Minimizing the Spread of Misinformation on Online Social Networks with Time and Budget Constraint

Manh M.Vu
University of Technology and Engineering
Vietnam National University, Hanoi, Vietnam
Email: vuminhmanh@gmail.com

Huan X.Hoang
University of Technology and Engineering
Vietnam National University, Hanoi, Vietnam
Email: huanhx@vnu.edu.vn

***Abstract*—In this article, we propose a linear threshold model to the problem of minimizing the spread of misinformation for cases where partial knowledge of misinformation sources is available by using monitors, the misinformation propagation time and the budget for placing monitors are restricted, and proved it is NP-Hardness. At the same time, we also suggest two greedy algorithms to solve the problem. Experimental results show the dominant advantages of the algorithms in comparison with other commonly used algorithms.**

I. Introduction

Information propagation models are the bases for researching how to limit the effects of misinformation. There are many different types of proposed diffusion models such as: threshold models [2], [3], cascading models [4], epidemic models [5], competitive influence diffusion models [6], especially the two models independent cascading (IC) model and linear threshold (LT) model proposed by Kempe [2] are being wildly utilized [1], [7], [8], [9], [10], [15].

Recently, there have been various approaches to the research of misinformation on online social networks (OSNs). Qazvinian [11] and Kwwon [12] identified misinformation and rumors. T.D. Nguyen [7] researched the problem of identifying the k most suspected users from the set of victims who are already influenced by the misinformation. To restrain the misinformation propagation, some authors suggested choosing a few initial nodes to inject good information and spread it on the same network to convince other users [13], [14], [15], where different many information propagation models such as: multi-campaign independent cascade model [13], competitive activation model [14], independent cascading model and linear threshold model [15] are used.

In particular, Zhang [1] suggested a research on determining the smallest set of nodes for placing monitors with the given misinformation emitting sources and the vulnerable r node needing protection so the probability of misinformation activating the r node is within the allowed threshold. In here, the authors used IC model and the nodes on which monitors are placed could obstruct misinformation emitted from them with the same monitor expenses on each node.

In this article, we used the LT model to examine the problem of foreknowing the source of misinformation S and the cost $c(u)$ to place monitors on each respective node u. The node u when activated with misinformation causes a damage $r(u)$, find the set of nodes I to place monitors so that the total installation cost does not exceed the given budget B with the expectation that the number of nodes activated with misinformation after d spread steps is minimal. After proving our problem is NP-Hardness, we introduce two greedy algorithms for this problem. Experimental results show that the suggested greedy algorithms are more effective than the two baseline algorithms Max Degree and Random.

The rest of this article is organized as follows. A summary of linear threshold model and a presentation using the equivalent live-arc graph model are displayed following the Introduction. Section 3 defines the problem of minimizing the spread of misinformation (MSM) and assesses its level of hardness. In Section 4, we present the two greedy algorithms to solve the MSM problem. The result of the experiment is given in Section 5. The last part concerns itself with conclusion and future work.

II. Diffusion Model

Foremost, we would like to give a summary introduction on the spread of information through the linear threshold model, for more details see [2], [16].

A. Linear threshold model

We model a social network as a directed graph $G = (V, E)$, where V is a finite set of vertices or nodes and $E \subseteq V \times V$ is the set of arcs or directed edges connecting pairs of nodes, $|V| = n, |E| = m$. A node represents an individual in the social network, while an arc $e = (u, v)$ in E represents the relationship between individuals u and v respectively. For each node $v \in V$, we denote by $N^{in}(v)$ the set of in-neighbors of v.

Active node and inactive node. The process of spreading information from the source set S to other nodes on OSNs develops through discrete time steps $t = 0, 1, 2, ...$ Each node $v \in V$ has two possible states, active and inactive. In each time step t, node v is active if v is a misinformation-emitting node in S or v receives misinformation from other active neighbors, really accepts it and continues to share and propagate it to other nodes; otherwise, node v is inactive. Denote by $S_t \subset V$ the set of active nodes at time t, obviously $S_0 = S$.

