Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI-10)

Optimal Social Trust Path Selection in Complex Social Networks

Guanfeng Liu, Yan Wang and Mehmet A. Orgun
Department of Computing
Macquarie University, NSW, Australia 2109
{gliu, yanwang, mehmet}@science.mq.edu.au

Abstract

Online social networks are becoming increasingly popular and are being used as the means for a variety of rich activities. This demands the evaluation of the trustworthiness between two unknown participants along a certain social trust path between them in the social network. However, there are usually many social trust paths between participants. Thus, a challenging problem is finding which social trust path is the optimal one that can yield the most trustworthy evaluation result.

In this paper, we first present a new complex social network structure and a new concept of Quality of Trust (QoT) to illustrate the ability to guarantee a certain level of trustworthiness in trust evaluation. We then model the optimal social trust path selection as a Multi-Constrained Optimal Path (MCOP) selection problem which is NP-Complete. For solving this problem, we propose an efficient approximation algorithm MONTE_K based on the Monte Carlo method. The results of our experiments conducted on a real dataset of social networks illustrate that our proposed algorithm significantly outperforms existing approaches in both efficiency and the quality of selected social trust paths.

Introduction

In recent years, online social networks have been attracting a large number of participants. In such social networks, each node represents a participant and each link corresponds to real world interactions or online interactions between participants. One participant can give a trust value to another based on their past interactions. For example, a trust rating can be given by one participant to another based on the quality of the movies recommended by the latter at FilmTrust[1]which is a social networking site for movie recommendations. As each participant usually interacts with many other participants, multiple trust paths may exist between two participants without any direct link between them, such as the trust path $A \rightarrow B \rightarrow E \rightarrow M$, $A \rightarrow C \rightarrow E \rightarrow M$ and $A \rightarrow D \rightarrow E \rightarrow M$ in Fig. 1. 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 participant based on the trust information between the intermediate participants along the path. This process is called the *trust propagation* (Golbeck and Hendler 2006). This path with trust information linking the source participant and the target participant is called a *social trust path* (Hang, Wang, and Singh 2009). For example, in Fig. 1, if A is an employer and M is an employee candidate in the social network, A can evaluate the trustworthiness of M along the social trust paths from A to M. We term A as the *source participant* and M as the *target participant*. In another example, in a new generation CRM system, if A is a retailer who wants to introduce new products to a person M who is not a customer but is a friend or friend's friend of some existing customers, A can evaluate the trustworthiness of M based on the trust information between the customers along the social trust paths from A to M. Nevertheless, there can be over tens of thousands of social trust paths between a source participant and the target one in large-scale social networks (Kunegis, Lommatzsch, and Bauckhang 2009). Evaluating trust values along all these social trust paths consumes huge computation time (Baase and Gelder 1999). Therefore, a problem is that among multiple social paths, which one is the optimal yielding the most trustworthy result of trust propagation. In the literature, Lin *et al.* (2009) 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 participant is selected as the optimal one. This method neglects *trust information* between participants. In the work by Hang *et al.* (2009), 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, or the one between an employer and an employee) and the *recommendation roles* of a participant (e.g., a supervisor as a referee in his postgraduate's job application)

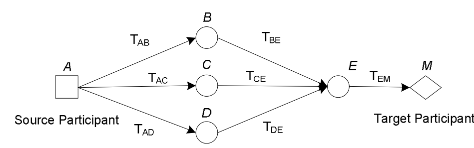

Figure 1: Social network

[1]http://trust.mindswap.org/filmtrust/

have significant influence on trust propagation (Adler 2001; Miller, Perlman, and Brehm 2007) and can be obtained by using data mining techniques in some specific social networks (Mccallum, Wang, and Corrada-Emmanuel 2007; Tang et al. 2008). But these factors have not been considered in existing social trust path selection methods.

In addition, different source participants usually have different preferences in evaluating the trustworthiness of the target participant. Thus, different constraints of the above factors should be set by each source participant to represent their preference in social trust path selection. But this is not supported in existing trust propagation models.

In this paper, we first introduce the structure of complex social networks which contains *trust information*, *social relationships* and *recommendation roles* of participants. We also present a new concept, *Quality of Trust* and model the social trust path selection as the Multi-Constrained Optimal Path (MCOP) selection problem, which is an NP-Complete problem (Korkmaz and Krunz 2001).

Since the characteristics of social networks are not considered in existing approximation algorithms (Korkmaz and Krunz 2001; Li, Wang, and Lim 2009; Yu, Zhang, and Lin 2007) for solving the MCOP selection problem, they can not deliver good performance. Therefore, we propose an efficient approximation algorithm, MONTE_K, based on the Monte Carlo method (Gentle, Hardle, and Mori 2004) and our optimization strategies. The experiments conducted on a real online social network, *Enron* email corpus[2], demonstrate the good performance of our proposed algorithm in both optimal social trust path selection and efficiency.

Related Work

Sociologists have been studying the properties of social networks for a long time. In the 1960's, Milgram (Milgram 1967) illustrated that the average path length between two Americans was about 6 hops in an experiment of mail sending and thus proposed the *small-world* characteristic in social networks. In the 1970's, Pool *et al.* (Pool and Kochen 1978) further analyzed the influences of small-world characteristic on human interactions. Mislove *et al.* (2007) analyzed several popular social networks including Facebook, MySpace, Flickr and Orkut, and validated the small-world and *power-law* (i.e. in a social network, the probability that a node has degree k is proportional to k^{-r}, $r > 1$) characteristics of online social networks.

