Rumor Containment by Blocking Nodes in Social Networks

Lan Yang, Ziyue Ma, Zhiwu Li, Alessandro Giua

Abstract

Rumor spreads fast in social networks and may seriously damage our society. In this paper we present a mathematical programming formulation based on integer linear programming (ILP) to minimize rumor spread by blocking a subset of nodes (called blockers) in complex social networks modeled as a Linear Threshold Model. We also propose a modified approach which solves the top-k blockers problem with a reduced computational effort and formally prove that its performance is still optimal. Then, the presented method is evaluated for its effectiveness of containing rumor spread in four different networks and its performance is compared with a greedy based and two centrality based approaches. The experimental analysis shows that the ILP based method outperforms the other three approaches and is applicable to large scale networks.

Key words: Rumor spread, blocking node, linear threshold model, integer linear programming.

Published as: L. Yang, Z.Y. Ma, Z.W. Li, A. Giua,"Rumor Containment by Blocking Nodes in Social Networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, Vol. 53, no. 7, pp. 3990–4002, 2023. DOI: 10.1109/TSMC.2023.3237933

⋆ L. Yang and Z. Ma are with the School of Mechano-Electric Engineering, Xidian University, Xi'an 710071, China (email: yanglan@xidian.edu.cn, maziyue@xidian.edu.cn). Z. Li is with School of Mechanical-Electrical Engineering, Xidian University, Xi'an 710071, China, and also with the Institute of Systems Engineering, Macau University of Science and Technology, Taipa, Macau (email: zhwli@xidian.edu.cn). A. Giua is with the Department of Electrical and Electronic Engineering, University of Cagliari, Piazza DArmi, 09123 Cagliari, Italy (email: giua@unica.it).
This work was supported by Shaanxi Provincial Natural Science Foundation (2021JQ-199) and the Fundamental Research Funds for the Central Universities (XJS210406).

1 Introduction

Many networked systems are concerned with propagation phenomena which are driven by the mutual influence among agents or individuals. For example, in social networks, people may be influenced by their friends to adopt an innovation, spread a piece of information, or participate a political activity. In infrastructure networks, a breakdown of a few components can lead to large scale temporary blackouts or even to a catastrophic failure. In large human or animal populations, epidemics spread rapidly among individuals. With the advance of communication technology since the beginning of the 21st century, online social networking sites have provided people new and easier ways to share information, exchange ideas, or even adopt innovations. Meanwhile, the analysis of influence propagation has become a hot research topic spanning several disciplines such as computer science, sociology, control theory, etc. This kind of research may provide new keys to understand the mechanism of influence dynamics in complex networks and develop techniques to predict the behaviors of individuals so as to prevent or control something undesirable.

Although propagation in networked systems is a complex phenomenon, there have been several successful attempts to describe it by mathematical models. Two widely-adopted discrete time diffusion models proposed by Kempe et al. [1] are the *Linear Threshold Model* (LTM) and the *Independent Cascade Model* (ICM), which consider directed graphs where a node represents a user or an individual and an arc represents a relationship between two users. At a certain time, each node can either be active or inactive. An active state means that the node adopts the innovation, otherwise, it is inactive. Initially, all nodes are inactive. At time 0, a few nodes in the network are activated to start the propagation according to the model dynamics.

Motivated by different practical applications in complex networks including rumor spread suppression [2, 3, 4, 5, 6, 7], advertisement in viral marketing [8, 9, 10, 11], etc., Kempe et al. [1] introduce the *influence maximization (IM) problem* that so far has been the most widely studied issue in this framework. The IM problem consists in identifying a small subset of early adopters (the so called seed set) to achieve a maximum influence propagation. They formalize the problem as a combinatorial optimization problem and prove it to be NP-hard for both the ICM and LTM models. To solve the IM problem, a greedy approach that can achieve a guaranteed approximate solution — albeit with a significant computational cost — is presented. The theoretical hardness of the IM problem triggers numerous researchers to design feasible algorithms for its solution. The documented approaches in the literature can be classified into four categories: approximation algorithms with provable guarantee [12, 13, 14, 15, 16], community-based algorithms [17], heuristic approaches [18, 19, 20], and metaheuristics [21].

Influence maximization problems aim to maximize the diffusion of desirable contents like valuable information. Undesirable or damaging contents, such as infrastructure failures and fake news, can also spread in networked systems and could produce catastrophic effects in man-made systems or in our society. As an example, the fake tweet "Two explosions in White House and Obama is injured" caused instability on US stock market and 10 billion USD losses in a short time before the rumor was clarified [22]. It is evident that the study on how to contain or control the diffusion of undesirable contents is meaningful in both societal and economic terms.

This work addresses the issue of controlling rumor spread in complex social networks. Normally, rumor control strategies fall into two types: *counterbalance strategy* and *network disruption strategy.*

1.1 Counterbalance strategy

Counterbalance strategy tries to counterbalance the negative influence of rumor by propagating anti-rumor or truth such that the two pieces of opposite information fight with each other. Minimizing rumor spread is addressed in [3, 4, 23, 24] by extending the LTM to describe the competitive propagation dynamics between the rumor and truth. The diffusion models involved in [23, 24] are assumed to be progressive, i.e., once a user adopts one type of information, it will disregard any other ones. However, Yang et al. [4] allow individuals that adopt rumor first to reconsider their belief, which is a mechanism that better fits with real individual behaviors. The rumor control problem is also studied in [25] and [26] using different extensions of the ICM. Recently, Manouchehri [27] address an extended version of the above problem, so called temporal rumor blocking problem, by adopting a competitive ICM.

1.2 Disruption strategy

Network disruption strategy, which can be done by removing (or blocking) some key nodes/edges from a network, tries to disconnect the inactive nodes from active nodes with the aim of suppressing the rumor spread.

Consider the edge-based rumor containment problem, Kimura et al. [28] and Khalil et al. [29] use a greedy algorithm to search for approximate solutions in the framework of the ICM and LTM respectively. The objective function of this problem is proved to be non-submodular for the ICM in [30]. Fortunately, the authors present submodular upper-bound and lower-bound such that they can apply a Sandwich algorithm. GPU-based approaches are proposed in [31] to control the cyber-epidemics spread considering both edge-based and node-based schemes for the LTM. In [32], Nandi et al. propose several methods of removing links to suppress infections spread that can also be applied to suppress rumor spread. Medya et al. [33] solve the influence limitation problem over links removals by approximation algorithms based on submodularity.

Blocking users in online social platforms may refer to denying access to some users such that they cannot see and propagate rumor [34]. From the network perspective, blocking a node is viewed as deleting it and all its connections with other nodes from a network, thus ensuring to interrupt the pathways from it to others. This kind of node-based scheme is considered in [35, 36, 37]. The node-based rumor containment problem, which aims to identify the most effective nodes (so called blockers), is formalized as an optimization problem in [35] and approximately solved by a natural greedy method. Yan et al. [36] propose a two-stage greedy heuristic including generating candidates and selecting the top-k blockers for general networks and a dynamic programming method for tree networks considering ICM as the diffusion model. Saha et al. [37] consider epidemic models and transform the problem of minimizing infection to the issue of minimizing spectral radius of the network via removing some key nodes or edges. Different from the above strategies, a group disbanded scheme is presented to address the misinformation spread minimization problem motivated by the echo-chamber effect within a chat group in [38].

We notice that the removal (or block) of nodes involves the removal of edges. However, the solutions to the edge-based rumor containment problem are not directly solvable for the node-based one. Furthermore, it is more reasonable and of interest to block nodes than block links in population networks. Therefore, in this work, we adopt the option of blocking nodes to contain rumor spread. The problem of finding the top-k blockers has a combinatorial nature and requires an exhaustive search in general. Most of the above mentioned results try to solve this problem by greedy approaches, which do not scale well for very large networks. Furthermore, the top-k blockers problem is neither submodular nor supermodular, implying that there is no approximation guarantee for greedy approaches. Beyond the greedy approaches, this paper aims to propose an algebraic way for providing optimal solutions and makes it scalable for large networks. The workflow chart is shown in Fig. 1.

This paper considers linear threshold models to mathematically characterize the propagation dynamics of rumor in social networks since threshold models are relatively appropriate to describe collective behaviors. From the network perspective, with the aim of identifying the effective blockers, instead of removing them and their connections, we assign them threshold values that are greater than 1. This strategy works the same as the removal of node since the LTM normalizes the sum of influence weight from one's in-neighbors to 1. Then, any node with a threshold greater than 1 stays inactive forever. Our contributions are multifold.

- A special LTM called *blocked linear threshold network* is defined, where some nodes' thresholds are larger than 1, denoting that the nodes are blocked.
- We show that in linear threshold networks the monotonicity holds for the objective function of the

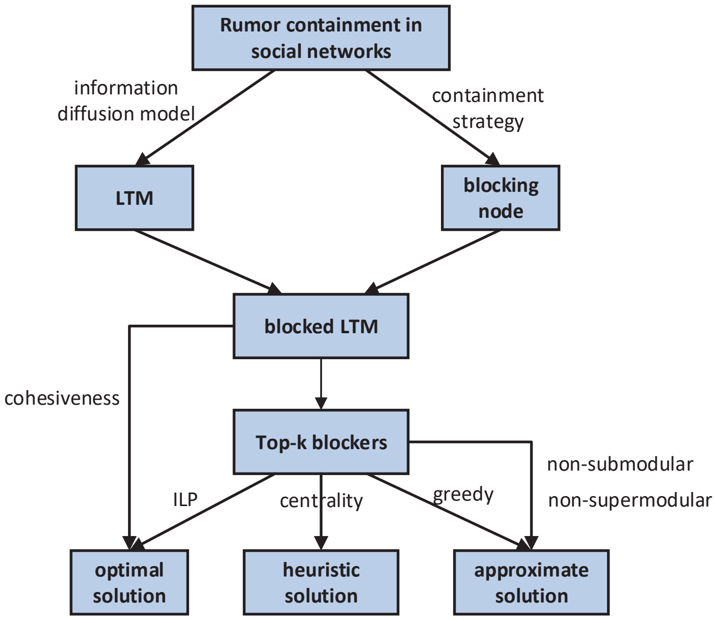


Fig. 1. Overall workflow diagram.

top-k blockers problem, however, the submodularity and supermodularity do not.

