Maximizing Circle of Trust in Online Social Networks

Yilin Shen, Yu-Song Syu, Dung T. Nguyen, My T. Thai
Department of Computer and Information Science and Engineering
University of Florida, USA
{yshen, yssyu, dtnguyen, mythai}@cise.ufl.edu

ABSTRACT

As an imperative channel for fast information propagation, Online Social Networks(OSNs) also have their defects. One of them is the information leakage, i.e., information could be spread via OSNs to the users whom we are not willing to share with. Thus the problem of constructing a circle of trust to share information with as many friends as possible without further spreading it to unwanted targets has become a challenging research topic but still remained open.

Our work is the first attempt to study the *Maximum Circle of Trust* problem seeking to share the information with the maximum expected number of poster's friends such that the information spread to the unwanted targets is brought to its knees. First, we consider a special and more practical case with the two-hop information propagation and a single unwanted target. In this case, we show that this problem is NP-hard, which denies the existence of an exact polynomial-time algorithm. We thus propose a *Fully Polynomial-Time Approximation Scheme* (FPTAS), which can not only adjust any allowable performance error bound but also run in polynomial time with both the input size and allowed error. FPTAS is the best approximation solution one can ever wish for an NP-hard problem. We next consider the number of unwanted targets is bounded and prove that there does not exist an FPTAS in this case. Instead, we design a *Polynomial-Time Approximation Scheme* (PTAS) in which the allowable error can also be controlled. Finally, we consider a general case with many hops information propagation and further show its #P-hardness and propose an effective *Iterative Circle of Trust Detection* (ICTD) algorithm based on a novel greedy function. An extensive experiment on various real-word OSNs has validated the effectiveness of our proposed approximation and ICTD algorithms.

Categories and Subject Descriptors

G.2.2 [**Graph Theory**]: Network problems, Graph algorithms; G.2.1 [**Combinatorics**]: Counting problems

HT'12, June 25–28, 2012, Milwaukee, Wisconsin, USA.
Copyright 2012 ACM 978-1-4503-1335-3/12/06 ...$10.00.

General Terms

Algorithms, Experimentation, Theory

Keywords

Online Social Networks, Circle of Trust, Computational Complexity, Approximation Algorithms

1. INTRODUCTION

The rapid growth of Online Social Networks (OSNs), such as Facebook, Twitters, and LinkedIn, has made them become one of the most important channels for fast information propagation and influence [6, 19]. Many individuals and companies use this popular media to share their messages with other users or advertise their products by leveraging the power of others' influences [6]. However, in spite of its benefits to information propagation, OSNs also have defects as a media to leak information, that is, the information can be spread to the users whom we do not want to share with.

Let us consider the following example in real life. Suppose that Bob is a PhD student and feels very upset with the progress of papers recently, he then decides to have a vacation during a business trip of his advisor Chuck. After coming back, Bob wants to share with his friends the pictures and stories during this vacation in Facebook, yet he is reluctant to let Chuck know about it. Although Chuck is not a friend of Bob in Facebook and he cannot see those pictures directly, Chuck could still see some pictures if they are marked favorite, replied or mentioned by Bob's friend Alice, who is also a friend of Chuck. Thus it raises a practical question: Is there any mechanism for Bob to share his pictures and stories to as many friends as possible without reaching to Chuck?

In an initial attempt to handle this information leakage problem, Facebook [4] developed a new function to customize the privacy for each user when he wants to share some message. In this function, a user can choose a range of friends to share with and also hide the message from some specific users. Later on, Google+ [5] further developed a concept of grouping users into circles such that a user can select a specific circle to share with whenever he starts to share a message. Superficially, the information leakage problem appears to have been overcome in Facebook and Google+ by tracking the message-ID to hide it from those whom he does not want to share with. However, Facebook and Google+ actually neglected an important channel of information propagation, that is, *mentioning* the message. Back to our example, when Bob's friend Alice posts a new message and mentions Bob's pictures and stories, this new message cannot be hidden from Chuck anymore since its ID is no longer the same as the original message from Bob. Consequently, Chuck will still see the message from Alice and know Bob's vacation.

Therefore, in our example, Bob needs to construct a circle of trust, a set of trust friends to share the information with so that the probability that Chuck will know it is very small. Meanwhile, one of the main purposes of posting messages on OSNs is to share the information with as many friends as possible. Thus, we formulate a new optimization problem, called *Maximum Circle of Trust* (MCT), to construct a circle of trust with the maximum number of *visible friends* for a user s so that once s posts a message to this CT, the probability of such friends in this CT spreading the message to *unwanted users* is under some certain threshold, where a friend of s is said to be visible to a message if the message appears on his wall, and the unwanted users are referred as those whom s does not want to share the information with.

According to the discovery by Cha *et al.* [8] that the information can only be propagated within a very limited number of hops and the number of unwanted users is usually very small, we first focus our attention on the bounded-2-MCT problem, a special and more practical case of MCT problem in which the information is propagated within two hops and the number of unwanted users is bounded. Even in this case, we showed that the bounded-2-MCT problem is NP-hard and thus a major thrust is the development of approximation algorithms of which one can theoretically prove the performance bound. In the case of NP-hard problems, the most desirable approximation algorithms are the full polynomial time approximation scheme (FPTAS) and polynomial time approximation scheme (PTAS) which can not only control any allowable errors but also run in polynomial time with the input size (also in polynomial time with error for FPTAS). FPTAS and PTAS are the best one can hope for an NP-hard optimization problem, assuming $P \neq NP$. Unfortunately, the design of such approximation schemes is very challenging and it may not exist for certain problems.

Our contributions are summarized as follows:

- This is the first attempt to study the maximum circle of trust problem tackling the information leakage in online social networks;
- In a special and more practical case of 2-hop information propagation and fixed number of unwanted users, we first prove the NP-hardness of a single unwanted user and then design an FPTAS approximation algorithm based on the idea of scaling and dynamic programming. For multiple unwanted users, we show that there is no FPTAS and thus design a PTAS algorithm, the best solution one can ever wish when the FPTAS does not exist.
- For the general MCT problem, we prove its #P-hardness when the information can be propagated more than 2 hops. Due to its #P-hardness, we design an efficient ICTD algorithm based on a novel greedy function.
- The performance of our proposed approximation and ICTD algorithms are validated on Facebook, Twitter, Foursquare and Flickr datasets.

The rest of this paper is organized as follows. In Section 2, we introduce a novel ISM propagation model and the formal definition of the MCT problem. Section 3 includes the complexity results and approximation algorithms for the bounded-2-MCT problem. For general MCT problem, its #P-hardness and the ICTD algorithm are provided in Section 4. The experimental evaluation is illustrated in Section 5 and related work is presented in Section 6. Section 7 concludes the whole paper.

2. MODEL AND PROBLEM DEFINITION

In this section, we first introduce a novel information leakage propagation model, namely *Independent Sharing-Mention* (ISM) propagation model, in the context of different diffusion channels. Based on this model, we introduce the formal definition of our MCT problem.

2.1 ISM Propagation Model

In our model, we consider two types of information leakage propagations in OSNs between two friends u and v as illustrated in Figure 1:

- *Sharing*: u shares the message using functions provided by OSNs. For example, u can use "retweet" or "reply" to share on Twitter and "share", "comment" or "like" to share on Facebook. In this case, the message will appear on u's own wall and then be seen by v;
- *Mention*: u can also propagate the information to v by mentioning it with the same content (or retyping using his own words).

Correspondingly, between users u and v, we refer to the probability of a sharing and mention propagation (*Sharing Probability* and *Mention Probability*) as a_{uv} and p_{uv}.