In social networks, since trust is one of the most important factors for participants decision-making (Kuter and Golbeck 2007), several trust management methods have been proposed. For example, Guha *et al.* (2004) propose a trust propagation model, where the number of hops in trust propagation is considered in calculating the propagated trust values between a source participant and the target one. Golbeck *et al.* (2006) propose a trust inference mechanism for the trust relation establishment between a source participant and the target one based on averaging trust values along the social trust paths. The above trust management strategies are based only on the ratings from participants. As pointed in social science theories (Adler 2001; Miller, Perlman, and Brehm 2007), both *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) have significant influence on participants' decision making. Therefore, they should be taken into account in trust propagation. In (Kuter and Golbeck 2007), though domain experts can determine the "confidence of trust", social relationships are not explicitly considered. In addition, in those reported works, the trust value between two participants without direct links is calculated based all social trust paths, consuming huge computation time. Thus they are not feasible for large-scale social networks.

For addressing the social path selection problem, Lin *et al.* (2009) propose a shortest optimal social path selection method. In their model, up to 16 social paths with no more than 6 hops between a source participant are selected and the shortest one is taken as the optimal path. In this method, some significant influence factors including *trust information*, *recommendation roles* and *social relationships* between participants are not taken into account. In another reported work (Hang, Wang, and Singh 2009), Hang *et al.* propose a social trust path selection method, where the social trust path with the maximum propagated trust value is selected as the optimal one that yields the most trustworthy results of trust propagation between a source participant and the target participant. In their model, although trust information is taken into account in social trust path selection, the other two important factors are ignored.

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

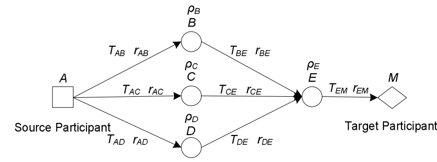

Figure 2: Complex social network

Complex Social Networks

The complex social networks proposed by us (Liu, Wang, and Orgun 2009) comprise the attributes of several impact factors, which influence the trustworthiness of trust propagation and hence the decision making of a source participant.

Trust between Participants

In the literature, many different trust definitions have been proposed addressing different aspects of trust. Alunkal *et al.* (2003) 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 *et al.* (2006) 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, the 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 that B's future action can lead to the expected outcome.

Social Intimacy Degree

A participant can have trust to the participants with whom he/she has more intimate social relationships than those with whom he/she has less intimate social relationships (Ashri et al. 2005). The following principle in social psychology illustrates the impact of social relationships on trust.

Principle 1: An intimate relationship is a particularly close interpersonal relationship, in which the participants are confidants and trust one another very well (Miller, Perlman, and Brehm 2007).

Therefore, the *Social Intimacy Degree* (SID) should be defined to describe the extent to which two participants have intimate social relationships. The SID values can be obtained by using data mining techniques.

Definition 2: $r_{AB} \in [0,1]$ is the *Social Intimacy Degree* between participant A and participant 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.

For example, in the work by Mccallum *et al.* (2007), through mining the subjects and contents of emails in *Enron* Corporation, the social relationships between two participants can be discovered and the corresponding SID values can be calculated based on probabilistic models. In addition, in academic social networks formed by large databases of Computer Science literature (e.g, DBLP or ACM Digital Library), some social relationships between two scholars (e.g., co-authors, supervisors and students) can be mined from their publications or from their home pages. The SID value between them can be calculated by applying the information of social relationships into the PageRank model (Tang et al. 2008).

Role Impact Factor

Rich activities of participants in social networks can be categorized into different domains (e.g., recruitment or product introduction) based on their characteristics (Wang and Varadharajan 2007). In a certain domain, recommendations from a domain expert are more credible than that from a beginner. For example, the recommendation from a *professor* to a textbook is more credible than that from a *student* in a subject taught by the professor. The following principle in social psychology illustrates the impact of recommendation roles on trust.

Principle 2: The effective growing knowledge intensity is a trend towards greater reliance on trust, especially relevant to particular social positions where one's actions weigh heavily on one's social position (Adler 2001).

Therefore, *Role Impact Factor* (RIF) should be defined to reflect the impact of a participant's recommendation role (e.g., *expert* or *beginner*) on trust propagation.

Definition 3: $\rho_A \in [0,1]$ is 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 build up comprehensive role hierarchies in all domains, it is realistic to build it up in a particular application. For example, in the *Enron* email dataset[2], the role of each email sender or receiver can be known through parsing or mining the contents of emails. The value of RIF can be calculated based on the role of a participant (e.g., *department manager* or *CEO*) by applying probabilistic models. In addition, in academic social networks, the roles of scholars can be mined from authors' publications or from the profile in their home pages. The value of RIF can also be calculated by applying the PageRank model (Tang et al. 2008).

Social Trust Path Selection

In this section, we propose a novel social trust path selection model with end-to-end *Quality of Trust* (QoT) constraints.

Quality of Trust (QoT)

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 (Franken 1996). For example, in broadband services, bit rate, delay and packet dropping probability are important for quality measurement. Similar to the QoS, we propose a new concept, *Quality of Trust* (QoT) as follows.

