Chapter 4
Rumor Blocking in Social Networks

4.1 Overview

Online social networks have many benefits as a medium for fast, widespread information dissemination. They provide fast access to large scale news data, sometimes even before the mass media. They also serve as a medium to collectively achieve a social goal. For instance with the use of group and event pages in Facebook, events such as Day of Action protests reached thousands of protestors [71]. While the ease of information propagation in social networks can be very beneficial, it can also have disruptive effects. One such example was observed in August, 2012, thousands of people in Ghazni province left their houses in the middle of the night in panic after the rumor of earthquake [127]. Another example is the fast spread of misinformation in twitter that the president of Syria is dead, leading to a sharp, quick increase in the price of oil [208]. There are lots of similar examples. Although social networks are the main source of news for many people today, they are not considered reliable due to such problems.

Clearly, in order for social networks to serve as a reliable platform for disseminating critical information, it is necessary to have tools to limit the effect of misinformation or rumors. Existing work in controlling rumor spread includes [30, 62, 63, 84, 98, 147]. In [98], Kimura et al. proposed to block a certain number of links in a network to reduce the bad effects of rumors. In the presence of a misinformation cascade, [30, 62, 63, 84, 147] aim to find a near-optimal way of disseminating good information that will minimize the devastating effects of a misinformation campaign. For instance, [84] seeks ways of making sure that most of the users of the social network hear about the correct information before the bad one, making social networks a more trustworthy or reliable source of information.

Related Work The identification of influential users in a social network is a problem that has received significant attention in recent research. For the influence maximization problem, given a probabilistic model of information diffusion such

W. Xu, W. Wu, *Optimal Social Influence*, SpringerBriefs in Optimization,
https://doi.org/10.1007/978-3-030-37775-5_4

as the independent cascade model, a network graph, and a budget k, the objective is to select a set S of size k for initial activation so that the expected value of $f(S)$ (size of cascade created by selecting seed set A) is maximized [37–39, 96, 97, 122, 139, 197, 207]. For learning more about this topic, please refer to our related book chapter of influence maximization.

In contrast to influence maximization problem which studies single-cascade influence propagation (only one kind of influence diffuses in a social network), there is a series of work that focus on multiple-cascade influence diffusion in social networks. Bharathi et al. [19] explored the multiple-cascade influence diffusion under the extension of the independent cascade (IC) model. In [26], Borodin et al. studied the multiple-cascade influence diffusion in several different models generated from the linear threshold (LT) model. In [190], Trpevski et al. proposed a two-cascade influence diffusion model based on the SIS (susceptible-infected-susceptible) model. Kostka et al. [104] considered the two-cascade influence diffusion problem from a game-theory aspect, where each cascade tries to maximize their influence among the social network. Then they studied it under a more restricted model than the IC model and the LT model. To learn more about the study in game-theoretic models where multiple decision-makers try to maximize their own objectives at the same time, interested readers are referred to [5, 33, 147, 191, 192, 195].

Among multiple-cascade influence diffusion, there is a special kind, rumor control related problem, in which there are only two kinds of cascades, one is called positive cascade, while the other is called negative cascade. The goal is to use the positive cascade diffusion to fight against the negative cascade diffusion. Budak et al. [30] and He et al. [84] focused on the problem: given a set of initial "bad" seeds, how to optimally choose the initial set of "good" seeds to limit the diffusion of their influence? In [30], the authors proved the NP-hardness of this optimization problem under the generalized IC model. They also established the submodularity of the objective function and therefore, the greedy algorithm was used as a constant-factor approximation algorithm. In [84], He et al. proposed a competitive linear threshold (CLT) model. They proved that the objective function is submodular and obtained a $(1-1/e)$-approximation ratio. To overcome the inefficiency of the greedy algorithm, they designed a heuristic algorithm which uses the local structure of the network.

Extending both the IC and LT models to two-cascade information diffusion model with a time deadline, Nguyen et al. in [147] studied the following problem: given bad influence sources, how to select the least number of nodes as good influence sources to limit bad influence propagation in the entire network, such that after T steps, the expected number of infected nodes is at most $1 - \beta$. The authors demonstrated several hardness results and proposed effective greedy algorithms and heuristic algorithms.

In this chapter, we will introduce two recent works about rumor blocking or rumor control in detail, including *Community-Based Least Cost Rumor Blocking* (Sect. 4.2) [63] and *Rumor Blocking Maximization with Constrained Time* (Sect. 4.3) [62].

To efficiently decontaminate the wide spread of rumors in a network, in Sect. 4.2, attention is drawn to exploiting communities, i.e., confine the rumor diffusion to its own community. In this work, we propose to initiate protectors to fight against rumors in social networks. That is, we select some individuals as initial protectors, let them spread true or credible information. Correspondingly, some individuals will be protected from rumors. In specific, we focus on protecting bridge ends using certain number of protectors. Here, bridge ends are boundary nodes of communities, which have relations with members in rumor community, and can be reached by rumors at an earlier stage.

In Sect. 4.3, we investigate the problem: given the number of initial protectors and deadline, how to select initial protectors such that the number of "really" protected members in social networks is maximized within deadline. We propose two models to capture competitive influence diffusion process, namely the Rumor-Protector Independent Cascade model with Meeting events (RPIC-M) model and the Rumor-Protector Linear Threshold model with Meeting events (RPLT-M) model. Three features are included in these two models: a time deadline, random time delay between information exchange, and personal interests regarding the acceptance of information. Under these two models, we study the Rumor Containment maximization with the constraints: time Deadline, Meeting events, and Personal interests (RC-DMP problem). We prove that the problem under these two models is both NP-hard. Moreover, we demonstrate that the objective functions for the problem under the two different models are both monotone and submodular. Therefore, we apply the greedy algorithm as a constant-factor approximation algorithm with performance guarantee ratio of $1 - \frac{1}{e}$.

In the last section, we will summarize our work of rumor blocking and future work in this field will also be discussed.

4.2 Community-Based Least Cost Rumor Blocking

We assume that rumors and protectors start diffusing at the same time, and also follow a same diffusion mechanism. Each node can only be in one of the three statuses: *protected*, *infected*, or *inactive*. When the two cascades, namely *cascade P* for protector and *cascade R* for rumor, arrive a node at the same time, we say that cascade P has priority over cascade R, in other words, the node is protected. By considering the community property of a social network, we identify certain kinds of nodes, which are located in boundaries of communities, as protection targets. Then the goal of the *Rumor Control (RC)* problem is to find the minimal number of initial protectors to protect certain fraction of these nodes. This is a novel perspective in constraining rumor dissemination.

The authors in [21] found that a social network is composed of a set of disjoint communities, and members in a same community have similar interests. Furthermore, we have the common knowledge that most of the time, rumors originate from individuals with similar interests. Therefore, we assume that rumors

originate from a same community of a social network. According to the community property that connections among individuals in a same community are denser than that across different communities, we know that influence spreads faster within a same community, while slower across different communities.

To simplify the description, we name the community that contains rumors as *rumor community* and a neighbor community of rumor community as a *R-neighbor community*. Considering the number of nodes that we can protect and the number of nodes that we need to use as initial protectors, it is practical for us to protect the members in R-neighbor communities. We focus on those nodes that exist in R-neighbor communities, and also can be reached first when cascade R arrives in their own communities. We name them as *bridge ends*.

We study the RC problem under an influence diffusion model: the *Deterministic One-Activate-Many* (DOAM) model. Considering the budget for launching the initial protectors, in the DOAM model, we focus on the *RC-D* problem, where we need to protect all the bridge ends.

Through proving the equivalence between the RC-D problem and the Set Cover (SC) problem, we propose the *Set Cover Based Greedy* (SCBG) algorithm. Then we demonstrate that there is no polynomial time $o(\ln n)$-approximation for the RC-D problem unless $P = NP$, and get a $O(\ln n)$-approximation ratio solution. Finally, we collect real-world data to validate our algorithms, and the experimental reports demonstrate that both the Greedy algorithm and the SCBG algorithm outperform the other two heuristics, respectively.

The rest of this section is organized as follows: in Sect. 4.2.1, we propose an influence diffusion models, namely, the DOAM model. In Sect. 4.2.2, we formulate the RC-D problem under the proposed model. In Sect. 4.2.3, as for the DOAM model, we prove that there is no polynomial time $o(\ln n)$-approximation for the RC-D problem unless $P = NP$, and propose the SCBG algorithm. In Sect. 4.2.4, we compare our algorithms with other heuristics and analyze the experimental results.

