a feasible trust aggregation method [6], [28], [39].

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

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

4.2.2 Social Intimacy Degree Aggregation

First, social intimacy between participants decays with the increasing number of hops between them in a social trust path [24], [35]. In addition, in the real-world, the intimacy degree decays fast when it approaches 1. In contrast, the intimacy degree decays slowly when it approaches zero [7], [19]. Namely, the decay speed of the social intimacy degree is nonlinear in social networks. The aggregated r value in path $p(a_1, \ldots, a_n)$ can be calculated by (2) whose function image is a *hyperbolic curve*, fitting the characteristic of social intimacy attenuation [35].

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

4.2.3 Role Impact Factor Aggregation

As illustrated in social psychology [33], in the same society, the role impact factor of a participant *does not decay* with the increase of transitivity hops. Thus, the aggregated ρ value of $p_{(a_1,\ldots,a_n)}$ can be calculated by

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

4.3 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 the arguments in

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

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 $\omega_T + \omega_r + \omega_\rho = 1$.

The goal of optimal social trust path selection is to select the path that satisfies multiple end-to-end QoT constraints and yields the best utility with the weights specified by the source participant.

5 Social Trust Path Selection Algorithms

The optimal social trust path selection with multiple end-to-end QoT constraints can be modeled as the classical Multiconstrained Optimal Path selection problem which has been proved to be NP-Complete [21]. In this section, we first analyze some existing approximation algorithms for the MCOP selection problem, including our earlier H_OSTP algorithm [29], and then propose a novel Multiple Foreseen Path-Based Heuristic algorithm for Optimal Social Trust Path selection, MFPB-HOSTP.

5.1 Existing Algorithms

5.1.1 H_MCOP

Korkmaz and Krunz [21] propose a heuristic algorithm H_MCOP for the multiple-constrained optimal path selection in service invocation. In this algorithm, both multi-constrained values and QoS attributes values are aggregated based on

$$g_\lambda(p) \triangleq \left(\frac{q_1(p)}{Q^1_{v_s,v_t}}\right)^\lambda + \left(\frac{q_2(p)}{Q^2_{v_s,v_t}}\right)^\lambda + \cdots + \left(\frac{q_m(p)}{Q^m_{v_s,v_t}}\right)^\lambda, \quad (5)$$

where $\lambda \geq 1$; $q_i(p)$ is the aggregated value of the ith QoS attribute of path p (e.g., the total cost of the services in a path formed by service invocations); $Q^i_{v_s,v_t}$ is the ith QoS constraint value of the selected path between v_s and v_t (e.g., $Q^{cost}_{v_s,v_t} \leq \$100$).

H_MCOP first adopts Dijkstra's shortest path algorithm [11] to find the path with the minimum g_λ from v_t to v_s, which intends to investigate whether there exists a feasible solution satisfying all end-to-end QoS constraints in a subnetwork. In this process, at each intermediated node v_k, the aggregated value of each QoS attribute for the identified path from v_k to v_t is computed and recorded. If there exists at least one feasible solution, then these aggregated values are used in another search from v_s to v_t, which intends to identify a feasible path from v_s to v_t with the minimal cost of services.

Before we proposed H_OSTP in 2010 [29], H_MCOP was one of the most promising algorithms for the MCOP selection problem as it outperformed prior existing algorithms in both algorithm efficiency and solution quality [21], [29].

5.1.2 MCSP_K

Based on H_MCOP, in the field of Service-Oriented Computing, Yu et al. [44] propose an approximation algorithm, MCSP_K, which keeps only K paths from a source node to each intermediate node, aiming to reduce the search space and execution time. In their service candidate graph, each node represents a service and all services are categorized into different service sets based on their functionality. There is a link between any two nodes in adjacent service sets and thus all the paths from a source node to an intermediate node can be enumerated when necessary, avoiding an exhaustive search. But if a network does not have such a typical structure, MCSP_K has to search all the paths from a source node to each intermediate node and hence the time complexity becomes exponential. Therefore, it does not scale up to large social networks.

5.1.3 H_OSTP

In [29], based on Dijkstra's shortest path algorithm [11], we developed a novel efficient Heuristic algorithm for the Optimal Social Trust Path selection, called H_OSTP, in complex social networks.

In H_OSTP, we first proposed the objective function given in (6) and adopted the *Backward_Search* procedure to identify the path with the minimal δ from v_t to v_s to investigate whether there exists a *feasible solution* where all end-to-end QoT constraints can be satisfied in the subnetwork, and to record the aggregated QoT attributes (i.e., T, r, and ρ) of the path identified from v_t to each intermediate node v_k

$$\delta(p) \triangleq max\left\{\left(\frac{1-T_p}{1-Q^T_{v_s,v_t}}\right), \left(\frac{1-r_p}{1-Q^r_{v_s,v_t}}\right), \left(\frac{1-\rho_p}{1-Q^\rho_{v_s,v_t}}\right)\right\}. \quad (6)$$

If a feasible solution exists, H_OSTP then adopts the *Forward_Search* procedure to search the network from v_s to v_t to deliver a near-optimal solution. This process adopts the information provided by *Backward_Search* to identify whether there is another path $p^{forward}_{v_s \to v_t}$ which satisfies QoT constraints. In this process, H_OSTP first searches the path with the 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 as the utility of the path from v_s to v_m (denoted as the *forward local path (FLP)* $p^{f(u)}_{v_s \to v_m}$) is maximal. Let $p^{b(\delta)}_{v_m \to v_t}$ denote the *backward local path* from v_m to v_t identified in the *Backward_Search* procedure. Then, a *foreseen path* from v_s to v_t via v_m (denoted as $fp^{f(u)+b(\delta)}_{v_s \to v_m \to v_t} = p^{f(u)}_{v_s \to v_m} + p^{b(\delta)}_{v_m \to v_t}$) is formed.

If $fp^{f(u)+b(\delta)}_{v_s \to v_m \to v_t}$ is feasible, then H_OSTP chooses the next node from v_m with the maximal $\mathcal{F}$ value which is calculated based on Dijkstra's shortest path algorithm. Otherwise, H_OSTP does not search the path from v_m and the link $v_s \to v_m$ is deleted from the subnetwork. Subsequently, H_OSTP performs the *Forward_Search* procedure to search the path from v_s in the subnetwork without the link $v_s \to v_m$.

5.1.4 Other Algorithms

Some other algorithms [45], [46] adopt integer linear programming to solve the service selection problem with multi-QoS constraints. But, in [44], they have been proved having low efficiency in finding a near-optimal solution in large-scale networks.

5.2 Advantages and Disadvantage of H_OSTP

Advantages. H_OSTP could detect whether there exists a feasible solution in a subnetwork, as it adopts a new objective function $\delta(p)$ which is better than that of H_MCOP. If there exists at least one feasible solution, H_OSTP does not deliver any solution that is worse in quality than that of H_MCOP, and could possibly deliver better solutions than H_MCOP. In addition, when a foreseen path is infeasible (i.e., at least one aggregated QoT attribute value of the path does not satisfy the corresponding QoT constraint), the corresponding link between nodes is deleted, which reduces the search space and makes H_OSTP more efficient than H_MCOP [29].

Disadvantage. Although H_OSTP significantly outperforms existing approximation algorithms in both the efficiency and the quality of identified social trust paths, it still has a disadvantage called the *imbalance problem of QoT attributes*, which may cause a failed feasibility estimation of a foreseen path in the forward search procedure from v_s to v_t, and deliver a solution with a low utility that is not near optimal. We analyze the disadvantage of H_OSTP below in detail.

If a *feasible solution* (i.e., a path where the aggregated value of each QoT attribute satisfies the corresponding QoT constraint) exists in the subnetwork between v_s and v_t, H_OSTP performs the *Forward_Search* procedure, where H_OSTP investigates the feasibility of the foreseen path $fp^{f(u)+b(\delta)}_{v_s \to v_k \to v_t}$ to estimate whether a feasible solution can be delivered by following $p^{f(u)}_{v_s \to v_k}$. But, this strategy may give a failed feasibility estimation. Namely, even if $fp^{f(u)+b(\delta)}_{v_s \to v_k \to v_t}$ is infeasible, there may still exist a feasible solution identified by following $p^{f(u)}_{v_s \to v_k}$ in the subnetwork.

We use the following example to illustrate the imbalance problem of QoT attributes in H_OSTP. Fig. 3 depicts a social network between v_s and v_t, which contains five intermediate nodes v_1 to v_5, and the aggregated QoT attribute values computed by the *Backward_Search* procedure at each of these

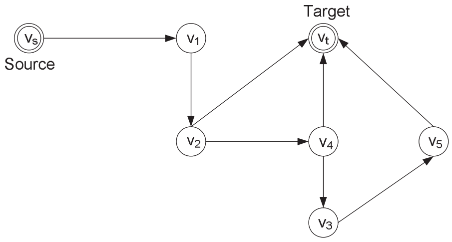

Fig. 3. Limitation of H_OSTP.

nodes are listed in Table 1. Suppose that v_s specifies the QoT constraints as $Q^T_{v_s,v_t} > 0.3$, $Q^r_{v_s,v_t} > 0.3$, and $Q^\rho_{v_s,v_t} > 0.2$. Based on the search strategy introduced in Section 5.1.3, at v_4, H_OSTP concatenates the social trust path $p^{f(u)}_{v_s \to v_4}$ with $p^{b(\delta)}_{v_4 \to v_t}$ to form a foreseen path $fp^{f(u)+b(\delta)}_{v_s \to v_4 \to v_t}$ with the aggregated QoT attributes values as $T = 0.2$, $r = 0.48$, and $\rho = 0.5$, which is infeasible (note: the aggregated $T = 0.2$ does not satisfy the corresponding constraint $Q^T_{v_s,v_t} > 0.3$). In such a situation, H_OSTP deletes the link $v_2 \to v_4$ in $p^{f(u)}_{v_s \to v_4}$ and selects another path $v_s \to v_1 \to v_2 \to v_t$ as the near-optimal social trust path between v_s and v_t. Suppose the QoT attributes have the same weights in the utility function, then the utility of this path is 0.35.

However, as shown in Fig. 3, the aggregated values of QoT attributes of another path $v_4 \to v_t$ (denoted as $p^{b(T)}_{v_4 \to v_t}$) are $T = 0.8$, $r = 0.45$, and $\rho = 0.5$. If we concatenate $p^{f(u)}_{v_s \to v_4}$ and $p^{b(T)}_{v_4 \to v_t}$ together, a new foreseen path $fp^{f(u)+b(T)}_{v_s \to v_4 \to v_t}$ is formed that is feasible. In such a situation, the path $v_s \to v_1 \to v_2 \to v_4 \to v_t$ with a utility of 0.39 is selected as the solution, which has a better quality than the one identified by H_OSTP (i.e., the $utility = 0.35$).