Definition 4: *Quality of Trust* (QoT) is the ability to guarantee a certain level of trustworthiness in trust propagation along a social trust path, taking trust (T), social intimacy degree (r), and role impact factor (ρ) as attributes.

In service composition, service consumers can set multiple end-to-end constraints for the attributes of QoS (e.g., cost, delay and availability) to satisfy their requirements of services. Different requirements have different constraints (e.g., total cost<\$20, delay<5s and availability>70%). In our model, a source participant can set multiple end-to-end constraints for QoT attributes (i.e., T, r and ρ) as the requirements of trust propagation in social trust paths. For example, in Fig. 2, source participant A can set the end-to-end QoT constraints for the social trust paths from A to M as $Q_{AM} = \{Q^T_{AM} > 0.5,\ Q^r_{AM} > 0.05,\ Q^\rho_{AM} > 0.5\}$ in the domain of *employment* or $Q_{AM} = \{Q^T_{AM} > 0.3,\ Q^r_{AM} > 0.4,\ Q^\rho_{AM} > 0.2\}$ in the domain of *product introduction*. Q^T_{AM}, Q^r_{AM} and Q^ρ_{AM} are the QoT constraints of T, r and ρ respectively.

QoT Attribute Aggregation

When specifying end-to-end QoT constraints of social trust path selection, we need to know the aggregated value of each QoT attribute in a social trust path.

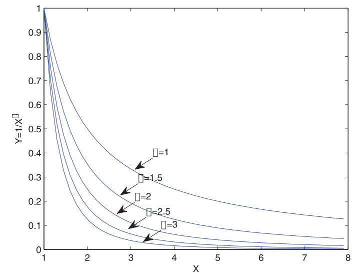

Figure 3: Attenuation equation

Trust Aggregation In our model, we adopt the strategy that the aggregated trust value between a source participant and the target participant is calculated as the multiplication of trust values between all adjacent participants along the social trust path. This strategy has been widely used in the literature as a feasible trust aggregation method (Li, Wang, and Lim 2009; Walter, Battiston, and Schweitzer 2008). In the model, if there are n participants $a_1, ..., a_n$ in order in a social trust path (denoted as $p(a_1, ..., a_n)$), the aggregated trust value is calculated as.

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

In identifying the optimal social trust path that yields the most trustworthy result of trust propagation, different from existing works (Hang, Wang, and Singh 2009; Lin et al. 2009), this aggregated trust value will be combined with social intimacy degree and role impact factor.

Social Intimacy Degree Aggregation Social intimacy between participants involves some extent of transitivity and it is attenuated with the increasing number of hops between them in a social trust path (Levinger 1983). In addition, in the real world, the intimacy degree is attenuated fast when approaching one. In contrast, the intimacy degree is attenuated slowly when approaching zero (Miller, Perlman, and Brehm 2007). Namely, the attenuation of social intimacy degree is non-linear in social networks. The attenuation equation can be defined in Eq. (2) (plotted in Fig. (3)).

$$Y = \frac{1}{X^\alpha} \tag{2}$$

where X and Y are independent variables and $\alpha \geq 1$ is used to control the attenuation speed.

Based on Eq. (2), the aggregated r value in path $p(a_1, ..., a_n)$ can be calculated by Eq. (3).

$$r_{p(a_1,...,a_n)} = \frac{\prod_{(a_i,a_{i+1})\in p(a_1,...,a_n)} r_{a_i\ a_{i+1}}}{\theta^\alpha} \tag{3}$$

where θ is the number of hops of path $p(a_1, ..., a_n)$.

Role Impact Factor Aggregation Since RIF does not have the transitivity property (Adler 2001), in this paper, we average the RIF values of intermediate recommending participants in path $p(a_1, ..., a_n)$ as the aggregated value based on Eq. (4)

$$\rho_{p(a_1,...,a_n)} = \frac{\sum_{i=2}^{n-1} \rho_{a_i}}{n-2} \tag{4}$$

Utility Function

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

$$\mathcal{F}_{p(a_1,...,a_n)} = \omega_T * T_{p(a_1,...,a_n)} + \omega_r * r_{p(a_1,...,a_n)} + \omega_\rho * \rho_{p(a_1,...,a_n)} \tag{5}$$

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

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

Social Trust Path Selection Algorithm

The optimal social trust path selection with multiple end-to-end QoT constraints can be modelled as the classical Multi-Constrained Optimal Path (MCOP) selection problem which is NP-Complete (Korkmaz and Krunz 2001). In this section, we first analyze some existing approximation algorithms for the MCOP selection problem and then propose an efficient approximation algorithm, MONTE_K, based on the Monte Carlo method (Gentle, Hardle, and Mori 2004) and our optimization strategies.

Existing Approximation Algorithms

In the literature, several approximation algorithms have been proposed for the MCOP selection problem.