- Considering the blocked linear threshold network, the top-k blockers problem is first formalized as a non-linear programming problem. A linearization of the non-linear formulation is then proposed such that the optimal top-k blockers can be identified by solving an integer linear programming problem. We can further reduce the complexity of the linearized programming by showing that the evolution of the entire network can be equivalently represented by its active sub-network.
- We compare the mathematical programming method with two centrality based approaches and a greedy based approach for the effectiveness of rumor containment in four different networks.

This paper is structured as follows. Section 2 first reviews the LTM and then discusses the effect of blocking nodes on network evolution. The top-k blockers problem is defined and analyzed in Section 3. Section 4 shows an optimal solution to the top-k blockers problem by solving a non-linear programming problem that can be further linearized as an integer linear programming problem. A greedy based approach and two centrality based approaches are presented in Section 5. Section 6 shows a series of experimental evaluation for our mathematical programming method and other three heuristics. Section 7 concludes this paper.

2 Linear threshold networks

We first introduce the LTM to explain how influence or information propagates in social networks. Table 1 lists some of the notations and explanations in this paper.

2.1 Linear threshold network

A *linear threshold network* N_{LT} is a four-tuple (V, E, θ, w), where

- $V = \{1, 2, ..., n\}$ denotes the set of nodes in the network;
- $E \subseteq V \times V$ denotes a set of directed arcs, we say $(i, j) \in E$ if there is an arc from node i to node j;
- Function $\theta : V \to (0, 1]$ is a mapping that assigns a *threshold value* $\theta_i \in (0, 1]$ to each node $i \in V$;
- Function $w : V \times V \to [0, 1]$ is a mapping that assigns an influence weight $w_{ij} \in (0, 1]$ to each arc $(i, j) \in E$ such that $w_{ij} = 0$ if $(i, j) \notin E$ and $\sum_{i \in V} w_{ij} = 1$ for all $j \in V$. We assume that arcs (i, i), i.e., self-loops, are only possible for source nodes (which have no in-neighbors) and in this case their

Table 1
Notations

Notation	Description
V $(\hat{V})$	set of nodes in network N_{LT} (active sub-network $\hat{N}_{LT}$)
E $(\hat{E})$	set of directed arcs in N_{LT} $(\hat{N}_{LT})$
W $(\hat{W})$	weighted adjacency matrix of N_{LT} $(\hat{N}_{LT})$
n $(\hat{n})$	number of nodes in $N_{LT}(\hat{N}_{LT})$: $n = \|V\|$ $(\hat{n} = \|\hat{V}\|)$
θ_i $(\hat{\theta}_i)$	threshold value of node $i \in V(\hat{V})$
w_{ij}	influence weight from i to j
$N_i(\hat{N}_i)$	in-neighbor set of node $i \in V(\hat{V})$
d	mean degree of network N_{LT}
ϕ_0	seed set: the set of nodes initially activated
S_b	blocker set: the set of blockers
Φ^*	set of final adopters
$\boldsymbol{x}(\boldsymbol{x}^*)$	characteristic vector of a (maximal) cohesive set
$\boldsymbol{y}$	characteristic vector of the seed set
$\boldsymbol{b}(\boldsymbol{b}^*)$	characteristic vector of a (optimal) blocker set
$\text{Diag}(\boldsymbol{a})$	a function creating diagonal matrix from vector $\boldsymbol{a}$

weight must be 1 to ensure that $\sum_{i\in V} w_{ij} = 1$.

Each node i in a network N_{LT} represents an individual. Its threshold value θ_i denotes the tendency of this node to behave like their neighbors [1]. The *in-neighbor set* of node $i \in V$ is defined as $\mathcal{N}_i = \{j|(j,i) \in E\}$. Arc (i,j) indicates that node i can influence node j.

We use $\Theta = \text{Diag}([\theta_1, \theta_2, \cdots, \theta_n])$ to denote the *threshold matrix* of the network: its diagonal elements are the nodes' threshold values and all non-diagonal elements are null. Let $W \in [0,1]^{n\times n}$ denote the *weighted adjacency matrix* of the network, where $W(i,j) = w_{ij}$.

Each node can either be inactive or active. Initially, all nodes are inactive. Let $\phi_0 \subseteq V$ be a *seed set* representing a set of nodes that are activated at step $t = 0$. The diffusion process originates from these seeds which try to influence other nodes step by step. Let ϕ_t denote the set of nodes that are *activated* at step t. The set of nodes that are *active* at step t, i.e., those that have been activated at any step in $[0,t]$, is denoted by $\Phi_t = \bigcup_{k=0}^{t} \phi_k$. By definition, we have $\Phi_0 = \phi_0$.

At each step $t = 1, 2, \ldots$, an inactive node i changes its state to be active if the sum of the influence weights of its active in-neighbors at step $t-1$ exceeds its threshold θ_i, i.e.,

$$i \in \phi_t \Longleftrightarrow \sum_{j\in\mathcal{N}_i\cap\Phi_{t-1}} w_{ji} \geq \theta_i \quad (\forall i \in V \setminus \Phi_{t-1})\,. \tag{1}$$

The activation propagates following Equation (1) in a progressive manner, that is, deactivation process is not considered and active nodes never switch their states. Thus, the network will reach a steady state in certain steps where all nodes will not switch their states. It is characterized by the set of *final adopters* and defined as $\Phi_\infty(N_{LT}, \phi_0) = \bigcup_{k=0}^{\infty} \phi_k$.

The set of final adopters can be obtained by simulating the above evolution process until the stop criterion is reached, i.e., no more inactive nodes can be activated, see Algorithm A1 in the appendix. This procedure has a computational time complexity $O(nd)$, where n denotes the number of nodes and d denotes the mean degree of the underlying network.

2.2 Properties of linear threshold network

Instead of simulating the evolution process of the network to determine the steady state, in this part we show a linear algebraic approach for its determination based on the concept of cohesiveness.

2.2.1 Cohesiveness

Cohesive set is originally defined in un-weighted linear threshold network by Acemoglu et al. [8]. We generalize its definition to weighted networks in the following.

Definition 2.1 *A subset $X \subseteq V$ is called a* cohesive set *if for all $i \in X$ it holds:*

$$\sum_{j \in X \cap \mathcal{N}_i} w_{ji} > 1 - \theta_i. \tag{2}$$

We can learn from Equation (2) an interesting property: if $\phi_0 \cap X = \emptyset$, then for all $t \geq 0$, $\phi_t \cap X = \emptyset$. That is to say, if no seed node exists in X, then all nodes in X will always remain inactive. Besides, the union of cohesive sets is also cohesive.

Lemma 2.1 *[8] Given an LTM network $N_{LT} = (V, E, \theta, w)$ and a seed set $\phi_0 \subseteq V$, let $M \subseteq V \setminus \phi_0$ be the* maximal cohesive set *contained in $V \setminus \phi_0$. The final adopter set is:*

$$\Phi_\infty(N_{LT}, \phi_0) = V \setminus M. \tag{3}$$

2.2.2 Algebraic characterization of $\Phi_\infty(N_{LT}, \phi_0)$

Definition 2.2 *Given a set $X \subseteq V$, its* characteristic vector *$\mathbf{x} \in \{0,1\}^n$ is defined as $\mathbf{x}_i = 1$ if $i \in X$; otherwise $\mathbf{x}_i = 0$, i.e.,*

$$\mathbf{x}_i = \begin{cases} 1, & \textit{if node } i \in X \\ 0, & \textit{otherwise.} \end{cases}$$

The sufficient and necessary condition for a cohesive set proposed in [9] can also be generalized to weighted networks.

Lemma 2.2 *[9] A set $X \subseteq V$ is cohesive if and only if its characteristic vector $\mathbf{x}$ satisfies*

$$\mathbf{x}^T W(\cdot, i) \geq 1 - \theta_i \quad (\forall i \in X). \tag{4}$$

where $W(\cdot, i)$ is the i-th column of the weighted adjacency matrix W.

Lemma 2.1 provides us an immediate way to compute the set of final adopters avoiding to determine the detailed evolution process. Then based on Lemmas 2.1 and 2.2, an algebraic characterization of the steady state is proposed in Proposition 2.1. Note that in the linear or non-linear programs of this paper, $\boldsymbol{x}$ denotes the decision variable vector (i.e., the characteristic vector of a cohesive set) while $\boldsymbol{x}^*$ denotes the optimal solution (i.e., the characteristic vector of the maximal cohesive set).

Proposition 2.1 *[9] Given an LTM network $N_{LT} = (V, E, \theta, w)$ with n nodes, let $\mathbf{y}$ be the characteristic vector of a seed set $\phi_0 \subseteq V$. The maximal cohesive set M contained in $V \setminus \phi_0$ has a characteristic vector $\mathbf{x}^*$ that is the optimal solution of the following integer linear programming problem (ILP):*

$$\begin{aligned} \max_{x} \quad & \boldsymbol{1}^T \mathbf{x} \qquad\qquad\qquad (ILP-1) \\ s.t. \quad & \boldsymbol{1} - \mathbf{x} \geq \mathbf{y} \\ & [I - \Theta - W^T]\mathbf{x} \leq \boldsymbol{0} \\ & \mathbf{x} \in \{0,1\}^n. \end{aligned}$$

where I is an $n \times n$ identity matrix. The set of final adopters is $\Phi_\infty(N_{LT}, \phi_0) = \{i \in V | x_i^* = 0\}$.

Remark: The weighted adjacency matrix W (as defined in Section 2) is different from the one adopted in [9] (let us call it $\bar{W}$) where no self-loops are allowed and thus all diagonal elements are null. Using the definition in [9], Proposition 1 only applies to networks without source nodes. However, using our definition of W, Proposition 1 also holds when there are source nodes. In the following, we show an example that computes the set of final adopters using (ILP-1) with the two different weighted adjacency matrices W and $\bar{W}$, respectively.