From the above example, we can see that the foreseen path formed by concatenating path $p^{f(u)}_{v_s \to v_k}$ with path $p^{b(\delta)}_{v_k \to v_t}$ may not accurately estimate whether there exists a feasible a solution identified by following $p^{f(u)}_{v_s \to v_k}$ in the forward search procedure. This is because during searching $p^{b(\delta)}_{v_k \to v_t}$, one of the aggregated values of the QoT attributes may be already close to the corresponding QoT constraints (e.g., $T = 0.5$ of $p^{b(\delta)}_{v_4 \to v_t}$ in Fig. 3). In such a situation, if the aggregated values of that QoT attribute is also close to the corresponding QoT constraint in $p^{f(u)}_{v_s \to v_k}$ (e.g., $T = 0.4$ of $p^{f(u)}_{v_4 \to v_t}$ in Fig. 3), the foreseen path at v_k is usually infeasible. This is the typical imbalance problem of QoT attributes (e.g., the imbalance problem of T at v_4 in Fig. 3), which may lead to a failed feasibility estimation of a foreseen path. In such a situation, H_OSTP cannot identify a social trust path with a high utility that is near optimal.

TABLE 1
Social Trust Paths and the Aggregated QoT Attributes Values

Path	Nodes and Links	T	r	ρ
$p^{f(u)}_{v_s \to v_4}$	$v_s \to v_1 \to v_2 \to v_4$	0.4	0.8	0.5
$p^{b(\delta)}_{v_4 \to v_t}$	$v_4 \to v_3 \to v_5 \to v_t$	0.5	0.6	0.5
$p^{b(T)}_{v_4 \to v_t}$	$v_4 \to v_t$	0.8	0.45	0.5
path $v_2 \to v_t$	$v_2 \to v_t$	0.75	0.4	0.4

6 Our Proposed MFPB-HOSTP Algorithm

Algorithm 1: MFPB-HOSTP

Data: $MT(v_s, v_t)$, $Q^T_{v_s,v_t}$, $Q^r_{v_s,v_t}$, $Q^\rho_{v_s,v_t}$
Result: $p^{forward}_{v_s \to v_t}$, $\mathcal{F}(p^{forward}_{v_s \to v_t})$

begin
- $p^{forward}_{v_s \to v_t} = \emptyset$, $p^{backward}_{v_s \to v_t} = \emptyset$
- **Backward_Search**($M(v_s, v_t)$, $Q^T_{v_s,v_t}$, $Q^r_{v_s,v_t}$, $Q^\rho_{v_s,v_t}$)
- **if** $\delta(p^{backward}_{v_s \to v_t}) > 1$ **then**
  - **Return** no feasible solution
- **else**
  - **Forward_Search**($MT(v_s, v_t)$, $AQ^\mu(p^{b(\delta)}_{v_k \to v_t})$, $AQ^\mu(p^{b(\mu)}_{v_k \to v_t})$, $AQ^\mu(p^{CBLP(\mu)}_{v_k \to v_t})$, $\mu \in \{T, r, \rho\}$, $Q^T_{v_s,v_t}$, $Q^r_{v_s,v_t}$, $Q^\rho_{v_s,v_t}$)
  - **Return** $p^{forward}_{v_s \to v_t}$ and $\mathcal{F}(p^{forward}_{v_s \to v_t})$

end

6.1 Algorithm Overview

We first introduce some definitions below that are used to describe our algorithm.

Definition 5 (Backward local path). *In a subnetwork from v_s to v_t, a Backward Local Path (BLP) is the path from v_t to an intermediate node v_k, identified by the backward search from v_t to v_s.*

Based on *Definition 5*, path $p^{b(\delta)}_{v_k \to v_t}$ identified by the backward search procedure is a BLP.

Definition 6 (Forward local path). *In a subnetwork from v_s to v_t, a Forward Local Path (FLP) is the path from v_s to an intermediate node v_k, identified by the forward search from v_s to v_t.*

Based on *Definition 6*, path $p^{f(u)}_{v_s \to v_t}$ identified by the forward search procedure is an FLP. A foreseen path can be formed at the same intermediate node v_k by concatenating an FLP that ends at node v_k and a BLP that starts from node v_k.

Definition 7 (Composite backward local path (CBLP)). *In a subnetwork between v_s and v_t, a Composite Backward Local Path is the path which is composed of the BLP with the minimal δ and the links of BLP with the maximal aggregated value for one of the QoT attributes.*

Based on the above definitions, we propose a novel Multiple Foreseen Path-Based Heuristic algorithm for Optimal Social Trust Path selection (MFPB-HOSTP) in complex social networks, which inherits the advantages of H_OSTP (i.e., the objective function) and aims to overcome its disadvantage (i.e., the imbalance problem of QoT attributes). Our MFPB-HOSTP also bidirectionally searches a subnetwork (i.e., by employing both a backward search and a forward search procedure) by adopting Dijkstra's shortest path algorithm [11]. But, our algorithm employs different search strategies from H_OSTP.

In the backward search procedure from v_t to v_s, at each intermediate node v_k, in addition to BLP $p^{b(\delta)}_{v_k \to v_t}$, MFPB-HOSTP first identifies the BLPs with the maximal aggregated T, r, and ρ values, respectively (denoted as $p^{b(\mu)}_{v_k \to v_t}, \mu \in \{T, r, \rho\}$). When facing with the imbalance problem of QoT attribute $\mu (\mu \in \{T, r, \rho\})$ at v_k (e.g., T at v_4 in Fig. 3), the identified BLPs $p^{b(\mu)}_{v_k \to v_t} (\mu \in \{T, r, \rho\})$ are concatenated with the identified FLP, forming other foreseen paths (e.g., $fp^{f(u)+b(T)}_{v_s \to v_4 \to v_t}$ in Fig. 3), helping avoid a failed feasibility estimation of a foreseen path and having a chance to deliver a better solution than H_OSTP (e.g., the path $v_s \to v_1 \to v_2 \to v_4 \to v_t$ in Fig. 3). However, greedily maximizing the aggregated value of the QoT attribute may cause a new imbalance problem of QoT attributes (see a detailed analysis in *Step 2* in the following section of *Algorithm Description*). Therefore, MFPB-HOSTP then identifies some CBLPs the number of which depends on the number of intermediate nodes of $p^{b(\mu)}_{v_k \to v_t} (\mu \in \{T, r, \rho\})$. When facing with the new imbalance problem of QoT attributes at v_k, these CBLPs are used to be concatenated with the FLP to balance QoT attributes in the newly formed foreseen paths, which could increase the probability of delivering a solution with high utility that is near optimal (see a detailed analysis in *Step 2* in the following section of *Algorithm Description*).

The backward search procedure could illustrate whether there exists a feasible solution in a subnetwork (it is proved in *Theorem 1* in the following section of *Algorithm Description*). If there exists at least one feasible solution, MFPB-HOSTP performs a forward search procedure from v_s to v_t. This procedure intends to identify the path with the maximal utility by using Dijkstra's shortest path algorithm [11]. When facing with the imbalance problem of QoT attributes at v_k, MFPB-HOSTP concatenates the FLP (i.e., $p^{f(u)}_{v_s \to v_k}$) with BLPs and CBLPs, forming multiple foreseen paths, instead of one foreseen path only in H_OSTP. This strategy could effectively help address the imbalance problem of QoT attributes in path selection, and thus helping avoid a failed feasibility estimation of a foreseen path in the social path selection.

6.2 Algorithm Description

In this section, we give a more detailed description of our proposed MFPB-HOSTP algorithm.

Backward_Search. In the *Backward_Search* procedure, MFPB-HOSTP searches the subnetwork from v_t to v_s to investigate whether there exists a feasible solution in the subnetwork. In this process, at each intermediate node v_k, several BLPs and CBLPs from v_t to v_k are identified. The identification of these paths can be divided into the following four steps.

Algorithm 2: Backward_Search ()

Data: $MT(v_s, v_t)$, $Q^T_{v_s,v_t}$, $Q^r_{v_s,v_t}$, $Q^\rho_{v_s,v_t}$

Result: $\delta(p^{backward}_{v_s \to v_t})$, $AQ^\mu(p^{b(\delta)}_{v_k \to v_t})$, $AQ^\mu(p^{b(\mu)}_{v_k \to v_t})$, $AQ^\mu(p^{CBLP(\mu)}_{v_k \to v_t})$, $(\mu \in \{T, r, \rho\})$

begin
- **Set** $v_x.d = \infty\ (v_x \neq v_t)$, $v_t.d = 0$, $S_x = \emptyset$, $p^{b(\delta)}_{v_t \to v_t} = v_t$
- **Add** v_t into S_x
- **while** $S_x \neq \emptyset$ **do**
  - $v_a.d = min(v^*_a.d)\ (v^*_a \in S_x)$
  - **for** *each* $v_b \in adj[v_a]$ **do**
    - **if** $v_b \notin S_x$ **then**
      - Put v_b into S_x
      - $p^{b(\delta)}_{v_b \to v_t} = v_b \to v_a + p^{b(\delta)}_{v_a \to v_t}$
    - **else if** $\delta(v_b \to v_a + p^{b(\delta)}_{v_a \to v_t}) < v_b.d$ **then**
      - Update $v_b.d$ and $AQ^\mu(p^{b(\delta)}_{v_b \to v_t})$, $(\mu \in \{T, r, \rho\})$
      - $p^{b(\delta)}_{v_b \to v_t} = v_b \to v_a + p^{b(\delta)}_{v_a \to v_t}$
  - **Remove** v_a from S_x
- $p^{backward}_{v_s \to v_t} = p^{b(\delta)}_{v_s \to v_t}$
- **if** $\delta(p^{backward}_{v_s \to v_t}) \leq 1$ **then**
  - Computing Max_T($MT(v_s, v_t)$, $Q^T_{v_s,v_t}$, $Q^r_{v_s,v_t}$, $Q^\rho_{v_s,v_t}$)
  - Computing Max_r($MT(v_s, v_t)$, $Q^T_{v_s,v_t}$, $Q^r_{v_s,v_t}$, $Q^\rho_{v_s,v_t}$)
  - Computing Max_ρ($MT(v_s, v_t)$, $Q^T_{v_s,v_t}$, $Q^r_{v_s,v_t}$, $Q^\rho_{v_s,v_t}$)

end

Algorithm 3: Computing Max_T ()