Korkmaz *et al.* (2001)propose a heuristic algorithm, H_MCOP which is one of the most promising algorithms in solving the MCOP selection problem as it outperforms prior algorithms in both efficiency and the quality of delivered solutions. In H_MCOP, multiple QoS constraints and QoS attributes are aggregated into a single value based on a non-linear function as in Eq. (6). H_MCOP adopts the Dijkstria's shortest path algorithm twice to search the path with the minimal $g_\lambda(p)$ value.

$$g_\lambda(p) \triangleq (\frac{q_1(p)}{Q_1})^\lambda + (\frac{q_2(p)}{Q_2})^\lambda + ... + (\frac{q_m(p)}{Q_m})^\lambda \tag{6}$$

where $\lambda \geq 1$; $q_i(p)$ is the aggregated value of the i^{th} QoS attribute of path p; Q_i is the i^{th} QoS constraint of path p.

Consequently, in the field of Service-Oriented Computing (SOC), Yu *et al.* (2007) propose an approximation algorithm, MCSP_K to solve the quality-driven service selection problem which is also the MCOP selection problem. This method keeps only K paths from a source node to each intermediate node, aiming to reduce the search space and execution time. Their K-path selection is based on Eq. (7).

$$\xi(p) \triangleq max\{(\frac{q_1(p)}{Q_1}), (\frac{q_2(p)}{Q_2}), ..., (\frac{q_m(p)}{Q_m})\} \tag{7}$$

From Eq. (7), if any QoS attribute value does not satisfy the corresponding QoS constraint in path p, then $\xi(p) > 1$.

In their search strategies, the paths with up to K minimum ξ values are kept at each intermediate node. This method never prunes any feasible path if it exists. In their service candidate graph, all services are categorized into different service sets based on their functionalities. Any two nodes in adjacent service sets have a link with each other and thus all 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 paths from a source to each intermediate node and hence the time complexity will become exponential. Therefore, the algorithm does not fit large-scale social networks.

Algorithm Description

Monte Carlo Method Monte Carlo method (Gentle, Hardle, and Mori 2004) is a computational algorithm which relies on repeated random sampling to compute results. The Monte Carlo method has been used in an algorithm called MCBA (Monte Carlo Based Algorithm) (Li, Wang, and Lim 2009) for solving the NP-Complete trustworthy service selection problem. But MCBA has some drawbacks when applied to the computation in networks: firstly, all neighbors of a node are regarded as candidates for selection. Thus at each node, there exists a large search space leading to low inefficiency. Secondly, in each simulation, there exists a certain probability to deliver a solution worse than that from previous simulations, leading to a lower probability to deliver the optimal solution.

MONTE_K In this paper, we propose MONTE_K, an efficient Monte Carlo approximation algorithm. In MONTE_K, we adopt two optimization strategies.

Strategy 1: *K-path selection*. Let v_s denote a source participant and v_t denote the target one. According to Eq. (7), the lower the ξ value of a path, the higher the probability for that path to be a feasible solution. Thus, given a partially identified social trust path from v_s to v_x ($v_x \neq v_t$), we calculate the ξ values of the paths from v_s to each neighboring node of v_x and record up to K neighboring nodes that yield up to K minimum ξ values as candidates for selection.

As this strategy selects no more than K neighbors at each step in social trust path selection, it can reduce the search space and deliver higher efficiency than MCBA.

Strategy 2: *Optimization at dominating nodes.* If the indegree of v_y ($v_y \neq v_s$) is greater than 1 in the social network, then node v_y is regarded as a *dominating node*. To obtain a near-optimal solution, MONTE_K performs multiple simulations. In the first simulation, if a social trust path from v_s to v_y (denoted as path p_1) is selected, we store the utility $\mathcal{F}$, ξ value and the aggregated value of each QoT attribute of p_1 at v_y. In all subsequent simulations, if a different social path from v_s to v_y (denoted as path p_y, where $y > 1$) is selected, the optimization is performed in the following situations.

Situation 1: If $v_y = v_t$ and $\mathcal{F}(p_y) < \mathcal{F}(p_1)$, it indicates p_y is worse than p_1. Thus we replace the values of p_y (i.e., T, r, ρ, $\mathcal{F}$ and ξ) with the one stored at v_y .

Situation 2: If $v_y = v_t$ and $\mathcal{F}(p_y) > \mathcal{F}(p_1)$, it indicates p_y is better than p_1. Thus, we store the values of p_y at v_y.

Situation 3: If $v_y \neq v_t$, $\mathcal{F}(p_y) < \mathcal{F}(p_1)$ and $\xi(p_y) > \xi(p_1)$, it indicates p_y is worse than p_1. Thus we replace the values of p_y with the one stored at v_y .

Situation 4: If $v_y \neq v_t$, $\mathcal{F}(p_y) > \mathcal{F}(p_1)$ and $\xi(p_y) < \xi(p_1)$, it indicates p_y is better than p_1. Thus, we store the values of p_y at v_y.

Following Strategy 2, the dominating node v_y records T, r, ρ, $\mathcal{F}$ and ξ values of the locally optimal social trust path from v_s to v_y. The optimization at v_y can guarantee that the delivered solution from v_s to v_y is locally optimal.

The process of MONTE_K is as follows.

Initialization: Mark the status of all nodes in the network as unvisited. Add v_s into set $temp\_P$ that stores the solution (i.e., identified social trust path). Let $Min\_K(\mu)$ be a set that stores up to K neighboring nodes of node μ (*lines 1 to 3 in Algorithm 1*).

Step 1: Get an unvisited node μ from $temp\_P$ and mark μ as visited. Select up to K neighboring nodes of μ based on *strategy 1* and put these nodes into $Min\_K(\mu)$ (*lines 4 to 12 in Algorithm 1*).