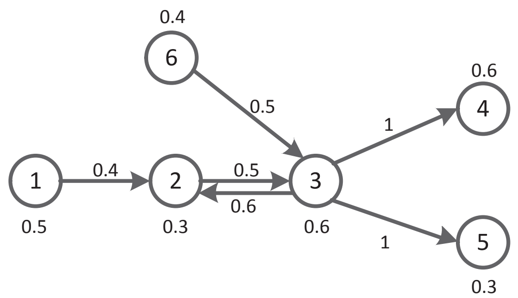


Fig. 2. A simple linear threshold network with source nodes.

Example 2.1 *Consider the linear threshold network N_{LT} in Fig. 2 with six nodes, among which nodes 1 and 6 are source nodes. The value next to each node is the threshold value and the value besides each arc is the associated influence weight. Assume that $\phi_0 = \{1\}$. Solving the integer linear programming (ILP-1) using $\bar{W}$ returns us the set of final adopters $\Phi_\infty(N_{LT}, \phi_0) = \{1, 2, 3, 4, 5, 6\}$. Instead, if we use W, we can obtain $\Phi_\infty(N_{LT}, \phi_0) = \{1, 2\}$. This is consistent with the result obtained by Algorithm A1, which also outputs $\Phi_\infty(N_{LT}, \phi_0) = \{1, 2\}$.* △

2.3 Network evolution with blocking of nodes

A set of blocked nodes $S_b \subseteq V \setminus \phi_0$ is called a *blocker set* in the literature. In this part, we discuss the effect of blocking nodes on the network evolution.

Given an LTM $N_{LT} = (V, E, \theta, w)$, a seed set $\phi_0 \subseteq V$, and a blocker set $S_b \subseteq V \setminus \phi_0$, we denote *the set of final adopters with blocker set S_b* by $\Phi_\infty(N_{LT}, \phi_0, S_b)$. In a network with blockers, the activation rule (1) only applies to nodes that are not blocked. We can compute $\Phi_\infty(N_{LT}, \phi_0, S_b)$ by slightly modifying Algorithm A1: initializing $\Phi_c = V \setminus (\phi_0 \cup S_b)$ in Line 3.

Given an LTM $N_{LT} = (V, E, \theta, w)$ and S_b, we can define an equivalent *blocked linear threshold network* $N'_{LT} = \mathcal{B}(N_{LT}, S_b) = (V, E, \theta', w)$ where θ' is a mapping that assigns a threshold value θ'_i for each node $i \in V$ such that

$$\theta'_i = \begin{cases} \theta_i + 1, & \text{if } i \in S_b \\ \theta_i, & \text{else.} \end{cases}$$

An LTM N_{LT} with blocker set S_b generates a blocked linear threshold network N'_{LT} that is a special LTM with some thresholds larger than 1. Blockers are those nodes with $\theta'_i > 1$ in N'_{LT}. The activation rule of a node in the standard LTM N_{LT} still works for N'_{LT}. The LTM assumes that the sum of influence weight from one's in-neighbors is equal to 1. According to Equation (1), inactive nodes with $\theta'_i > 1$ can never be activated. This result turns out to coincide with the goal of blocking a node, i.e., preventing it from being activated and in turn triggering its inactive followers.

We will show below that the set of final adopters with blocker set in an LTM N_{LT} is equivalent to the set of final adopters in the corresponding blocked LTM N'_{LT}.

Let $\phi_t(N_{LT}, \phi_0, S_b)$ and $\phi_t(N'_{LT}, \phi_0)$ denote the sets of nodes activated at step t in the network N_{LT} with blocker set S_b and the corresponding blocked LTM N'_{LT}, respectively.

Proposition 2.2 *Given an LTM network $N_{LT} = (V, E, \theta, w)$, a seed set $\phi_0 \subseteq V$, and a blocker set $S_b \subseteq V \setminus \phi_0$, the evolution process of network N_{LT} with blocker set S_b is equivalent to that of its blocked linear threshold network $N^{'}_{LT} = \mathcal{B}(N_{LT}, S_b) = (V, E, \theta^{'}, w)$, i.e., $\phi_t(N_{LT}, \phi_0, S_b) = \phi_t(N^{'}_{LT}, \phi_0)$.*

PROOF. *For $t = 0$, the two sets are identical by definition. We can prove the result by showing that for any $t \geq 1$, it holds $\phi_t(N_{LT}, \phi_0, S_b) = \phi_t(N^{'}_{LT}, \phi_0)$.*

We start from $t = 1$, for any node i, we have $i \in \phi_1(N_{LT}, \phi_0, S_b)$ iff it holds $\sum_{j \in \phi_0 \cap \mathcal{N}_i} w_{ji} \geq \theta_i$, that is, $\sum_{j \in \phi_0 \cap \mathcal{N}_i} w_{ji} \geq \theta^{'}_i$ since $i \notin S_b$ and $\theta^{'}_i = \theta_i$. This implies $i \in \phi_1(N^{'}_{LT}, \phi_0)$. Then, we have $\phi_1(N_{LT}, \phi_0, S_b) = \phi_1(N^{'}_{LT}, \phi_0)$. This statement also holds for step $t = 2$ and then by recursion, for any $t > 2$, it holds $\phi_t(N_{LT}, \phi_0, S_b) = \phi_t(N^{'}_{LT}, \phi_0)$.

Let $\Phi_\infty(N^{'}_{LT}, \phi_0)$ denote the steady state in $N^{'}_{LT}$ given ϕ_0. We have $\Phi_\infty(N^{'}_{LT}, \phi_0) = \bigcup_{k=0}^{\infty} \phi_t(N^{'}_{LT}, \phi_0)$. Then, we can obtain the following result from Proposition 2.2.

Corollary 2.1 *Given a seed set $\phi_0 \subseteq V$, the set of final adopters with blocker set $S_b \subseteq V \setminus$ in an LTM $N_{LT} = (V, E, \theta, w)$ is identical to the set of final adopters in $N^{'}_{LT} = \mathcal{B}(N_{LT}, S_b) = (V, E, \theta^{'}, w)$, i.e., $\Phi_\infty(N_{LT}, \phi_0, S_b) = \Phi_\infty(N^{'}_{LT}, \phi_0)$.*

We give a simple example below to illustrate how nodes blocking interferes with the diffusion process of the original network.

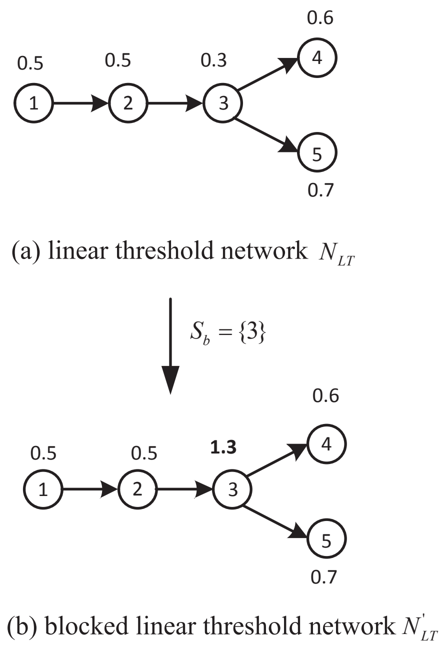


Fig. 3. Generation of a sample blocked linear threshold network.

Example 2.2 *Consider the linear threshold network in Fig. 3a by assuming that $\phi_0 = \{1\}$ and $S_b = \{3\}$. The corresponding blocked linear threshold network is depicted in Fig. 3b. If there is no blocking, we have $\Phi_\infty(N_{LT}, \phi_0) = \{1, 2, 3, 4, 5\}$. However, the set of final adopters with blocker set S_b is $\Phi_\infty(N_{LT}, \phi_0, S_b) = \Phi_\infty(N^{'}_{LT}, \phi_0) = \{1, 2\}$. We also show in Table 2 the detailed evolutions of the LTM with blocking nodes and the blocked LTM, which are the same. Due to the limited space in Table 2, we denote $\Phi_t(N_{LT}) = \bigcup_{k=0}^{t} \phi_t(N_{LT}, \phi_0, S_b)$ and $\Phi_t(N^{'}_{LT}) = \bigcup_{k=0}^{t} \phi_t(N^{'}_{LT}, \phi_0)$.* △

Table 2
Evolutions of LTM and blocked LTM of Example 2.2

t	$\phi_t(N_{LT}, \phi_0, S_b)$	$\Phi_t(N_{LT})$	$\phi_t(N'_{LT}, \phi_0)$	$\Phi_t(N'_{LT})$
0	{1}	{1}	{1}	{1}
1	{2}	{1,2}	{2}	{1,2}
2	$\emptyset$	{1,2}	$\emptyset$	{1,2}

We discuss a property of the blockers in the following, which implies that any blocker is cohesive itself in a blocked linear threshold network and therefore can never be activated by others.

Proposition 2.3 *Given a linear threshold network $N_{LT} = (V, E, \theta, w)$ and a seed set $\phi_0 \subseteq V$, any node in a blocker set $S_b \subseteq V \setminus \phi_0$ is cohesive in the corresponding blocked LTM $N'_{LT} = (V, E, \theta', w)$.*

PROOF. *For any node $i \in S_b$, we have $\theta'_i > 1$. Therefore, it always holds $\sum_{j \in X \cap \mathcal{N}_i} w_{j,i} > 1 - \theta'_i$ where X denotes any subset contained in V, which implies that i is cohesive in network N'_{LT}.*

Proposition 2.3 implies $S_b \subseteq M'$, where M' is the maximal cohesive set contained in $V \setminus \phi_0$ of the blocked LTM.