4.2.1 Deterministic One-Activate-Many (DOAM) Propagation Model

A social network can be modeled as a directed graph $G = (V, E)$, in which V denotes the node set and E denotes the edge set. In the context of influence diffusion, V represents the individuals in this network and E represents the relationships among these individuals. Furthermore, a node $u \in V$ is an in-neighbor of a node $v \in V$ if there exists an edge $e_{uv} \in E$ (i.e., the edge from u to v exists in graph G). A node v is called an out-neighbor of u if u is an in-neighbor of v. Based on this special structure, influence can diffuse among individuals in social networks. Since under different situations, influence spreads with different mechanisms. In our paper, we introduce a new influence diffusion model.

Here, we first introduce some denotations and three properties of the model. Let R represent rumor cascade and P denote protector cascade. A node is said to be *infected* (*protected*) if it is influenced by *rumors* (*protectors*) either initially or sequentially from one of its neighbors, or *inactive* otherwise. We also denote the initial set of infected (protected) nodes for R (P) as S_r (S_p).

Right now, we introduce three properties of the proposed model: (1) There are two kinds of cascades R (for rumor) and P (for protector); (2) when R and P reach a node u at the same time, P has the priority to influence u, meaning u will always be protected; (3) R or P diffuses *progressively*, that is, nodes can switch from inactive to infected or protected, but cannot switch in the other direction, that is, once an inactive node is infected or protected, it will never change its status. Property (1) makes sense since it happens in reality. Property (2) is reasonable since people are likely to believe in the truth. Property (3) originates from [93].

In the following, we describe the proposed model in detail.

Given the initial rumor set S_r, an initial protector set S_p is selected and protected at step $t = 0$. At any step $t \geq 1$, when a node u first becomes infected (protected), it will infect (protect) all of its currently inactive out-neighbors successfully. And u only has one chance to influence its out-neighbors, that is, at step $t + 1$, u will not influence its out-neighbors. This influence diffuses in discrete time and continues until no new inactive nodes become protected or infected.

This influence propagation process is actually the information broadcast (one-to-many) phenomenon in social networks, under which situation, each person is able to spread the information to many persons simultaneously. Obviously, the information diffusion speed under the DOAM model is very fast.

4.2.2 Rumor Control Problem

In this section, we define the Rumor Control (RC) problem in social networks. It is known that a social network is composed of individuals and connections between individuals. We notice that social networks have community property, that is, they divide into groups of members, where connections within the same group are dense while across different groups are sparse. It is common sense that individuals form communities based on their common interests, and they are more likely to communicate with members in their own communities than with members in other communities. Therefore, the connections within the same communities are dense while across different communities are sparse. Thus, it is impossible that information can spread fast from one community to other communities.

Based on the community property, to efficiently control the wide spread of rumors originated from one community, we try to prevent them from spreading out to other communities. To realize it, we only need to protect all the members in R-neighbor communities. Bridge ends are the nodes that exist in R-neighbor communities and can be reached first when cascade R arrives in their own communities. Therefore, to protect all the members in R-neighbor communities, it is enough to protect all the bridge ends.

In the following, we give some problem-related definitions and the formal definitions of our problems.

Definition 4.1 A social network is a directed graph $G(V, E, C)$, where each node $v_i \in V$ denotes an individual in the network, and a directed edge $(v_i, v_j) \in E$ denotes the event that individual v_i has influence on individual v_j. Here $C = \{C_1, C_2, \cdots, C_k\}$ is a set of disjoint communities that form the network, satisfying $\bigcup_{r=1}^{k} V(C_r) = V$, where $V(C_r)$ denotes the individuals in community C_r.

Definition 4.2 Rumor Control (RC) problem: Given a community C_k in $G(V, E, C)$, an initial rumor set $S_r \subseteq V(C_k)$ ($C_k \in C$ is the rumor community and is predetermined), and bridge ends B, our goal is to select a least number of nodes as the initial protectors, such that at least α ($0 \leq \alpha \leq 1$) fraction of the bridge ends are protected in the end of influence diffusion.

Considering the influence propagation speed under the DOAM model, we introduce the RC-D problem for the DOAM model. It is because under the DOAM model, rumors propagate very fast in a social network. In other words, within short time, rumors can infect a large amount of individuals in a social network. Considering the budget in launching the initial protectors, the goal of the problem requires to protect all the bridge ends.

Definition 4.3 The RC-D problem: Given a community C_k in $G(V, E, C)$, an initial rumor set $S_r \subseteq V(C_k)$ ($C_k \in C$ is the rumor community and is predetermined), and bridge ends B, under the DOAM model, our goal is to select a least number of nodes as the initial protectors, such that all the bridge ends ($\alpha = 1$ in the RC problem.) are protected in the end of influence diffusion.

Since the *Set Cover (SC)* problem will be used in the RC-D problem, here we give its definition below.

Definition 4.4 Set Cover (SC) Problem: Given a set of elements $U = \{v_1, v_2, \cdots, v_n\}$ and a set of m subsets of U, called $S = \{S_1, S_2, \cdots, S_m\}$, find a "least cost" (minimum size) collection $\mathscr{C}$ of sets from S such that $\mathscr{C}$ covers all the elements in U. That is, $\bigcup_{S_i \in \mathscr{C}} S_i = U$.

4.2.3 Set Cover Based Greedy Algorithm for DOAM Model

In this section, we first prove that under the DOAM model, the RC-D problem is equivalent to the SC problem. Following the seminal result of [64] that the SC problem is NP-hard, we propose an approximation algorithm called *Set Cover Based Greedy* (SCBG) algorithm for the RC-D problem.

In the following, we show the equivalence between the RC-D problem and the SC problem under the DOAM model, and subsequently, we propose the SCBG algorithm for the RC-D problem.

4.2.3.1 Performance for the RC-D Problem

Theorem 4.1 ([64]) *There is a polynomial time $O(\ln n)$-approximation algorithm for the RC-D problem, where n is the number of bridge ends B.*

Proof Assume that we have an input of RC-D instance $\mathscr{A}$. For each vertex v_i of B, use BFS (Breadth First Search) method to find all vertices that can reach v_i before v_i is infected, this can be done in polynomial time. Assume we have a candidate root set S, for each vertex r_j in S, use BFS method to find all vertices of B that are reachable from r_j before they are infected. Obviously, each root can protect a subset of vertices of B, then the problem becomes a SC problem, i.e., use the least number of roots to cover all vertices of B. Therefore, it has a polynomial time $O(\ln n)$-factor approximation, where n is the number of nodes in B.

Theorem 4.2 *If the RC-D problem has an approximation algorithm with ratio $k(n)$ if and only if the SC problem has an approximation algorithm with ratio $k(n)$.*

Proof Assume $S_1, \cdots, S_m$ is the list of sets for the SC problem and $S_1 \bigcup S_2 \bigcup \cdots \bigcup S_m = \{a_1, \cdots, a_n\}$, we construct a social network as follows.

1. For each set S_i, create a vertex u_i. For each a_j, create a vertex v_j, add directed edges from u_i to v_j if $a_j \in S_i$. An edge from u_i to v_j means v_j can be protected by u_i.
2. Create a social network with a constant number of individuals and an infected node r, add directed edges from r to $v_1, v_2, \cdots, v_n$.
3. Let B be the set of bridge ends including vertices $v_1, v_2, \cdots, v_n$ that need to be protected.
4. The SC problem is converted into the RC-D problem. Thus, it is reasonable to point out that the RC-D problem has a $k(n)$-approximation if and only if the SC problem has a $k(n)$-approximation.

Corollary 4.1 *There is no polynomial time $o(\ln n)$-approximation for the RC-D problem unless $P = NP$.*

Proof It follows from Theorem 4.2 and the well-known inapproximability result for the SC problem [64].

4.2.3.2 The SCBG Algorithm

Now we introduce the SCBG algorithm described in Algorithm 2. The main idea is that we first convert the RC-D problem into the SC problem, then, we apply the greedy algorithm used for the SC problem to obtain the initial protectors for our problem.

The brief description is as follows: given the initial rumor set S_r and bridge end set B. For each node $v \in B$, by using BFS method, we construct v's Bridge End Backward Search Tree (BBST) T_v, in which v is the root of the tree. Denote by $T_1, T_2, \cdots, T_{|B|}$ the BBSTs for corresponding bridge ends. Here $1, 2, \cdots, |B|$

represent the roots of these T_vs, respectively. For $u \in T_i \backslash (S_r \bigcup_{k=0}^{i-1} T_k)$, define $T_0 = \emptyset$, search $T_{i+1}, \cdots, T_{|B|}$ to find the ones that contain u, and record all their corresponding roots and i's corresponding root in the set TR_u. Finally, we apply Algorithm 1 to select sets from TR_u's to cover all the nodes in B, and for all these selected sets TR_vs, all these v's form the solution to the RC-D problem.

To simplify our expression, here we define the ith level out-neighbors of a node u: let $N^0(u) = u$, and $N^i(u) = N(N^{i-1}(u))$. Since we know the first level out-neighbors of a node, we can easily get the ith level out-neighbors of a node.