Influence weight. Every arc $(u, v) \in E$ is associated with an influence weight $w(u, v) \in [0, 1]$, indicating the influence of information from user u on user v. The weights are normalized such that for all v satisfies:

$$\sum_{u \in N^{in}(v)} w(u,v) \leq 1 \quad (1)$$

Threshold value. Depending on the characteristics of the corresponding users, every node $v \in V$ has a threshold θ_v demonstrating the node v that will be activated by active neighbors and thus becomes active. The threshold values θ_v are assigned uniformly at random from the interval $[0, 1]$ and are updated during the spread process reflects our lack of knowledge of the individuals' internal thresholds. Therefore, this model belongs to the stochastic diffusion model.

Linear threshold propagation. At each time step t, every node v in inactive state will be activated if the weighted sum of its active neighbors exceeds its threshold θ_v, meaning:

$$\sum_{u \in N^{in}_{active}(v)} w(u,v) \geq \theta_v \quad (2)$$

where $N^{in}_{active}(v)$ is the set of active in-neighbors of v.

If the initial information source set S is foreknown, the process of information propagation follows discrete time steps as follows:

- At step $t = 0$, we have: the active set is the source set $S_0 = S$.
- At step $t \geq 1$, every inactivate node v will be activated if its weighted sum of its active in-neighbors exceeds θ_v, meaning Eq. 2 is satisfied. Once a node is activated, it stays active in the next propagation steps.
- The diffusion process ends when no extra node is activated.

B. Model equivalence

Chen [16] showed that the LT model is equivalent to live-arc graph model with proportional arc selection, which means the active sets $\{S_t\}_{t=1}^{T}$ are the same in both models. Later on, we use live-arc graph model to analyze the problem.

In the LT model, given a social graph $G = (V, E)$ and the influence weights $w(.)$ on all arcs, we select a random live-arc graph $G_L = (V, E_{G_L})$ such that for each node $v \in V$, at most one incoming arc of v is selected with probability $p(u,v) = w(u,v)$, and no arc is selected as a live arc with probability $1 - \sum_{u \in N^{in}(v)} p(u,v)$. The selection of the incoming arc of v is independent of the selection of incoming arcs of other nodes and is called live-arc, the rest is called blocked-arc. Therefore, G_L is a graph with node set V and arc set containing live-arc.

Graph G_L selection probability. Let $\mathcal{G}$ denote the set of all possible live-arc graphs of $G = (V, E)$, and let $Pr(G_L)$ denote the probability that G_L is selected from among all live-arc graphs in $\mathcal{G}$, we have:

$$Pr(G_L) = \prod_{v \in V} p(v) \quad (3)$$

where

$$p(v) = \begin{cases} p(u,v) = w(u,v) & \text{If } \exists u : (u,v) \in E_{G_L} \\ 1 - \sum_{u \in N^{in}(v)} p(u,v) & \text{Otherwise} \end{cases}$$

Estimating the number of activated nodes. Denote by $\sigma(S)$ the expected number of activated nodes caused by misinformation source S once the propagation process ends and $R(G_L, S)$ is the reachable set of nodes from set S in live-arc graph G_L. $\sigma(S)$ can be determined by the equation:

$$\sigma(S) = \sum_{G_L \in \mathcal{G}} Pr(G_L)|R(G_L, S)| \quad (4)$$

where $|X|$ is the number of elements of set X.

Damage caused by misinformation source. When a node v is activated by misinformation source S, meaning the correspondent user believes in this information will cause damages quantified by $r(u) \geq 0$. Since it is hard to estimate the exact amount of damages for each node; in this problem we assume the amount of damage of all nodes are the same. Denoted by $\mathcal{D}(S)$ is the expected amount of damages integrated from active nodes caused by the misinformation source set S throughout the propagation process. Therefore, $\mathcal{D}(S)$ is proportional to $\sigma(S)$ short as damages caused by source S. Without loss of generalization, assume $r(u) = 1$ for all active nodes u, then $\mathcal{D}(S)$ coincides with $\sigma(S)$, meaning:

$$\mathcal{D}(S) = \sigma(S) = \sum_{G_L \in \mathcal{G}} Pr(G_L)|R(G_L, S)| \quad (5)$$

However, subsequently we still use the term *damage* to refer to both of these quantities.