3 Problem statement and properties

Blocking some critical users is an effective way to contain rumor spread in social networks. However, it may be impossible to control all users in a network due to the high cost. Thus, we address the rumor containment problem with a limited budget what we call the *top-k blockers problem.*

3.1 Problem statement

Problem 3.1 (Top-k blockers problem) *Given an LTM network $N_{LT} = (V, E, \theta, w)$ and a seed set $\phi_0 \subseteq V$, let k be a positive integer. Find a set of at most k nodes denoted by $S_b \subseteq V \setminus \phi_0$ to minimize the cardinality of the set of final adopters with blocker set S_b, i.e.,*

$$\begin{aligned} \min_{S_b} \quad & |\Phi_\infty(N_{LT}, \phi_0, S_b)| \\ s.t. \quad & |S_b| \leq k \qquad (a) \\ & S_b \subseteq V \setminus \phi_0 \qquad (b) \end{aligned}$$

3.2 Properties

A set function $f : 2^V \to \mathbb{R}$ is said to be *monotone increasing* (resp., *monotone decreasing*) if for any two subsets $V_1 \subseteq V_2 \subseteq V$, it holds $f(V_1) \leq f(V_2)$ (resp., $f(V_1) \geq f(V_2)$). A set function f is said to be *submodular* (resp., *supermodular*) if for any subsets $V_1 \subseteq V_2 \subseteq V$ and for all $v \in V \setminus V_2$, it holds $f(V_1 \cup \{v\}) - f(V_1) \geq f(V_2 \cup \{v\}) - f(V_2)$ (resp., $f(V_1 \cup \{v\}) - f(V_1) \leq f(V_2 \cup \{v\}) - f(V_2)$). As evident from the definition, if f is a monotone decreasing and supermodular set function, then $-f$ is monotone increasing and submodular. Hence, maximizing a monotone increasing and submodular function is equivalent to minimizing a monotone decreasing and supermodular function. In addition, a solution obtained by a greedy approach for maximizing a monotone increasing and submodular function can approximate the optimal solution with a provable guarantee 63% [1].

In the following, we show that the monotonicity holds for the objective function of the top-k blockers problem, however, the submodularity and supermodularity do not. This means that the solution obtained by the greedy approach has no provable guarantee.

Proposition 3.1 *Given an LTM network* $N_{LT} = (V, E, \theta, w)$ *and a seed set* ϕ_0*, the objective function of the top-k blockers problem, i.e.,* $|\Phi_\infty(N_{LT}, \phi_0, S_b)|$*, is* monotone decreasing *but* non-supermodular *and* non-submodular *w.r.t* S_b*.*

PROOF. *We first show the monotonicity, i.e., for any two subsets* $S_b^{'} \subseteq S_b^{''} \subseteq V \setminus \phi_0$*, it holds* $|\Phi_\infty(N_{LT}, \phi_0, S_b^{''})| \leq |\Phi_\infty(N_{LT}, \phi_0, S_b^{'})|$*. Let* $N_{LT}^{'} = (V, E, \theta^{'}, w)$ *and* $N_{LT}^{''} = (V, E, \theta^{''}, w)$ *denote the blocked linear threshold networks generated based on blocker sets* $S_b^{'}$ *and* $S_b^{''}$ *respectively such that (1) for any node* $i \in V \setminus S_b^{''}$*,* $\theta_i^{'} = \theta_i^{''} = \theta_i$*; (2) for node* $i \in S_b^{'}$*,* $\theta_i^{'} = \theta_i^{''} = \theta_i + 1$*; (3) and for node* $i \in S_b^{''} \setminus S_b^{'}$*,* $\theta_i^{''} = \theta_i + 1$*. We can conclude that* $\theta_i^{''} \geq \theta_i^{'}$ *for any* $i \in V$*. Due to Proposition 2.2, it is equivalent to prove* $|\Phi_\infty(N_{LT}^{''}, \phi_0)| \leq |\Phi_\infty(N_{LT}^{'}, \phi_0)|$*.*

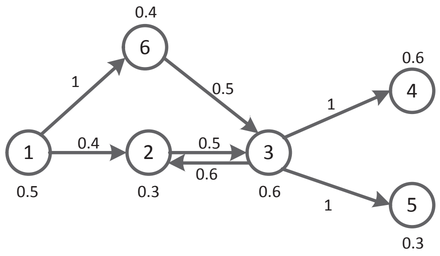


Fig. 4. Counterexample for submodularity.

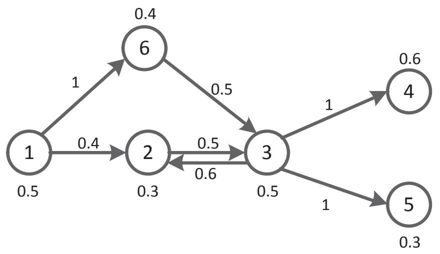


Fig. 5. Counterexample for supermodularity.

Assume that $M^{'}$ *and* $M^{''}$ *are the maximal cohesive sets contained in* $V \setminus \phi_0$ *for networks* $N_{LT}^{'}$ *and* $N_{LT}^{''}$*, respectively. Therefore, for any node* $i \in M^{'}$*, we have* $\sum_{j \in M^{'} \cap \mathcal{N}_i} w_{j,i} \geq 1 - \theta_i^{'} \geq 1 - \theta_i^{''}$*. This implies that* $M^{'}$ *is a cohesive set in* $N_{LT}^{''}$ *and then we have* $M^{'} \subseteq M^{''}$*. Thanks to Lemma 2.1 for the characterization of the final adopter set, it holds that* $\Phi_\infty(N_{LT}^{''}, \phi_0) \subseteq \Phi_\infty(N_{LT}^{'}, \phi_0)$ *and therefore* $|\Phi_\infty(N_{LT}^{''}, \phi_0)| \leq |\Phi_\infty(N_{LT}^{'}, \phi_0)|$*.*

We then show the non-submodularity and non-supermodularity by presenting two counterexamples.

Non-submodular: Consider the network in Fig. 4. Assume that the seed set for rumor is $\phi_0 = \{1\}$*,* $S_b^{'} = \emptyset \subseteq S_b^{''} = \{4\}$*, and* $v = \{2\}$*. Then we have* $|\Phi_\infty(N_{LT}, \phi_0, S_b^{'} \cup \{v\})| - |\Phi_\infty(N_{LT}, \phi_0, S_b^{'})| = -4 < |\Phi_\infty(N_{LT}, \phi_0, S_b^{''} \cup \{v\})| - |\Phi_\infty(N_{LT}, \phi_0, S_b^{''})| = -3$*.*

Non-supermodular: Consider the network in Fig. 5. Assume that the seed set for rumor is $\phi_0 = \{1\}$*,* $S_b^{'} = \emptyset \subseteq S_b^{''} = \{6\}$*, and* $v = \{2\}$*. Then we have* $|\Phi_\infty(N_{LT}, \phi_0, S_b^{'} \cup \{v\})| - |\Phi_\infty(N_{LT}, \phi_0, S_b^{'})| = -1 > |\Phi_\infty(N_{LT}, \phi_0, S_b^{''} \cup \{v\})| - |\Phi_\infty(N_{LT}, \phi_0, S_b^{''})| = -4$*.*

4 Optimal solutions

On the basis of cohesiveness in $N_{LT}^{'}$, the top-k blockers can be identified by solving a mathematical programming problem in a linear threshold network.

4.1 Non-linear formulation

Let $\boldsymbol{b} \in \{0,1\}^n$ be the characteristic vector of S_b where $b_i = 1$ denotes that node $i \in V$ is blocked and 0 not blocked, i.e.,

$$b_i = \begin{cases} 1, & \text{if node } i \text{ is blocked} \\ 0, & \text{otherwise.} \end{cases}$$

Given an LTM N_{LT} and a blocker set $S_b \subseteq V \setminus \phi_0$ with characteristic vector $\boldsymbol{b}$, the threshold of each node $i \in V$ in $N_{LT}^{'}$ can be written as

$$\theta_i^{'} = \theta_i + b_i.$$

Since networks N_{LT} and $N_{LT}^{'}$ have the same network structure (V and E) and influence weights w, we can use the same notation W to denote the weighted adjacency matrix for $N_{LT}^{'}$. Then we have the following sufficient and necessary condition for a cohesive set in $N_{LT}^{'}$.

Corollary 4.1 *Given a linear threshold network $N_{LT} = (V, E, \theta, w)$, a seed set $\phi_0 \subseteq V$, and a blocker set $S_b \subseteq V \setminus \phi_0$ with characteristic vector $\mathbf{b}$, a set X is cohesive in network $N_{LT}^{'} = (V, E, \theta^{'}, w)$ if and only if its characteristic vector $\mathbf{x}$ satisfies*

$$\mathbf{x}^T W(\cdot, i) \geq 1 - (\theta_i + b_i) \quad (\forall i \in X). \tag{5}$$

PROOF. *By Lemma 2, a set X in network $N_{LT}^{'} = (V, E, \theta^{'}, w)$ is cohesive if and only if $\mathbf{x}^T W(\cdot, i) \geq 1 - \theta_i^{'}$ where $\theta_i^{'} = \theta_i + b_i$.*

Let $\Theta^{'} = \text{Diag}([\theta_1^{'}, \theta_2^{'}, \cdots, \theta_n^{'}])$ denote the threshold matrix in network $N_{LT}^{'}$ and can be written as:

$$\Theta^{'} = \Theta + \text{Diag}(\boldsymbol{b}).$$

Based on Corollary 4.1, we can derive the following proposition that transforms the top-k blockers problem to a non-linear programming problem.