Data: $MT(v_s, v_t)$, $Q^T_{v_s,v_t}$, $Q^r_{v_s,v_t}$, $Q^\rho_{v_s,v_t}$

Result: $AQ^\mu(p^{b(T)}_{v_k \to v_t})$ and $AQ^\mu(p^{CBLP(T)}_{v_k \to v_t})$, $(\mu \in \{T, r, \rho\})$

begin
- **Set** $v_x.d = \infty\ (v_x \neq v_t)$, $v_t.d = 0$, $S_x = \emptyset$, $p^{b(T)}_{v_t \to v_t} = v_t$, $p^{CBLP(T)}_{v_t \to v_t} = v_t$
- **Add** v_t into S_x
- **while** $S_x \neq \emptyset$ **do**
  - $v_a.d = min(v^*_a.d)\ (v^*_a \in S_x)$
  - **for** *each* $v_b \in adj[v_a]$ **do**
    - $obj = 1/AQ^T(p^{b(\delta^T)}_{v_a \to v_t} + v_a \to v_b)$
    - **if** $v_b \notin S_x$ **then**
      - Put v_b into S_x
      - $p^{b(T)}_{v_b \to v_t} = v_b \to v_a + p^{b(T)}_{v_a \to v_t}$
    - **else if** $obj < v_b.d$ **then**
      - Update $AQ^T(p^{b(T)}_{v_b \to v_t})$
      - $v_b.d = obj$
      - $p^{b(T)}_{v_b \to v_t} = v_b \to v_a + p^{b(T)}_{v_a \to v_t}$
    - **for** $i = 1\ to\ M$ **do**
      - $p^{CBLP^i(T)}_{v_b \to v_t} = p^{CBLP^i(T)}_{v_a \to v_t}$
      - $AQ^\mu(p^{CBLP^i(T)}_{v_b \to v_t}) = AQ^\mu(p^{CBLP^i(T)}_{v_a \to v_t})$
    - $p^{CBLP^{M+1}(T)}_{v_b \to v_t} = v_b \to v_a + p^{b(\delta)}_{v_a \to v_t}$
  - **Remove** v_a from S_x

end

Algorithm 4: Computing Max_r ()

Data: $MT(v_s, v_t)$, $Q^T_{v_s,v_t}$, $Q^r_{v_s,v_t}$, $Q^\rho_{v_s,v_t}$

Result: $AQ^\mu(p^{b(r)}_{v_k \to v_t})$ and $AQ^\mu(p^{CBLP(r)}_{v_k \to v_t})$, $(\mu \in \{T, r, \rho\})$

begin
- **Set** $v_x.d = \infty\ (v_x \neq v_t)$, $v_t.d = 0$, $S_x = \emptyset$, $p^{b(r)}_{v_t \to v_t} = v_t$, $p^{CBLP(r)}_{v_t \to v_t} = v_t$
- **Add** v_t into S_x
- **while** $S_x \neq \emptyset$ **do**
  - $v_a.d = min(v^*_a.d)\ (v^*_a \in S_x)$
  - **for** *each* $v_b \in adj[v_a]$ **do**
    - $obj = 1/AQ^r(p^{b(\delta^r)}_{v_a \to v_t} + v_a \to v_b)$
    - **if** $v_b \notin S_x$ **then**
      - Put v_b into S_x
      - $p^{b(r)}_{v_b \to v_t} = v_b \to v_a + p^{b(r)}_{v_a \to v_t}$
    - **else if** $obj < v_b.d$ **then**
      - Update $AQ^r(p^{b(r)}_{v_b \to v_t})$
      - $v_b.d = obj$
      - $p^{b(r)}_{v_b \to v_t} = v_b \to v_a + p^{b(r)}_{v_a \to v_t}$
    - **for** $i = 1\ to\ M$ **do**
      - $p^{CBLP^i(r)}_{v_b \to v_t} = p^{CBLP^i(r)}_{v_a \to v_t}$
      - $AQ^\mu(p^{CBLP^i(r)}_{v_b \to v_t}) = AQ^\mu(p^{CBLP^i(r)}_{v_a \to v_t})$
    - $p^{CBLP^{M+1}(r)}_{v_b \to v_t} = v_b \to v_a + p^{b(\delta)}_{v_a \to v_t}$
  - **Remove** v_a from S_x

end

Algorithm 5: Computing Max_ρ ()

Data: $MT(v_s, v_t)$, $Q^T_{v_s,v_t}$, $Q^r_{v_s,v_t}$, $Q^\rho_{v_s,v_t}$

Result: $AQ^\mu(p^{b(\rho)}_{v_k \to v_t})$ and $AQ^\mu(p^{CBLP(\rho)}_{v_k \to v_t})$, $(\mu \in \{T, r, \rho\})$

begin
- **Set** $v_x.d = \infty\ (v_x \neq v_t)$, $v_t.d = 0$, $S_x = \emptyset$, $p^{b(\rho)}_{v_t \to v_t} = v_t$, $p^{CBLP(\rho)}_{v_t \to v_t} = v_t$
- **Add** v_t into S_x
- **while** $S_x \neq \emptyset$ **do**
  - $v_a.d = min(v^*_a.d)\ (v^*_a \in S_x)$
  - **for** *each* $v_b \in adj[v_a]$ **do**
    - $obj = 1/AQ^\rho(p^{b(\delta^r)}_{v_a \to v_t} + v_a \to v_b)$
    - **if** $v_b \notin S_x$ **then**
      - Put v_b into S_x
      - $p^{b(\rho)}_{v_b \to v_t} = v_b \to v_a + p^{b(\rho)}_{v_a \to v_t}$
    - **else if** $obj < v_b.d$ **then**
      - Update $AQ^r(p^{b(\rho)}_{v_b \to v_t})$
      - $v_b.d = obj$
      - $p^{b(\rho)}_{v_b \to v_t} = v_b \to v_a + p^{b(\rho)}_{v_a \to v_t}$
    - **for** $i = 1\ to\ M$ **do**
      - $p^{CBLP^i(\rho)}_{v_b \to v_t} = p^{CBLP^i(\rho)}_{v_a \to v_t}$
      - $AQ^\mu(p^{CBLP^i(\rho)}_{v_b \to v_t}) = AQ^\mu(p^{CBLP^i(\rho)}_{v_a \to v_t})$
    - $p^{CBLP^{M+1}(\rho)}_{v_b \to v_t} = v_b \to v_a + p^{b(\delta)}_{v_a \to v_t}$
  - **Remove** v_a from S_x

end

Algorithm 6: Path_Selection ()

Data: $MT(v_s, v_t)$, S_y, v_a, v_b
Result: $p^{f(u)}_{v_s \to v_b}$, $AQ^{\mu}(p^{f(u)}_{v_s \to v_b})$, $\mu \in \{T, r, \rho\}$
begin
  if $v_b \notin S_y$ **then**
    Put v_b into S_y and $p^{f(u)}_{v_s \to v_b} = p^{f(u)}_{v_s \to v_a} + v_a \to v_b$
  else if $1/\mathcal{F}(p^{f(u)}_{v_s \to v_a} + v_a \to v_b) < v_b.d$ **then**
    Update $AQ^{\mu}(p^{f(u)}_{v_s \to v_b})$
    $p^{f(u)}_{v_s \to v_b} = p^{f(u)}_{v_s \to v_a} + v_a \to v_b$
end

Step 1 (Identify the BLP with the minimal δ). In social trust path selection, if a path satisfies multiple QoT constraints, the aggregated value of each QoT attribute (i.e., T, r, or ρ) of that path should be larger than the corresponding QoT constraint. From (6), we can see that if any aggregated QoT attribute value of a social trust path does not satisfy the corresponding QoT constraint, then $\delta(p) > 1$. Otherwise, $\delta(p) \leq 1$.

To investigate whether there exists a feasible solution in a subnetwork, in this step, MFPB-HOSTP identifies the path from v_t to v_s with the minimal δ (i.e., $p^{b(\delta)}_{v_s \to v_t}$) based on Dijkstra's shortest path algorithm [11]. In the searching process, at each intermediate node v_k, BLP $p^{b(\delta)}_{v_k \to v_t}$ is identified and the aggregated QoT attribute values of these paths (i.e., $T_{p^{b(\delta)}_{v_k \to v_t}}$ $r_{p^{b(\delta)}_{v_k \to v_t}}$ and $\rho_{p^{b(\delta)}_{v_k \to v_t}}$) are computed and recorded. According to the following *Theorem 1*, the *Backward_Search* procedure can investigate whether there exists a feasible solution in the subnetwork.

Theorem 1. *In the* Backward_Search *procedure, the process of identifying the path with the minimal δ can guarantee finding a feasible solution if at least one feasible solution exists in a subnetwork.*

Proof. Let $p^{backward}_{v_s \to v_t} = p^{b(\delta)}_{v_s \to v_t}$ be a path from v_t to v_s with the minimal δ, and p_* be a feasible solution. Then, $\delta(p^{backward}_{v_s \to v_t}) \leq \delta(p_*)$. Assume $p^{backward}_{v_s \to v_t}$ is not a feasible solution, then $\exists \varphi \in \{T, r, \rho\}$ that $\varphi_{p^{backward}_{v_s \to v_t}} < Q^{\varphi}_{v_s, v_t}$. Hence, $\delta(p^{backward}_{v_s \to v_t}) > 1$. Since p_* is a feasible solution, then $\delta(p_*) \leq 1$ and $\delta(p^{backward}_{v_s \to v_t}) > \delta(p_*)$. This contradicts $\delta(p^{backward}_{v_s \to v_t}) \leq \delta(p_*)$. Therefore, $p^{backward}_{v_s \to v_t}$ is a feasible solution. □

The *Backward_Search* procedure can always identify the path with the minimal δ. If $\delta_{min} > 1$, it indicates there is no feasible solution in the subnetwork, then the algorithm terminates. If $\delta_{min} \leq 1$, it indicates there exists at least one feasible solution and the identified path is a feasible solution. In such a case, the algorithm will perform the following steps to deliver a near-optimal solution.

Step 2 (Identify the BLP with the maximal aggregated T value and the corresponding CBLPs). In this step, at each intermediate node v_k, MFPB-HOSTP first identifies the BLP with the maximal aggregated T value (i.e., $p^{b(T)}_{v_k \to v_t}$), and then identifies several corresponding CBLPs which are composed of part of $p^{b(T)}_{v_k \to v_t}$ and a BLP with the minimal δ from v_t to each intermediate node in $p^{b(T)}_{v_k \to v_t}$.