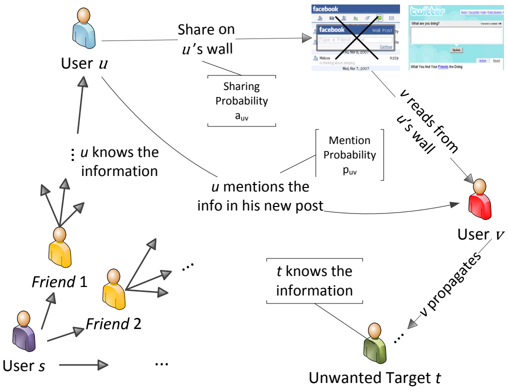


Figure 1: ISM Propagation Model

Therefore, each link (u, v) in OSNs has a two-tuple probabilities $\langle a_{uv}, p_{uv} \rangle$. Although a_{uv} and p_{uv} are not necessarily to be independent, our ISM model is independent in terms of the following two aspects: the independence between different links; the independence among the current propagation, the history propagation, and the future propagation.

As can be seen, this model can reflect the information propagation on a majority of existing OSNs by only choosing different parameters. For example, as Facebook provided a function in *Custom Privacy* to hide a certain information from specific users, which narrows down the sharing probability to 0 on Facebook if the source user blocks all his unwanted targets. Thus, the information propagation on

Facebook only depends on mention probabilities. If we take Twitter as another example, the information propagations are dependent on both sharing probability and mention probability. That is, we can define p_{uv} as an alternative sharing-mention probability instead of mention probability for each link (u, v).

2.2 Problem Definition

In OSNs, when a user s posts some message m, his aim is usually to share it with as many friends as possible, i.e., to maximize the visibility of his message to his friends while preventing it from reaching to some *unwanted targets*. Considering that information can be spread at most δ hops, we study the following *δ-Hop-Propagation Maximum Circle Of Trust* (δ-MCT) problem, which constructs a circle of trust *to maximize the expected visible friends of s* as well as *to restrict the leakage probability of each unwanted target to a certain degree* so that s can safely post his message to this CT.

PROBLEM 1 (δ-MCT PROBLEM). *Given a directed graph $G = (V, E)$ with $|V|$ users and $|E|$ edges underlying an OSN, where each edge (u, v) is associated with a tuple of sharing probability and mention probability $\langle a_{uv}, p_{uv}\rangle$. Let $T = \{t_1, \ldots, t_k\}$ be the set of $k = |T|$ unwanted targets and s be the source user with $|N(s) \setminus T| = S_n$ neighbors. The δ-MCT problem constructs a circle of trust (CT) with the maximum expected visible friends of s (Size of CT) such that the probability of each unwanted target t_i can see the message m posted by s after at most δ hops propagation is at most its leakage threshold τ_j, which lies in $[0, 1)$.*

In our paper, we assume that the source user s is *rational*. That is, he will neither tell the message to his unwanted targets nor share the message in online social networks with them. Then we immediately have the following lemma.

LEMMA 1. *When source user s is rational, all unwanted targets T must be at least two hops from s.*

3. BOUNDED-2-MCT PROBLEM

In this section, we consider the following two special and more practical factors in information propagations:

- The limited propagation hops: According to Cha *et al.* [8], majority of the messages are propagated within 2 hops in OSNs. Moreover, with recent new block functions of many OSNs, the chance of message being leaked by more than 2 hops is very limited.
- Once a user wants to post a message, the number of his unwanted targets is usually very small.

Motivated by these practical observations, we focus on the *Bounded-2-MCT* problem in which the message m can be spread at most 2 hops and the number of unwanted targets is bounded by some constant κ. We further refer to this problem as *Single-2-MCT* when there is only a single unwanted target. In this section, we show the NP-hardness of Single-2-MCT and present an FPTAS approximation algorithm. For multiple unwanted targets $k \geq 2$, we further prove the non-existence of FPTAS algorithms and provide a PTAS approximation algorithm.

We note that FPTAS and PTAS are the most desirable solution for a NP-hard problem by trading accuracy for running time. That is, we can decide how to choose the error parameter based on the allowed time. For example, we can allow more errors when the time is limited and less errors otherwise. In particular, FPTAS is even better since it requires the algorithm to be polynomial in both the problem size and error parameter. Now we first show the following lemma, which can be obtained using contradiction method.

LEMMA 2. *For any user u except the unwanted targets, its propagation can lead to the information leakage if and only if it receives m directly from s when $\delta = 2$.*

PROOF. This is trivial to see by using the contradiction method. Assume that u receives m from some other users rather than s, then message m must have be propagated more than 2 hops from s in order to reach to the unwanted targets, contradicting to the fact that $\delta = 2$. □

3.1 NP-Completeness for Single-2-MCT

THEOREM 1. *Single-2-MCT is NP-complete.*

PROOF. In the proof, it is easy to see that the decision version of Single-2-MCT∈NP. To prove that Single-2-MCT is NP-hard, we reduce the known NP-hard subset sum problem to it, which asks if there exists a non-empty subset whose sum is Z given a set of integers $(z_1, z_2, \ldots, z_n)$ and an integer Z. Let I be an arbitrary instance of subset sum problem, our construction is as illustrated in Fig. 2. We construct two terminal nodes s, t and n nodes N_i, $i = 1 \ldots n$ for each item in I. For each node N_i, we construct an edge from s to it with a sharing probability $a_{si} = \frac{z_i}{\sum_i z_i}$ and mention probability $p_{si} = 1 - e^{-\frac{z_i}{\sum_i z_i}}$; and another edge from N_i to t with mention probability $p_{it} = 1$. We set the leakage threshold $\tau = 1 - e^{-\frac{Z}{\sum_i z_i}}$ for the target t. We show that there is a subset sum of I iff our reduced instance has a Single-2-MCT with the expected visible users at least $\frac{Z}{\sum_i z_i}$.

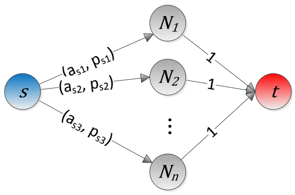


Figure 2: Single-2-MCT Reduction

First, suppose that R is a yes instance of I. Now let us consider a set $R' = \{N_i \mid i \in R\}$. If s posts his message m to R', then the leakage probability to t is exactly $1 - \prod_{i \in R} a_{si} p_{it} = \tau$ and the size of CT is $\sum_{i \in R} \frac{z_i}{\sum_i z_i} = \frac{Z}{\sum_i z_i}$, implying R' is a yes instance of Single-2-MCT.

Conversely, suppose that R' is a Single-2-MCT instance in G with respect to s and t with the leakage probability $\tau = 1 - \prod_{i \in R'} a_{si} p_{it}$, that is, $\sum_{i \in R'} \frac{z_i}{\sum_i z_i} \leq \frac{Z}{\sum_i z_i}$. Then $R = \{i \mid N_i \in R'\}$ is a subset sum of I. This is because the expected visible friends of R' is at least $\sum_{i \in R'} \frac{z_i}{\sum_i z_i} \geq \frac{Z}{\sum_i z_i}$. Thus, $\sum_{i \in R} \frac{z_i}{\sum_i z_i} = \frac{Z}{\sum_i z_i}$. □

3.2 FPTAS Algorithm for Single-2-MCT

Since the NP-hardness denies the existence of any polynomial algorithms for Single-2-MCT problem, we then focus on designing an effective Algorithm to solve it. The analysis shows that this algorithm is an FPTAS, which is the best approximation solution for an NP-hard problem.

According to Lemma 1, we only need to consider the case that t is two hops away from s in $G[E \setminus \{s, T\}]$ since t cannot see m if he is at least 3 hops away from s while $\delta = 2$. In this case, the probability that m will be leaked to t is $1 - \prod_{i \in N(s) \setminus \{t\}} (1 - a_{si} p_{it})^{x_i}$, which is implied by Lemma 2.