Proposition 4.1 *Given an LTM network $N_{LT} = (V, E, \theta, w)$ with n nodes, let $\mathbf{y}$ be the characteristic vector of the seed set $\phi_0 \subseteq V$ and $k \in \mathbb{R}_+$ a constant. Consider the following non-linear programming (NLP) with binary variable vectors $\mathbf{x}$ and $\mathbf{b}$:*

$$\begin{array}{lll} \max\limits_{\mathbf{x},\mathbf{b}} & \boldsymbol{1}^T\mathbf{x} & (NLP-1) \\ s.t. & \boldsymbol{1} - \mathbf{x} \geq \mathbf{y} & (a) \\ & \mathbf{x} \geq \mathbf{b} & (b) \\ & \boldsymbol{1}^T\mathbf{b} \leq k & (c) \\ & [I - (\Theta + Diag(\boldsymbol{b})) - W^T]\mathbf{x} \leq \boldsymbol{0} & (d) \\ & \mathbf{x}, \mathbf{b} \in \{0,1\}^n & (f) \end{array}$$

Let $\mathbf{x}^*, \mathbf{b}^* \in \{0,1\}^n$ be the global optimal solution of (NLP-1). Then the set of nodes $S_b^* = \{i \in V | b_i^* = 1\}$ is the optimal blocker set of the top-k blockers problem. The set of final adopters with blocker set S_b^* is $\Phi_\infty(N_{LT}, \phi_0, S_b^*) = \Phi_\infty(N_{LT}^{'}, \phi_0) = \{i \in V | x_i^* = 0\}$.

PROOF. *Constraint (b) insures that each blocked node must be also cohesive (see Proposition 2.3). Constraints (a) and (b) imply $\mathbf{b} \leq \mathbf{x} \leq \boldsymbol{1} - \mathbf{y}$, indicating that the blocker set S_b and the seed set ϕ_0 are disjoint: this coincides with Constraint (b) in the problem definition. Constraint (c) insures that a* *set S_b with characteristic vector* $\mathbf{b}$ *is a feasible solution to the top-k blockers problem: this coincides with Constraint (a) in the problem definition. Constraint (a) insures that the set M with characteristic vector* $\mathbf{x}$ *is a cohesive set in* $N_{LT}^{'} = (V, E, \theta^{'}, w)$*. Then Constraint (d) and the objective function insure that M is the maximal one contained in* $V \setminus \phi_0$*. Hence* $\Phi_\infty(N_{LT}^{'}, \phi_0)$ *has characteristic vector* $\mathbf{1} - \mathbf{x}$ *thanks to Lemma 2.1. Finally, the objective function maximizing M (that is equivalent to minimize the number of nodes that will be finally activated) insures that the set* S_b^* *with characteristic vector* $\mathbf{b}^*$ *is an optimal solution to the* top-k blockers problem*.*

4.2 Linearized formulation

We note that programming problem (NLP-1) contains only one non-linear vector constraint, that is, Constraint (d). Now we consider Constraint (d) entry-wise, i.e., for all $i = 1, 2, \cdots, n$, it holds that,

$$(1 - \theta_i^{'})x_i - \sum_{j=1}^{n} W(j,i)x_j \leq 0 \tag{6}$$

$$\Longleftrightarrow (1 - \theta_i)x_i - b_i x_i - \sum_{j=1}^{n} W(j,i)x_j \leq 0 \tag{7}$$

As we see in Constraint (7), the non-linear term is the product of two binary variables, i.e., $b_i x_i$.

We then show how Constraint (7) can be linearized by introducing some additional binary variables. We define new binary variables

$$z_i \overset{def}{=} b_i x_i \tag{8}$$

for all $i \in V$. The number of new binary variables that we need to introduce in total is n.

Since b_i and x_i are binary variables, Equation (8) can be linearized by the following set of linear constraints (9)–(11).

$$z_i \leq b_i \tag{9}$$
$$z_i \leq x_i \tag{10}$$
$$z_i \geq b_i + x_i - 1 \tag{11}$$

Then, the non-linear Constraint (7) can be replaced by the following linear constraints:

$$(1 - \theta_i)x_i - z_i - \sum_{j=1}^{n} W(j,i)x_j \leq 0 \tag{12}$$
$$z_i \leq b_i \tag{13}$$
$$z_i \leq x_i \tag{14}$$
$$z_i \geq b_i + x_i - 1 \tag{15}$$

Let vector $\boldsymbol{z} = \{z_i\}^n$. Based on the above linearization of (NLP-1) and Proposition 4.1, we can find the optimal solution by solving an integer linear programming problem.

Proposition 4.2 *Given an LTM network* $N_{LT} = (V, E, \theta, w)$ *with n nodes, let* $\boldsymbol{y}$ *be the characteristic vector of the seed set* $\phi_0 \subseteq V$ *and* $k \in \mathbb{R}_+$ *a constant. Consider the following integer linear programming*

(ILP) with binary variable vectors $\mathbf{x}$, $\mathbf{b}$, *and* $\mathbf{z}$*:*

$$\begin{aligned}
&\max_{\mathbf{x},\mathbf{b},\mathbf{z}} \quad \mathbf{1}^T\mathbf{x} && (ILP-2)\\
&s.t. \quad \mathbf{1}-\mathbf{x} \geq \mathbf{y} && (a)\\
&\qquad \mathbf{x} \geq \mathbf{b} && (b)\\
&\qquad \mathbf{1}^T\mathbf{b} \leq k && (c)\\
&\qquad [I-\Theta-W^T]\mathbf{x}-\mathbf{z} \leq \mathbf{0} && (d)\\
&\qquad \mathbf{z} \leq \mathbf{b} && (e)\\
&\qquad \mathbf{z} \leq \mathbf{x} && (f)\\
&\qquad \mathbf{z} \geq \mathbf{b}+\mathbf{x}-\mathbf{1} && (g)\\
&\qquad \mathbf{x},\mathbf{b},\mathbf{z} \in \{0,1\}^n && (h)
\end{aligned}$$

and let $\mathbf{x}^*, \mathbf{b}^*, \mathbf{z}^* \in \{0,1\}^n$ *be the optimal solution of (ILP-2). Then the set of nodes* $S_b^* = \{i \in V | b_i^* = 1\}$ *is the optimal blocker set of the top-k blockers problem. The set of final adopters with blocker set* S_b^* *is* $\Phi_\infty(N_{LT}, \phi_0, S_b^*) = \{i \in V | x_i^* = 0\}$.

PROOF. *The result follows from Proposition 5 and the linearization of the non-linear programming (NLP-1) presented before.*

The non-linear formulation (NLP-1) has $2n$ decision variables and $3n+1$ constraints. By linearization, the number of decision variables and constraints of (ILP-2) are $3n$ and $6n+1$ respectively which are still linear with n.

4.3 Reducing the complexity of the problem

We know that integer linear programming problems with a huge number of decision variables and constraints are computationally hard to solve. Therefore, solving the integer linear programming (ILP-2) for finding the top-k blockers cannot scale well to large-sized networks. In this part, we show that the optimal k blockers identified by traversing the entire network are equivalent to the optimal k blockers identified by traversing a part of the network. Then the number of decision variables and constraints of (ILP-2) dramatically decreases and hence is much easier to solve. This idea can be also applied to other searching methods, e.g., Greedy and centrality approach to be used in Section 5.

We search for the optimal blocker set in the entire network by solving (ILP-2). However, the linear threshold network is deterministic and the set of final adopters activated by the seeds (without blocking) can be easily computed. We know that only activated nodes may contribute to the rumor spread. Therefore, we just need to search for the blockers among the nodes that will be activated if no blocking is carried out.

The final adopter set $\Phi_\infty(N_{LT}, \phi_0)$ can be determined either by Algorithm A1 or solving (ILP-1). Then, we can extract the active sub-network $\hat{N}_{LT} = (\hat{V}, \hat{E}, \hat{\theta}, \hat{w})$ from the original network $N_{LT} = (V, E, \theta, w)$ by defining $\hat{V} = \Phi_\infty(N_{LT}, \phi_0) \subseteq V$, $\hat{E} = (\hat{V} \times \hat{V}) \cap E$, $\hat{\theta}_u = \theta_u$ for all $u \in \hat{V}$, and $\hat{w}_{u,v} = w_{u,v}$ for any $(u,v) \in \hat{E}$.

Since the seed sets for networks N_{LT} and $\hat{N}_{LT}$ are the same, we omit ϕ_0 in the following notations. Let $\phi_t(N_{LT})$ (resp., $\phi_t(\hat{N}_{LT})$) and $\Phi_\infty(N_{LT})$ (resp., $\Phi_\infty(\hat{N}_{LT})$) denote the set of nodes activated at step t in network N_{LT} (resp., $\hat{N}_{LT}$) and the final adopter set in network N_{LT} (resp., $\hat{N}_{LT}$) respectively, and $\mathcal{N}_v(N_{LT})$ (resp., $\mathcal{N}_v(\hat{N}_{LT})$) denote the in-neighbor set in network N_{LT} (resp., $\hat{N}_{LT}$). Then we have $\mathcal{N}_v(\hat{N}_{LT}) \subseteq \mathcal{N}_v(N_{LT})$. Moreover, we show in the following that the evolution process of network N_{LT} can be represented by that of the active sub-network $\hat{N}_{LT}$.

Theorem 4.1 *Given an LTM network $N_{LT} = (V, E, \theta, w)$ and a seed set $\phi_0 \subseteq V$, the evolution process of network N_{LT} is equivalent to that of its corresponding active sub-network $\hat{N}_{LT} = (\hat{V}, \hat{E}, \hat{\theta}, \hat{w})$, i.e., $\phi_t(N_{LT}) = \phi_t(\hat{N}_{LT})$ for all $t \geq 0$.*

PROOF. *The statement obviously holds for $t = 0$ since the seed sets are the same. We then prove the result for $t \geq 1$ by induction.*

We start from $t = 1$. For any node v, $v \in \phi_1(N_{LT})$ iff it holds $\sum_{u \in \phi_0 \cap \mathcal{N}_v(N_{LT})} w_{u,v} \geq \theta_v$. We also have $\phi_0 \cap \mathcal{N}_v(N_{LT}) = (\phi_0 \cap \Phi_\infty(N_{LT})) \cap \mathcal{N}_v(N_{LT}) = \phi_0 \cap (\Phi_\infty(N_{LT}) \cap \mathcal{N}_v(N_{LT}))$ and $\phi_0 \cap \mathcal{N}_v(\hat{N}_{LT}) = \phi_0 \cap (\mathcal{N}_v(N_{LT}) \cap \Phi_\infty(N_{LT}))$ since the in-neighbor set of a node v in the active sub-network $\hat{N}_{LT}$ is identical to active in-neighbor set in the steady state of the original network N_{LT}. Therefore, $\phi_0 \cap \mathcal{N}_v(N_{LT}) = \phi_0 \cap \mathcal{N}_v(\hat{N}_{LT})$, and

$$\sum_{u \in \phi_0 \cap \mathcal{N}_v(\hat{N}_{LT})} w_{u,v} = \sum_{u \in \phi_0 \cap \mathcal{N}_v(N_{LT})} w_{u,v} \geq \theta_v$$

which implies $v \in \phi_1(\hat{N}_{LT})$. Then we have $\phi_1(N_{LT}) = \phi_1(\hat{N}_{LT})$.