Step 2: For each $v_i \in Min\_K(\mu)$, calculate the probability of v_i for selection, based on the utility of the social trust path from v_s to v_i via μ (denoted as path p_{v_i}). The probability of v_i to be selected is $\mathcal{P}(p_{v_i}) = \frac{\mathcal{F}(p_{v_i})}{\sum_{i=1}^{size(Min\_K(\mu))} \mathcal{F}(p_{v_i})}$ (*lines 13 to 17 in Algorithm 1*).

Step 3: Select v_j from set $Min\_K(\mu)$ based on a random number $rand \in [0, 1]$ and $\{\mathcal{P}(p_{v_i})\}$. If $\xi(p_{v_j}) \leq 1$ and the indegree of v_j is greater than 1, then v_j is a dominating node and thus perform the optimization at v_j based on *Strategy 2*. If $\xi(p_{v_j}) > 1$, it indicates that no feasible solution has been delivered in this simulation (*lines 18 to 25 in Algorithm 1 and lines 1 to 18 in Algorithm 2*).

Step 4: If $v_j \neq v_t$, add v_j into $temp\_P$ and go to *Step 1*. If $v_j = v_t$, return $temp\_P$ and $\mathcal{F}(temp\_P)$ (*lines 26 to 30 in Algorithm 1*).

According to the *power-law* characteristic (Mislove et al. 2007), only a few nodes have a large outdegree in social networks (e.g., in *Enron* email corpus[2], 94.7% nodes have an outdegree less than 15). Therefore, in MONTE_K, each node can keep a small search space without pruning a large number of neighboring nodes (i.e., candidates) of a node in K-path selection, which results in high efficiency and a higher probability of finding the optimal solution. The time complexity of MONTE_K is $O(mluK)$, where m is the number of simulations; l is the average length of the shortest social trust paths from a source participant to the target one in social networks; u is the maximal outdegree of nodes in social networks and K is the argument specified for K-path selection. In social networks, usually $l < 7$ according to the *small-world* characteristic (Mislove et al. 2007). Thus the time complexity of MONTE_K is $O(muK)$. By both addressing the characteristics of social networks and adding optimization strategies in the algorithm design, MONTE_K can deliver better solutions with less execution time than existing methods.

Algorithm 1: MONTE_K

Data: $G = (V, E)$, QoT constraints, v_s, v_t, K-path number
Result: $\mathcal{F}$, P_{st}
/*$G = (V, E)$: the sub-network between a source agent and the target agent; v_s, v_t: a source agent and the target agent; $\mathcal{F}$: utility; P_{st}: identified social trust path; $temp\_P$: partly identified social trust path; $De(v)$: the indegree of v; $\mathcal{P}(v)$: the probability of v to be selected; $adj[\mu]$: neighboring nodes of μ; $Min\_K(\mu)$: the set stores up to K neighboring nodes of μ; $\mathcal{F}_{old}(p_v), \xi_{old}(p_v), T_{old}(p_v), r_{old}(p_v), \rho_{old}(p_v), temp_{old}\text{-}P(p_v)$: values stored at dominating node v; */
begin
  Mark the status of all node as **unvisited**, $P_{st} = \emptyset, temp\_P \longleftarrow v_s$
  while *(unvisited node exists in* $temp\_P$*)* **do**
    Get unvisited node μ from $temp\_P$
    Mark μ as **visited**
    for *each* $v_i \in adj[\mu]$ **do**
      Calculate $T(p_{v_i}), r(p_{v_i}), \rho(p_{v_i})$ and $\xi(p_{v_i})$
    if *size(*$adj[\mu]$*)¿K* **then**
      Put v_i with K minimum $\xi(p_{v_i})$ into $Min\_K(\mu)$
    else
      Put $v_i \in adj[\mu]$ into $Min\_K(\mu)$
    for *each* $v_i \in Min\_K(\mu)$ **do**
      $\mathcal{F}(p_{v_i}) = \omega_T * T(p_{v_i}) + \omega_r * r(p_{v_i}) + \omega_\rho * \rho(p_{v_i})$
      $\mathcal{P}(v_i) = \mathcal{F}(p_{v_i}) / \sum_{i=1}^{size(Min\_K(\mu))} \mathcal{F}(p_{v_i})$
    Generate a random number $rand \in [0, 1]$
    Select the m^{th} node v_j such that $rand \leq \sum_{i=1}^{m} \mathcal{P}(v_i)$
    if $\xi(p_{v_j}) > 1$ **then**
      Break
    else
      if $De(v_j) > 1$ **then**
        Optimization at Dominating Nodes $(\mathcal{F}(p_{v_j}), \xi(p_{v_j}), T(p_{v_j}), r(p_{v_j}), \rho(p_{v_j}), temp\_P)$
      if $v_j = v_t$ **then**
        $\mathcal{F} = \mathcal{F}(p_{v_j}), P_{st} = temp\_P$
        Return $\mathcal{F}$ and P_{st}
      else
        $T(p_{v_j}) = T_{new}(p_{v_j}), r(p_{v_j}) = r_{new}(p_{v_j})$
        $\rho(p_{v_j}) = \rho_{new}(p_{v_j}), temp\_P = temp\_P_{new}$
        Put v_j into $temp\_P$
end