III. PROBLEM STATEMENT AND HARDNESS

A. Problem statement

Consider an OSN presented with a directed graph $G = (V, E)$ with LT model as above and the original misinformation source S. Here, we are concerned with the problem of preventing misinformation from spreading in d time steps (deadline constraint), if we do not prevent this early then the number of activated users will rise rapidly due to the rapid speed of the spread. On the other hand, in various situations arises the problem of preventing the spread before a specific point time. For example, before major political events, opposing organizations and individuals usually disperse misinformation on social networks with the intention of baffling those events. Therefore, it is compulsory prevent that information from spreading throughout the social networks with the view to assuring the success of those events.

Denote by $R_d(G_L, S)$ the set of reachable nodes from S in graph G_L, after d spread steps or d time steps. The shortest distance of all paths from node set S to a node v in graph G_L is denoted by $d_{G_L}(S, v)$ (if no such path exists, we define $d_{G_L}(S, v) = \infty$, and if $v \in S$ then $d_{G_L}(S, v) = 0$), we have:

$$R_d(G_L, S) = \{v \in V \mid d_{G_L}(S, v) \leq d\} \quad (6)$$

Then, from Eq. 5 we determine the damage $\mathcal{D}_d^S$ caused by misinformation source S after d spread steps using Eq. 7:

$$\mathcal{D}_d^S = \sum_{G_L \in \mathcal{G}} Pr(G_L)|R_d(G_L, S)| \quad (7)$$

Suppose, we cannot interfere with source S but we can place monitors on other nodes to limit misinformation propagation. The monitors suggested by Zhang [1] are used to prevent misinformation from spreading from S to the given nodes needing protecting.

Monitor. Monitor is a content-filter system that detects misinformation from users (nodes) placed to prevent the share and propagation of misinformation from this users. The placing of monitor on node v is equivalent to removing the node and its adjacent edges from graph G and requires $c(u)$ cost respectively.

We consider the problem of finding the node set I to place monitors such that the installation budget does not exceed the given budget B with minimal damages after d spread steps of propagating misinformation.

Let $G(I)$ be the sub-graph of G after removing the node set I and the set of it adjacent edges. Then, the damages caused by source S on graph G after placing monitors on the node set I are equal to those caused by source S on graph $G(I)$.

Denote by $\mathcal{G}(\mathcal{I})$ the set of all possible live-arc graphs of $G(I)$ and call $\mathcal{D}_d^S(I)$ the damage function caused by source S after d spread steps when the monitors are placed on node set I. Then, from Eq. 7 we have:

$$\mathcal{D}_d^S(I) = \sum_{G_L \in \mathcal{G}(\mathcal{I})} Pr(G_L)|R_d(G_L, S)| \quad (8)$$

With the process misinformation propagation by the LT model, the problem of Minimizing the Spread of Misinformation on OSNs is stated as follows.

Definition 1 (Minimizing the Spread of Misinformation - MSM). *Given a social graph $G = (V, E)$ under the LT model. $S \subset V$ is the source of misinformation. Each node $u \in V$ has a cost $c(u) \geq 0$ to place the monitors and damage $r(u) = 1$ when activated by misinformation, with limited budget $B > 0$ and the given number of misinformation spread steps $d \in \mathbb{Z}_+$, the objective of the problem is to find the node set $I \subset V \setminus S$ to place monitors with the total cost not exceeding budget B, $\sum_{u \in I} c(u) \leq B$ to minimize the function $\mathcal{D}_d^S(I)$.*

The MSM problem can be shortened as: Find set $I \subset V \setminus S$ to minimize the function $\mathcal{D}_d^S(I)$ with the condition that $\sum_{u \in I} c(u) \leq B$.

B. Hardness

In this part, we show that the problem MSM is NP-Hardness by reducing it from the decision version of Set Cover problem defined as follows.

The decision version of Set Cover problem. Given a universe $\mathcal{U}$ of m elements, $\mathcal{U} = \{e_1, e_2, ..., e_m\}$, and a collection $\mathcal{S}$ of n subsets of the universe set, $\mathcal{S} = \{S_1, S_2, ..., S_n\}$, such that $\bigcup_i S_i = \mathcal{U}$. Given a positive integer $k \leq n$, the question is there a collection of at most k of these subsets whose union equals $\mathcal{U}$.

Theorem 1. *The MSM problem is NP-Hardness.*

Proof. To prove that the MSM problem is NP-Hardness, first we construct a reduction from a known NP-Complete problem which is decision version of Set Cover problem. Next, we prove the equivalence between the instances of MSM problem and decision version of Set Cover problem.