At time step $t = 2$, for any node v, $v \in \phi_2(N_{LT})$ iff it holds $\sum_{u \in \phi_0 \cup \phi_1(N_{LT}) \cap \mathcal{N}_v(N_{LT})} w_{u,v} \geq \theta_v$. Since $\phi_1(N_{LT}) = \phi_1(\hat{N}_{LT})$, we have $\sum_{u \in \phi_0 \cup \phi_1(\hat{N}_{LT}) \cap \mathcal{N}_v(N_{LT})} w_{u,v} =$ $\sum_{u \in \phi_0 \cup \phi_1(N_{LT}) \cap \mathcal{N}_v(N_{LT})} w_{u,v} \geq \theta_v$ which implies $v \in \phi_2(\hat{N}_{LT})$ and $\phi_2(N_{LT}) = \phi_2(\hat{N}_{LT})$.

Then, at any time step $t \geq 2$, for any node v,

$$\begin{aligned} v \in \phi_t(N_{LT}) \iff & \sum_{u \in \bigcup_{k=0}^{t-1} \phi_k(\hat{N}_{LT}) \cap \mathcal{N}_v(N_{LT})} w_{u,v} \\ = & \sum_{u \in \bigcup_{k=0}^{t-1} \phi_k(N_{LT}) \cap \mathcal{N}_v(N_{LT})} w_{u,v} \\ \geq & \, \theta_v \\ \iff & v \in \phi_t(\hat{N}_{LT}) \\ \iff & \phi_t(N_{LT}) = \phi_t(\hat{N}_{LT}). \end{aligned}$$

This completes the proof.

Since the top-k blockers problem is concerned with blocking some nodes that will be activated if there is no blocking, i.e., $S_b \subseteq \Phi_\infty(N_{LT}, \phi_0)$, and thanks to Theorem 4.1 we have the following result.

Corollary 4.2 *Given an LTM network $N_{LT} = (V, E, \theta, w)$ and seed set $\phi_0 \subseteq V$, the optimal solution to the top-k blockers problem in network N_{LT} is equivalent to the optimal solution in its active sub-network $\hat{N}_{LT} = (\hat{V}, \hat{E}, \hat{\theta}, \hat{w})$.*

Let $\hat{n} = |\Phi_\infty(N_{LT}, \phi_0)|$ denote the number of the final adopters if there is no blocking. Let $\hat{\Theta} \in [0, 1]^{\hat{n} \times \hat{n}}$ and $\hat{W} \in [0, 1]^{\hat{n} \times \hat{n}}$ denote the threshold submatrix and weighted adjacency submatrix of the active sub-network $\hat{N}_{LT}$, respectively.

The characteristic vector $\hat{\boldsymbol{x}}$ of a set $\hat{X} \subseteq \hat{V}$ is defined with the universal set $\hat{V}$, i.e., $\hat{\boldsymbol{x}} \in \{0, 1\}^{\hat{n}}$. For example, assume that $\hat{V} = \{1, 2, 3, 4, 5, 7, 9\}$ and $\hat{X} = \{2, 7, 9\}$. Then its characteristic vector is $\hat{\boldsymbol{x}} = [0, 1, 0, 0, 0, 1, 1]^T$.

The top-k blockers problem can be solved by Algorithm 1. As we can see, (ILP-3) is much easier to solve than (ILP-2) since it has $3\hat{n}$ decision variables and $6\hat{n} + 1$ constraints and normally we have $\hat{n} \ll n$. Acemoglu et al. [8] claim an upper bound on the number of final adopters, i.e., $\hat{n} \leq \sum_{k=1}^{|\phi_0|} |M_k|$, if the

Algorithm 1 Identification of top-k blockers

1: **Input:** An LTM network $N_{LT} = (V, E, \theta, w)$, a seed set $\phi_0 \subseteq V$, constant k
2: **Output:** The optimal blocker set S_b
3: Computing the set of final adopters $\Phi_\infty(N_{LT}, \phi_0)$ without blocking by either Algorithm A1 or solving (ILP-1).
4: Extracting the active sub-network $\hat{N}_{LT} = (\hat{V}, \hat{E}, \hat{\theta}, \hat{w})$.
5: Solving the following integer linear programming:

$$
\begin{aligned}
\max_{\hat{\boldsymbol{x}}, \hat{\boldsymbol{b}}, \hat{\boldsymbol{z}}} \quad & \mathbf{1}^T \hat{\boldsymbol{x}} && (ILP-3) \\
\text{s.t.} \quad & \mathbf{1} - \hat{\boldsymbol{x}} \geq \hat{\boldsymbol{y}} && (a) \\
& \hat{\boldsymbol{x}} \geq \hat{\boldsymbol{b}} && (b) \\
& \mathbf{1}^T \hat{\boldsymbol{b}} \leq k && (c) \\
& [\hat{I} - \hat{\Theta} - \hat{W}^T]\hat{\boldsymbol{x}} - \hat{\boldsymbol{z}} \leq \mathbf{0} && (d) \\
& \hat{\boldsymbol{z}} \leq \hat{\boldsymbol{b}} && (e) \\
& \hat{\boldsymbol{z}} \leq \hat{\boldsymbol{x}} && (f) \\
& \hat{\boldsymbol{z}} \geq \hat{\boldsymbol{b}} + \hat{\boldsymbol{x}} - \mathbf{1} && (g) \\
& \hat{\boldsymbol{x}}, \hat{\boldsymbol{b}}, \hat{\boldsymbol{z}} \in \{0, 1\}^{\hat{n}} && (h)
\end{aligned}
$$

6: The set of final adopters with blocking of nodes in S_b: $\Phi_\infty(N_{LT}, \phi_0, S_b) = \{i \in \hat{V} | \hat{x}_i = 0\}$.
7: Let $S_b = \{i \in \hat{V} | \hat{b}_i = 1\}$.
8: Return S_b.

network N_{LT} can be partitioned into $r \geq |\phi_0|$ disjoint cohesive sets and $\{M_k\}_{k=1}^r$ are in a descending order w.r.t. their cardinalities.

Note that the matrix $\hat{I}$ in constraint (d) of (ILP-3) is no longer an identity matrix, but a diagonal matrix whose diagonal element is the sum of each column of $\hat{W}$, i.e., $\hat{I}(i,i) = \sum_{j \in \hat{V}} \hat{W}(j, i)$.

5 Approximate or Heuristic solutions

To better evaluate our ILP approaches presented above, we also consider some approximate or heuristic approaches for selecting a set of top-k blockers.

5.1 Greedy

Instead of searching for top-k blockers in the entire network whose size n sometimes is very large, we can just focus on the active sub-network in which we can find the same set of blockers thanks to Corollary 4.2. The algorithm first computes the set of final adopters assuming that there is no blocking (line 3) which can be done in time $O(nd)$ by Algorithm A1. We then extract the active sub-network (line 4). In the active sub-network, we select iteratively one blocker that maximally decreases the rumor spread if it is blocked (line 7). The blockers identification process stops until k blockers are found. In each iteration, one needs to determine the final adopter set for any node $i \in \hat{V} \setminus (\phi_0 \cup S_b)$ to be blocked. Such a computation process may require to check all nodes' states and thus this process can be finished in time $O(\hat{n}^2 \hat{d})$ where $\hat{d}$ denotes the mean degree of the active sub-network. Therefore, the Greedy approach shown in Algorithm A2 in the appendix has time complexity of order $O(nd + k\hat{n}^2\hat{d})$.

5.2 Degree

Degree centrality is one kind of simple and widely used measure of node's importance in a network. One's out-degree is exactly the number of its out-neighbors. All the nodes are sorted by their out-degree and top-k nodes excepting the rumor seeds in the active sub-network are selected as the blockers, i.e., the first k nodes in $V_d \cap (\hat{V} \setminus \phi_0)$ where V_d is an ordered set of V according to nodes' degrees. The complexity of computing all nodes' degrees is $O(nd)$.

5.3 Source-aware betweenness centrality (Source-BC)

We then present a heuristic based on betweennesss centrality that is a shortest path based centrality measure.

The fastest known approach [39] to compute the betweenness centrality requires $O(nm)$ time where m denotes the number of edges. However, in order to find the top-k blockers, there is no need to enumerate all the shortest paths between any two nodes as Brandes' algorithm [39] was doing. We are only interested in that which node is passed by the most times from the rumor seeds to all inactive nodes. Therefore, we only need to enumerate the shortest paths from subset S to subset $V \setminus S$. We call this approach *source-aware betweenness centrality.* Denote the source-aware betweenness centrality for any node $u \in V$ as $c_b(u|S)$ that can be computed by:

$$c_b(u|S) = \sum_{s \in S, u \neq t \in V \setminus S} \frac{\sigma_{st}(u)}{\sigma_{st}},$$

where S denotes the set of source nodes.

We can learn from the above analysis that it requires $O(|S|m)$ (equally $O(|S|nd)$) to compute the source-aware betweenness centrality by adopting Brandes' algorithm where $|\cdot|$ is the cardinality of a set. Top-k blockers by this approach are the first k nodes with highest source-aware betweenness centrality in the active sub-network, i.e., the first k nodes in $V_{sbc} \cap (\hat{V} \setminus \phi_0)$ where V_{sbc} is an ordered set of V according to nodes' values of source-aware betweenness centrality.