The basic idea of FPTAS algorithm with $k = 1$ has two main phases: (1) the scaling of sharing probability; (2) dynamic programming to find the minimum leakage probability w.r.t. the scaled sharing probabilities.

First, since all a_{si} are rational values, we can rewrite each of them with $\frac{an_{si}}{ad_{si}}$ where both an_{si} and ad_{si} are integers. We then define Ad be the least common multiple of all denominators ad_{si}. Thus, $a_{si} = \frac{an_{si} Ad/ad_{si}}{Ad}$, where the numerator is clearly an integer. Then, in the first phase, in order to avoid the case that $an_{si}Ad/ad_{si}$ is exponentially larger than S_n, we scale the sharing probability a_{si} for each s's neighbor by the factor $A = \frac{\varepsilon \max\left\{\frac{an_{si}Ad}{ad_{si}} | a_{si}p_{it} \leq \tau\right\}}{S_n}$ and define its corresponding scaled sharing probability to be $a'_{si} = \lfloor \frac{an_{si}}{A} \rfloor$.

In the second step, we consider using dynamic programming to solve a complex problem by breaking the problem down into simpler subproblems in a recursive manner. That is, to solve the MCT problem w.r.t. the scaled sharing probabilities, we only need to define the recursion function as follows. Let $L_i(a)$ be the minimum leakage probability of a subset of s's first i friends with the circle of trust of size equal to a. Thus, the recursion can be written as

$$L_i(a) = \begin{cases} L_{i-1}(a), & \text{if } a < a_{si} \\ \min\left\{L_{i-1}(a), L_{i-1}(a - a'_{si}) + w_i\right\}, & \text{if } a \geq a_{si} \end{cases} \tag{1}$$

where $w_i = -\log(1 - a_{si}p_{it})$ corresponding to the neighbor i of s. The detail of FPTAS Algorithm is shown in Algorithm 1.

```
Input  : Directed graph $G$ with a tuple of probability
         $\langle a_{uv}, p_{uv} \rangle$ in each edge $(u, v)$, source user $s$,
         unwanted target $t$ and leakage probability $\tau$
Output: Circle of Trust $C$
$a_{si} \leftarrow an_{si}/ad_{si}$ for each $i$;
$Ad \leftarrow$ the least common multiple of all denominators $ad_{si}$;
// Phase 1: Scaling
For some $\varepsilon > 0$, let $A \leftarrow \frac{\varepsilon \max\left\{\frac{an_{si}Ad}{ad_{si}} | a_{si}p_{it} \leq \tau\right\}}{S_n}$;
For each neighbor $i \in N(s)$, define $a'_{si} = \left\lfloor \frac{an_{si}}{A} \right\rfloor$;
// Phase 2: Dynamic Programming
$A_u \leftarrow \sum_{i \in N(S) \setminus \{t\}} a'_{si}$;
$L(a) = A_u + 1$ for all integers $a$ less than $A_u$;
for $a \leftarrow 1$ to $S_n$ do
    Apply dynamic programming to find the CT $C$ using
    the recursion (1) to obtain $C_a$;
end
$C \leftarrow \arg\max_{1 \leq a \leq S_n} \{C_a | L(a) \leq \tau\}$;
return $C$;
```

Algorithm 1: FPTAS for Single-2-MCT

Now, we prove that Algorithm 1 is indeed an FPTAS algorithm, that is, we need to prove the approximation ratio $(1 - \varepsilon)$ and the time complexity is polynomial in both the input size and error parameter.

Lemma 3. *Algorithm 1 is a $(1 - \varepsilon)$-approximation algorithm of Single-2-MCT.*

Proof. Let C^* be the optimal set of CT, π_1^ε be the expected size of CT C obtained by Algorithm 1, and π^* be the the optimal solution of Single-2-MCT. Then, we have

$$\begin{aligned} \pi_1^\varepsilon &= \sum_{i \in C} a_{si} \geq \frac{1}{Ad} \sum_{i \in C} A \left\lfloor \frac{a_{si}Ad}{A} \right\rfloor \geq \frac{1}{Ad} \sum_{i \in C^*} A \left\lfloor \frac{a_{si}Ad}{A} \right\rfloor \\ &\geq \frac{1}{Ad} \sum_{i \in C^*} A \Big(\frac{a_{si}Ad}{A} - 1 \Big) = \sum_{i \in C^*} \Big(a_{si} - \frac{A}{Ad} \Big) \\ &\geq \pi^* - \varepsilon \frac{|C^*| \max\{a_{si} | a_{si}p_{it} \leq \tau\}}{S_n} \geq (1 - \varepsilon)\pi^* \end{aligned}$$

where the last step holds since $\max\{a_{si} | a_{si}p_{it} \leq \tau\} \leq \pi^*$ and $|C^*| \leq S_n$. □

Lemma 4. *Algorithm 1 has the running time of $O(S_n^3/\varepsilon)$.*

Proof. The running time of Algorithm 1 is dependent on the second phase of dynamic programming, which has its running time $O(S_n A_u)$. That is,

$$S_n A_u \leq S_n \cdot S_n \frac{an_{si}}{A} \leq S_n^2 \frac{S_n}{\varepsilon} = \frac{S_n^3}{\varepsilon}$$

The proof is complete. □

The results of Lemma 3 and 4 imply the following theorem:

Theorem 2. *Algorithm 1 is an FPTAS approximation algorithm for Single-2-MCT.*

3.3 No FPTAS for Any $k \geq 2$

As 2-MCT is NP-complete, one will question how tightly we can approximate the solution when $k \geq 2$. In this section, we further investigate that there is no FPTAS approximation algorithm of 2-MCT with any $k \geq 2$.

Theorem 3. *There is no FPTAS for 2-MCT problem with any $k \geq 2$ unless P=NP.*

Proof. We reduce the 2-MCT problem from EQUIPARTITION problem, which asks if there exists a subset of items R satisfying both $|R| = n/2$ and $\sum_{j \in R} \varpi_j = \sum_{j \notin R} \varpi_j$ given n items with integer weight ϖ_j for $j = 1, \ldots, n$ and even n. EQUIPARTITION problem has been proven to be NP-hard in [10]. Let a set of even number of n items with each integer weight ϖ_j be an arbitrary instance I of EQUIPARTITION. We must construct in polynomial time an instance of 2-MCT such that if we have a FPTAS to solve the 2-MCT on this instance, this algorithm can be applied to solve the EQUIPARTITION problem on I in polynomial time.

Our construction is as follows. Given n items, we construct $n + 3$ nodes for graph G: node u_i for each item; a source node s and 2 unwanted targets t_1 and t_2. The mention probability from s to each u_i is 1. For each u_i, the mention probability from him to t_1 and t_2 are $p_{i1} = 1 - e^{-\frac{\varpi_i}{\sum_i \varpi_i}}$ and $p_{i2} = 1 - e^{-\frac{\varpi_{\max} - \varpi_i}{n\varpi_{\max} - \sum_i \varpi_i}}$ respectively. Moreover, we set τ_1, τ_2 to be $1 - e^{-1/2}$ and all sharing probabilities $a_{sN(s)} = 1$. We first show that there is an EQUIPARTITION of I iff our reduced instance has 2-MCT of size at least $n/2$.