1. **Identify the BLPs with the maximal T.** MFPB-HOSTP first identifies the path from v_t to v_s with the maximal aggregated T value (i.e., $p^{b(T)}_{v_s \to v_t}$) based on Dijkstra's shortest path algorithm [11]. In the searching process, at each intermediate node v_k, BLP $p^{b(T)}_{v_k \to v_t}$ (e.g., BLP $v_4 \to v_t$ in Fig. 3) and the aggregated QoT attributes' values of $p^{b(T)}_{v_k \to v_t}$ are computed and recorded. When facing with the imbalance problem of T at v_k, BLP $p^{b(T)}_{v_k \to v_t}$ is concatenated with the FLP $p^{f(u)}_{v_k \to v_t}$, forming a new foreseen path $fp^{f(u)+b(T)}_{v_s \to v_k \to v_t}$ (e.g., the foreseen path $v_1 \to v_2 \to v_4 \to v_t$ in Fig. 3). This foreseen path could be used as a reference to estimate whether there exists a feasible solution identified by following $p^{f(u)}_{v_s \to v_k}$. This strategy could help avoid a failed feasibility estimation of a foreseen path caused by the imbalance problem of T at v_k.
2. **Identify the CBLPs based on the BLPs with the maximal T.** Greedily maximizing the aggregated T value without considering other QoT attributes values in $p^{b(T)}_{v_k \to v_t}$ may lead to the new imbalance problem of QoT attributes (i.e., r and ρ). Therefore, in addition to $p^{b(T)}_{v_k \to v_t}$, suppose there are M intermediate nodes (denoted as $v_l, l \in [1, M]$) in path $p^{b(T)}_{v_k \to v_t}$, MFPB-HOSTP then identifies M *Composite Backward Local Paths* at v_k (denoted as $p^{CBLP^M(T)}_{v_k \to v_t}$) which are composed of $p^{b(T)}_{v_k \to v_l} l \in [1, M]$ and $p^{b(\delta)}_{v_l \to v_t}, l \in [1, M]$. For example, as shown in Fig. 4, since there is no intermediate node between v_4 and v_t in BLP $p^{b(T)}_{v_4 \to v_t}$ (i.e., $M = 0$), MFPB-HOSTP only identifies one BLP $p^{b(T)}_{v_4 \to v_t} = v_4 \to v_t$. Since there exists an intermediate node v_4 between v_2 and v_t in BLP $p^{b(T)}_{v_2 \to v_t}$ (i.e., $M = 1$), in addition to $p^{b(T)}_{v_2 \to v_t}$, MFPB-HOSTP identifies one CBLP $p^{CBLP^1(T)}_{v_2 \to v_t} = (v_2 \to v_4) + p^{b(\delta)}_{v_4 \to v_t}$. Similarly, at v_1 there exist two intermediate nodes between v_1 and v_t in BLP $p^{b(T)}_{v_1 \to v_t}$ (i.e., $M = 2$), MFPB-HOSTP identifies two CBLPs. They are CBLP $p^{CBLP^1(T)}_{v_1 \to v_t} = (v_1 \to v_2 \to v_4) + p^{b(\delta)}_{v_4 \to v_t}$ and CBLP $p^{CBLP^2(T)}_{v_1 \to v_t} = (v_1 \to v_2) + p^{b(\delta)}_{v_2 \to v_t}$. When facing with the new imbalance caused by the BLP with the maximal T, the M CBLPs at v_k are concatenated with the FLP $p^{f(u)}_{v_s \to v_k}$. This strategy could help avoid a failed feasibility estimation of a foreseen path caused by the new imbalance problem of other two QoT attributes (i.e., r and ρ) at v_k. Next, we use an example to illustrate the effectiveness of CBLPs in solving the new imbalance problem of QoT attributes.

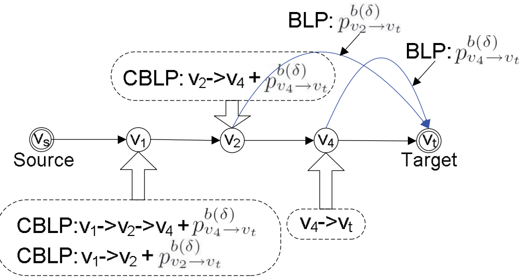

Fig. 4. Multiple CBLPs in backward search procedure.

Fig. 5 depicts a subnetwork between v_s and v_t. Table 2 lists the FLP at v_2, the BLP at v_2, the corresponding CBLP at v_2, and the aggregated values of QoT attributes of these paths. Suppose that the QoT constraints specified by source

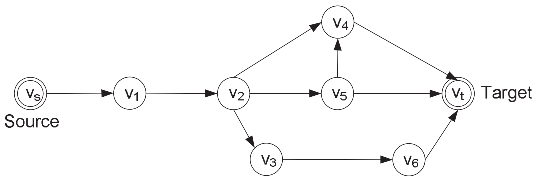

Fig. 5. The CBLP in path selection.

participant v_s are $Q^T_{v_s,v_t} = 0.12$, $Q^r_{v_s,v_t} = 0.15$ and $Q^\rho_{v_s,v_t} = 0.3$. We could see that the foreseen path $fp^{f(u)+b(\delta)}_{v_s\to v_2\to v_t}$ is infeasible due to the imbalance problem of T at v_2 ($T = 0.075 < Q^T_{v_s,v_t} = 0.12$). Then, MFPB-HOSTP concatenates the FLP with BLP $p^{b(T)}_{v_2\to v_t}$ to form another foreseen path $fp^{f(u)+b(T)}_{v_s\to v_2\to v_t}$.

However, we could see there arises a new imbalance problem of r, where the aggregated r value of $fp^{f(u)+b(T)}_{v_s\to v_2\to v_t}$ does not satisfy the corresponding QoT constraint ($r = 0.08 < Q^r_{v_s,v_t} = 0.15$) and thus the foreseen path is infeasible. In such a situation, suppose $p^{b(\delta)}_{v_5\to v_t} = v_5 \to v_t$, at v_2, MFPB-HOSTP identifies the CBLP $p^{CBLP^1(T)}_{v_2\to v_t} = v_2 \to v_5 \to v_t$ and concatenates it with the FLP to balance the aggregated r value. In such a situation, the foreseen path $fp^{f(u)+CBLP^1(T)}_{v_s\to v_2\to v_t}$ is feasible. Assume the QoT attributes have the same weight in the utility function, with the assistance of CBLP $p^{CBLP^1(T)}_{v_2\to v_t}$, MFPB-HOSTP could select the path $v_s \to v_1 \to v_2 \to v_5 \to v_t$ with the utility of 0.117 as the solution. Otherwise, the path $v_s \to v_1 \to v_3 \to v_6 \to v_t$ with the utility of 0.107 will be selected, which is worse than the one (i.e., utility is 0.117) identified with the assistance of CBLPs.

From this example, we could see that when facing with the new imbalance problem of QoT attributes caused by greedily maximizing the aggregated QoT attributes values in BLPs, CBLPs could help avoid a failed feasibility estimation caused by a new imbalance problem of QoT attributes. Thus, with the assistance of CBLPs, MFPB-HOSTP could deliver a better solution in some cases. In the process of identifying these BLPs and CBLPs, if there exist two overlapping paths (i.e., they have the same aggregated QoT attributes values), MFPB-HOSTP keeps only one of them for further search, saving execution time.

Step 3 (Identify the BLP with the maximal aggregated r value and the corresponding CBLPs).

1. **Identify the BLPs with the maximal r.** Similar to *Step 2*, in order to avoid the imbalance problem of r, in this step, at each intermediate node v_k, MFPB-HOSTP first identifies the BLP with the maximal aggregated r value (denoted as $p^{b(r)}_{v_k\to v_t}$) based on Dijkstra's shortest path algorithm [11]. In this search process, at v_k, the aggregated values of QoT attributes of $p^{b(r)}_{v_k\to v_t}$ are computed and recorded. When facing with the imbalance problem of r at v_k, BLP $p^{b(r)}_{v_k\to v_t}$ is concatenated with the FLP $p^{f(u)}_{v_s\to v_k}$, forming a new foreseen path $fp^{f(u)+b(r)}_{v_s\to v_k\to v_t}$. This foreseen path is used as a reference to estimate whether there exists a feasible solution identified by following $p^{f(u)}_{v_s\to v_k}$. This strategy could avoid a failed feasibility estimation of a foreseen path caused by the imbalance problem of r at v_k.

TABLE 2
BLPs, CBLPs, and the Aggregated QoT Attributes Values

Path	Nodes and Links	T	r	ρ
$p^{f(u)}_{v_s\to v_2}$	$v_s \to v_1 \to v_2$	0.3	0.8	0.5
$p^{b(\delta)}_{v_2\to v_t}$	$v_2 \to v_4 \to v_t$	0.25	0.5	0.4
$p^{b(T)}_{v_2\to v_t}$	$v_2 \to v_5 \to v_4 \to v_t$	0.7	0.1	0.3
$p^{CBLP^1(T)}_{v_2\to v_t}$	$v_2 \to v_5 \to v_t$	0.5	0.2	0.3
path $v_3 \to v_t$	$v_3 \to v_6 \to v_t$	0.4	0.2	0.3

2. **Identify the CBLPs based on the BLPs with the maximal r.** To avoid the new imbalance problem of QoT attributes caused by greedily maximizing r value, MFPB-HOSTP then identifies M CBLPs at each intermediate node v_k, which are composed of $p^{b(r)}_{v_k\to v_l}, l \in [1, M]$ and $p^{b(\delta)}_{v_l\to v_t}, l \in [1, M]$. When facing with the new imbalance problem of QoT attributes caused by maximizing r value, the identified M CBLPs at v_k are concatenated with the FLP $p^{f(u)}_{v_s\to v_k}$, to estimate whether there exists a feasible solution identified by following the FLP. This could help avoid a failed feasibility estimation of a foreseen path caused by the new imbalance problem of the other two QoT attributes (i.e., T and ρ) at v_k.