Table 3
General time complexity of each heuristic approach

Approach	Complexity
Greedy	$O(nd + k\hat{n}^2\hat{d})$
Source-BC	$O(\|S\|nd)$
Degree	$O(nd)$

Note that the solution obtained by the heuristics including Greedy has no approximation guarantee because of the non-submodularity and non-supermodularity of the objective function. In the rest of this work, we denote by ILP the approach that uses the full programming problem (ILP-2) and by R-ILP the approach that uses the reduced programming problem (ILP-3). The numbers of binary decision variables of ILP and R-ILP are $3n$ and $3\hat{n}$, respectively. We expose the general time complexity of the above heuristics in Table 3.

6 Experimental evaluation

6.1 Experimental setup

In this part, we perform a series of experiments on one synthetic network and three real-world networks scaling from small to large to evaluate the performance for our mathematical programming method and three heuristics on solving the top-k blockers problem for the LTM. The synthetic dataset has some specific structural properties as defined below.

- *Scale-free*: A network is called scale-free if the nodes' degrees follow the power-law degree distribution. Typical real-world examples of scale-free networks are collaboration networks, financial networks, etc. A benchmark network of this type with 500 nodes is generated by adopting the Barabási-Albert model [40].

The real-world networks are adopted from SNAP [1] .

- LastFM: This network consists of users of a music website (called LastFM) from Asian countries.
- Deezer: This network consists of Deezer users from European countries.
- DBLP: This is a DBLP co-authorship network in computer science.

Table 4
Datasets

Dataset	Number of nodes	Number of edges	Average degree	Maximum degree
Scale-free	500	3,965	7.93	66
LastFM	7,624	27,806	7.29	216
Deezer	28,281	92,752	6.56	172
DBLP	317,080	1,049,866	6.62	343

Table 4 summarizes the statistical information of these networks that are all assumed to be undirected, i.e., two nodes connected by an undirected arc are in-neighbors of each other. To fairly estimate and compare how the presented approaches perform, for a given rumor seed set ϕ_0 and k, we run them on 10 randomly generated linear threshold networks $N_{LT} = (V, E, \theta, w)$. In each run a set of top-k blockers S_b of size k is determined and we finally take the average value of $|\Phi_\infty(N_{LT}, \phi_0, S_b)|$ over the 10 random LTM networks for each dataset. All LTM networks generated for a given dataset have the same network structure, i.e., V, E, and influence weights w, but different thresholds θ_u that are selected uniformly at random in $(0, 1]$. In addition, we assume that $|\phi_0| = 10$ and k ranges from 1 to 20 for all networks. The rumor seeds are determined by their degrees. That is, rumor spreads from ten nodes with highest degrees in each run. Note that we can change the rumor seed set size to different values without affecting the relative performance on rumor containment and the computational time of all the approaches.

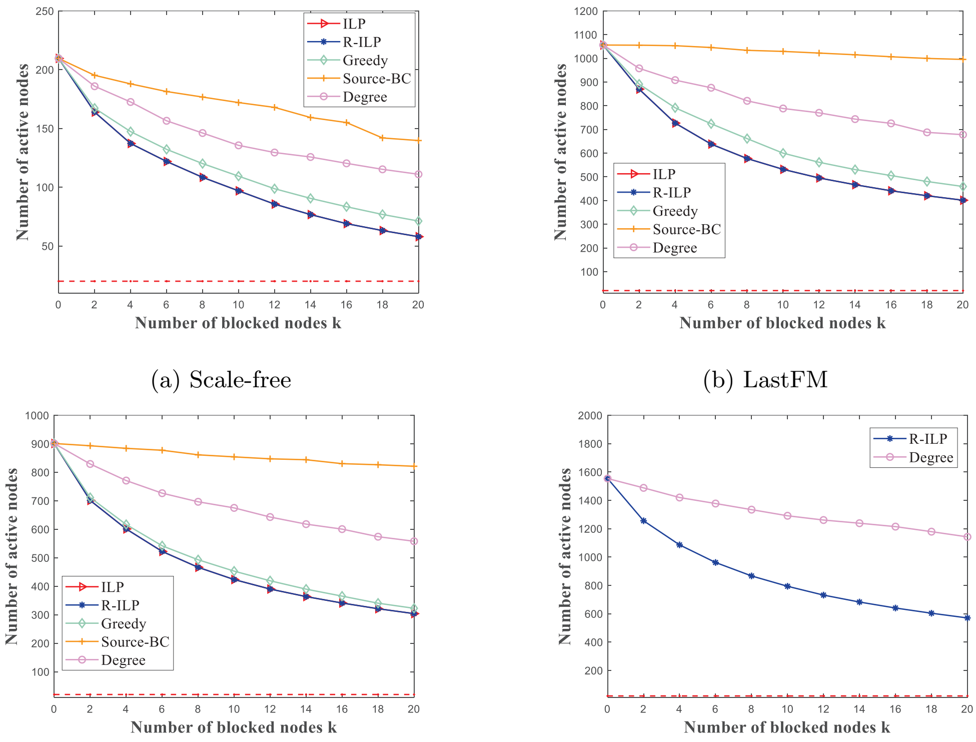
All experiments are coded with MATLAB on a 3.00 GHz Intel(R) Core(TM) i7-9700U CPU and 16GB memory. All integer linear programs are solved using toolbox SCIP, whose academic license is available at https://www.inverseproblem.co.nz/OPTI/index.php/Solvers/SCIP.

6.2 Experimental results and discussions

The five different approaches are evaluated in terms of rumor suppression and their computational time. The rumor spread with different numbers of blockers on four networks with non-uniform weights (that are randomly generated but assure that for all $v \in V$, we have $\sum_{u \in N_v} w_{uv} = 1$) is shown in Fig 6. The computational time and standard deviation for selecting 20 blockers by each approach are shown in Table 5. We make the following observations.

- In small networks, i.e., Scale-free and LastFM, R-ILP runs three to eight times faster than ILP. However, their gap increases with the network size, e.g., R-ILP runs 55 times faster than ILP in medium-sized Deezer network. R-ILP can find an optimal blocker set in less than one second in networks Scale-free, LastFM, and Deezer, where the rumor spread can be reduced by about 70%. In the DBLP network that contains 0.3 million of nodes, ILP and Greedy cannot give a result in a reasonable time, therefore, we only compare R-ILP and Degree which performs better and faster than Source-BC. R-ILP finds the solution in several seconds which reduces the rumor spread by 68%.
- In all networks, Greedy computes a solution approximating the optimal solution obtained by ILP and R-ILP. However, as mentioned before, the objective function $\Phi_\infty(\phi_0, S_b)$ is monotone decreasing but non-supermodular w.r.t. $S_b \subseteq V \setminus \phi_0$. Therefore, the solution obtained by Greedy has no provable approximate guarantee with respect to the optimal solution [1]. Considering its computational time, Greedy runs much slower than R-ILP. and the gap of their computation will increase with the number of nodes in underlying network or its sub-active network. Furthermore, we can see from Table 5 that in terms of computational time Greedy has a large standard deviation. That is to say, Greedy's computational time heavily depends on the size of the sub-active network. In each run for a network with

[1] http://snap.stanford.edu/data/



(a) Scale-free (b) LastFM

(c) Deezer (d) DBLP

Fig. 6. Effectiveness of rumor containment by blocking nodes in different networks with non-uniform weights.

randomly selected thresholds, the number of final adopters may be significantly different and therefore the corresponding computational time changes.

- The centrality based approaches, i.e., Degree and Source-BC, do not perform well on rumor containment even though they run much faster than R-ILP in small network Scale-free. However, their computational time increases with the network size and even surpasses R-ILP in network Deezer. Degree outperforms Source-BC in all networks. In networks Scale-free, LastFM, and Deezer, blocking 20 nodes selected by Degree (resp., Source-BC) can reduce the rumor spread by 47% (resp., 33%), 36% (resp., 6%), and 38% (resp., 9%), respectively.
- The presence of uniform or non-uniform influence weights has no impact on the results of each approach.

Table 5
Mean time and standard deviation over 10 runs for selecting 20 blockers in networks with non-uniform weights

M(±SD) Dataset / Approach	Scale-free	LastFM	Deezer	DBLP
ILP	0.39s(±0.04s)	8.0s(±5.58s)	28.5s(±7.77)s	–
R-ILP	0.18s(±0.03s)	0.98s(±0.93s)	0.52s(±0.11s)	3.09s(±1.78s)
Greedy	26.44s(±13.61s)	5.4min(±2.17min)	74min(±50.51min)	–
Source-BC	0.03s(±0.001s)	0.38s(±0.004s)	1.75s(±0.02s)	–
Degree	0.01s(±0.00s)	0.01s(±0.00s)	0.019s(±0.00s)	0.04s(±0.00s)

7 Conclusion and future work

To identify the top-k blockers for rumor containment, we first propose a non-linear formulation of this problem in linear threshold networks, and then introduce some new decision variables to linearize the non-linear formulation. We can further reduce the complexity of the integer linear programming by showing that given a seed set, the evolution process in the whole network is equivalent to that in its active sub-network. To better evaluate the effectiveness of our improved integer linear programming method, we compared it with a greedy based approach and two centrality based approaches in one synthetic and three real-world networks. We can learn from the experimental results: our integer linear programming approach outperforms greedy and centrality approaches and is applicable for large scale networks due to its reasonable execution time.

We are aware that ILP based solutions do not scale well for very large networks. For this reason our future work will consist in exploring reduced complexity approach for finding solutions close to the optimum by combining the ILP with a divided-and-conquer scheme. Another future direction on rumor containment could take multiplexity into consideration, motivated by the fact that there can be more than one mode of connection between agents in real networks. Social networks are usually coupled since they have some overlapping users. The multiplexity facilitates the information spread in real social scenarios, which makes it much harder to control rumor in multiplex networks[41].

References

[1] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data mining*, Washington, USA, 2003, pp. 137–146.