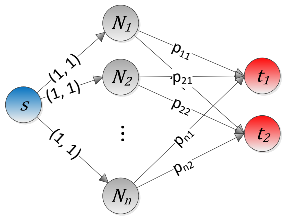


Figure 3: 2-MCT Reduction G from EQUIPARTITION (All edges from s to blue nodes have probability 1; an edge from N_i to t_1 and t_2 has probability $p_{i1} = 1 - e^{-\frac{\varpi_i}{\sum_i \varpi_i}}$ and $p_{i2} = 1 - e^{-\frac{\varpi_{\max} - \varpi_i}{n\varpi_{\max} - \sum_i \varpi_i}}$)

First, suppose that R is a yes instance of I. Clearly, $|R| = n/2$ and $\sum_{j\in R} \varpi_j = \sum_{j \notin R} \varpi_j$. Now let us consider a set $R' = \{N_i | i \in R\}$. If s posts his message m to R', then the leakage probability to t_1 and t_2 are

$$1 - \prod_{i\in R}\Big(1 - p_{i1}\Big) = 1 - \prod_{i\in R} e^{-\frac{\varpi_i}{\sum_i \varpi_i}} = 1 - e^{-1/2}$$

and

$$1 - \prod_{i\in R}\Big(1 - p_{i2}\Big) = 1 - \prod_{i\in R} e^{-\frac{\varpi_{\max} - \varpi_i}{n\varpi_{\max} - \sum_i \varpi_i}} = 1 - e^{-1/2}$$

which are no larger than τ_1 and τ_2. And the circle of trust has its size $n/2$, implying R' is a yes instance of 2-MCT.

Conversely, suppose that R' is a 2-MCT instance in G with respect to s and t. By satisfying

$$1 - \prod_{i\in R'} e^{-\frac{\varpi_i}{\sum_i \varpi_i}} \le 1 - e^{-1/2}$$

and

$$1 - \prod_{i\in R'} e^{-\frac{\varpi_{\max} - \varpi_i}{n\varpi_{\max} - \sum_i \varpi_i}} \le 1 - e^{-1/2}$$

we immediately have $\sum_{i\in R'} \varpi_i \le \frac{1}{2}\sum_{i\in N(s)} \varpi_i$ and $\sum_{i\in R'}\Big(\varpi_{\max} - \varpi_i\Big) \le \frac{1}{2}\Big(n\varpi_{\max} - \sum_{i\in N(s)} \varpi_i\Big)$. After summing these two inequalities up, we obtain $|R'| \le n/2$. Since the size of CT is at least $n/2$, i.e., $|R'| \ge n/2$, we obtain $|R'| = n/2$. Then, substituting $|R'| = n/2$ into the second above inequality, we have $\sum_{i\in R'} \varpi_i \ge \frac{1}{2}\sum_{i\in N(s)} \varpi_i$. Combining with the first one, $\sum_{i\in R'} \varpi_i = \frac{1}{2}\sum_{i\in N(s)} \varpi_i$. Thus, $R = \{i | N_i \in R'\}$ is a EQUIPARTITION of I.

Then, suppose that there is an FPTAS for 2-MCT, we show that this polynomial time algorithm can be applied to solve the NP-complete EQUIPARTITION problem, which leads to the contradiction. Let $\mathcal{A}$ be an FPTAS algorithm generating an $(1 - \varepsilon)$-approximation algorithm for 2-MCT for any $\varepsilon > 0$ in polynomial time with respect to both n and $1/\varepsilon$. When choosing $\varepsilon = \frac{1}{n+1}$, we have the following relations between the solution of $\pi^{\mathcal{A}}$ and optimal solution π^* as

$$\pi^{\mathcal{A}} \ge (1 - \varepsilon)\pi^* > \pi^* - \pi^*/n \ge \pi^* - 1$$

where the last step follows from a trivial observation that $\pi^* \le n$. Due to the equivalence between EQUIPARTITION and 2-MCT in our above reduction, we can obtain a solution $\pi^{\mathcal{A}} > \pi^* - 1$ for EQUIPARTITION. However, the integrality of solution to EQUIPARTITION implies that $\pi^* = \lceil \pi^{\mathcal{A}} \rceil$, which means that $\mathcal{A}$ can solve the EQUIPARTITION problem in polynomial time. This contradicts the fact that EQUIPARTITION is NP-hard. □

3.4 PTAS Algorithm for Bounded-2-MCT

Because of the non-existence of FPTAS for the Bounded-2-MCT problem, we now focus our attention on designing a PTAS solution, which is the best approximation solution we can expect. We first formulate the *Integer Linear Programming* (ILP) formulation for this problem and then propose the PTAS algorithm based on its relaxed LP formulation.

3.4.1 ILP Formulation

First, let us define an indicator variable x_i for each friend $i \in N(s)$ of s as $x_i = 1$ if i is visible to m, and 0 otherwise. Clearly, we have our objective to maximize the circle of trust, i.e., the expected number of visible friends of s. Thus, it can be written as the sum of sharing probabilities of s's friends except unwanted targets, that is, $\max \sum_{i\in N(s)\setminus T} a_{si}x_i$.

According to Lemma 2, which can be easily proven using contradiction method, the message will be leaked to t_j iff an s's neighbor i is informed with probability a_{si} and i further leaks to t_j with probability p_{it_j}. Therefore, the constraint w.r.t. each unwanted target t_j can be written as $1 - \prod_{i\in N(s)\setminus T}(1 - a_{si}p_{it_j})^{x_i} \le \tau_j$. After rearranging and choosing the logarithm of both sides in each constraint and relaxing $x_i \in \{0, 1\}$ to $x_i \ge 0$, we can obtain the following linear programming (LP):

$$\begin{array}{lll} \max & \sum\limits_{i\in N(s)\setminus T} a_{si}x_i & \\ \text{s.t.} & \sum_{i\in N(s)\setminus T} w_{ij}x_i \le c_j, & \forall j \in T \\ & x_i \ge 0 & \end{array} \qquad (2)$$

where $w_{ij} = -\log(1 - a_{si}p_{it_j})$ and $c_j = -\log(1 - \tau_j)$.

3.4.2 PTAS Algorithm for Bounded-2-MCT

Our PTAS algorithm for 2-MCT consists of two phases with respect to a threshold $\beta = \min\{\lceil \frac{k}{\varepsilon} \rceil - (k-1), |N(s)\setminus T|\}$ with k unwanted targets: (1) when the number of visible neighbors of s is less than β, we enumerate the solution and select a feasible solution π which induces a maximum visibility; (2) after initializing the current optimal solution as the one in the first phase, we check each combination of size β. For each combination Ω, we first use the LP rounding algorithm (as shown in Algorithm 3) to obtain a bounded solution π_Ω of the subproblem of 2-MCT in terms of the neighbor set $N(s)' = \{i | a_{si} \le \min i \in \Omega\}$ and $c_j' = c_j - \sum_{i\in\Omega} w_{ij}$. Then, we update the new optimal solution if $\sum_{i\in\Omega} a_{si} + \pi_\Omega > \pi$. The detail of PTAS algorithm is shown as Algorithm 2.

The subroutine of LP rounding algorithm, as shown in Algorithm 3, starts with a basic solution of LP (2) consisting of k fractional x_i^{LP}. Between the sum of a_{si} on integers x_i^{LP} and a_{sj} with the maximum fraction value x_j^{LP}, the algorithm returns the larger value as its solution.

Let π^ε be the expected size of CT C obtained by Algorithm 2, π^k be the expected size of intermediate CT C_I obtained by Algorithm 3, and π^{LP}, π^* be the optimal LP solution and the optimal solution of 2-MCT. We first show that Algorithm 3 has an $1/(k+1)$ approximation guarantee.