Step 4 (Identify the BLP with the maximal aggregated ρ value and the corresponding CBLPs):

1. **Identify the BLPs with the maximal ρ.** To avoid the imbalance problem of ρ, in this step, at each intermediate node v_k, MFPB-HOSTP first identifies the BLP with the maximal aggregated ρ value (denoted as $p^{b(\rho)}_{v_k\to v_t}$) based on Dijkstra's shortest path algorithm [11]. In this search process, at each v_k, the aggregated QoT attributes values of $p^{b(\rho)}_{v_k\to v_t}$ are computed and recorded. When facing with the imbalance problem of ρ at v_k, BLP $p^{b(\rho)}_{v_k\to v_t}$ is concatenated with the FLP $p^{f(u)}_{v_k\to v_t}$, forming a new foreseen path $fp^{f(u)+b(\rho)}_{v_s\to v_k\to v_t}$. This strategy could help avoid a failed feasibility estimation of a foreseen path caused by the imbalance problem of ρ at v_k.
2. **Identify the CBLPs based on the BLPs with the maximal ρ.** To avoid the new imbalance problem of QoT attributes caused by greedily maximizing ρ value, MFPB-HOSTP then identifies M CBLPs at each intermediate node v_k, which are composed of $p^{b(\rho)}_{v_k\to v_l}, l \in [1, M]$ and $p^{b(\delta)}_{v_l\to v_t}, l \in [1, M]$. When facing with the new imbalance problem of QoT attributes caused by the BLP with the maximal ρ at v_k, the M CBLPs at v_k are concatenated with the FLP $p^{f(u)}_{v_s\to v_k}$, to estimate the feasibility of searching by following the FLP. This could avoid a failed feasibility estimation of a foreseen path caused by the new imbalance problem of the other two QoT attributes (i.e., T and r) at v_k.

In summary, the *Backward_Search* procedure can illustrate whether there exists a feasible solution in a subnetwork. In addition, if a feasible solution exists, compared with the *Backward_Search* procedure of H_OSTP, MFPB-HOSTP identifies the BLP with the maximal aggregated value of each of the QoT attributes. Furthermore, to solve a new imbalance problem of QoT attributes caused by greedily maximizing the aggregated values of QoT attributes, MFPB-HOSTP also identifies several CBLPs, which are composed of part of the BLP with the minimal δ and part of the BLP with the maximal aggregated value of each of the QoT attributes. When facing with an imbalance problem of QoT attributes, the identified BLPs and CBLPs will be used in the following *Forward_Search* procedure aiming to avoid a failed feasibility estimation of a foreseen path in H_OSTP and deliver a near-optimal solution. Next, we discuss the search strategies adopted in the following *Forward_Search* procedure of MFPB-HOSTP.

Forward_Search. In the forward search from v_s to v_t, MFPB-HOSTP uses the BLPs and CBLPs identified by the above *Backward_Search* procedure to investigate whether there exists another path $p^{forward}_{v_s \to v_t}$, which is better in quality than the above path $p^{backward}_{v_s \to v_t} = p^{b(\delta)}_{v_s \to v_t}$ returned in the *Backward_Search* procedure (i.e., whether $\mathcal{F}(p^{forward}_{v_s \to v_t}) > \mathcal{F}(p^{backward}_{v_s \to v_t})$).

Algorithm 7: Forward_Search ()

Data: $MT(v_s, v_t)$, $AQ^{\mu}(p^{b(\delta)}_{v_k \to v_t})$, $AQ^{\mu}(p^{b(\mu)}_{v_k \to v_t})$, $AQ^{\mu}(p^{CBLP(\mu)}_{v_k \to v_t})$, $\mu \in \{T, r, \rho\}$, $Q^T_{v_s, v_t}$, $Q^r_{v_s, v_t}$, $Q^{\rho}_{v_s, v_t}$
Result: $p^{forward}_{v_s \to v_t}$, $\mathcal{F}(p^{forward}_{v_s \to v_t})$

begin
  Set $v_y.d = \infty$ $(v_y \neq v_s)$, $v_s.d = 0$, $S^1_y = S^2_y = \emptyset$, $p^{f(u)}_{v_s \to v_s} = v_s$
  Add v_s into S^1_y and S^2_y
  while $S^1_y \neq \emptyset$ *and* $S^2_y \neq \emptyset$ **do**
    $v^1_a.d = min(v^*_a.d)$ $(v^*_a \in S^1_y)$
    $v^2_a.d = min(v^{2*}_a.d)$ $(v^{2*}_a \in S^2_y)$
    if $v^1_a = v^2_a$ *and* $v^1_a.d^1 = v^2_a.d^2$ **then**
      for *each* $v_b \in adj[v^1_a]$ **do**
        if $fp^{f(u)+b(\delta)}_{v_s \to v_b \to v_t}$ *is feasible* **then**
          Path_Selection($MT(v_s, v_t)$, S^1_y, v^1_a, v_b)
        else if $fp^{f(u)+b(\delta)}_{v_s \to v_b \to v_t}$ *is infeasible* **then**
          if *one of* $\{fp^{f(u)+b(\mu)}_{v_s \to v_j \to v_t}\}$ *and* $\{fp^{f(u)+CBLP^M(\mu)}_{v_s \to v_j \to v_t}\}$ *is feasible* **then**
            Path_Selection($MT(v_s, v_t)$, S^2_y, v^1_a, v_b)
    else
      for *each* $v_b \in adj[v^1_a]$ **do**
        if $fp^{f(u)+b(\delta)}_{v_s \to v_b \to v_t}$ *is feasible* **then**
          Path_Selection($MT(v_s, v_t)$, S^1_y, v^1_a, v_b)
      for *each* $v_b \in adj[v^2_a]$ **do**
        if *one of* $\{fp^{f(u)+b(\mu)}_{v_s \to v_j \to v_t}, fp^{f(u)+CBLP^M(\mu)}_{v_s \to v_j \to v_t}\}$ *is feasible* **then**
          Path_Selection($MT(v_s, v_t)$, S^2_y, v^2_a, v_b)
    Remove v^1_a from S^1_y and v^2_a from S^2_y
  Return $p^{forward}_{v_s \to v_t}$ =max_utility($p^{f(u)}_{v_s \to v^1_a \to v_t}$, $p^{f(u)}_{v_s \to v^2_a \to v_t}$) and $\mathcal{F}(p^{forward}_{v_s \to v_t})$
end

In this procedure, MFPB-HOSTP searches the path with the maximal $\mathcal{F}$ value from v_s to v_t. Assume node $v_m \in \{neighboring\ nodes\ of\ v_s\}$ is selected based on Dijkstra's shortest path algorithm (i.e., FLP $p^{f(u)}_{v_s \to v_m}$ is identified). Then, MFPB-HOSTP concatenates the FLP with BLP $p^{b(\delta)}_{v_m \to v_t}$ to form a foreseen path $fp^{f(u)+b(\delta)}_{v_s \to v_m \to v_t}$. If the foreseen path is feasible, MFPB-HOSTP then chooses the next node from v_m with the maximal $\mathcal{F}$ value. Otherwise, MFPB-HOSTP concatenates the FLP with the BLPs with the minimal T, r, and ρ, respectively, to form three foreseen paths $\{fp^{f(u)+BLP(\mu)}_{v_s \to v_m \to v_t}(\mu \in \{T, r, \rho\})\}$. According to the feasibility of these foreseen paths, MFPB-HOSTP adopts the following search strategies.

Situation 1. If one of $\{fp^{f(u)+b(\mu)}_{v_s \to v_m \to v_t}(\mu \in \{T, r, \rho\})\}$ is feasible, MFPB-HOSTP adopts the following two strategies to identify two social trust paths and selects the feasible social trust path with the higher utility value as the final solution:

1. **Strategy 1.** MFPB-HOSTP identifies one path by choosing the next node from v_m with the maximal $\mathcal{F}$ value.
2. **Strategy 2.** MFPB-HOSTP identifies another path by searching another neighboring node of v_s with the maximal $\mathcal{F}$, which is the same as the search strategy adopted in H_OSTP [29].

Situation 2. If all $\{fp^{f(u)+b(\mu)}_{v_s \to v_m \to v_t} \mu \in \{T, r, \rho\}\}$ are infeasible, then at v_m, MFPB-HOSTP concatenates the FLP with the CBLPs to form the foreseen paths (i.e., $\{fp^{f(u)+CBLP^M(\mu)}_{v_s \to v_m \to v_t}(\mu \in \{T, r, \rho\})\}$). According to the feasibility of these foreseen paths, MFPB-HOSTP adopts the following search strategies:

1. **Subsituation 2.1.** If one of $\{fp^{f(u)+CBLP^M(\mu)}_{v_s \to v_m \to v_t}(\mu \in \{T, r, \rho\})\}$ is feasible, MFPB-HOSTP identifies two social trust paths based on *Strategies 1 and 2* in the above *Situation 1*, and selects the feasible social trust path with the higher utility as the final solution.
2. **Subsituation 2.2.** If all of $\{fp^{f(u)+CBLP^M(\mu)}_{v_s \to v_m \to v_t}(\mu \in \{T, r, \rho\})\}$ are infeasible, MFPB-HOSTP does not search the path from v_m. Instead, MFPB-HOSTP performs the *Forward_Search* procedure to search the path from v_s in the subnetwork without taking link $v_s \to v_m$ into consideration.

The following *Theorem 2* illustrates that the social trust path $p^{forward}_{v_s \to v_t}$ identified by the *Forward_Search* procedure cannot be worse than the feasible social trust path $p^{backward}_{v_s \to v_t}$ identified by the *Backward_Search* procedure. Namely, $\mathcal{F}(p^{forward}_{v_s \to v_t}) \geq \mathcal{F}(p^{backward}_{v_s \to v_t})$.

Theorem 2. *With the social trust path $p^{backward}_{v_s \to v_t}$ identified by the* Backward_Search *procedure and the social trust path $p^{forward}_{v_s \to v_t}$ identified by the* Forward_Search *procedure in MFPB-HOSTP, if $p^{backward}_{v_s \to v_t}$ is a feasible solution, then $p^{forward}_{v_s \to v_t}$ is feasible and $\mathcal{F}(p^{forward}_{v_s \to v_t}) \geq \mathcal{F}(p^{backward}_{v_s \to v_t})$.*

Proof. Assume that path $p^{backward}_{v_s \to v_t}$ consists of $n+2$ nodes $v_s, v_1, \ldots, v_n, v_t$. In the *Forward_Search* procedure, H_OSTP searches the neighboring nodes of v_s and chooses v_1 from these nodes when a foreseen path from v_s to v_t via v_1 is feasible and the current path from v_s to v_1 has the maximal $\mathcal{F}$. This step is repeated at all the nodes between v_1 and v_n until a social trust path $p^{forward}_{v_s \to v_t}$ is identified. If at each search step, only one node of $\{v_1, \ldots, v_n\}$ has a feasible foreseen path, then $p^{forward}_{v_s \to v_t}$ is the only feasible solution in the subnetwork between v_s and v_t. According to *Theorem 1*, then $p^{forward}_{v_s \to v_t} = p^{backward}_{v_s \to v_t}$. Thus, $\mathcal{F}(p^{forward}_{v_s \to v_t}) = \mathcal{F}(p^{backward}_{v_s \to v_t})$. Otherwise, if $p^{forward}_{v_s \to v_t} \neq p^{backward}_{v_s \to v_t}$, it can lead to $\mathcal{F}(p^{forward}_{v_s \to v_t}) > \mathcal{F}(p^{backward}_{v_s \to v_t})$ by maximizing the $\mathcal{F}$ value in all candidate nodes which have feasible foreseen paths based on Dijkstra's shortest path algorithm. Therefore, *Theorem 2* is proved. $\square$

TABLE 3
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	0.01	0.01	0.01
2	0.05	0.05	0.05
3	0.1	0.1	0.1
4	0.15	0.15	0.15
5	0.2	0.2	0.2
6	0.25	0.25	0.25
7	0.3	0.3	0.3
8	0.35	0.35	0.35
9	0.4	0.4	0.4
10	0.2	0.05	0.05
11	0.05	0.2	0.05
12	0.05	0.05	0.2
13	0.25	0.05	0.05
14	0.05	0.25	0.05
15	0.05	0.05	0.25
16	0.3	0.05	0.05
17	0.05	0.3	0.05
18	0.05	0.05	0.3
19	0.35	0.05	0.05
20	0.05	0.35	0.05
21	0.05	0.05	0.35
22	0.4	0.05	0.05
23	0.05	0.4	0.05
24	0.05	0.05	0.4

If there exists only one feasible solution in the subnetwork, it can be identified by both the *Backward_Search* procedure and the *Forward_Seach* procedure, and it is the optimal solution. Otherwise, if there exist more than one feasible solution in the subnetwork, then the solution identified by the *Forward_Seach* procedure is near-optimal or optimal, which is better than the one identified by the *Backward_Search* procedure.