[2] S. Wen, J. Jiang, Y. Xiang, S. Yu, W. Zhou, and W. Jia, "To shut them up or to clarify: Restraining the spread of rumors in online social networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 12, pp. 3306–3316, 2014.

[3] L. Yang, Z. Li, and A. Giua, "Rumor containment by spreading correct information in social networks," in *Proceedings of 2019 American Control Conference*, Philadelphia, USA, 2019, pp. 5608–5613.

[4] L. Yang, Z. Li, and A. Giua, "Containment of rumor spread in complex social networks," *Information Sciences*, vol. 506, pp. 113–130, 2020.

[5] Y. Lin, X. Wang, F. Hao, Y. Jiang, Y. Wu, G. Min, D. He, S. Zhu, and W. Zhao, "Dynamic control of fraud information spreading in mobile social networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 6, pp. 3725–3738, 2021.

[6] L. Ding, P. Hu, Z.-H. Guan, and T. Li, "An efficient hybrid control strategy for restraining rumor spreading," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 11, pp. 6779–6791, 2021.

[7] L. Zhu, M. Liu, and Y. Li, "The dynamics analysis of a rumor propagation model in online social networks," *Physica A: Statistical Mechanics and its Applications*, vol. 520, pp. 118–137, 2019.

[8] D. Acemoglu, A. Ozdaglar, and E. Yildiz, "Diffusion of innovations in social networks," in *Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference*, Orlando, USA, 2011, pp. 2329–2334.

[9] D. Rosa and A. Giua, "On the spread of innovation in social networks," in *Proceedings of the 4th IFAC Workshop on Distributed Estimation and Control in Networked Systems*, Rhine-Moselle-Hall, Germany, 2013, pp. 322–327.

[10] L. Yang, Z. Li, and A. Giua, "Influence minimization in linear threshold networks," *Automatica*, vol. 100, pp. 10–16, 2019.

[11] L. Yang, A. Giua, and Z. Li, "Minimizing the influence propagation in social networks for linear threshold models," *IFAC-PapersOnline*, vol. 50, no. 1, pp. 14 465–14 470, 2017.

[12] W. Chen, Y. Wang, and S. Yang, "Efficient influence maximization in social networks," in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Paris, France, 2009, pp. 199–208.

[13] W. Chen, Y. Yuan, and L. Zhang, "Scalable influence maximization in social networks under the linear threshold model," in *Proceedings of the 10th IEEE International Conference on Data Mining*, Sydney, Australia, 2010, pp. 88–97.

[14] A. Goyal, W. Lu, and L. V. Lakshmanan, "Simpath: An efficient algorithm for influence maximization under the linear threshold model," in *Proceedings of the 11th IEEE International Conference on Data Mining*, Vancouver, Canada, 2011, pp. 211–220.

[15] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance, "Cost-effective

outbreak detection in networks," in *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Jose, USA, 2007, pp. 420–429.
[16] C. Borgs, M. Brautbar, J. Chayes, and B. Lucier, "Maximizing social influence in nearly optimal time," in *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 2014, pp. 946–957.
[17] G. Song, X. Zhou, Y. Wang, and K. Xie, "Influence maximization on large-scale mobile social network: a divide-and-conquer method," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 5, pp. 1379–1392, 2015.
[18] K. Saito, M. Kimura, K. Ohara, and H. Motoda, "Super mediator–a new centrality measure of node importance for information diffusion over social network," *Information Sciences*, vol. 329, pp. 985–1000, 2016.
[19] R. Narayanam and Y. Narahari, "A shapley value-based approach to discover influential nodes in social networks," *IEEE Transactions on Automation Science and Engineering*, vol. 8, no. 1, pp. 130–147, 2011.
[20] K. Kandhway and J. Kuri, "Using node centrality and optimal control to maximize information diffusion in social networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 7, pp. 1099–1110, 2016.
[21] M. Gong, J. Yan, B. Shen, L. Ma, and Q. Cai, "Influence maximization in social networks based on discrete particle swarm optimization," *Information Sciences*, vol. 367, pp. 600–614, 2016.
[22] Bogus ap tweet about explosion at the white house wipes billions off us markets. [Online]. Available: https://www.telegraph.co.uk/finance/markets/10013768
[23] X. He, G. Song, W. Chen, and Q. Jiang, "Influence blocking maximization in social networks under the competitive linear threshold model," in *Proceedings of the 2012 SIAM Internatinal Conference on Data Mining*, Anahelm, USA, 2012, pp. 463–474.
[24] 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 International Conference on Computational Social Networks*, Beijing, China, 2015, pp. 35–47.
[25] C. Budak, D. Agrawal, and A. El Abbadi, "Limiting the spread of misinformation in social networks," in *Proceedings of the 20th International Conference on World Wide Web*, Hyderabad, India, 2011, pp. 665–674.
[26] G. Tong, W. Wu, L. Guo, D. Li, C. Liu, B. Liu, and D.-Z. Du, "An efficient randomized algorithm for rumor blocking in online social networks," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 2, pp. 845–854, 2020.
[27] M. A. Manouchehri, M. S. Helfroush, and H. Danyali, "Temporal rumor blocking in online social networks: A sampling-based approach," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, no. 7, pp. 4578–4588, 2022.
[28] M. Kimura, K. Saito, and H. Motoda, "Blocking links to minimize contamination spread in a social network," *ACM Transactions on Knowledge Discovery from Data*, vol. 3, no. 2, pp. 1–23, 2009.
[29] E. Khalil, B. Dilkina, and L. Song, "Cuttingedge: influence minimization in networks," in *Proceedings of Workshop on Frontiers of Network Analysis: Methods, Models, and Applications at NIPS*, 2013, pp. 1–13.
[30] R. Yan, Y. Li, W. Wu, D. Li, and Y. Wang, "Rumor blocking through online link deletion on social networks," *ACM Transactions on Knowledge Discovery from Data*, vol. 13, no. 2, pp. 1–26, 2019.
[31] H. T. Nguyen, A. Cano, T. Vu, and T. N. Dinh, "Blocking self-avoiding walks stops cyber-epidemics: a scalable gpu-based approach," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 7, pp. 1263–1275, 2019.
[32] A. K. Nandi and H. R. Medal, "Methods for removing links in a network to minimize the spread of infections," *Computers & Operations Research*, vol. 69, pp. 10–24, 2016.
[33] S. Medya, A. Silva, and A. Singh, "Approximate algorithms for data-driven influence limitation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 6, pp. 2641–2652, 2022.
[34] B. Wang, G. Chen, L. Fu, L. Song, and X. Wang, "Drimux: Dynamic rumor influence minimization with user experience in social networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 10, pp. 2168–2181, 2017.

[35] S. Wang, X. Zhao, Y. Chen, Z. Li, K. Zhang, and J. Xia, "Negative influence minimizing by blocking nodes in social networks," in *Proceedings of Workshops at the 27th AAAI Conference on Artificial Intelligence*, Bellevue, USA, 2013, pp. 134–136.

[36] R. Yan, D. Li, W. Wu, D.-Z. Du, and Y. Wang, "Minimizing influence of rumors by blockers on social networks: Algorithms and analysis," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 3, pp. 1067–1078, 2020.

[37] S. Saha, A. Adiga, B. A. Prakash, and A. K. S. Vullikanti, "Approximation algorithms for reducing the spectral radius to control epidemic spread," in *Proceedings of the 2015 SIAM International Conference on Data Mining*, Vancouver, Canada, 2015, pp. 568–576.

[38] J. Zhu, P. Ni, G. Wang, and Y. Li, "Misinformation influence minimization problem based on group disbanded in social networks," *Information Sciences*, vol. 572, pp. 1–15, 2021.

[39] U. Brandes, "A faster algorithm for betweenness centrality," *Journal of Mathematical Sociology*, vol. 25, no. 2, pp. 163–177, 2001.

[40] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[41] A. I. E. Hosni, K. Li, and S. Ahmad, "Minimizing rumor influence in multiplex online social networks based on human individual and social behaviors," *Information Sciences*, vol. 512, pp. 1458–1480, 2020.

Algorithm A1 Computation of $\Phi_\infty(N_{LT}, \phi_0)$

Input: An LTM network $N_{LT} = (V, E, \theta, w)$, a seed set $\phi_0 \subseteq V$
Output: The set of final adopters $\Phi_\infty(N_{LT}, \phi_0)$
Let $\Phi = \phi_0$, $\Phi_c = V \setminus \phi_0$, $\Phi_{old} = \emptyset$.
Let $t = 0$.
while $\phi_t \neq \emptyset$ **do**
    Let $t = t + 1$, $\phi_t = \emptyset$.
    Let $\Phi_{old} = \Phi$.
    for $v \in \Phi_c$ **do**
        if $\sum_{u \in \Phi_{old} \cap \mathcal{N}_v} w_{uv} \geq \theta_v$ **then**
            $\phi_t = \phi_t \cup \{v\}$.
        end if
    end for
    Let $\Phi = \Phi \cup \phi_t$.
    Let $\Phi_c = \Phi_c \setminus \phi_t$.
end while
Let $\Phi_\infty(N_{LT}, \phi_0) = \Phi$.

Algorithm A2 Greedy approach

Input: An LTM network $N_{LT} = (V, E, \theta, w)$, a seed set $\phi_0 \subseteq V$, constant k
Output: The blocker set S_b
Computing the set of final adopters without blocking, i.e., $\Phi_\infty(N_{LT}, \phi_0)$, by Algorithm A1.
Extracting the active sub-network $\hat{N}_{LT} = (\hat{V}, \hat{E}, \hat{\theta}, \hat{w})$.
Initialize $S_b \leftarrow \emptyset$.
while $|S_b| \leq k$ **do**
    $u = \underset{i \in \hat{V} \setminus (\phi_0 \cup S_b)}{argmin} \; \Phi_\infty(\hat{N}_{LT}, \phi_0, S_b \cup \{i\})$.
    $S_b = S_b \cup \{u\}$.
end while
Return S_b.