Consider the decision version of MSM problem: Given a social graph $G = (V, E)$ under the LT model. $S \subset V$ is the source of misinformation. Each node $u \in V$ has a cost $c(u) \geq 0$ to place the monitors and damage $r(u) = 1$ when activated by misinformation, with limited budget $B > 0$ and the given number of misinformation spread steps $d \in \mathbb{Z}_+$, there exists or not node set $I \subset V \setminus S$ to place monitors with $\sum_{u \in I} c(u) \leq B$, such that $\mathcal{D}_d^S(I) \leq t$, where t is a positive integer.

Reduction. Given an instance $\mathcal{I}_{SC} = \{\mathcal{U}, \mathcal{S}, k\}$ of the decision version of Set Cover problem, we construct an instance $\mathcal{I}_{MSM} = \{G, w(u, v), \theta_v, S, c(u), r(u), d, B, t\}$ of the MSM problem as follows.

- For each set $S_i \in \mathcal{S}$, we construct a source node of misinformation $s_i \in S$ and a node u_i, connecting these two nodes with a directed edge (s_i, u_i) with weight $w(s_i, u_i) = 1$, activated threshold $\theta_{u_i} = 1$.
- For each element $e_j \in \mathcal{U}$, we construct a node v_j. If $e_j \in S_i$, we build a directed edge of (u_i, v_j) with an influence weight $w(u_i, v_j) = \frac{1}{d_+(v_j)}$, where $d_+(v_j)$ is the in-degree of v_j, the activated threshold $\theta_{v_j} = 1$.
- Damage done on each node activated by misinformation is $r(u_i) = r(v_j) = 1$ $(i = 1..n, j = 1..m)$.
- The cost of placing monitors on each node $c(u_1) = c(u_2) = ... = c(u_n) = 1, c(v_1) = c(v_2) = ... = c(v_m) = +\infty$

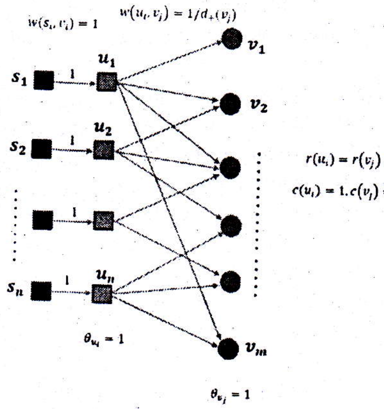


Fig. 1: Reduction from the decision version of Set Cover problem to the MSM problem

The reduction is illustrated in Fig. 1. Finally, put $B = k$, $d = 2$, $t = n - k$. It is easy to see that the reduction is implemented in polynomial time of n.

We now will prove the two instances $\mathcal{I}_{SC} = \{\mathcal{U}, \mathcal{S}, k\}$ and $\mathcal{I}_{MSM} = \{G, w(u, v), \theta_v, S, c(u), r(u), d, B, t\}$ are equivalent to each other.

Suppose that the instance $\mathcal{I}_{SC}$ has the solution $\mathcal{S}'$ with $|\mathcal{S}'| \leq k$ and it can cover all the elements of $\mathcal{U}$. By the above-mentioned reduction, we choose the node set $I = \{u_i \mid S_i \in \mathcal{S}'\}$ to place the monitors. Then, we have $\sum_{u_i \in I} c(u_i) \leq k = B$. Because $\mathcal{S}'$ covers all the elements of $\mathcal{U}$, all nodes v_j $(j = 1..m)$ are adjacent to at least one node $u_i \in I$. Thus, when placing a monitor on the set I we have $\sum_{u_i \in N^{in}_{active}(v_j)} w(u_i, v_j) < 1 = \theta_{v_j}$, therefore no node v_j is activated. The nodes $u_i \notin I$ and are directly adjacent to the nodes in S will be activated, so we have $\mathcal{D}^S_d(I) = n - k$.