Algorithm 1 Greedy algorithm in SCBG algorithm

Input: B, T_i and TR_j, where $i = 1, \cdots, |B|$, $j = 1, \cdots, |\bigcup_{k=1}^{|B|} T_k \setminus S_r|$
Output: S_p.

```
Initialize L = ∅ and S_p = ∅
while |L| < |B| do
    Select u = arg max_{v ∈ ⋃_{k=1}^{|B|} T_k \ S_r} |TR_v \ L|
    S_p = S_p ∪ {u} and L = L ∪ TR_u
end while
return S_p.
```

Algorithm 2 SCBG algorithm-select initial protectors

Input: A directed graph $G = (V, E, C)$, a given community C_m and a set of initial rumors $S_r = \{r_1, r_2, \cdots, r_M\} \subseteq V(C_m)$;
Output: Initial protectors $S_p \subseteq V$;

```
for all r ∈ S_r do
    construct Rumor Forward Search Tree (RFST) by BFS method to find all bridge ends in G,
    which are the leaves of the RFSTs, and denote them by a set B;
end for
for all node v ∈ B do
    construct Bridge End Backward Search Tree (BBST) by BFS method to find all the protector
    candidates,
    record all the in-neighbors x ∈ N^i(v) of v, where i is determined by the value of the shortest
    paths between v and any node y ∈ S_r,
    Denote all the nodes in this tree as a set T_v;
end for
List all T_v s as T_1, ··· , T_{|B|}.
for all u ∈ T_i \ (S_r ⋃_{k=0}^{i-1} T_k) do
    define T_0 = ∅,
    search T_{i+1}, ··· , T_{|B|} to find the ones that contain u,
    record all their corresponding roots and i's corresponding root in the set TR_u;
end for
Apply Algorithm 1 on TR_u s to cover B;
return Output of Algorithm 1.
```

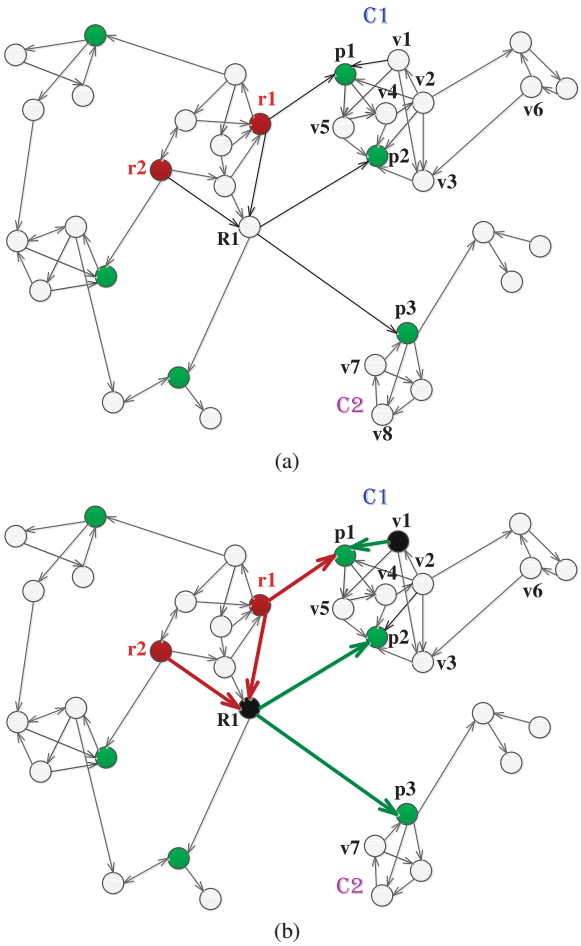

Fig. 4.1 (**a**) Rumor community and its R-neighbor communities, red nodes are rumors and green nodes are bridge ends; (**b**) Initial protectors v1 and R1 for bridge ends in R-neighbor communities C1 and C2

We use Fig. 4.1 to show the bridge ends and the corresponding initial protectors for them. In Fig. 4.1a, the red nodes $r1$ and $r2$ are initial rumors. All green nodes are bridge ends. In Fig. 4.1b, for simplification, we only illustrate an optimal initial protectors for R-neighbor communities $C1$ and $C2$, respectively, which are black vertices $R1$ and $v1$. As seen from Fig. 4.1b, among rumor community and its two R-neighbor communities $C1$ and $C2$, the green edges form the paths generated by cascade P ($R1$ and $v1$ are the initial protectors), while the red edges form the paths generated by cascade R ($r1$ and $r2$ are the initial rumors). Figure 4.2a is Forward search tree for rumor $r1$ with respect to Fig. 4.1a, and Fig. 4.2b is Backward search tree for bridge end $p2$ with respect to Fig. 4.1a.

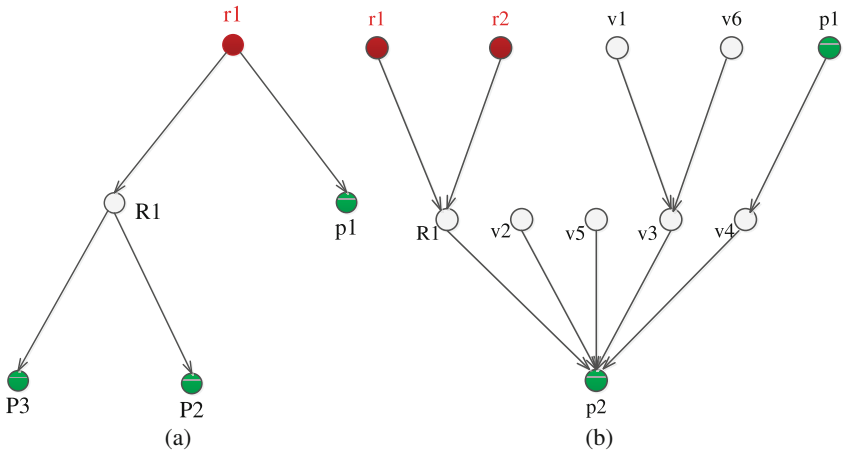

Fig. 4.2 (**a**) Forward search tree for rumor $r1$ with respect to Fig. 4.1a, and bridge ends are $p1$, $p2$, $p3$; (**b**) Backward search tree for bridge end $p2$ with respect to Fig. 4.1a, all nodes in this tree except $r1$ and $r2$ can protect $p2$

4.2.4 Experiment Setup and Evaluation

We execute experiments on our algorithms as well as two heuristics in two real-world networks. Our experiments aim at valuating our algorithms from the following aspects: (a) effectiveness with respect to different network density, where network density means the average node degree; (b) effectiveness with respect to different community size, where community size denotes the number of nodes in this community; (c) effectiveness with respect to different number of initial rumors.

4.2.4.1 Datasets

We obtain data from two real-world networks. One network, namely Enron Email communication network, is the same as used in [103, 116]. The other is a collaboration network, which is used in the experimental study in [114], and this network has been shown to capture many key features of social networks in [143].

Enron Email Communication Network

This network covers all the email communications within a dataset of around half million emails. Nodes of the graph represent email addresses and a directed edge from i to j means i sends at least one email to j. This dataset contains 36,692 nodes connected by 367,662 edges with an average node degree of 10.0.

Collaboration Network

Hep collaboration network is extracted from the e-print arXiv, and covers scientific collaborations between authors with papers submitted to High Energy Physics. In this network, nodes stand for authors and an undirected edge between i and j implies that i co-authors a paper with j. Since our problems are based on directed graph, we represent each undirected edge (i, j) by two directed edges (i, j) and (j, i). This dataset contains 15,233 nodes connected by 58,891 edges with an average node degree of 7.73.

To run our experiments, first, we need to obtain the community structure of a social network, since the community partition problem is not a main point in our work, we use a community partition approach proposed by Blondel et al. in [21], and the performance of this approach has been verified in [110]. After obtaining the community structure of a network, we choose different sizes of rumor communities and compute the number of corresponding bridge ends from the two networks. From the Enron Email network, we select two communities, one with 2631 nodes and 2250 bridge ends, and the other with 80 nodes and 135 bridge ends. From the collaboration network, we select a community with 308 nodes and 387 bridge ends.

Finally, we evaluate the performance of our algorithms in comparison with two heuristics: MaxDegree and Proximity. The experimental results are shown in two aspects: (1) Number of selected protectors under the DOAM model; (2) Number of infected nodes under the DOAM model.

We compared the following algorithms to confirm the effectiveness of our algorithms.

MaxDegree A basic algorithm, which simply chooses the nodes according to the decreasing order of node degree as the initial protectors.

Proximity A simple heuristic algorithm, in which the direct out-neighbors of rumors are chosen as the initial protectors.

We do not include the random algorithm due to its poor performance. Instead, a NoBlocking line is included to reflect the performances of these algorithms.