Algorithm 2: Optimization at Dominating Nodes

Data: $\mathcal{F}(p_{v_j}), \xi(p_{v_j}), v, temp\_P, T(p_{v_j}), r(p_{v_j}), \rho(p_{v_j})$
Result: $\mathcal{F}_{new}(p_{v_j}), \xi_{new}(p_{v_j}), temp\_P_{new}, T_{new}(p_{v_j}), r_{new}(p_{v_j}), \rho_{new}(p_{v_j})$
begin
  Get $(node(v_j).\mathcal{F}, \xi, T, r, \rho, temp\_P)$ that stored at v_j
  Put these values into $(\mathcal{F}_{old}(p_{v_j}), \xi_{old}(p_{v_j}), T_{old}(p_{v_j}), r_{old}(p_{v_j}), \rho_{old}(p_{v_j}), temp\_P_{old})$
  if $v_j \neq v_t$ **then**
    if $\mathcal{F}(p_{v_j}) < \mathcal{F}_{old}(p_{v_j})$ *and* $\xi(p_{v_j}) > \xi_{old}(p_{v_j})$ **then**
      Put $\{\mathcal{F}_{old}(p_{v_j}), \xi_{old}(p_{v_j}), T_{old}(p_{v_j}), r_{old}(p_{v_j}), \rho_{old}(p_{v_j}), temp\_P_{old}\}$ **into** $\{\mathcal{F}_{new}(p_{v_j}), \xi_{new}(p_{v_j}), T_{new}(p_{v_j}), r_{new}(p_{v_j}), \rho_{new}(p_{v_j}), temp\_P_{new}\}$
    else
      if $\mathcal{F}(p_{v_j}) > \mathcal{F}_{old}(p_{v_j})$ *and* $\xi(v) < \xi_{old}(p_{v_j})$ **then**
        Put $\{\mathcal{F}(p_{v_j}), \xi(p_{v_j}), T(p_{v_j}), r(p_{v_j}), \rho(p_{v_j}), temp\_P\}$ **into** $\{\mathcal{F}_{new}(v), \xi_{new}(v), T_{new}(v), r_{new}(v), \rho_{new}(v), temp\_P_{new}\}$
        Update$(node(v_j).\mathcal{F}, \xi, T, r, \rho, temp\_P)$ with these values
  else
    if $\mathcal{F}(p_{v_j}) < \mathcal{F}_{old}(p_{v_j})$ **then**
      Put $\{\mathcal{F}_{old}(p_{v_j}), \xi_{old}(p_{v_j}), T_{old}(p_{v_j}), r_{old}(p_{v_j}), \rho_{old}(p_{v_j}), temp\_P_{old}\}$ **into** $\{\mathcal{F}_{new}(p_{v_j}), \xi_{new}(p_{v_j}), T_{new}(p_{v_j}), r_{new}(p_{v_j}), \rho_{new}(p_{v_j}), temp\_P_{new}\}$
    else
      Put $\{\mathcal{F}(p_{v_j}), \xi(p_{v_j}), T(p_{v_j}), r(p_{v_j}), \rho(p_{v_j}), temp\_P\}$ **into** $\{\mathcal{F}_{new}(p_{v_j}), \xi_{new}(p_{v_j}), T_{new}(p_{v_j}), r_{new}(p_{v_j}), \rho_{new}(p_{v_j}), temp\_P_{new}\}$
      Update $(node(v_j).\mathcal{F}, \xi, T, r, \rho, temp\_P)$ with these values
  Return$(\mathcal{F}_{new}(p_{v_j}), \xi_{new}(p_{v_j}), T_{new}(p_{v_j}), r_{new}(p_{v_j}), \rho_{new}(p_{v_j}), temp\_P_{new})$
end

Experiments

Experiment Settings

The *Enron* email dataset[2] has been proved to possess the small-world and power-law characteristics of social networks and has been widely used in the studies of social networks (Goldstein et al. 2006; Mccallum, Wang, and Corrada-Emmanuel 2007; Yoo et al. 2009). In addition, the social intimacy degree between participants and the role impact factor of participants can be calculated through mining the subjects and contents of emails (Mccallum, Wang, and Corrada-Emmanuel 2007). Therefore, in contrast to other real social network datasets (e.g., Epinions[3] and FilmTrust[1]), the *Enron* email dataset fits our proposed complex social network structure very well. Thus, to verify our proposed algorithm, we select the *Enron* email corpus[2] with 87,474 nodes (participants) and 30,0511 links (formed by sending and receiving emails) as the dataset, and conduct experiments on it.

As the complexity of MCSP_K (Yu, Zhang, and Lin 2007) is exponential in finding an optimal social trust path in social networks, it is ignored in our experiments. Instead, we compare MONTE_K with H_MCOP (Korkmaz and Krunz 2001) and MCBA (Li, Wang, and Lim 2009) in both execution time and the utilities of identified social paths. Since this paper does not focus on detailed data mining techniques, in our experiments, the T, r and ρ values are randomly generated. The end-to-end QoT constraints are set as $Q = \{Q^T \geq 0.05,\ Q^r \geq 0.001,\ Q^\rho \geq 0.3\}$ and the weights of attributes in utility function are set as $\omega_t = 0.25$, $\omega_r = 0.25$ and $\omega_\rho = 0.5$.