Conversely, suppose the instance $\mathcal{I}_{MSM}$ has the solution of the node set for placing monitors $I \subset V \backslash S$ with $\sum_{u \in I} c(u) = B = k$ such that $\mathcal{D}^S_d(I) \leq n - k$. By the above-mentioned reduction, we determine the set $\mathcal{S}' = \{S_i \mid u_i \in I\}$. Then, we have $|\mathcal{S}'| = k$. Because $c(v_j) = +\infty > B$ $(j = 1..m)$, then it is impossible to place monitors on nodes v_j $(j = 1..m)$. Thus the set I can only consist of u_i $(i = 1..n)$. Besides, the fact that $\mathcal{D}^S_d(I) \leq n - k$ proves the nodes v_j $(j = 1..m)$ cannot be activated by misinformation. Thence, each node v_j is adjacent to at least one node $u_i \in I$. In other words, the set $\mathcal{S}'$ covers the entire elements of set $\mathcal{U}$. □

IV. THE SUGGESTED ALGORITHMS

Regarding the problem of information propagation, ones can use the baseline algorithms Max Degree and Random to find a good enough solution. These two baseline algorithms are commonly used to evaluate the effectiveness in comparison with newly suggested algorithms [1], [2], [8], [15].

Denote by $N_k(S)$ the set of nodes with distance no greater than k starting from the misinformation source set S in graph G. When $k = 1$, $N_k(S)$ is the out-neighbors of S. To prevent the propagation of misinformation after d time steps then the nodes chosen to place monitors need to be in set $N_d(S)$ as well.

In this article, we propose two greedy algorithms for the MSM problem, the first one is based on the characteristics of the function $f(I)$ (given by Eq. 9) measuring the damage reduction level after determining the node set I for placing monitors; the second algorithm use the function of $\alpha(v)$ (given by Eq. 10) measuring the increment of $f(I)$ over a cost unit when adding a new node v to set I.

Damage-reducing function. For each set $I \subset N_d(S)$, we define the damage-reducing function $f(I)$ as:

$$f(I) = \mathcal{D}^S_d(\emptyset) - \mathcal{D}^S_d(I) = \mathcal{D}^S_d - \mathcal{D}^S_d(I) \qquad (9)$$

implicitly, $\mathcal{D}^S_d(\emptyset) = \mathcal{D}^S_d$.

$f(I)$'s value-increasing function over one cost unit. With each given set I, function $\alpha(v)$ measures the increment of $f(I)$ over a cost unit when adding a new node $v \in N_d(S)$ to set I, determined by:

$$\alpha(v) = \frac{(f(I \cup \{v\}) - f(I))}{c(v)} \qquad (10)$$

A. Greedy algorithm based on damage-reducing function

The goal of the MSM problem is to minimize misinformation propagation, which means minimizing the value of the function $\mathcal{D}^S_d(I)$, equivalently, maximizing damage reduction, or, maximizing $f(I)$. Therefore, we can use the $f(I)$ as a substituting objective function in the MSM problem. This algorithm works on gradually supplementing the set I using the greedy method.

Idea of the algorithm. Initialize $I = \emptyset$, then repeat the act of choosing the node $v \in N_d(S)$ so that the function $f(I \cup \{v\})$ is maximal. If the current total cost of monitor placing does not exceed budget threshold B then add v to I, on the contrary, stop the algorithm and return the set I as the result. This process ends when the total cost of monitor placing for the set I exceeds the given budget B or when all the elements in $N_d(S)$ have been checked. The details of the algorithm are shown in Algorithm 1 pseudo-code.

Algorithm 1: Greedy algorithm based on function $f(I)$

```
Input : G = (V, E), w(u, v), d, B, the misinformation source set S.
Output: Nodes set I is the solution to the MSM problem.
begin
    I ← ∅;
    N ← N_d(S);
    C ← 0;
    while (C < B) and (N ≠ ∅) do
        u ← argmax_{v∈N} f(I ∪ {v}); //Use Eq. 9 to calculate f(I)
        if C + c(u) ≤ B then
            I ← I ∪ {u};
            C ← C + c(u);
        N ← N \ {u};
    Return I;
end
```

In the worst case, Algorithm 1 executes maximally n_1^2 iterations of calculating the value of the function $f(I)$, where $n_1 = |N_d(S)|$. However, in order to calculate the value of $f(I)$, we need to calculate the expected number of nodes activated by misinformation after d spread steps. Computing exactly the expected number of nodes activated by misinformation is #P-Hard [16], [17]. To resolve this problem, Wei Chen [16], [17] used the Monte Carlo simulations of the diffusion process to estimate the expected number of activated nodes. The estimation of the value of $\mathcal{D}^S_d(I)$ using the Monte Carlo method is presented in Algorithm 2.