4.2.4.2 Experimental Results

To simplify our presentation, we denote by $|R|$ the number of the initial rumors, $|P|$ the number of the initial protectors, $|C|$ the number of nodes in the rumor community, $|B|$ the number of the bridge ends, $|N|$ the number of nodes in the entire network. To show the simulation results clearly, we adopt the log-time chart.

Table 4.1 Comparison results under the DOAM model

Dataset/$\|N\|/\|C\|$	$\|R\|$	SCBG	Proximity	MaxDegree
Hep/15233/308	$1\%\|C\|$	32.9	25.3	140.6
	$5\%\|C\|$	42.1	74.3	147.8
	$10\%\|C\|$	48.9	133.8	152.6
Email/36692/80	$5\%\|C\|$	6.2	43.7	72.7
	$10\%\|C\|$	8.2	46.9	79.3
	$20\%\|C\|$	13.8	62.9	91.1
Email/36692/2631	$1\%\|C\|$	20.4	289.3	1208.8
	$5\%\|C\|$	50.9	1067.6	1350.2
	$10\%\|C\|$	68.4	1422.6	1683.8

Number of Selected Protectors Under the DOAM Model

In Table 4.1, for each rumor community and fixed number of initial rumors (selected randomly), each decimal represents the average number of initial protectors selected by each algorithm (we randomly choose initial rumors for several times and each time we can get a solution). You can see that our SCBG algorithm almost selects the least number of initial protectors no matter where the community is selected and how many initial rumors in it. There is only one exception, in which the rumor community is selected from the Hep network, and has 308 nodes with 3 initial rumors. The reason is that the average node degree is low in Hep network. When the number of initial rumors is pretty small, only a few initial protectors are needed to control the spread of these rumors. Therefore, choosing the direct neighbors of initial rumors is an efficient strategy, that is, Proximity is a good choice.

Furthermore, we also notice that Proximity always performs better than MaxDegree, it is because that Proximity pay attention to the location of initial rumors, thus it can control rumor propagation before they infect a large number of nodes; while MaxDegree only focuses on current influential nodes (nodes having high degree) regardless of the initial rumors. Therefore, it has to choose more initial protectors than Proximity under regular situations. Meanwhile, we also observe that the performance difference among these three algorithms varies under different situations.

Note that among these three communities, the number of initial protectors selected by our algorithms varies much less than that in the other two heuristics. Particularly, in the third community, which is selected from the email network, and has 2631 nodes, when the number of initial rumors increases from 27 ($1\%|C|$) to 132, the number of initial protectors selected by our algorithm increases from 20.4 to 50.9 (average value), with the absolute change of 30.5. However, the change in the number of initial protectors is 778.3 and 141.4 for Proximity and MaxDegree, respectively. The results in this community clearly shows that the SCBG algorithm significantly outperforms both Proximity and MaxDegree in networks with large number of nodes and high average node degree.

Number of Infected Nodes Under the DOAM Model

In this part, we focus on testing the effectiveness of these algorithms in protecting nodes in the entire social networks. In other words, for the same number of initial protectors, we want to evaluate the performance of these algorithms. To do this, firstly, for different test cases (different community sizes with different initial rumor sizes), we determine the numbers of initial protectors, respectively, and these numbers are slightly larger than those selected by the SCBG algorithm. Then, for each test, from corresponding solutions, we randomly choose predetermined size of nodes as initial protectors. Thirdly, we run the three algorithms using selected initial protectors. Since each predetermined number is larger than the number of nodes selected by our algorithm, besides using the nodes in its solution, our algorithm also has to use some randomly selected nodes. From Figs. 4.3, 4.4, and 4.5, we observe that rumors propagate very fast within the first four steps while after the fourth step, almost no new nodes are infected over all test cases.

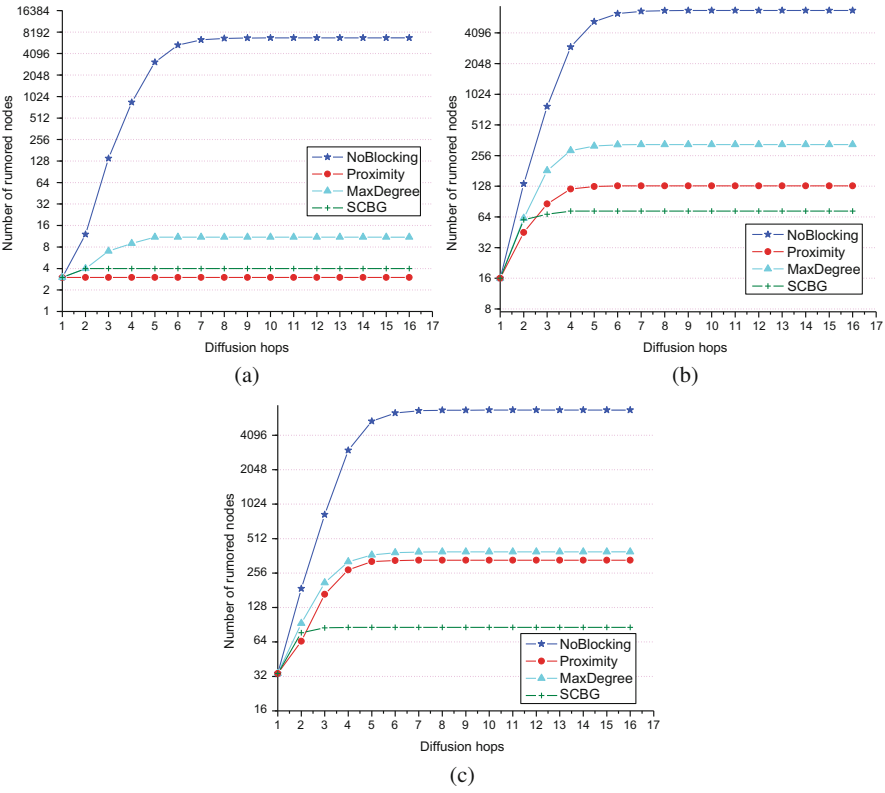

Fig. 4.3 Infected nodes under the DOAM model on Hep collaboration network with $|N| = 15{,}233$, $|C| = 308$, $|B| = 387$. (**a**) $|R| = 1\%|C|$, $|P| = 34$. (**b**) $|R| = 5\%|C|$, $|P| = 44$. (**c**) $|R| = 10\%|C|$, $|P| = 55$

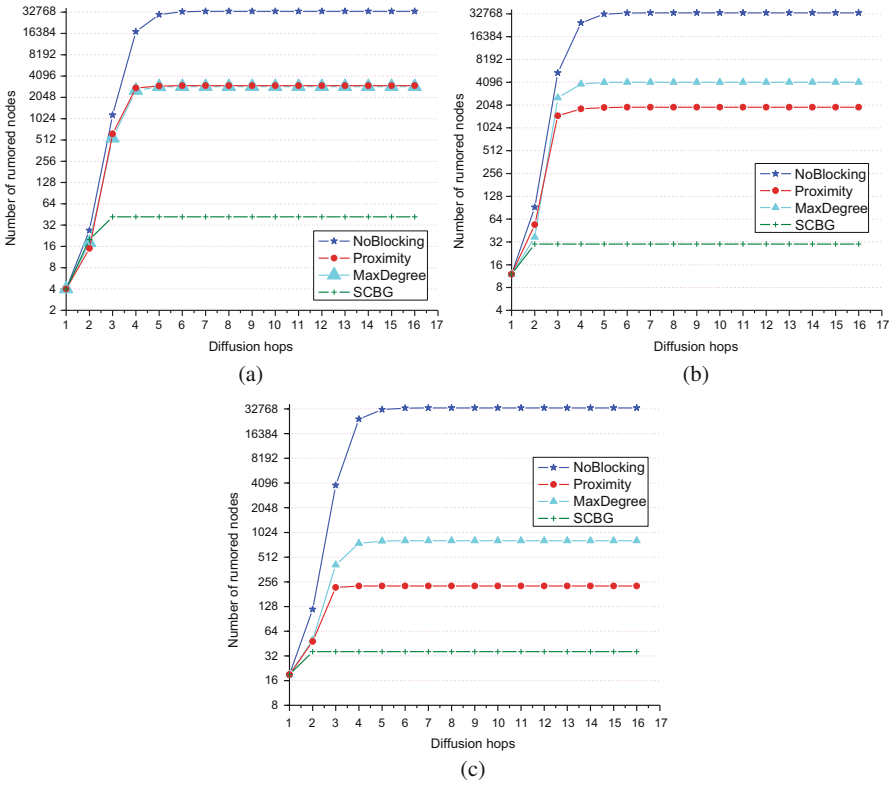

Fig. 4.4 Infected nodes under the DOAM model on Enron Email network with $|N| = 36{,}692$, $|C| = 80$, $|B| = 135$. (**a**) $|R| = 5\%|C|, |P| = 8$. (**b**) $|R| = 10\%|C|, |P| = 11$. (**c**) $|R| = 20\%C, |P| = 14$

Except Fig. 4.3a, in which the Proximity protects one more node than the SCBG algorithm due to small size of initial rumors and low network density, the SCBG algorithm always protects the most number of nodes in comparison with the other two heuristics. Therefore, we believe that our algorithm can be applied to those problems that aim at either protecting targeted nodes with least number of protectors or reducing the number of nodes infected in the entire networks at the end of cascade diffusion, or both of them.