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

Performance in Social Trust Path Selection

In this experiment, in order to evaluate the performance of our proposed approximation algorithm in the sub-networks of different scales and structures, we first randomly select 16 pairs of source and target nodes from *Enron* email dataset[2]. We then extract the corresponding 16 sub-networks between them by using the exhaustive search method. Among them, the maximal length of a social trust path varies from 4 to 7 hops following the *small-world* characteristic. The properties of these sub-networks are listed in Table 1.

The number of simulations of MONTE_K and MCBA in each sub-network is also listed in Table 1. The average outdegree in all these sub-networks is 3.77 and 95.5% nodes have an outdegree less than 15. Hence we set $K = 15$ in the K-path selection, without pruning a large number of neighboring nodes of most nodes following the *power-law* characteristic (Mislove et al. 2007). Secondly, we per-

[3]http://epinions.com/

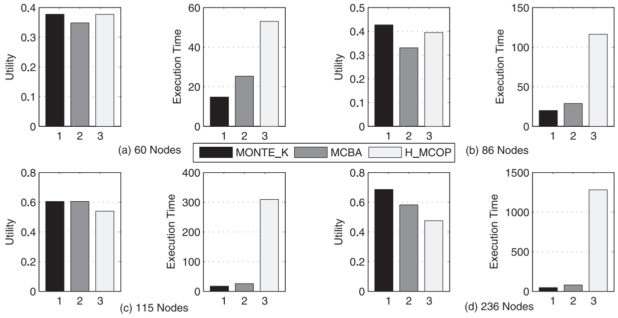

Figure 4: Maximal length of paths is 4 hops

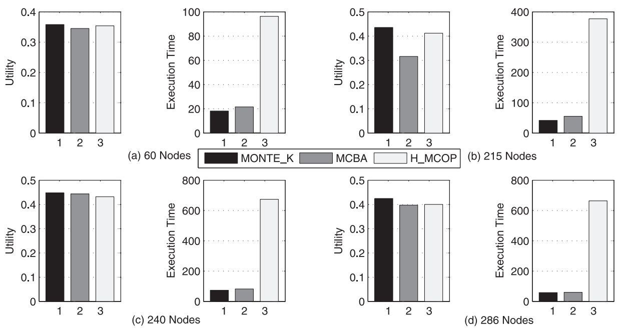

Figure 5: Maximal length of paths is 5 hops

form 500 repeated experiments for MONTE_K and MCBA in each sub-network and record the utilities of the identified social trust paths in each experiment. The maximal utilities of the social trust paths identified in all 500 experiments by MONTE_K and MCBA are selected for the comparison with that yielded by H_MCOP. The average execution time of each of MONTE_K and MCBA in each sub-network is recorded based on 500 repeated experiments. The execution time of H_MCOP is averaged based on 5 independent executions. The results are plotted in Fig. 4 to Fig. 7.

Table 1: Properties of different social networks

ID	Max Hops	Number of Nodes	Number of Links	Max Outdegree	Max Indegree	Simulation Times
1	4	60	113	16	15	100
2	4	86	192	20	32	100
3	4	115	257	41	82	150
4	4	236	1321	74	91	200
5	5	60	107	9	18	100
6	5	215	528	34	48	200
7	5	240	655	32	49	250
8	5	286	749	31	84	300
9	6	61	124	17	32	100
10	6	161	355	43	46	200
11	6	371	1623	56	48	350
12	6	651	2475	173	151	450
13	7	137	373	48	18	200
14	7	321	860	39	38	350
15	7	551	3265	122	91	400
16	7	793	3411	83	89	500

Utility: We can see that in any of 16 cases, MONTE_K does not yield any utility worse than that of H_MCOP while in most sub-networks, the utilities of social trust paths identified by MONTE_K are better than those of H_MCOP (*see Fig. 4(a, c, d), Fig. 5(a) to (d), Fig. 6 (a) to (d) and Fig. 7 (b) to (d)*). The sum of utilities computed by MONTE_K is 12.23% more than that of H_MCOP in 4 hops sub-networks, 4.27% more in 5 hops, 60.62% more in 6 hops and 41.51% more in 7 hops. This is because when a trust path with the maximal utility is a feasible solution, H_MCOP can identify it as the optimal solution. However, when the identified trust path is not a feasible solution, H_MCOP can hardly find a near-optimal solution and some times yields an infeasible one even when a feasible solution exists (*see Fig. 6(b) where the utility computed by H_MCOP is 0*).