Algorithm 2: The algorithm to estimate the value of the function $\mathcal{D}^S_d(I)$

```
Input : G = (V, E), w(u, v), the misinformation source set S, the node set I
        for placing monitors.
Output: The estimated value of the function D^S_d(I).
begin
    Graph G(I) obtained after removing the node set I from graph G;
    count ← 0;
    for j = 1 to R do
        Simulating the misinformation propagation process from the source set
        S on graph G(I);
        n_a ← the number of activated nodes after d spread steps;
        count ← count + n_a;
    Return count/R;
end
```

Given seed set S, we can simulate the randomized diffusion process with seed set S for R times. Each time, we count the number of activated nodes after d spread steps, and then take the average of these counts over the R times. We can increase R to get arbitrarily high accuracy in our estimate of $\mathcal{D}_d^S(I)$.

Hence, in the worst case, Algorithm 1 runs in time $\mathcal{O}(n_1^2 R)$, with $n_1 = |N_d(S)|$ and R is the number of simulations.

B. Greedy algorithm based on the $f(I)'s$ value-increasing function

Previously, Algorithm 1 is based on the idea of picking out the nodes yielding maximum damage reduction to add to the node set needing to be placed with monitors. However, in this section, we propose another algorithm based on picking out the nodes yielding maximum damage reduction but the cost expensed is considered.

Idea of the algorithm. Initialize $I = \emptyset$, then repeat the act of choosing the node $v \in N_d(S)$ so that the function $\alpha(v)$ is maximal. If the current total cost of monitor placing does not exceed budget threshold B then add v to I, on the contrary, stop the algorithm and return the set I as the result. This process ends when the total cost of monitor placing for the set I exceeds the given budget B or when all the elements in $N_d(S)$ have been checked. The details of the algorithm are shown in Algorithm 3 pseudo-code.

In the worst case, Algorithm 3 runs in time $\mathcal{O}(n_1^2 R)$, with $n_1 = |N_d(S)|$ and R is the number of simulations.

Algorithm 3: Greedy algorithm based on function $\alpha(v)$

```
Input  : G = (V, E), w(u, v), d, B, the misinformation source set S.
Output: Nodes set I is the solution to the MSM problem.
begin
    I ← ∅;
    N ← N_d(S);
    C ← 0;
    while (C < B) and (N ≠ ∅) do
        α(v) = (f(I ∪ {v}) − f(I)) / c(v), ∀v ∈ N; //Use Eq. 10 to calculate α(v)
        u ← argmax_{v∈N} α(v);
        if C + c(u) ≤ B then
            I ← I ∪ {u};
            C ← C + c(u);
        N ← N \ {u};
    Return I;
end
```

V. Experiment and Evaluation

To evaluate the effectiveness of the two proposed algorithms, we conducted experiments on real-world networks and compared with the baseline algorithms Max Degree and Random. These two baseline algorithms are commonly used in experiments to compare with suggested algorithms [1], [2], [8], [15].

Max Degree Algorithm is performed by selecting the highest-degree nodes to place monitors while the Random algorithm is performed by randomly selecting nodes to place monitors.

Comparing the effectiveness in decreasing the number of nodes activated by misinformation with variable budget B, $B = \{10, 25, 35, 50, 70, 110\}$ and the variable size of the set S, $|S| = \{5, 10, 15, 20, 25\}$.

The influence weight $w(u, v)$ in the LT model is setup as follows: Each edge incoming the node v has the influence weight of $1/d(v)$ where $d(v)$ is the in-degree of v. This means all edges make the same contribution to activating the node v and the sum of influence weights of the edges incoming the node v is 1. The cost of monitor placing on each node is randomly chosen in $[1.0, 3.0]$. Moreover, in all algorithms that use the Monter-Carlo method, the number of simulations is set to 10000.

A. Dataset

We use three real-world networks that are commonly utilized in the research on information diffusion, including the Email, CollegeMsg, and Gnutella whose descriptions can be found in the provided references [18]. The statistics of these datasets are summarized in Table I.