We also notice that Proximity outperforms MaxDegree for different sizes of initial rumors in Figs. 4.3 and 4.4. However, we can see in Fig. 4.5, MaxDegree performs better than Proximity. The reason is that this network has much higher average node degree.

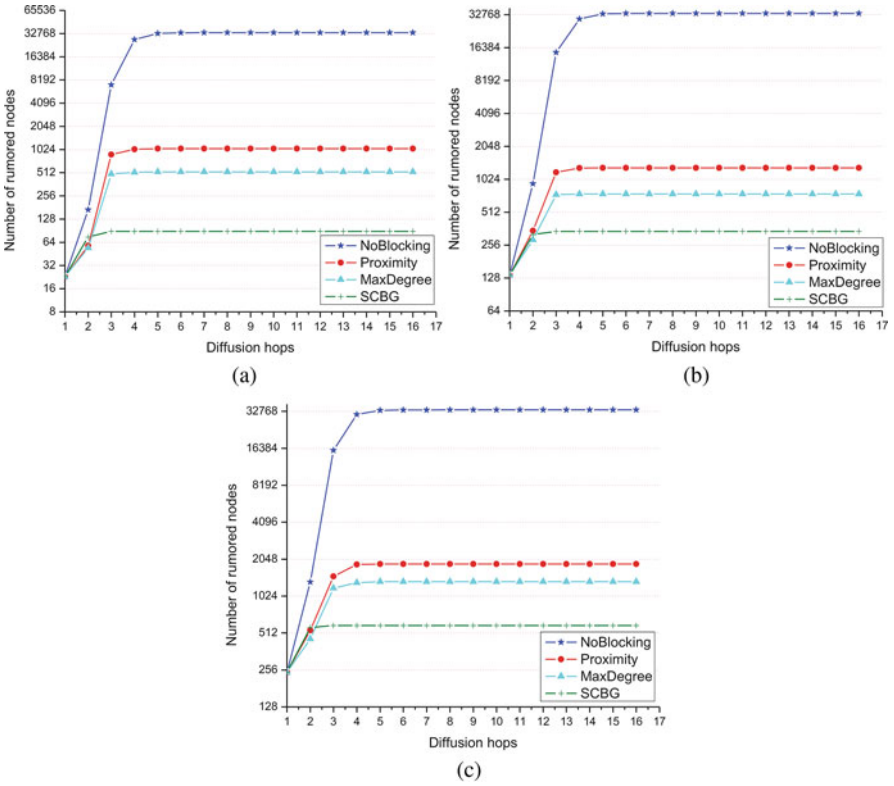

Fig. 4.5 Infected nodes under the DOAM model on Enron Email network with $|N| = 36{,}692$, $|C| = 2631$, $|B| = 2250$. (**a**) $|R| = 1\%|C|$, $|P| = 21$. (**b**) $|R| = 5\%|C|$, $|P| = 52$. (**c**) $|R| = 10\%|C|$, $|P| = 69$

4.3 Rumor Blocking Maximization with Constrained Time

In this section, we seek effective strategy to stop the diffusion of rumor in a network considering the following three factors:

1. A constraint on how much time we can use to control the spread of rumor in the network.
2. There is usually a random time delay in influencing a friend when a person accepts new information.
3. Individuals make decisions based on relationships with their informed friends, but also on their personal judgement about the piece of information that is being diffused.

Here we propose two general models to capture diffusion of rumor and truth (protector) in a social network and formally define the rumor containment problem under these two models as two optimization problems. NP-hardness results are

established for these two optimization problems. We prove the submodularity of the objective functions in these two problems. This enables us to use the greedy algorithm as a constant-factor approximation algorithm (for both problems) with a performance guarantee $1 - \frac{1}{e}$.

The rest of this chapter is organized as follows: Sect. 4.3.1 presents the two propagation models. In Sect. 4.3.2, we define the rumor blocking maximization problem formally, prove its NP-hardness under the two models, and establish the submodularity of the objective functions. Section 4.3.3 gives the formal description of the greedy algorithm.

4.3.1 Propagation Models

Social network can be modeled as a directed graph $G = (V, E)$, where V is the node set and E is the edge set. In the context of influence diffusion, V represents the individuals in this network and E represents the relationships among these individuals. Furthermore, a node $v \in V$ is an out-neighbor of a node $u \in V$ if there exists an edge $e_{uv} \in E$ (i.e., the edge from node u to node v exists in graph G). A node u is called an in-neighbor of v if v is an out-neighbor of u.

In reality, two individuals in a network may not interact/exchange information every day. If someone gets influenced by a certain event, then her friends may learn about this fact *several* days later, and get influenced also. In other words, there is a random time delay between the influence friends have on each other. In this paper, we model this phenomenon using the meeting probability among two nodes in the graph: the meeting action between a node u and its neighbor v happens stochastically at any time step with probability m_{uv}, independent of everything else. Moreover, each edge e_{uv} is assigned an influence weight (probability) IW_{uv}. In the following two models, we will explain this influence weight separately in detail.

4.3.1.1 Rumor-Protector Independent Cascade Model with Meeting Events (RPIC-M)

We first describe one of the most basic and well-studied diffusion models in [93], namely the independent cascade (IC) model. Then, we describe a generalized model which models the following additional features: competitive influence diffusion, meeting events, and personal interest.

In the IC model, a network is considered as a directed graph $G = (V, E)$, where V denotes individuals in the network and E denotes the relationship between individuals. Each edge $e_{uv} \in E$ is assigned an influence probability p_{uv}, indicating the possibility that node u influences node v successfully. For $e_{uv} \notin E$, let $p_{uv} = 0$. Each node can only be in one the following two statuses: inactive or active. Once a node becomes active, it will remain active forever. The diffusion process unfolds in discrete time steps. Starting with an initial set of active nodes A_0, at any step $t \geq 1$,

when node u first becomes active in step t, it has a single chance to activate any of its currently inactive neighbors. For neighbor node v, it succeeds with probability p_{uv}. If u succeeds in activating v, then v will become active in steps $t + 1$, and if u fails in activating v, then v will remain inactive. Regardless of the activation outcome, u cannot make any further attempts to activate v in all subsequent rounds. The process continues until no more activations are possible. If multiple newly activated nodes are in-neighbors of the same inactive node, then their attempts are sequenced in an arbitrary order.

We now describe our new model that incorporates competitive influence (the influence of protector and rumor) diffusion, as well as meeting events and personal interest. This model extends the models proposed in [30]. We denote it by RPIC-M (Rumor-Protector Independent Cascade model with Meeting events). Let P (for "protector") and R (for "rumor") denote the two cascades. The initial set of protected (resp., infected) nodes is denoted by A_p and (resp., A_r). Each node u has personal interests in the information (PI_u). This parameter PI_u is a probability and plays an role in activating node u. Each node is either inactive, infected, or protected. Each edge e_{uv} is associated with a meeting probability m_{uv} and an influence probability p_{uv} (if $e_{uv} \notin E, m_{uv} = 0$ and $p_{uv} = 0$).

Given rumor seed set A_r, as in the IC model, a protector seed set A_p is selected and activated at step $t = 0$. At any step $t \geq 1$, a protected (resp., infected) node u meets any of its currently inactive neighbors v independently with probability $m(u, v)$. Since u's activation, if a meeting event happens between u and v for the first time, then u has a single chance to try protecting (resp., infecting) v with an influence probability $\min\{1, p_{uv} + PI_v\}$, given that no other neighbor of v tries protecting or infecting v at the same step.

If the attempt from u succeeds, v becomes influenced (protected or infected) at step t and will start influencing (protect or infect) its inactive neighbors from time $t + 1$ onwards. If there are two or more nodes trying to influence v simultaneously, at most one of them can succeed. The attempts from the same cascade are ordered arbitrarily. As for the attempts from different cascades (P and R), we assume that all the attempts in R have priority over P. Once a node becomes protected or infected, it will never change its status. The diffusion process continues until no more nodes can be protected or infected.

4.3.1.2 Rumor-Protector Linear Threshold Model with Meeting Events (RPLT-M)

Again, we first describe another basic and well-studied diffusion models in [93], namely the linear threshold (LT) model. Next, we describe a generalized model which models the following additional features: competitive influence diffusion, meeting events, and personal interest.