LEMMA 5. *Algorithm 3 is a $\frac{1}{k+1}$ approximation algorithm of Bounded-2-MCT.*

```
Input  : Directed graph $G$ with a tuple of probability
         $\langle a_{uv}, p_{uv}\rangle$ in each edge $(u, v)$, source user $s$,
         unwanted target $T$ and leakage probability $\tau_j$ for
         each $t_j \in T$
Output: Circle of Trust $C$
$\beta \leftarrow \min\{\lceil \frac{k}{\varepsilon} \rceil - (k+1), |N(s) \setminus T|\}$;
$w_{ij} \leftarrow -\log(1 - p_{si}p_{it_j})$;
$c_j \leftarrow -\log(1 - \tau_j)$;
// Phase 1
foreach $\Lambda \subset N(s) \setminus T$ such that $|\Lambda| < \beta$ do
    if $\sum_{i \in \Lambda} w_{ij} \leq c_j$ for all $j \in T$ then
        if $\sum_{i \in \Lambda} a_{si} > \pi^{\varepsilon}$ then
            $C \leftarrow \Lambda$;
        end
    end
end
// Phase 2
foreach $\Omega \subset N(s) \setminus T$ such that $|\Omega| = \beta$ do
    if $\sum_{i \in \Omega} w_{ij} \leq c_j$ then
        Obtain the solution $C^k_\Omega$ of the subproblem with
        $N(s)' = \{j | c_j \leq \min\{c_i | i \in \Omega\}\} \setminus \Omega$ and
        $c'_j = \sum_{i \in \Omega} w_{ij}$ using Algorithm 3 ;
        if $\sum_{i \in \Omega \cup C^k_\Omega} a_{si} > \pi^{\varepsilon}$ then
            $C \leftarrow \Omega \cup C^k_\Omega$;
        end
    end
end
return $C$;
```

Algorithm 2: PTAS for Bounded-2-MCT

```
Input  : Directed graph $G$ with a tuple of probability
         $\langle a_{uv}, p_{uv}\rangle$ in each edge $(u, v)$, source user $s$,
         unwanted target $T$ and leakage probability $\tau_j$ for
         each $t_j \in T$
Output: Intermediate Circle of Trust $C_I$
Obtain an optimal basic solution $x^{LP}$ by solving the LP
(2) with $|\{i | 0 < x^{LP}_i < 1\}| \leq k$;
$I \leftarrow \{i | x^{LP}_i = 1\}$;
$F \leftarrow \{i | 0 < x^{LP}_i < 1\}$;
if $\sum_{i \in I} a_{si} > \max\{a_j | j \in F\}$ then
    $C_I \leftarrow I$;
end
else
    $C_I \leftarrow \{j\}$;
end
return $C_I$;
```

Algorithm 3: LP Rounding Algorithm

Proof. According to Luenberger [13], each LP formulation with n variables and d constraints has a basic optimal solution with at most $\min\{d, n\}$ fractional values. We can obtain such a basic optimal solution x^* in the first step. Then

$$\pi^* \leq \pi^{LP} \leq \sum_{i \in I} a_{si} + kF_{\max} \leq (k+1)\pi^k$$

where the last step follows from Algorithm 3. □

Now, we prove that Algorithm 2 is indeed a PTAS algorithm, that is, we need to prove the approximation ratio $(1 - \varepsilon)$ and the time complexity is polynomial in the input size.

Lemma 6. *Algorithm 2 is a $(1-\varepsilon)$-approximation algorithm of Bounded-2-MCT.*

Proof. If π^* has less than β neighbors, we can obtain the optimal solution in the first phase by enumerating all possible combinations. This certainly leads to the optimal solution. When $\pi^* > \beta$, after defining Ω^* to be the β neighbors having the maximum circle of trust in optimal solution, we consider two cases as follows:

Case 1: $\sum_{i \in \Omega^*} a_{si} \geq \frac{\beta}{\beta+k+1}\pi^*$

From the last step and the condition of this case, we have

$$\begin{aligned} \pi^{\varepsilon} &\geq \sum_{i \in \Omega^*} a_{si} + \pi^k_\Omega \geq \sum_{i \in \Omega^*} a_{si} + \frac{1}{k+1}\pi^*_\Omega \text{ (Lemma 5)} \\ &\geq \sum_{i \in \Omega^*} a_{si} + \frac{1}{k+1}\Big(\pi^* - \sum_{i \in \Omega^*} a_{si}\Big) \text{(Definition of } \Omega^*) \\ &\geq \frac{1}{k+1}\pi^* + \frac{k}{k+1}\frac{\beta}{\beta+k+1}\pi^* = \frac{\beta+1}{\beta+k+1}\pi^* \end{aligned}$$

Case 2: $\sum_{i \in \Omega^*} a_{si} < \frac{\beta}{\beta+k+1}\pi^*$

First, among all these β neighbors of s, there is at least one having sharing probability less than $\frac{1}{\beta+k+1}\pi^*$. According to the definition of Ω^*, i.e., all neighbors in Ω^* have higher sharing probability than others, all neighbors in π^k_Ω have $a_{si} \leq \frac{1}{\beta+k+1}\pi^*$.

$$\pi^*_\Omega \leq \pi^{LP}_\Omega \leq \pi^k_\Omega + \frac{k}{\beta+k+1}\pi^*_\Omega$$

where the last step follows from the upper bound of all k fractional values according to Luenberger [13]. Therefore,

$$\pi^* = \sum_{i \in \Omega^*} a_{si} + \pi^*_\Omega \leq \pi^{\varepsilon} + \frac{k}{\beta+k+1}\pi^*_\Omega$$

Then, we have

$$\pi^{\varepsilon} \geq \frac{\beta+1}{\beta+k+1}\pi^* \geq \frac{\lceil \frac{k}{\varepsilon} \rceil - k}{\lceil \frac{k}{\varepsilon} \rceil}\pi^* \geq \frac{\frac{1}{\varepsilon} - 1}{\frac{1}{\varepsilon}}\pi^* = (1-\varepsilon)\pi^*$$

where the second step follows from the fact that $\frac{\beta+1}{\beta+k+1}$ is monotonously increasing with respect to β. □

Lemma 7. *Algorithm 2 has the running time of $O(S_n^{\lceil \kappa/\varepsilon \rceil})$, where constant κ is the upper bound of the number of unwanted targets k.*

Proof. It is easy to see that the first phase has the running time at most $S_n^{\lceil \frac{k}{\varepsilon} \rceil - (k+1)}$. For the second phase, we need to solve LP (2) S_n^β times. According to Megiddo *et al.* [14], LP (2) S_n^β can be solved in $O(S_n)$ when k is upper bounded by some constant κ. Hence, the overall running time of Algorithm 3 is $O(S_n^{\lceil \kappa/\varepsilon \rceil})$. □

The results of Lemma 6 and 7 imply the following theorem:

Theorem 4. *Algorithm 2 is a PTAS approximation algorithm for Bounded-2-MCT.*

4. GENERAL δ-MCT PROBLEM

When the message can be propagated more than 2 hops, i.e., $\delta > 2$, one will be interested to see how hard a general MCT problem is and how to develop an efficient approach to solve it. In this section, we first prove that the MCT problem is #P-hard when $\delta > 2$. Due to its extreme challenge to design a fully polynomial-time randomized approximation scheme (FPRAS) for a #P-hard problem, we propose an effective ICTD algorithm based on a novel greedy function. The performance of our ICTD algorithm is further evaluated in the next section.