TABLE I: Statistics of Three Real Networks

Dataset	Type	Nodes	Edges	Avg. Degree
Email	Directed	986	332,334	25.2
CollegeMsg	Directed	1,899	59,835	10.6
Gnutella	Directed	6,301	20,777	3.2

B. Experimental Results

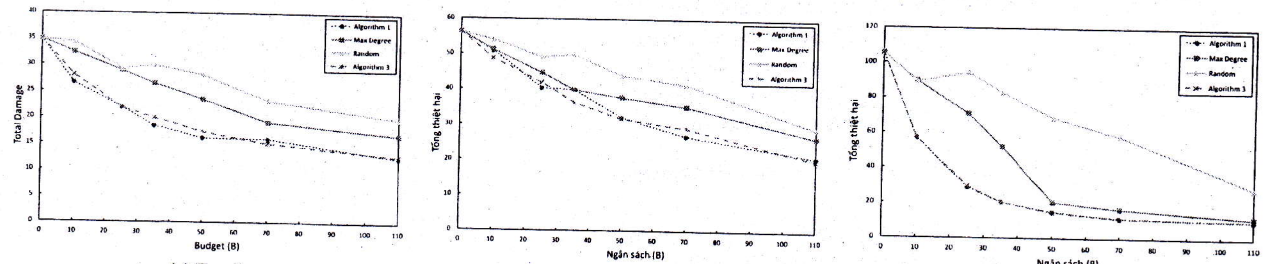
Impact of buget. Fig. 2 shows the total damages caused by misinformation after placing monitor on the node set I by Algorithm 1, Algorithm 3, Max Degree algorithm and Random algorithm in the case that budget B changes, $B = \{10, 25, 35, 50, 70, 110\}$, $d = 6$, and the misinformation source set is randomly initialized with the size of $|S| = 10$. Under all circumstances, Algorithm 1 and Algorithm 3 yielded more desirable results than the remaining algorithms. The damage reduction is from 1.017 times to 3.478 times higher in comparison with Max Degree algorithm. Especially in Fig. 2(a) when $B = 10$, Algorithm 1 and Algorithm 3 prove to be 3.478 and 2.87 times, respectively, more effective compared to Max Degree algorithm. In Fig. 2(c), when $B = 10$, Algorithm 1 and Algorithm 3 prove to be 3.052 and 3.028 times, respectively, more effective compared to the Max Degree algorithm.

Impact of sources. Fig. 3 presents the damage reduction before and after placing monitors on node I by Algorithm 1 and Algorithm 3 as well as the baseline algorithms Max Degree and Random when the size of source S changes, $|S| = \{5, 10, 15, 20, 25\}$, $d = 5$, with a fixed budget of $B = 25$. We can see that, in all scenarios, Algorithm 1 and Algorithm 3 are more effectual than the two baseline algorithms. In Fig. 3(a) when $|S| = 5$, Algorithm 1 proves 2.563 times more effective than Max Degree algorithm. In Fig. 3(c), once $|S| = 20$, Algorithm 1 proves 3.98 times more effective than Max Degree algorithm.

In general, Algorithm 1 is almost as effective as Algorithm 3 on the three datasets.

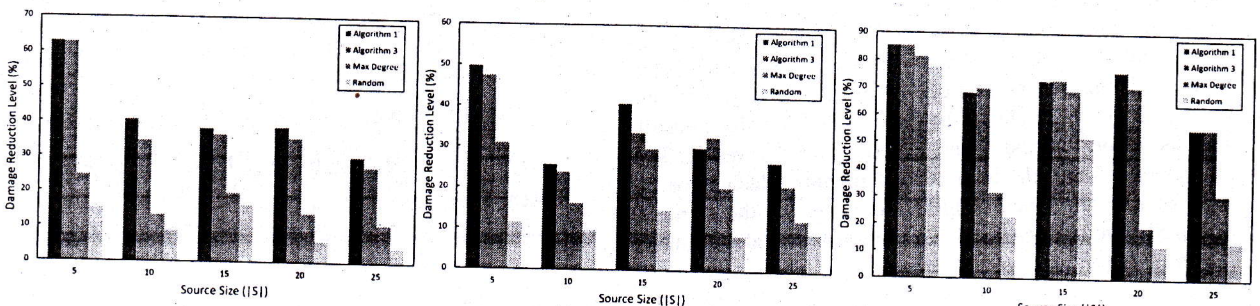
VI. Conclusion and Future Work

In conclusion, we consider the MSM problem of determining the optimal position for placing monitors with a view to