In the LT model, a social network is viewed as a directed graph $G = (V, E)$, where V denotes individuals in the network and E denotes the relationship between individuals. Each edge $e_{uv} \in E$ is assigned a non-negative weight w_{uv}, which

represents the impact that node u has on node v. For $e_{uv} \notin E$, let $w_{uv} = 0$. For each $v \in V$, $\sum_{u \in V} w_{uv} \leq 1$. Each node v is associated with a random threshold θ_v, which is drawn independently from a uniform distribution with support [0, 1]. Each node is either active or inactive. Once a node becomes active, it remains active forever. The diffusion process unfolds in discrete time steps. Starting with an initial set of active nodes A_0, at any step $t \geq 1$, a node v will become active if and only if the total weight coming from its active in-neighbors exceeds its threshold θ_v, i.e., $\sum_{u \in A_{t-1}} w_{uv} \geq \theta_v$, where A_{t-1} is the set of active nodes by time step $t - 1$. This diffusion process continues until no more nodes can be activated.

We now describe our new model that incorporates competitive influence (the influence of protector and rumor) diffusion, as well as meeting events and personal interest. This model extends the models proposed in [84]. We denote it by RPLT-M (Rumor-Protector Linear Threshold model with Meeting events). Let P (for "protector") and R (for "rumor") denote the two cascades. The initial set of protected (resp., infected) nodes is denoted by A_p and (resp., A_r). Each node v has personal interests in the information from protector (PI_{pv}) and rumor (PI_{rv}). These two parameters are probabilities and play a role in activating node v. Each node is either inactive, infected, or protected. Each edge e_{uv} is associated with a meeting probabilities m_{uv} as well as two weights $w_{uv,p}$ and $w_{uv,r}$ (if $e_{uv} \notin E$, $m_{uv} = 0$ and $w_{uv,p} = w_{uv,r} = 0$). We assume that for all node $v \in V$, $\sum_{u \in V} w_{uv,p} + PI_{pv} \leq 1$ and $\sum_{u \in V} w_{uv,r} + PI_{rv} \leq 1$. Each node u chooses two independent thresholds, namely θ_{pu} (for P) and θ_{ru} (for R) randomly from the uniform distribution with support [0, 1].

Given rumor seed set A_r, as in the LT model, a protector seed set A_p is selected and activated at step $t = 0$. At any step $t \geq 1$, a protected (resp., infected) node u keeps its status and meets any of its currently inactive neighbors v with probability $m(u, v)$. Since u's activation, if a meeting event happens between u and v for the first time, then we say that u's influence (protect or infect) to v is valid. An inactive node v is protected (resp., infected) if the total valid weight from its protected (resp., infected) in-neighbors plus its own interest PI_{pv} (resp., PI_{rv}) exceeds its threshold θ_{pv} (resp., θ_{rv}), given that v has not been activated (infected or protected) yet. If at step t, v is both successfully influenced by P and R, then the diffusion of R has priority over that of P, and v becomes protected. Once a node becomes protected or infected, it will never change its status. The diffusion process continues until no more nodes can be protected or infected.

4.3.2 Rumor Containment with Constraints

In Sect. 4.3.2.1, we define the problem of Rumor Containment maximization with the following additional constraints: time Deadline, Meeting events, and Personal interests (RC-DMP). Subsequently in Sect. 4.3.2.1, we show that RC-DMP under the RPIC-M and RPLT-M models are both NP-hard. Finally in Sect. 4.3.2.1, we prove that the objective functions of RCM-DM under the above-mentioned two

models are monotone and submodular. Following the seminal result of [141], the greedy algorithm is a constant-ratio approximation algorithm for RC-DMP with performance guarantee $1 - \frac{1}{e}$.

4.3.2.1 Problem Definition

We note that previous research on rumor blocking (see, e.g., [30, 62, 84, 147]) fails to address the following additional features:

1. there is often a time deadline on how long the diffusion process can last.
2. people do not interact with each other (influence each other) every day (i.e., the influence between individuals happens randomly, instead of deterministically at every time step).
3. In addition to the influence coming from an individual's friends, she has her own personal interests/opinions about the information that is being diffused. This may also affect how well she accepts the information.

We consider these three factors in our RC-DMP problem.

The problem is formally defined as follows: given a directed graph $G = (V, E)$, a rumor seed set A_r, and two positive integers k and T, our goal is to find a protector seed set A_p ($|A_p| \leq k$) to minimize the expected number of infected nodes by the end of time deadline T. We denote the objective function for the RC-DMP problem by $RC_T(S)$, which is the number of nodes that will be infected within deadline T by the diffusion of rumor, if instead of the set S, the empty set is chosen as the protector seed set.

NP-Hardness of RC-DM

In this section, we prove that the RC-DMP problem under our two proposed models is NP-hard.

NP-Hardness of RC-DMP Under the RPIC-M Model

Theorem 4.3 *Problem RC-DMP under the RPIC-M model is NP-hard.*

Proof Consider the following special case of Problem RC-DMP: The time deadline $T = +\infty$, all the meeting probabilities are equal to 1, all the personal interests are 0 and $p_{uv} = 1$ for all $e_{uv} \in E$. We note that this special case of Problem RC-DMP is identical to the Problem LCRB-D considered in [62] except the tie-breaking rule. A similar reduction from the Set Cover problem as that in the proof of Theorem 3 of [62] can be used to prove this result.

Next, we describe the reduction formally. Given an integer k, a ground set $N = \{m_1, m_2, \ldots, m_n\}$ and a list of subsets of N: $S_1, S_2, \ldots, S_m$ such that $S_1 \cup S_2 \cup \ldots S_m = \{m_1, m_2, \ldots, m_n\}$, the Set Cover problem wishes to find k subsets of N from the list, such that the union of these subsets covers the entire ground set. We reduce the Set Cover problem to our problem by constructing a directed graph as follows:

1. For each subset $S_i, i = 1, 2, \ldots, m$, create a vertex u_i. For each element m_j, $j = 1, 2, \ldots, n$, create a vertex v_j. Add a directed edge from u_i to v_j if $m_j \in S_i$.
2. Create a rumor node r. Add directed edges from r to $u_1, u_2, \ldots, u_m$.

Now it is easy to see that the set cover instance is a "yes" instance if and only if in our problem we can find a S such that $RC_T(S) \geq n + k$. The result follows.

NP-Hardness of RC-DMP Under the RPLT-M Model

Theorem 4.4 *Problem RC-DMP under the RPLT-M model is NP-hard.*

Proof Consider the following special case of Problem RC-DMP: The time deadline $T = +\infty$, all the meeting probabilities are equal to 1, and all the personal interests are 0. We note that this special case of Problem RC-DMP is identical to the Problem IBM under the CLT model considered in [84], which is shown to be NP-hard. The result follows.

Submodularity of RC-DMP A set-based function $f : 2^S \rightarrow \mathsf{R}$ is called submodular if it has the property of diminishing marginal returns, that is, $f(A \cup \{u\}) - f(A) \geq f(B \cup \{u\}) - f(B), \forall A \subseteq B \subset S, \forall u \in S \setminus B$. Furthermore, f is monotone if it satisfies $f(A) \leq f(B)$ when $A \subseteq B \subset S$. In the following, we prove that the objective functions of our RC-DMP problem under the RPIC-M and the RPLT-M models are monotone and submodular. To maximize a non-negative, monotone, and submodular function, we can use the well-known greedy hill-climbing algorithm [141] to obtain a constant approximation ratio of $1 - \frac{1}{e}$.

Submodularity of RC-DMP Under the RPIC-M Model

Theorem 4.5 *Function $RC_T(\cdot)$ is monotone and submodular for any instance of RC-DMP under the RPIC-M model.*

Proof Similarly to the proof in [93], we establish the "*live-path*" graph to demonstrate the submodularity of our objective function. Since the cascade process under the RPIC-M model is random, we can suppose that before the cascade starts, a set of outcomes for all meeting events as well as the *live* or *blocked* assignment for all edges are already determined. The "live-path" graph G_{live} is constructed by combining the two outcomes. Specifically, a live edge e_{uv} is added to G_{live} in the event that u is activated (infected or protected) and is meeting the inactive v for the first time.

For each meeting event (an edge e_{uv} and a time step t in $[1, T]$), we flip a coin with bias m_{uv} to determine if u will meet v at t. Similarly, for each edge e_{uv}, we flip a coin once with bias $\min\{1, p_{uv} + PI_v\}$, and we declare the edge "live" with probability $\min\{1, p_{uv} + PI_v\}$, or "blocked" with probability $1 - \min\{1, p_{uv} + PI_v\}$. All the two operations with coin-flips are independent.

Given an instance S_M of outcomes of all meeting events ($\forall e_{uv} \in E, \forall t \in [1, T]$), and also an instance S_{LB} of live or blocked assignments for all edges, since the process for meeting events and that of the live or blocked assignment are different, and moreover, all flips in the two processes are independent, a possible instance S of all the random outcomes of our problem can be obtained by combing S_M and S_{LB}.