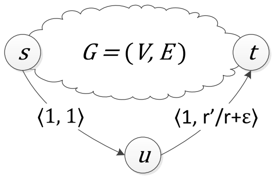
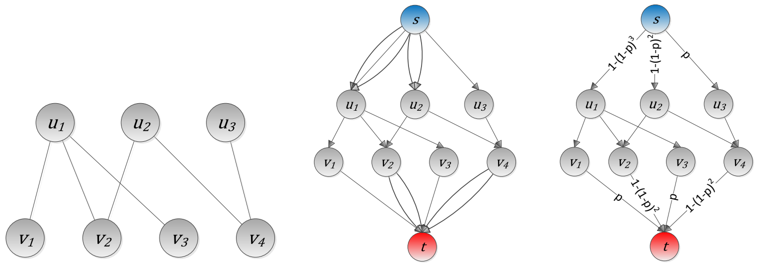


Figure 4: δ-MCT Reduction



(a) An instance G (b) Graph H (c) Reduced G'

Figure 5: An Example of 3-Conn$_2$ Reduction

4.1 #P-Hardness when $\delta \geq 3$

Theorem 5. *δ-MCT problem is #P-hard when $\delta \geq 3$.*

Proof. We will show the reduction from 3-Conn$_2$ problem, which is defined in Definition 1 and proven to be #P-hard in Lemma 8. First of all, we notice that 3-Conn$_2$ problem can be polynomially solved if we can determine that 3-Conn$_2$ $\leq r'/r$ in a graph G for any integer $r' \leq r$. Since each $p(u, v)$ in G is a rational number which can be represented by a numerator and a denominator which are integers, we can define r to be the least common multiple of all the denominators such that a simple binary search from 1 to r can be finished within a polynomial time with respect to the input size.

Therefore, let G, s and t be an arbitrary instance of 3-Conn$_2$, we must construct in a polynomial time a graph $G' = (V', E')$, a source user s and a set of unwanted targets T along with their leakage thresholds τ_j for each of them such that if we have a polynomial-time algorithm to solve the δ-MCT problem on our reduced instance, this algorithm can be applied to determine the upper bound of 3-Conn$_2$ problem on G.

As shown in Figure 4, our construction is as follows. First, we choose $s' = s$ and $T = \{t\}$. Then we set the sharing probability in each edge of G to be $1/|N(s)| + \epsilon_1$ where $0 < \epsilon_1 < \frac{1}{|N(s)|(|N(s)|-1)}$ and $|N(s)|$ is the number of neighbors of s. The mention probability is set to $p(i, j)$ in G for each edge. Then, we add a two-hop disjoint path between s and t onto the graph G with the intermediate node u. Both edges (s, u) and (u, t) have the sharing probability to be 1. And $p_{su} = 1$ and $p_{ut} = r' \leq r + \epsilon_2$ for any integer $r' \leq r$ and $\epsilon_2 < 1/r$. Besides, we set $T = \{t\}$ and its leakage threshold to $r'/r + \epsilon_2$.

Assume that $\mathcal{A}$ is a polynomial algorithm solving δ-MCT problem in our reduced instance. Let's consider two cases:

- If $\mathcal{A}$ returns the circle of trust with size larger than 1, we know all neighbors of s in G except u is visible to the message. That is, the 3-Conn$_2$ in G, s and t is less than or equal to r'/r;
- If $\mathcal{A}$ returns the circle of trust with size equal to 1, that is, $\mathcal{A}$ selected only one neighbor u of s, since the visibility $\left(\frac{1}{|N(s)|} + \epsilon_1\right)(|N(s)| - 1) < 1$ when $\epsilon_1 < \frac{1}{|N(s)|(|N(s)|-1)}$ if only selecting $N - 1$ neighbors of s in G. Clearly, 3-Conn$_2$ in G, s and t is larger than r'/r.

Thus, $\mathcal{A}$ can be used to decide if 3-Conn$_2$ is less than r'/r, implying that our δ-MCT problem is at least as hard as 3-Conn$_2$. □

Definition 1 (3-Conn$_2$). *Given a directed graph G with $|V| = n$ nodes and a probability $p(u, v)$ for each pair of nodes denoting the probability of u being able to connect to v. Let s and t be two terminals in G. 3-Conn$_2$ asks for the probability that there is a path from source s to destination t in G and the path has its length no larger than 3 hops.*

Lemma 8. *3-Conn$_2$ problem is #P-hard.*

Proof. In this proof, we reduce the 3-Conn$_2$ problem from Counting Bipartite Independent Set (CBIS) problem, which asks for the total number of independent sets in a bipartite graph $G = (U, V; E)$. CBIS has been proven to be #P-hard by Provan *et al.* [17]. Let a graph $G = (U, V; E)$ be an arbitrary instance of CBIS. We must construct in polynomial time a probabilistic graph G' and two terminals s, t such that if we have a polynomial-time algorithm to solve the 3-Conn$_2$ problem on the reduced instance, this algorithm can be applied to solve CBIS problem on G.

Our reduction is two phases: First, we construct the probabilistic graph H by adding two terminals s and t onto G. Between s and each $u \in U$, we add $\deg(u)$ number of edges where $\deg(u)$ is the degree of u in G. Similarly, between each $v \in U$ and t, we add $\deg(v)$ number of edges. And all edges in H have probability p with $0 < p < 1$. Secondly, we construct the probabilistic graph G' on H by replacing the multi-edges between each pair of nodes (u, v) with an edge of probability $1 - (1 - p)^\gamma$ where γ is the number of multi-edges between u and v. Note that the paths between s and t in G' are at most 3-hops. This reduction is depicted for an example in Figure 5.

Then we first show that CBIS in G is equivalent to counting the minimum cardinality $s - t$ cutsets in H. It is easy to see that the construction ensures that the $s - t$ cutsets contain at least $|E|$ edges. Also, it is clear that if there are $\deg(u) > 1$ edges between s and u, a minimum cutset includes either all or none of them. In addition, we note that (u, v) must not an edge of G if both (s, u) and (v, t) are included in a minimum cut since we can reduce the size of cutset by simply replacing the multi-edges in (s, u) and (v, t) with the edges incident to u and v in G, which contradicts that the cutset is minimum.

Therefore, suppose that $I_u \cup I_v$ is an independent set in G where $I_u \subseteq U$ and $I_v \subseteq V$, we have the cutset consisting of (s, u) for all $u \in I_u$, $(v, t) \in I_v$ for all $v \in I_v$ and all edges in E not incident to $I_u \cup I_v$. Conversely, suppose C is a minimum cutset in H, according to our above arguments, the endpoints (except s and t) incident to the edges in $C \setminus E$ forms an independent set in G.

Furthermore, it is easy to see that the probability that there is a path from s to t is the same in H and G' since only multi-edges in H are replaced in G' with simple edges with the same probability. According to Charles [9], if the 3-Conn$_2$ between s and t can be determined in G' (also H), this is suffices to obtain the $s-t$ pathset. Thus, the minimum $s-t$ cutsets can be further counted using the pathset, implying that 3-Conn$_2$ is at least as hard as CBIS. □

4.2 ICTD Algorithm

As can be seen in the above proofs, the general MCT problem with $\delta \geq 3$ is related to a set of problems in network reliability [9], which is a long standing open problem. Therefore, it is extremely challenging to design a FPRAS algorithm, which may not even exist. Instead, we propose an effective *Iterative Circle of Trust Detection* (ICTD) algorithm.

The idea of ICTD algorithm is to iteratively eliminate one of s's neighbors until each unwanted target t_j can see m with the leakage probability less than τ_j. Due to the objective of maximizing the circle of trust, we define the greedy function $f(v)$ to maximize

$$\frac{\sum_{t_j \in T, \tau_j(C) > \tau_j} |\tau_j(C) - \tau_j|}{a_{sv} + \sum_{i \in C_v} a_{si}} - \frac{\sum_{t_j \in T, \tau_j(C_v) > \tau_j} |\tau_j(C_v) - \tau_j|}{\sum_{i \in C_v} a_{si}}$$

where $\tau_j(C)$ is the expected probability that unwanted target t_j knows the information when s only chooses the subset C to share, and $C_v = C \setminus \{v\}$. Intuitively, we do not want to remove very close friends of s, whose sharing probabilities with s are relatively high. Therefore, the normalization factor is to ensure that the removed neighbor does not have a high sharing probability to s. In addition, it is not hard to see that this greedy function can reflect the impact on a user to the leakage by calculating the difference between before and after removing him from the circle of trust.