(a) Email (b) CollegeMsg (c) Gnutella

Fig. 2: Total damages when budget B changes, $d = 6$, $|S| = 10$



(a) Email (b) CollegeMsg (c) Gnutella

Fig. 3: Damage reduction when the size of source S changes, $d = 5$, $B = 25$

minimizing the propagation of misinformation on OSNs in d spread steps on a limited budget for the installation of monitors. Besides proving MSM is NP-Hardness, we suggested two greedy algorithms based on the damage-reducing function $f(I)$ and the $f(I)$'s value-increasing function over one monitors-placing cost unit. The result of the experiment shows that the suggested algorithms are better than the baseline algorithms Max Degree and Random. In the future, we will expand the problem to the case in which each node activated with misinformation causes different damages and, at the same time, consider the problem on various other information propagation models.

REFERENCES

[1] H. Zhang, M. Alim, X. Li, M. T. Thai, and H. Nguyen. Misinformation in Online Social Networks: Catch Them All with Limited Budget. ACM Transactions on Information Systems, 2016, pp. 18:1-18:24.

[2] D. Kempe, J. Kleinberg, E. Tardos. Maximizing the spread of influence through a social network. In Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2003, pp. 137-146.

[3] M. S. Granovetter. Threshold models of collective behavior. The American Journal of Sociology, 1978, pp. 1420-1443.

[4] J. Goldenberg, B. Libai, and E. Muller. Talk of the network: A complex systems look at the underlying process of word-of-mouth. Marketing Letters, 2001, pp. 211-223.

[5] J. Leskovec, M. Mcglohon, C. Faloutsos, N. Glance, and M. Hurst. Patterns of cascading behavior in large blog graphs. In Proc. SDM, 2007, pp. 551-556.

[6] T. Carnes, R. Nagarajan, S. M. Wild, and A. V. Zuylen. Maximizing influence in a competitive social network: A follower's perspective. In Proceedings of the ninth international conference on Electronic Commerce, 2007, pp. 351-360.

[7] D. T. Nguyen, N. P. Nguyen, and M. T. Thai. Sources of Misinformation in Online Social Networks: Who to Suspect?. In Proceedings of the IEEE Military Communications Conference, 2012, pp. 1-6.

[8] A. Goyal, W. Lu, and L. V. S. Lakshmanan. Simpath: An efficient algorithm for influence maximization under the linear threshold model. In Proc. ICDM, 2011, pp. 211-220.

[9] W. Chen, Y. Yuan, L. Zhang. Scalable Influence Maximization in Social Networks under the Linear Threshold Model. In Proc. ICDM, 2010, pp. 88-97.

[10] A. Goyal, W. Lu, L. V.S. Lakshmanan. Celf++: optimizing the greedy algorithm for influence maximization in social networks. In Proc. WWW, 2011, pp. 47-48.

[11] V. Qazvinian, E. Rosengren, D. R. Radev, and Qiaozhu Mei. Rumor has it: Identifying misinformation in microblogs. In Proc. EMNLP, 2011, pp. 1589-1599.

[12] S. Kwon, M. Cha, K. Jung, W. Chen, and Y. Wang. 2013. Prominent features of rumor propagation in online social media. In Proc. ICDM, 2013, pp. 1103-1108.

[13] C. Budak, D. Agrawal, A. El Abbadi. Limiting the spread of misinformation in social networks. In Proceedings of the 20th International Conference on World Wide Web, 2011, pp. 665-674.

[14] H. Zhang, H. Zhang, X. Li, and M. T. Thai. Limiting the Spread of Misinformation while Effectively Raising Awareness in Social Networks. In Proceedings of the 4th International Conference on Computational Social Networks, 2015, pp. 35-47.

[15] N. P. Nguyen, G. Yan, and M. T. Thai. Analysis of Misinformation Containment in Online Social Networks. Elsevier Computer Networks-Towards a Science of Cyber Security, 2013, pp. 2133-2146.

[16] W. Chen, L.V.S. Lakshmanan, C. Castillo. Information and Influence Propagation in Social Networks. Synthesis Lectures on Data Management, 2013, pp. 1-177.

[17] Chen, Wei, Wei Lu, and Ning Zhang. Time-critical influence maximization in social networks with time-delayed diffusion process. In Proc. AAAI, 2012, pp. 1-5.

[18] J. Leskovec. Stanford Large Network Dataset Collection, http://snap.stanford.edu/data/.