For a fixed S, the two cascades unfold deterministically. Let $DM_T^S(A)$ denote by the end of time step T, the node set that will be infected if instead of A, the empty set is chosen as the initial protector seed set. Note that by definition, we have that

$$RC_T(A) = \sum_S Prob(S) \cdot |DM_T^S(A)|.$$

In the classic IC model, given outcome S, for a live edge e_{uv} in the graph, node u can reach node v with one hop. However, in our model with meeting events, given outcome S, for a live edge e_{uv} in the graph, u will reach node v with $t_v - t_u$ hops, where t_u is the step in which u itself is activated, and t_v is the first step when u meets v, after t_u.

Hence, we say that v is reachable from a seed set A if and only if

- There exists at least one path consisting entirely of live edges (called live-path) from some node in A to v.
- The collective number of hops along the shortest live-path from S to v is no greater than T.

For any given outcome S, consider the graph $G_{live} = (V, E')$, where V is the vertex set of graph G, and E' is the set of live edges in E (determined by S). Both rumor and protector can propagate in this graph. Let V' denote the nodes that can be reached by rumor seed set A_r via live edges within T time steps. Then we construct another graph $G' = (V'', E'')$, where $V'' = \{v | v \in V \text{ and } v \notin A_r\}$ and $E'' = \{e_{uv} | u, v \in V'' \text{ and } e_{uv} \in E'\}$. Since the rumor seed set is given and the meeting event at each time step for pair of nodes is determined by S, we can determine the time step t_u that $u \in V'$ is infected. Similarly, for a protector seed set A_p, we can also determine the time step t'_u that $u \in V'$ is protected.

To construct the protector reachability graph, we do as follows: If $t'_u < t_u$, then we keep the live-path from A_p to u. Otherwise, we delete the path. For all the nodes in V', we determine whether there exists a live-path from A_p. Let $A \subseteq B \subseteq V''$, consider the quantity $|DM_T^S(A \cup \{u\})| - |DM_T^S(A)|$. This is the number of nodes that can be reached by node u but cannot be reached by any node in set A. This is at least as large as the number of nodes that can be reached by node u but cannot be reached by any node in set B. In other words, $|DM_T^S(A \cup \{u\})| - |DM_T^S(A)| \geq |DM_T^S(B \cup \{u\})| - |DM_T^S(B)|$, indicating that $|DM_T^S(\cdot)|$ is submodular. Taking expectation over all possible S, we conclude that the function $RC_T(\cdot)$ is also submodular.

Submodularity of RC-DMP Under the RPLT-M Model

We follow the general idea in [93] for the proof, that is, we prove that the influence diffusion process guided by the RPLT-M model is equivalent to the one guided by a random live or blocked assignment process. Since we have meeting events in our model, we need to incorporate them into the live or blocked assignment process. We now describe the live or blocked assignment process that we use.

Since the meeting event associated with each edge is random, we can determine them for each edge e_{uv} at any time step t by pre-flipping a coin. Given an outcome

S_M of all the random meeting events, for each edge $e_{uv} \in E$, the outcome of meeting events at each time step is determined by S_M. Based on the original graph $G = (V, E)$, we construct two random graphs, namely $G_{S_M R} = (V, E_R)$ and $G_{S_M P} = (V, E_P)$ for rumor diffusion and protector diffusion, respectively.

To construct $G_{S_M P}$, for each node $v \in V$, with probability $w_{uv,p}$, only in-edge e_{uv} is selected and marked as live. With probability PI_{pv}, we mark all of its in-edges as live, and with probability $1 - (\sum_{u \in V} w_{uv,p} + PI_{pv})$ no in-edge is selected as live. (Note that our live-or-blocked assignment process differs with [93] in the sense that for a node v, multiple in-edges can be selected as live. While in [93], at most one edge can be selected as live.)

Similarly, to obtain graph $G_{S_M R}$, for each node $v \in V$, with probability $w_{uv,r}$, only in-edge e_{uv} is selected and marked as live. With probability PI_{rv}, we mark all of its in-edges as live, and with probability $1 - (\sum_{u \in V} w_{uv,r} + PI_{rv})$ no in-edge is selected as live.

We define the concept of an effective live edge as follows: At any step t, live edge e_{uv} becomes *effective* when v meets with its selected neighbor u for the first time, and u has been activated at some earlier step $t' < t$.

In $G_{S_M R}$ (resp. $G_{S_M P}$), given a rumor seed set A_r (resp. protector seed set A_p), for a node $v \in V$, if its selected live edges for rumor diffusion (resp. protector diffusion) connect some node u in A_r (resp., A_p), and in S_M, u meets v before deadline T, then edge e_{uv} becomes effective. If u is not in A_r (resp., A_p), but u has been influenced at t_{ru} (resp., $t_{pu}^{A_p}$ since A_p is not a fixed set), and in S_M, u meets v before deadline T, then edge e_{uv} also becomes effective. If a node u cannot be activated by rumor diffusion (resp., protector diffusion) by the end of time step T, then we define $t_{ru} = \infty$ (resp., $t_{pu}^{A_p} = \infty$), meaning no effective live rumor path (resp., protector path) exists between A_r and u. We say that a node u is *protected* if $t_{pu}^{A_p}, t_{ru} < \infty$ and $t_{pu}^{A_p} < t_{ru}$, and u is *infected* if $t_{ru} < \infty$ and $t_{ru} \leq t_{pu}^{A_p}$.

The following lemma states that the distribution over the final activated (protected or infected) nodes are identical for our RPLT-M and the above live or blocked assignment process.

Lemma 4.1 *For a given protector seed set A_p and rumor seed set A_r, the distribution over the sets of nodes that are infected and protected is identical in the following two models:*

1. *RPLT-M model.*
2. *the live or blocked assignment process.*

Proof We prove this lemma by proving this equivalence under any fixed outcome S_M of the meeting events.

To proceed, we first look at the diffusion process under the RPLT-M model for a given S_M. Recall that the diffusion unfolds in discrete time steps. In each step, some nodes change from inactive to active (protected or infected). For all $t \in [0, T]$, let $A_t^p(v)$ be the set of nodes that are already protected and have met v at least once after their activation by the end of step t, and $A_t^r(v)$ be the set of nodes that are

already infected and have met v at least once since their activation by the end of step t. Consider a node v that has not been activated by the end of time step t. The probability that v becomes protected in $t+1$ equals to the probability that the incremental weight contributed by $A_t^p(v) \setminus A_{t-1}^p(v)$ pushes it over the threshold θ_{pv} (and the incremental weight contributed by $A_t^r(v) \setminus A_{t-1}^r(v)$ does not push it over the threshold θ_{rv}), given that it is not activated by the end of step t. This probability is:

$$\frac{\sum_{u \in A_t^p(v) \setminus A_{t-1}^p(v)} w_{uv,p} \left[1 - \sum_{u \in A_t^r(v) \setminus A_{t-1}^r(v)} w_{uv,r}\right]}{\left[1 - (\sum_{u \in A_{t-1}^p(v)} w_{uv,p} + PI_{pv})\right] \cdot \left[1 - (\sum_{u \in A_{t-1}^r(v)} w_{uv,r} + PI_{rv})\right]}.$$

Similarly, the probability that node v becomes infected in $t+1$ given that v is inactive from step 0 to t is:

$$\frac{\sum_{u \in A_t^r(v) \setminus A_{t-1}^r(v)} w_{uv,r}}{\left[1 - (\sum_{u \in A_{t-1}^p(v)} w_{uv,p} + PI_{pv})\right] \cdot \left[1 - (\sum_{u \in A_{t-1}^r(v)} w_{uv,r} + PI_{rv})\right]}.$$

Next, we look at the live or blocked assignment process for the same fixed outcome S_M of the meeting events. Let B_0^p and B_0^r denote protector seed set and rumor seed set, respectively. For each $t \in [1, T]$, let B_t^p denote the set that contains any $v \notin B_{t-1}^p \cup B_{t-1}^r$ such that v has one effective live in-edge from some node in B_{t-1}^p but no effective live in-edge from any node in B_{t-1}^r. For each $t \in [1, T]$, let B_t^r denote the set containing any $v \notin B_{t-1}^p \cup B_{t-1}^r$ such that v has one effective live in-edge from some node in B_{t-1}^r.