To calculate $\tau_j(C)$ in each iteration, we use the *Monte Carlo Sampling* method due to its #P-hardness according to Theorem 5. In the sampling subroutine, according to the ISM propagation model, the information is propagated via each edge (u, v) with mention probability p_{uv} on G until no newly informed users can be found or the message has been propagated δ hops. Then, in order to seek for the subset of unwanted targets in T knowing the information at the end propagation, we repeat the sampling 20,000 times and obtain average leaking probability for each unwanted target $t_j \in T$. The whole ICTD algorithm, shown as Algorithm 4, terminates until the average probability in sampling is less than τ_j for each unwanted target t_j. Clearly, our ICTD algorithm runs at most constant times of the multiply of the maximum sampling time and S_n^2.

```
Input  : 2-MCT instance
Output: visible friends $\Pi^h$ and size of CT $\pi^h$
$C \leftarrow N(s) \setminus T$;
while $\exists \tau_j(C) \geq \tau_j$ do
    Find $v \in C$ using Monte Carlo Sampling which
    maximizes $f(v)$;
    $C \leftarrow C \setminus \{v\}$;
end
$\Pi^h \leftarrow C$;
$\pi^h \leftarrow \sum_{i \in \Pi_h} a_{si}$;
```

Algorithm 4: ICTD Algorithm

5. EXPERIMENTAL EVALUATION

5.1 Dataset and Metrics

Table 1: Dataset

Dataset	Nodes	Edges	Density	Source
Facebook	63,731	905,565	4.46%	Ref [18]
Twitter	88,484	2,364,322	3.02%	Sampling in [7]
Foursquare	44,832	1,664,402	8.28%	Our data
Flickr	80,513	5,899,882	18.2%	DMML [2]

* Facebook and Flickr are undirected networks; Twitter and Foursquare are directed networks.

We examine the performance of our proposed algorithms on different real-world OSNs, including Facebook, Twitter, Foursquare, and Flickr, with different sizes and density as shown in Table 1. Here we omit the detailed descriptions of Facebook and Flickr datasets, which can be found in the provided references shown in Table 1. For Twitter, we used the unbiased sampling approach [11] to sample a portion of Twitter network from the complete Twitter network, which is provided by Cha *et al.* [7]. And for Foursquare, we initially picked a seed set consisting of entrepreneurs and investors, from whom we used Foursquare API [1] to obtain the users and links within their two-hop neighbors.

For each dataset, we randomly assign sharing probability a_{uv} and propagation probability p_{uv} to each edge respectively, in which both of them lie in the interval $[0, 1]$. Then we evaluate the following metrics on these four datasets according to the application illustrated in Figure **??**:

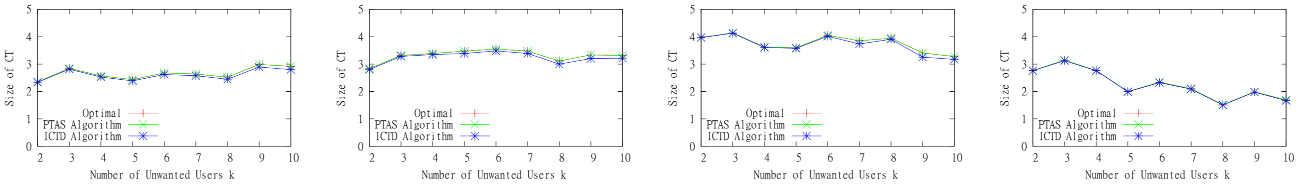
(1) *Size of CT*: defined as $\sum_{j \in CT} a_{sj}$. This is used to measure the expected visible neighbors of s;

(2) *Running Time*: the time a user needs to wait for the construction of CT before he posts a message. An effective solution should construct a CT within a second.

5.2 Performance of Proposed Algorithms

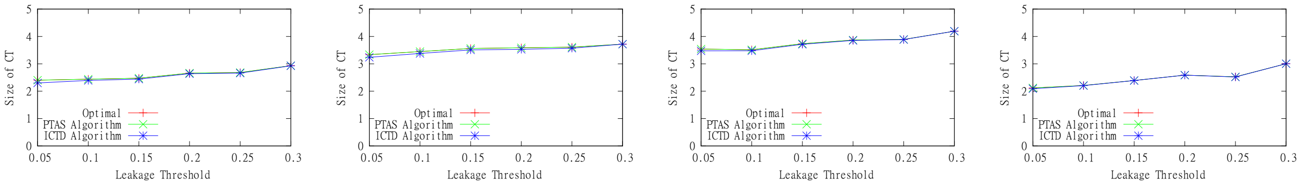
We compare our proposed PTAS algorithm and ICTD algorithm with the optimal solution in the above four datasets. Due to the impracticability to obtain optimal solution for an #P-hard problem, we select $\delta = 2$. In addition, since the leakage threshold is usually small, we set all of them to 0.1, i.e., $\tau_j = 0.1$ for any unwanted target t_j. The allowable error ε for PTAS algorithm is set to 0.01, that is, the result obtained from PTAS should be close to the optimal solution since the approximation ratio is close to 1. In our experiments, we test different numbers of unwanted targets and different leakage thresholds in each dataset. In each network, for a specific number of unwanted targets, we randomly choose a source and unwanted targets and perform the experiments 100 times. The size of CT is then averaged. To obtain the optimal solution, we solve the IP as in Section 3.4.1 using CPLEX optimization suite from ILOG [3].

As revealed in Figure 6, the solution of our PTAS algorithm is almost identical with the optimal solution when the allowable error is small. This empirical result once again claims that Algorithm 2 algorithm is PTAS. Also, in all four datasets, the expected size of CT obtained from our ICTD algorithm is at most 1% less than the optimal solution for different number of unwanted targets. It indicates that ICTD is very effective and our greedy function can actually reflect the influence of information leakage for each user in every



(a) Facebook (b) Twitter (c) Foursquare (d) Flickr

Figure 6: Comparison Among Optimal Solution, PTAS Algorithm and ICTD on k when $\delta = 2$



(a) Facebook (b) Twitter (c) Foursquare (d) Flickr

Figure 7: Comparison Among Optimal Solution, PTAS Algorithm and ICTD on τ when $\delta = 2$

iteration. When we fix the number of unwanted targets to be 5, Figure 7 reports the similar results in terms of different leakage thresholds τ_j from 0.05 to 0.3. In addition, as illustrated in Table 2, the running time of ICTD is very low, thus it is suitable for the design of this application. Even in Flickr, whose density is up to 20%, ICTD algorithm can finish detecting CT within 1 second when $\delta = 2$.

5.3 Findings using ICTD

With the effectiveness of ICTD observed through the above experiments, we confidently use ICTD to further analyze the real-world traces and exploit some insight properties with respect to the securities in OSNs. In our experiments, we perform the following procedure on each dataset: We randomly select 40 source users. For each source user, we further randomly select 5 unwanted targets and deploy the ICTD algorithm to obtain the CT while δ is 2 and 3. We repeat this experiment 100 times and obtain the size of CT by taking the average value. Since the MCT problem is #P-hard when $\delta = 3$, calculating the leakage probability is time consuming. Therefore, we use a relatively larger leakage threshold $\tau_j = 0.15$ here to alleviate the running time of ICTD.

Propagation Hops and Size of CT: Our first observation, as revealed in Figure 8, shows that the higher δ is, the smaller the circle of trust we have. This intuitively agrees with what we expected since the information is more likely to leak to unwanted targets when it can propagate further. In our experiments, the sizes of CT in Facebook and Flickr are 20%-30% lower than the other two since they are undirected on which the information is easier to propagate than on directed networks Twitter and Foursquare. Again note that the size of CT refers to the expected visible friends who can see the message if posted to CT. When $\delta = 3$, the sizes of CT on Facebook and Flickr drop roughly 20%-30%, which is less than the decrease percentages on Twitter and Foursquare, i.e., more than 50%. This can again be explained due to their undirected properties. In addition, Facebook has its size of CT larger than Flickr since it has lower density. Therefore, it is not hard to see that the size of CT is quite sensitive to the information propagation hops.