6.3 Summary

Based on the above discussion, during the *Backward_Search* procedure, MFPB-HOSTP could illustrate whether there exists a feasible solution in a subnetwork (it is proved by *Theorem 1*). If a feasible solution exists, MFPB-HOSTP then identifies several BLPs and CBLPs at each intermediate node rather than only one BLP in H_OSTP. During the *Forward_Search* procedure, MFPB-HOSTP delivers a near-optimal solution which is no worse than the one returned by the *Backward_Search* procedure (it is proved by *Theorem 2*). In this search process, the identified BLPs and CBLPs are used to concatenate with the FLP, forming multiple foreseen paths rather than one foreseen path only in H_OSTP. These foreseen paths could help avoid a failed feasibility estimation of a foreseen path caused by the imbalance problem of QoT attributes.

In the *Backward_Search* procedure, in order to identify four BLPs for the minimal δ and the maximal value of each QoT attribute (i.e., T, r, and ρ), MFPB-HOSTP adopts Dijkstra's shortest path algorithm four times with the time complexity of $O(4*(NlogN+E))$ [11] (N is the number of nodes and E is the number of links). In addition, in the worst case, the time complexity of identifying the CBLPs for three QoT attributes by MFPB-HOSTP is $O(3*(KN))$, where K is the maximal path length in a subnetwork. So, the time complexity of the Backward_Search procedure is $O(4*(NlogN+E)+3*KN)$.

In the Forward_Search procedure, in the worst case, MFPB-HOSTP adopts Dijkstra's shortest path algorithm twice with the time complexity of $O(2*(NlogN+E))$ [11]. In addition, in the worst case, the time complexity of evaluating the feasibility of foreseen paths is $O(KE)$. So, the time complexity of MFPB-HOSTP is $O(NlogN+KE)$.

In social networks, following the *small-world*[12] characteristic, it is usually the case that $K \leq 7$ [34]. Therefore, the time complexity of MFPB-HOSTP is $O(NlogN+E)$, which is the same as that of H_OSTP. But our proposed heuristic algorithm has better search strategies than H_OSTP. Thus, MFPB-HOSTP delivers a solution no worse than that of H_OSTP, and as our experiments confirm, MFPB-HOSTP can deliver better solutions than that of H_OSTP in some cases (*see a detailed analysis in Section 7.2*).

7 Experiments

7.1 Experiment Settings

The *Enron* e-mail data set 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 [15], [28], [29], [32], [43]. In addition, as we explained in Section 3, the social intimacy degree between participants and the role impact factor of participants can be calculated through mining the subjects and contents of e-mails in the *Enron* e-mail data set [32]. Therefore, in contrast to other real social network data sets, the *Enron* e-mail data set fits our proposed complex social network structure better. Thus, to validate our proposed algorithm, we select the *Enron* e-mail data set with 87,474 nodes (participants) and 30,0511 links (formed by sending and receiving e-mails) for our experiments.

As we analyzed in Section 5.1, our previously proposed H_OSTP outperforms prior algorithms in both efficiency and the quality of identified social trust path [29]. Therefore, in order to study the performance of our proposed algorithm, we compare MFPB-HOSTP with H_OSTP in both execution time and the utilities of the identified social trust paths (*see Section 7.2*). In our experiments, since the detailed mining method of QoT attribute values (i.e., T, r, and ρ) is out of the scope of this paper, and they could have different values in different applications, the QoT attribute values are randomly generated by using *rand()* in Matlab.

As illustrated in Section 3, trust is domain dependent. Therefore, in our model, source participants may specify different QoT constraints for the social trust path selection in different domains. In order to investigate the performance of MFPB-HOSTP with different QoT constraints values, 24 sets of QoT constraints are specified and listed in Table 3, which cover some possible settings of QoT constraints. In some cases (i.e., constraint IDs 1 to 9), the values of QoT constraints are the same, and in the rest of the cases (i.e., constraint IDs 10 to 24), the constraint of one QoT attribute (i.e., T, r, or ρ) is larger than the values of the other two QoT attributes. In addition, in order to investigate the performance of MFPB-HOSTP in path selection with different weights of QoT attributes in the utility function, three sets of weights are specified and listed in Table 4, where T, r, and ρ are given a lager weight than other two QoT attributes, respectively.

In order to study the performance of our proposed heuristic algorithm in the subnetworks of different scales and structures, we first randomly select 80 pairs of source

12. The average path length between any two nodes is about 6 hops in a social network.

TABLE 4
The Setting of the Weights of QoT Attributes

Weight ID (WID)	w_T	w_r	w_ρ
1	0.5	0.25	0.25
2	0.25	0.5	0.25
3	0.25	0.25	0.5

TABLE 5
The Properties of the Simplest and the Most Complex Subnetworks in Each Group of Hops

Hops	The simplest sub-network			The most complex sub-network		
	ID	Nodes	Links	ID	Nodes	Links
4	1	33	56	20	393	1543
5	1	49	90	20	680	2670
6	1	48	74	20	1300	6396
7	1	40	64	20	964	4955

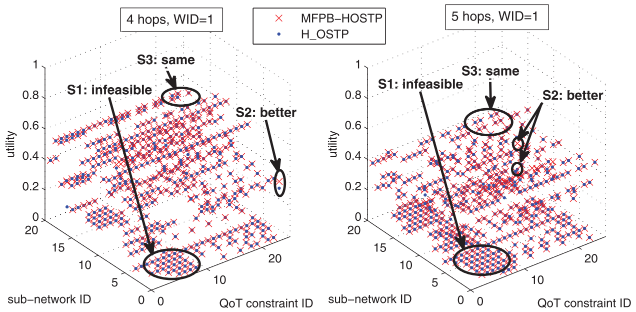

Fig. 6. The path utilities of subnetworks with 4 and 5 hops based on WID = 1.

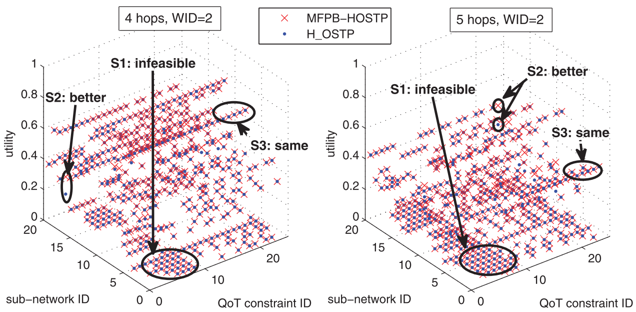

Fig. 7. The path utilities of subnetworks with 4 and 5 hops based on WID = 2.

and target participants from the *Enron* e-mail data set. We then extract the corresponding 80 subnetworks 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 subnetworks are grouped by the number of hops. In each group, they are ordered by the number of nodes in them. Table 5 lists the properties of the simplest and the most complex subnetworks in each group of hops. The simplest subnetwork has 33 nodes and 56 links (4 hops), while the most complex subnetwork has 1,300 nodes and 6,396 links (6 hops). With each subnetwork, we run MFPB-HOSTP and H_OSTP three times independently to calculate the average execution time.

Both MFPB-HOSTP and H_OSTP are implemented using Matlab R2008a running on an IBM ThinkPad SL500 laptop with an Intel Core 2 Duo T5870 2.00 GHz CPU, 3 GB RAM, Windows XP SP3 operating system and MySql 5.1.35 database.

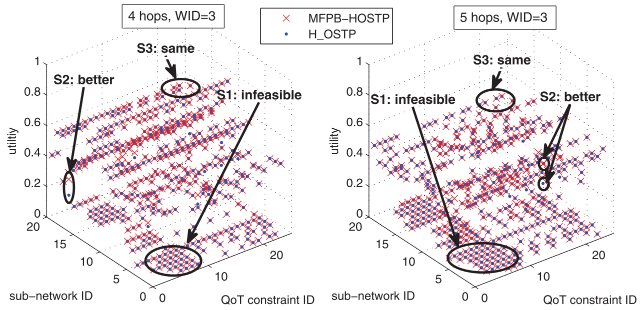

Fig. 8. The path utilities of subnetworks with 4 and 5 hops based on WID = 3.

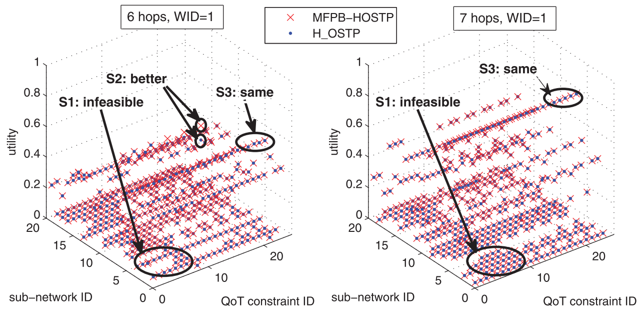

Fig. 9. The path utilities of subnetworks with 6 and 7 hops based on WID = 1.