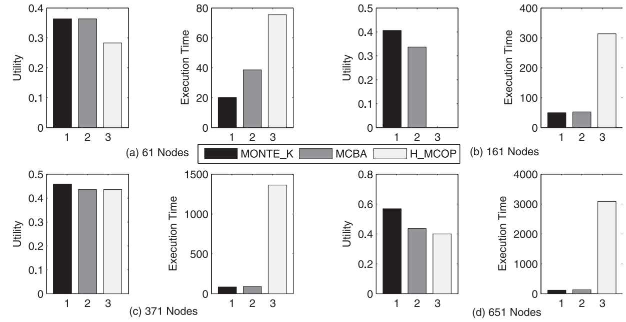

Figure 6: Maximal length of paths is 6 hops

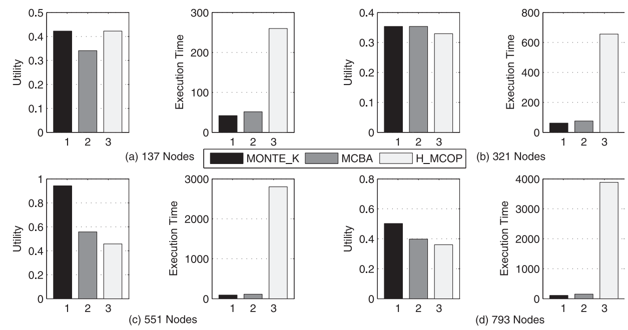

Figure 7: Maximal length of paths is 7 hops

Regarding the utility of identified paths, MONTE_K also outperforms MCBA in most cases and is no worse than MCBA in all cases. The sum of utilities computed by MONTE_K is 17.25% more than that of MCBA in 4 hops sub-networks, 10.89% more in 5 hops, 14.30% more in 6 hops and 34.60% more in 7 hops. This is because *Strategy 2* in MONTE_K guarantees that the solutions identified by later simulations will be no worse than the current one.

Execution Time: From Fig. 4 to Fig. 7, we can observe that the execution time of MONTE_K is significantly less than that of H_MCOP in all sub-networks. The total execution time of MONTE_K is only 5.92% of that of H_MCOP in 4 hops sub-networks, 10.58% in 5 hops, 5.63% in 6 hops and 4.05% in 7 hops. In particular, in the most complex sub-network with 793 nodes, 3411 links and 7 hops (*see the last row of Table 1*), the execution time of MONTE_K is only 2.88% of that of H_MCOP (*see Fig. 7 (d)*). From the above results, we can see that MONTE_K is much more efficient than H_MCOP for identifying the optimal social trust path, especially in larger scale sub-networks. (*see Fig. 4(d), Fig. 5(c, d), Fig. 6(c, d) and Fig. 7(b) to (d)*).

In addition, the execution time of MONTE_K is also less than MCBA. The total execution time of MONTE_K is 91.48% of that of MCBA in 4 hops sub-networks, 87.72% in 5 hops, 86.94% in 6 hops and 78.25% in 7 hops. This is because in MONTE_K, when any QoT constraint of a trust path from the source participant to an intermediate node v can not be satisfied, MONTE_K starts a new simulation, rather than searching the social trust path from v to the target. Thus, the execution time of MONTE_K is less than MCBA in all sub-networks. And the more the number of hops, the less the execution time of MONTE_K than MCBA.

Through the above experiments conducted in sub-networks with different scales and structures, we can see that our proposed approximation algorithm, MONTE_K, addresses the characteristics of online social networks well and thus can deliver better near-optimal solutions with less execution time than existing approximation algorithms.

Conclusion

In this paper, we have presented the complex social network structure, reflecting the real-world situations better. The optimal social trust path selection with QoT constraints in complex social networks is an NP-Complete problem. For solving this challenging problem, we proposed MONTE_K, an approximation algorithm. The results of the experiments conducted on the real dataset of social networks demonstrate that MONTE_K significantly outperforms existing ones in both execution time and optimal social trust path selection.

In our future work, we will develop a new generation CRM system, which maintains a database of both customers and the complex social network containing them. In this system, our proposed method will be applied, for instance, to help a retailer identify new trustworthy customers and introduce products to them.

References

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

Alunkal, B.; Valjkovic, I.; and Laszewski, G. 2003. Reputation-based grid resource selection. In *Proceedings of the Workshop on Adaptive Grid Middleware*, 28–35.

Ashri, R.; Ramchurn, S.; Sabater, J.; Luck, M.; and Jennings, N. 2005. Trust evaluation through relationship analysis. In *AAMAS'05*, 1005–1011.

Baase, S., and Gelder, A. 1999. *Computer Algorithms Introduction to Design and Analysis*. Addision Wesley.

Franken, L. 1996. Quality of service management: A model-based approach. *PhD Thesis, Centre for Telematics and IT.*

Gentle, J.; Hardle, W.; and Mori, Y. 2004. *Handbook of Computational Statistics*. Springer.

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

Goldstein, J.; Kwasinksi, A.; Kingsbury, P.; Sabin, R. E.; and McDowell:, A. 2006. Annotating subsets of the enron email corpus. In *Proceedings of the Third Conference on Email and Anti-Spam*, 37–38.

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

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

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

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

Kuter, U., and Golbeck, J. 2007. Sunny: A new algorithm for trust inference in social networks using probabilistic confidence model. In *AAAI'07*, 1377–1382.

Levinger, G. 1983. Development and change. *Close Relationships* 315–359.

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

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

Liu, G.; Wang, Y.; and Orgun, M. 2009. Trust inference in complex trust-oriented social networks. In *IEEE CSE'09*, 996–1001.

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

Milgram, S. 1967. The small world problem. *Psychology Today* 1(1):61–67.

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

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

Pool, I., and Kochen, M. 1978. Contacts and influence. *Social Networks* 1(1):5–51.

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

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

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

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

Yu, T.; Zhang, Y.; and Lin, K. 2007. Efficient algorithms for web services selection with end-to-end qos constraints. *ACM Transactions on the Web* 1(1):Article 6.