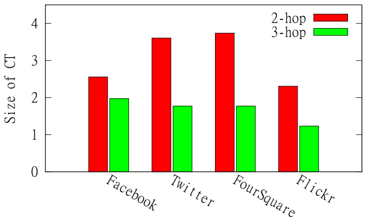


Figure 8: Propagation Hops and Size of CT

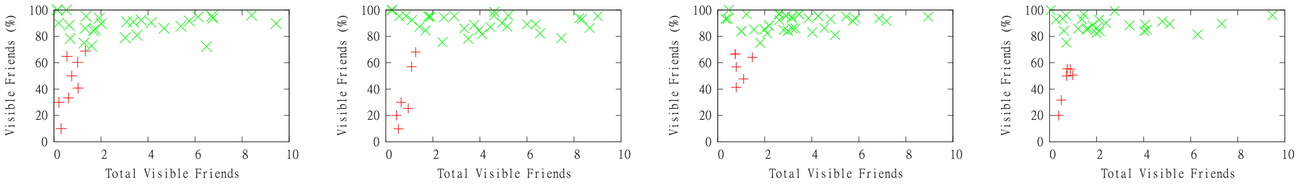
Next, we take a look into the impact of information propagation hops on the running time of ICTD. As is shown in Table 2, although the running time for $\delta = 3$ is about from 20 to 30 times as large as that for $\delta = 2$, the detection of CT can still be finished around 1 second in Facebook, Twitter and Foursquare even when $\delta = 3$. In Flickr, our ICTD algorithm spends 25 seconds to construct a CT, which cannot be avoided due to its #P-hardness as proven in Theorem 5.

Table 2: Time (s) and Propagation Hops

Dataset/Hops	2-Hop	3-Hop	Increase Ratio
Facebook	0.03	0.94	30
Twitter	0.04	1.07	27
FourSquare	0.06	1.32	22
Flickr	0.8	25.2	30

Popular Source Users and Their CTs: Consider popular source users (those who have a lot of friends) and other source users, we now are interested in finding relations between them and the size of their CTs. Intuitively, when a user is more popular, there is a higher chance that his friends will forward the information further. Thus, to avoid information leakage, the CT of popular sources possibly includes only a small fraction of their friends to CT.

To test our hypothesis, we select a set of source users with different degrees in each network. For each source, we choose ten random sets of five unwanted targets and com-



(a) Facebook (b) Twitter (c) Foursquare (d) Flickr

Figure 9: Total Friends and Percentage of Visible Friends

pute the average size of CT. Figure 9 reports the relations between *Total Visible Friends* ($\sum_{j\in N(s)\setminus T} a_{sj}$) and *Visible Friends(%)* ($\frac{\sum_{j\in CT} a_{sj}}{\sum_{j\in N(s)\setminus T} a_{sj}}$). As we can see, the tested users are differentiated into two groups for each network. The red users have relatively smaller total friends and smaller percentage of CT. These users usually have fewer friends but some are gossipy. The elimination of each friends helps to reduce the leakage probability and size of CT as well. The other set of users, green users, usually have various kinds of friends such that their percentage of CT is always larger than 70% no matter whether or not they are popular.

6. RELATED WORK

This work is the first attempt to address the smartly sharing information in OSNs without leaking them to unwanted targets, thus there is not many related work. The most relevant works are the set of papers studied on the privacy issues in OSNs [12, 15, 16]. Lam *et al.* [12] showed that, in current OSNs, no matter how much efforts a user puts to protect his personal information, it cannot be prevented from being revealed by some malicious users by examining their "public" interactions with friends. Later on, for the sake of such unintentional information spreading, Ngoc *et al.* [15] then presented a new metric to quantify the privacy. Noting the potential risks by disclosing information to OSN companies, Nigusse *et al.* [16] proposed an *information flow model*, which made the existing privacy techniques more practical. However, these studies only focus on the users' personal profile, i.e., name, address, etc., but not on the information sharing and posting. In addition, they neglected the information leakage led by multi-hops diffusions.

7. CONCLUSION

In this paper, we study the optimization problem of constructing circle of trust to maximize the expected visible friends such that the probability of information leakage is reduced to some degree. In a special and more practical case of 2-hop information propagation and fixed number of unwanted targets, we prove the NP-hardness and design an FPTAS approximation algorithm for one unwanted target. Then we show the non-existence of FPTAS and design a PTAS approximation algorithm for multiple unwanted targets. In a general case, we further show its #P-hardness and provide the ICTD algorithm using a novel greedy function. The experiments on real-world datasets not only show the effectiveness of our proposed algorithms but also reveal the relations of information leakage with propagation hops and popular source users, which illustrates some crucial characteristics of OSNs that one may pay attention when investigating many security related problems in OSNs, especially the study of information leakage and tracing the misbehaving users.

8. REFERENCES

[1] https://developer.foursquare.com.

[2] http://socialcomputing.asu.edu/datasets/flickr.

[3] http://www-01.ibm.com/software/integration /optimization/cplex-optimizer/.

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

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

[6] http://www.marketingcharts.com.

[7] M. Cha, H. Haddadi, F. Benevenuto, and K. P. Gummadi. Measuring User Influence in Twitter: The Million Follower Fallacy. In *Proceedings of ICWSM '10*, May 2010.

[8] M. Cha, A. Mislove, and K. P. Gummadi. A measurement-driven analysis of information propagation in the flickr social network. In *Proceedings of WWW '09*, pages 721–730, New York, NY, USA, 2009. ACM.

[9] C. J. Colbourn. *The Combinatorics of Network Reliability*. Oxford University Press, Inc., New York, NY, USA, 1987.

[10] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness (Series of Books in the Mathematical Sciences)*. W. H. Freeman, first edition edition, Jan. 1979.

[11] M. Gjoka, M. Kurant, C. T. Butts, and A. Markopoulou. Walking in Facebook: A case study of unbiased sampling of OSNs. In *Proceedings of IEEE INFOCOM 2010*, pages 1–9. IEEE, Mar. 2010.

[12] I.-F. Lam, K.-T. Chen, and L.-J. Chen. Involuntary information leakage in social network services. In *Proceedings of IWSEC '08*, pages 167–183, Berlin, Heidelberg, 2008. Springer-Verlag.

[13] D. G. Luenberger. *Linear and Nonlinear Programming, Second Edition*. Springer, 2nd edition, Sept. 2003.

[14] N. Megiddo and A. Tamir. Linear time algorithms for some separable quadratic programming problems. *Operations Research Letters*, 13:203–211, 1993.

[15] T. H. Ngoc, I. Echizen, K. Komei, and H. Yoshiura. New approach to quantification of privacy on social network sites. In *Proceedings of AINA '10*, pages 556–564, Washington, DC, USA, 2010.

[16] G. Nigusse and B. D. Decker. Privacy codes of practice for the social web: The analysis of existing privacy codes and emerging social-centric privacy risks. In *AAAI Spring Symposium Series*, 2010.

[17] S. J. Provan and M. O. Ball. The Complexity of Counting Cuts and of Computing the Probability that a Graph is Connected. *SIAM Journal on Computing*, 12(4):777–788, 1983.

[18] B. Viswanath, A. Mislove, M. Cha, and K. P. Gummadi. On the Evolution of User Interaction in Facebook. In *Proceedings of WOSN'09*, Aug. 2009.

[19] S. T. Walters and C. Neighbors. Feedback interventions for college alcohol misuse: what, why and for whom? *Addict Behav*, 30(6):1168–82, 2005.