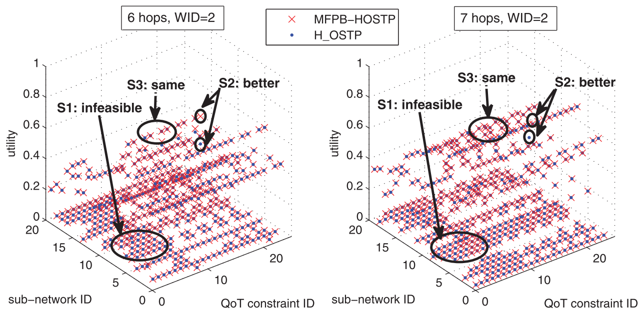

Fig. 10. The path utilities of subnetworks with 6 and 7 hops based on WID = 2.

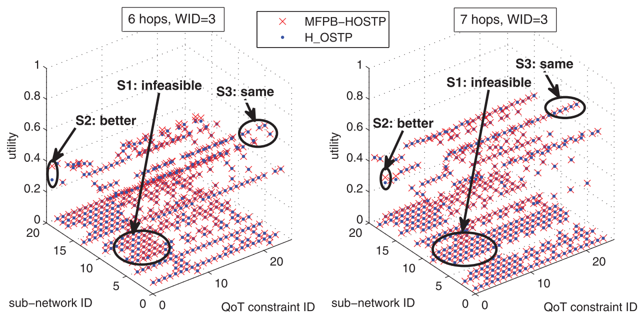

Fig. 11. The path utilities of subnetworks with 6 and 7 hops based on WID = 3.

7.2 Experimental Results

Results and analysis of path utility. Figs. 6, 7, 8, 9, 10, and 11 plot the path utilities of the identified social trust paths in the subnetworks categorized in groups of hops. From these figures, we can observe that if there are no feasible solutions in a subnetwork, both of MFPB-HOSTP and H_OSTP can investigate the infeasibility (e.g., case S1 in Figs. 6, 7, 8, 9, 10, and 11). This is because both of them perform a backward search from v_t to v_s to identify the social trust path with the

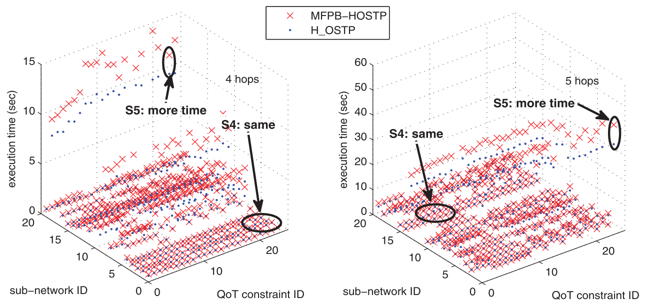

Fig. 12. The execution time of subnetworks with 4 and 5 hops.

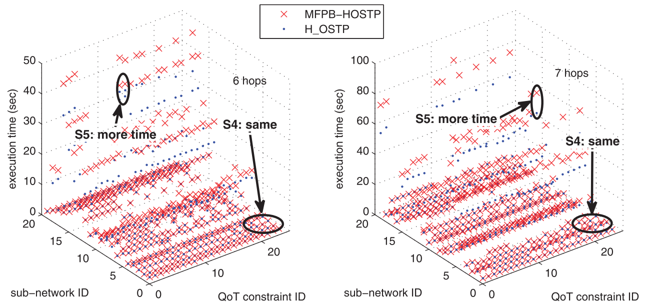

Fig. 13. The execution time of subnetworks with 6 and 7 hops.

minimal δ. It has been proved in *Theorem 1* that this procedure can always investigate whether there exists a feasible solution in a subnetwork.

From Figs. 6, 7, 8, 9, 10, and 11, we can see that in all cases of the 80 subnetworks, our MFPB-HOSTP does not yield any feasible social trust path with a utility worse than that of H_OSTP (e.g., cases S2 and S3 in Figs. 6, 7, 8, 9, 10, and 11). This is because in the *Forward_Search* procedure, if there is no imbalance problem of QoT attributes, MFPB-HOSTP identifies the same social trust path with H_OSTP. When facing with an imbalance problem of QoT attributes, MFPB-HOSTP identifies two social trust paths, out of which one path is identified by using the same search strategy adopted in H_OSTP (see *Strategy 2* of *Situation 1* in *Section 6.2*), and selects the feasible path with the higher utility as the solution. Therefore, MFPB-HOSTP does not yield any solution worse than that of H_OSTP in any cases.

According to our experimental results, in 27 out of 75 subnetworks with feasible solutions (i.e., 36 percent of total subnetworks with feasible solutions), MFPB-HOSTP can deliver better social trust paths than H_OSTP (e.g., case S2 in Figs. 6, 7, 8, 9, 10, and 11). The sums of utilities computed by MFPB-HOSTP and H_OSTP in these subnetworks with each group of hops are listed in Table 7, where we can see that the sum of utilities of our proposed MFPB-HOSTP algorithm is 15.94 percent more than that of H_OSTP in 4 hops subnetworks, 46.51 percent more in 5 hops, 12.63 percent more in 6 hops, and 17.79 percent more in 7 hops. This is because when facing with an imbalance problem of QoT attributes at an intermediate node v_k, in addition to $p^{b(\delta)}_{v_k \to v_t}$, more BLPs are concatenated with the FLP identified by the forward search procedure, forming multiple foreseen paths and helping avoid a failed feasibility estimation. Thus, MFPB-HOSTP can deliver a better solution than H_OSTP in some cases.

TABLE 6
The Comparison of Execution Time

Algorithms	The sum of execution time (sec)				
	4 hops	5 hops	6 hops	7 hops	total
MFPB-HOSTP	7.6478e+003	2.3537e+004	2.5621e+004	4.2355e+004	9.9161e+004
H_OSTP	5.7831e+003	1.8529e+004	1.9903e+004	3.2776e+004	7.6991e+004
ratio	1.3224	1.2703	1.2873	1.2922	1.2880

TABLE 7
The Comparison of Path Utility

Algorithms	The sum of path utility (sec)				
	4 hops	5 hops	6 hops	7 hops	total
MFPB-HOSTP	11.7634	11.2517	6.3161	2.1140	31.4452
H_OSTP	10.1459	7.6797	5.6076	1.7947	25.2279
comparison	15.94%more	46.51%more	12.63%more	17.79%more	24.64%more

Results and analysis of the execution time. Figs. 12 to 13 plot the average execution time of the social trust path selection with three different weights of QoT attributes. From these figures, we can see that in most cases (i.e., $3{,}082/5{,}760 = 53.5\%$ of total cases), MFPB-HOSTP has the same execution time as that of H_OSTP (e.g., case S4 in Figs. 12 to 13). This is because if no feasible solution exists in the subnetwork, based on *Theorem 1*, both of MFPB-HOSTP and H_OSTP can identify this and stop the search process, resulting in the same execution time. In addition, in the rest of the cases, MFPB-HOSTP consumes more execution time than H_OSTP (e.g., case S5 in Figs. 12 to 13). This is because if a feasible solution exists in a subnetwork, at each intermediate node v_k, in addition to $p^{b(\delta)}_{v_s \to v_k}$, MFPB-HOSTP identifies multiple BLPs (i.e., the BLPs with the maximal aggregated value of each of QoT attribute and M CBLPs for each QoT attribute) in the *Backward_Search* procedure, rather than one BLP only in H_OSTP (see Section 6.2). Moreover, when facing with the imbalance problem of QoT attributes at v_k, MFPB-HOSTP needs to identify two social trust paths. The total execution time of each of MFPB-HOSTP and H_OSTP in subnetworks with each group of hops is listed in Table 6, where we conclude that the ratio of the execution time between MFPB-HOSTP and H_OSTP is similar in subnetworks with each group of hops. On average, the execution time of MFPB-HOSTP is 1.288 times of that of H_OSTP.

Through the above experiments conducted on subnetworks with different scales and structures, we can see that on average MFPB-HOSTP consumes 1.288 times of the execution time of H_OSTP while delivering better solutions in subnetworks. Since MFPB-HOSTP has the same polynomial time complexity (i.e., $O(NlogN + E)$) as H_OSTP, MFPB-HOSTP is superior to H_OSTP when applied to large-scale social networks.

8 Conclusions

In this paper, we have presented a complex social network structure that takes trust information, social relationships and recommendation roles into account, reflecting the real-world situations better. For selecting the optimal social trust path with end-to-end QoT constraints in complex social networks, which is an NP-Complete problem, we first analyzed the advantages and the disadvantage (i.e., the imbalance problem of QoT attributes) of our previously proposed H_OSTP that is one of the most promising algorithms for the MCOP selection problem. Based on H_OSTP, we then proposed MFPB-HOSTP, an efficient heuristic algorithm, where multiple foreseen paths are formed, helping avoid a failed feasibility estimation of a foreseen path caused by the imbalance problem of QoT attributes. The results of experiments conducted on a real data set demonstrate that MFPB-HOSTP outperforms existing methods in optimal social trust path selection with good efficiency.

For our future work, we plan to develop a social network based trust-oriented service and service provider search engine, which maintains a database of participants and the complex social network among them. In this system, our proposed method will be applied, for instance, to help a buyer identify the most trustworthy one 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*, vol. 12, no. 12, pp. 215-234, 2001.

[2] B. Alunkal, I. Valjkovic, and G. Laszewski, "Reputation-Based Grid Resource Selection," *Proc. Workshop Adaptive Grid Middleware*, Sept. 2003.

[3] R. Ashri, S. Ramchurn, J. Sabater, M. Luck, and N. Jennings, "Trust Evaluation Through Relationship Analysis," *Proc. Fourth Int'l Joint Conf. Autonomous Agents and Multiagent Systems (AAMAS '05)*, pp. 1005-1011, 2005.

[4] P. Bedi, H. Kaur, and S. Marwaha, "Trust Based Recommender System for Semantic Web," *Proc. Int'l Joint Conf. Artificial Intelligence (IJCAI)*, pp. 2677-2682, 2007.

[5] E. Berscheid and H.T. Reis, "Attraction and Close Relationships," *The Handbook of Social Psychology*, fourth ed., vol. 2, Oxford Univ., 1998.

[6] J. Bi, J. Wu, and W. Zhang, "A Trust and Reputation Based Anti-SPIM Method," *Proc. IEEE INFOCOM*, pp. 2485-2493, Apr. 2008.

[7] R.S. Burt, "Decay Functions," *Social Networks*, vol. 22, no. 4, pp. 1-28, 2000.

[8] Y. Cho, G. Steeg, and A. Galstyan, "Co-Evolution of Selection and Influence in Social Networks," *Proc. Conf. Artificial Intelligence (AAAI)*, pp. 779-784, 2011.

[9] B. Christianson and W.S. Harbison, "Why Isn't Trust Transitive?" *Proc. Int'l Workshop Security Protocols*, pp. 171-176, 1996.

[10] P. Cui and F. Wang, "Item-Level Social Influence Prediction with Probabilistic Hybrid Factor Matrix Factorization," *Proc. Conf. Artificial Intelligence (AAAI)*, pp. 331-336, 2011.

[11] E. Dijkstra, "A Note on Two Problems in Connexion with Graphs," *Numerische Mathematik*, vol. 1, pp. 269-271, 1959.

[12] S. Fiske, *Social Beings: Core Motives in Social Psychology*. John Wiley and Sons, 2009.

[13] L. Franken, "Quality of Service Management: A Model-Based Approach," PhD thesis, Centre for Telematics and Information Technology, 1996.

[14] J. Golbeck and J. Hendler, "Inferring Trust Relationships in Web-Based Social Networks," *ACM Trans. Internet Technology*, vol. 6, no. 4, pp. 497-529, 2006.

[15] J. Goldstein, A. Kwasinksi, P. Kingsbury, R.E. Sabin, and A. McDowell, "Annotating Subsets of the Enron Email Corpus," *Proc. Third Conf. Email and Anti-Spam*, 2006.

[16] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins, "Propagation of Trust and Distrust," *Proc. Int'l Conf. World Wide Web (WWW '04)*, pp. 403-412, 2004.

[17] S. Guo, M. Wang, and J. Leskovec, "The Role of Social Networks in Online Shopping Information Passing, Price of Trust, and Consumer Choice," *Proc. ACM Conf. Electronic Commerce (EC '11)*, pp. 130-137, 2011.

[18] C. Hang, Y. Wang, and M. Singh, "Operators for Propagating Trust and Their Evaluation in Social Networks," *Proc. Int'l Conf. Autonomous Agents and Multiagent Systems (AAMAS '09)*, pp. 1025-1032, 2009.

[19] Y. Hu, Y. Fan, and Z. Di, "Orientation in Social Networks," *Physics and Soc.*, vol. 1, no. 1, 2009.

[20] M. Jamali and M. Ester, "TrustWalker: A Random Walk Model for Combining Trust-Based and Item-Based Recommendation," *Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD '09)*, pp. 29-42, 2009.

[21] T. Korkmaz and M. Krunz, "Multi-Constrained Optimal Path Selection," *Proc. IEEE INFOCOM*, pp. 834-843, 2001.

[22] J. Kunegis, A. Lommatzsch, and C. Bauckhang, "The Slashdot Zoo: Mining a Social Network with Negative Edges," *Proc. Int'l Conf. World Wide Web (WWW '09)*, pp. 741-750, 2009.

[23] U. Kuter and J. Golbeck, "Sunny: A New Algorithm for Trust Inference in Social Networks Using Probabilistic Confidence Model," *Proc. Conf. Artificial Intelligence (AAAI '07)*, pp. 1377-1382, 2007.

[24] G. Levinger, "Development and Change," *Close Relationships*, pp. 315-359, Freeman, 1983.

[25] L. Li, Y. Wang, and E.-P. Lim, "Trust-Oriented Composite Services Selection and Discovery," *Proc. Int'l Joint Conf. Service-Oriented Computing (ICSOC '09)*, pp. 50-67, 2009.

[26] C. Lin, N. Cao, S. Liu, S. Papadimitriou, J. Sun, and X. Yan, "SmallBlue: Social Network Analysis for Expertise Search and Collective Intelligence," *Proc. Int'l Conf. Data Eng. (ICDE '09)*, pp. 1483-1486, 2009.

[27] G. Liu, Y. Wang, and M. Orgun, "Quality of Trust for Social Trust Path Selection in Complex Social Networks," *Proc. Int'l Conf. Autonomous Agents and Multiagent Systems (AAMAS '10)*, pp. 1575-1576, 2010.

[28] G. Liu, Y. Wang, and M.A. Orgun, "Optimal Social Trust Path Selection in Complex Social Networks," *Proc. Conf. Artificial Intelligence (AAAI '10)*, pp. 1397-1398, 2010.

[29] G. Liu, Y. Wang, M.A. Orgun, and E.-P. Lim, "A Heuristic Algorithm for Trust-Oriented Service Provider Selection in Complex Social Networks," *Proc. IEEE Int'l Conf. Services Computing (SCC '10)*, pp. 130-137, 2010.

[30] H. Liu, E.-P. Lim, H.W. Lauw, M.-T. Le, A. Sun, J. Srivastava, and Y.A. Kim, "Predicting Trusts among Users of Online Communities: An Epinions Case Study," *Proc. ACM Conf. Electronic Commerce (EC)*, pp. 310-319, 2008.

[31] H. Ma, T. Zhou, M. Lyu, and I. King, "Improving Recommender Systems by Incorporating Social Contextual Information," *ACM Trans. Information Systems*, vol. 29, no. 2, article 9, 2011.

[32] A. Mccallum, X. Wang, and A. Corrada-Emmanuel, "Topic and Role Discovery in Social Networks with Experiments on Enron and Academic Email," *J. Artificial Intelligence Research*, vol. 30, no. 1, pp. 249-272, 2007.

[33] K. Merton, "The Role Set Problems in Sociological Theory," *British J. Sociology*, vol. 8, no. 2, pp. 110-113, 1957.

[34] S. Milgram, "The Small World Problem," *Psychology Today*, vol. 1, pp. 61-67, 1967.

[35] R. Miller, D. Perlman, and S. Brehm, *Intimate Relationships*, fourth ed. McGraw-Hill College, 2007.

[36] A. Mislove, M. Marcon, K. Gummadi, P. Druschel, and B. Bhattacharjee, "Measurement and Analysis of Online Social Networks," *Proc. ACM SIGCOMM Conf. Internet Measurement (IMC '07)*, pp. 29-42, 2007.

[37] I. Pool and M. Kochen, "Contacts and Influence," *Social Networks*, vol. 1, pp. 1-48, 1978.

[38] J. Tang, J. Zhang, L. Yan, J. Li, L. Zhang, and Z. Su, "ArnetMiner: Extraction and Mining of Academic Social Networks," *Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD '08)*, pp. 990-998, 2008.

[39] F. Walter, S. Battiston, and F. Schweitzer, "A Model of a Trust-Based Recommendation System on a Social Network," *Autonomous Agent Multi-Agent System J.*, vol. 16, no. 1, pp. 57-74, Feb. 2008.

[40] G. Wang and J. Wu, "Multi-Dimensional Evidence-Based Trust Management with Multi-Trusted Paths," *Future Generation Computer Systems*, vol. 17, pp. 529-538, 2011.

[41] Y. Wang and V. Varadharajan, "Role-Based Recommendation and Trust Evaluation," *Proc. IEEE Int'l Conf. E-Commerce Technology and Fourth IEEE Int'l Conf. Enterprise Computing, E-Commerce, and E-Services (EEE '07)*, pp. 278-295, 2007.

[42] S. Yang, J. Zhang, and I. Chen, "Web 2.0 Services for Identifying Communities of Practice," *Proc. IEEE Int'l Conf. Services Computing (SCC '07)*, pp. 130-137, 2007.

[43] S. Yoo, Y. Yang, F. Lin, and I. Moon, "Mining Social Networks for Personalized Email Prioritization," *Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD '09)*, pp. 967-976, 2009.

[44] T. Yu, Y. Zhang, and K. Lin, "Efficient Algorithms for Web Services Selection with End-to-End QoS Constraints," *ACM Trans. Web*, vol. 1, no. 1, article 6, 2007.
[45] L. Zeng, B. Benatallah, M. Dumas, J. Kalagnanam, and Q. Sheng, "Quality Driven Web Services Composition," *Proc. Int'l Conf. World Wide Web (WWW '03)*, pp. 411-421, 2003.
[46] L. Zeng, B. Benatallah, A. Ngu, M. Dumas, J. Kalagnanam, and H. Chang, "QoS-Aware Middleware for Web Service Composition," *IEEE Trans. Software Eng.*, vol. 30, no. 5, pp. 311-327, May 2004.

Guanfeng Liu received the BEng degree in computer science and technology from the Qingdao University of Science and Technology (QUST), P.R. China in 2005, and the MEng degree in computer software and theory from Qingdao University (QDU), P.R. China in 2008. He is currently working toward the PhD degree in computer science at Macquarie University, Australia. His current research focuses on trust management in online social networks. He is a student member of the IEEE.

Yan Wang received the BEng, MEng, and DEng degrees in computer science and technology in 1988, 1991, and 1996, respectively, from the Harbin Institute of Technology (HIT), P.R. China. He is currently an associate professor in the Department of Computing, Macquarie University, Sydney, Australia. Prior to joining Macquarie University, he was a research fellow in the Department of Computer Science, National University of Singapore, from 1999 to 2003. He is now on the editorial board of *Service-Oriented Computing & Applications* (SOCA) and was a guest coeditor of the special track on e-commerce of *IEEE Internet Computing* in 2008. He also served as a program cochair of the 2011 IEEE International Conference on Services Computing (SCC 2011) and the Eighth International Conference on Autonomic and Trusted Computing (ATC 2011). His research interests include trusted computing, e-commerce, software agent, and security. He is a senior member of the IEEE.

Mehmet A. Orgun received the BSc and MSc degrees in computer science and engineering from Hacettepe University, Ankara, Turkey, in 1982 and 1985, respectively, and the PhD degree in computer science from the University of Victoria, Canada, in 1991. He is currently a professor at Macquarie University, Sydney, Australia. Prior to joining Macquarie University in September 1992, he worked as a postdoctoral fellow at the University of Victoria. He researches in the broad area of intelligent systems with specific research interests in knowledge discovery, multiagent systems, trusted systems, and temporal reasoning. His professional service includes editorial and review board memberships of leading journals and program committee or senior program committee memberships of numerous national and international conferences. He is a senior member of the IEEE.

Ee-Peng Lim received the BSc degree in computer science from the National University of Singapore and the PhD degree from the University of Minnesota, Minneapolis, in 1994. He is a professor at the School of Information Systems of the Singapore Management University (SMU). He is also the codirector of the Living Analytics Research Centre at SMU. His research interests include social network/web mining, information integration, and digital libraries. He is currently an associate editor of the *ACM Transactions on Information Systems*, *Information Processing and Management*, *Social Network Analysis and Mining*, *Journal of Web Engineering, IEEE Intelligent Systems*, and *International Journal of Digital Libraries*. He is a member of the ACM publications board and serves on the steering committees of several conferences, including the International Conference on Asian Digital Libraries (ICADL), Pacific Asia Conference on Knowledge Discovery and Data Mining (PAKDD), and International Conference on Social Informatics (SocInfo). He is a senior member of the IEEE.