According to the definition of random live or blocked assignment process, the probability that a node v is in $B_{t+1}^p \setminus B_t^p$ conditioned on that v is not in $B_t^p \cup B_t^r$ is:

$$\frac{\sum_{u \in A_t^p(v) \setminus A_{t-1}^p(v)} w_{uv,p} \left[1 - \sum_{u \in A_t^r(v) \setminus A_{t-1}^r(v)} w_{uv,r}\right]}{\left[1 - (\sum_{u \in A_{t-1}^p(v)} w_{uv,p} + PI_{pv})\right] \cdot \left[1 - (\sum_{u \in A_{t-1}^r(v)} w_{uv,r} + PI_{rv})\right]}.$$

Similarly, the probability that a node v is in $B_{t+1}^r \setminus B_t^r$ conditioned on that v is not in $B_t^p \cup B_t^r$ is:

$$\frac{\sum_{u \in A_t^r(v) \setminus A_{t-1}^r(v)} w_{uv,r}}{\left[1 - (\sum_{u \in A_{t-1}^p(v)} w_{uv,p} + PI_{pv})\right] \cdot \left[1 - (\sum_{u \in A_{t-1}^r(v)} w_{uv,r} + PI_{rv})\right]}.$$

The above conditional probabilities are the same as that obtained from the RPLT-M model. Since $A_p = B_0^p$ and $A_r = B_0^r$, we conclude our proof.

With the help of the equivalence result in Lemma 4.1, we can now prove the monotonicity and submodularity of rumor blocking in the random live or blocked

assignment process. Given a fixed outcome S_M of the meeting events, a rumor seed set A_r and an instance S_L of the live or blocked assignment process (where the outcomes of all the live edge selections are determined), let $X = (S_M, S_L)$ and $RC_T^X(A)$ denote the node set that will be infected if instead of A, the empty set is chosen as the initial protector seed set. Then the objective function for our rumor blocking problem is

$$RC_T(A) = \mathbb{E}_X |RC_T^X(A)|.$$

Given a graph $G(V, E)$ and a set $S \subset V$, for a node $u \notin S$, we say that there is a unique path from S to u if there exists some path from a node in S to u. For any two paths from any two nodes in S to u, one path must be a sub-path of the other.

Next, we establish the following lemmas to prove the submodularity of $RC_T^X(A)$.

Lemma 4.2 *In an effective rumor path graph $G_{S_M}R$ ($G_{S_M}P$), given a protector seed set A, for any node u, if $t_{ru} < \infty$ (resp., $t_{pu}^A < \infty$), then there is a unique effective rumor (resp., protector) path from some node in A_r (resp., A) to v.*

Lemma 4.3 *The sufficient and necessary condition for $v \in RC_T^X(A)$ is:*

- *There exists a unique effective rumor path from A_r to v;*
- *there exists at least one node u in the unique rumor path, such that a unique effective protector path exists between A and u with $t_{pu}^A < t_{ru}$.*

Lemma 4.4 *The sufficient and necessary condition for $v \in RC_T^X(B \cup \{u\}) \setminus RC_T^X(B)$ is:*

- *There exists a unique effective rumor path from A_r to v;*
- *There exists at least one node w on the unique effective rumor path from A_r to v, such that a unique effective protector path exists between $B \cup \{u\}$ and w with $t_{pw}^{B\cup\{u\}} < t_{rw}$;*
- *for all node x on the unique effective rumor path from A_r to v, it holds that $t_{rx} \leq t_{px}^B$.*

Lemma 4.5 *The cardinality set function $|RC_T^X(A)|$ for an instance $X = (S_M, S_L)$ is monotone and submodular.*

Proof First we show that $|RC_T^X(A)|$ is monotone. That is, for any node $u \in V \setminus (A \cup A_r)$ where $A \subseteq V$, we need to prove that $|RC_T^X(A)| \leq |RC_T^X(A \cup \{u\})|$, which is equivalent to showing that $RC_T^X(A) \subseteq RC_T^X(A \cup \{u\})$. Consider any node $v \in RC_T^X(A)$, we have that $t_{rv} < \infty$, meaning that there exists a node w in the unique effective rumor path from A_r to v such that $t_{pw}^A < t_{rw}$. We also know that $t_{pw}^{A\cup\{u\}} \leq t_{pw}^A$, therefore, we have that $t_{pw}^{A\cup\{u\}} < t_{rw}$. Thus, we have that $v \in RC_T^X(A \cup \{u\})$.

To prove the submodularity of $|RC_T^X(A)|$, we show that for any $A \subseteq B \subseteq V$, and $u \in V \setminus B$, we have that $RC_T^X(B \cup \{u\}) \setminus RC_T^X(B) \subseteq RC_T^X(A \cup \{u\}) \setminus RC_T^X(A)$. That is, we only need to show that for any $v \in RC_T^X(B \cup \{u\}) \setminus RC_T^X(B)$, we have $v \in RC_T^X(A \cup \{u\}) \setminus RC_T^X(A)$. Since we know that there exists a node w

on the unique effective rumor path from A_r to v, and $t_{pw}^{B\cup\{u\}} < t_{rw}$. And for all node x on the effective rumor path from A_r to v, $t_{px}^{B} \geq t_{rx}$. Therefore, for node w, $t_{pw}^{B\cup\{u\}} < t_{rw} \leq t_{pw}^{B}$, meaning that the influence from node u can reach w earlier than all the nodes in B and A_r. Therefore, the influence from u can reach w earlier than all the nodes in A, that is, $t_{pw}^{A\cup\{u\}} = t_{pw}^{B\cup\{u\}} < t_{rw}$. Since for any node x on the unique effective rumor path from A_r to v, we have that $t_{px}^{B} \geq t_{rx}$, and $A \subseteq B$, it is clear that $t_{px}^{A} \geq t_{px}^{B}$, thus, $t_{px}^{A} \geq t_{rx}$, thus, we have demonstrated that $RC_T^X(B \cup \{u\}) \setminus RC_T^X(B) \subseteq RC_T^X(A \cup \{u\}) \setminus RC_T^X(A)$.

Since taking expectation preserves submodularity, we have established the following result.

Theorem 4.6 *Function $RC_T(\cdot)$ is monotone and submodular for any instance of RC-DMP under the RPLT-M model.*

4.3.3 Possible Solutions

From Theorems 4.3 and 4.4, we know that Problem RC-DMP is NP-hard under the two proposed models (RPIC-M and RPLT-M). This motivates our consideration for approximation algorithm for Problem RC-DMP. Moreover, from Theorem 4.6, we know that the objective function $RC_T(\cdot)$ of Problem RC-DMP under the RPIC-M and the RPLT-M models is monotone and submodular. Furthermore, by definition, $RC_T(\cdot)$ is non-negative and $RC_T(\emptyset) = 0$. Consequently, we can apply the seminal result in [141] and use the greedy algorithm as a constant-factor approximation algorithm with performance guarantee ratio of $1 - \frac{1}{e}$. We formally present the greedy algorithm in Algorithm 3. Note that variable R in the algorithm controls the number of Monte Carlo simulations.

Algorithm 3 Greedy algorithm

Input: Given a graph $G = (V, E)$, A_r, k and T
Output: Protector seed set $A_p \subseteq V$.

Initialize $A_p = \emptyset$, $R = Num\_Simulations$
for $i = 1$ **to** k **do**

  for all $u \in PV \setminus A_p$ **do**
    $IF(u) = 0$
  end for
  for $j = 1$ **to** R **do**
    $IF(u) += RC_T(A_p \cup \{u\})$
  end for
  $IF(u) = IF(u)/R$
  $A_p = A_p \cup \arg\max_{u \in V \setminus A_p}\{IF(u)\}$
end for
Output A_p.

Since in our problem, rumor and protector diffuse with time deadline T, meaning that we only need to search certain area for computation of the seed set of protectors. Let $N_{in}(u)$ denote one hop in-edge neighbors of node u, and $N_{in}^2(u) = \{w|e_{wv} \in E \cap v \in N_{in}(u)\}$ denote two hops in-edge neighbors of node u, thus $N_{in}^t(u)$ are the t hops in-edge neighbors of node u. Moreover, we denote $R^t(A_r) = \bigcup_{u \in A_r} N_{in}^t(u)$ and $R = \bigcup_{t=1}^{t=T} R^t(A_r)$. $P^t = \bigcup_{u \in R^t(A_r)} N_{in}^{t-1}(u)$, where $t \in [1, T]$, and $P = \bigcup_{t=1}^{t=T} P^t$. As a result, $PV = P \cup R$ is the valid nodes that we only need to compute in our objective function.

4.4 Conclusion

In this chapter we performed an extensive study of the problem of limiting the spread of misinformation/rumors in a social network. We investigated efficient solutions to the following question: Given a social network where a (bad) information campaign is spreading, who are the influential people to start a counter-campaign if our goal is to block the effect of the bad campaign efficiently?

In Sect. 4.2, we formulated the Rumor Control (RC) problem under the DOAM model and prove that it is equivalent to the Set Cover problem. To address the problem, Set Cover Based Greedy (SCBG) algorithm is presented, which contains two parts: first, transfer the RC-D problem into the SC problem; second, apply the greedy algorithm used for the SC problem to the obtained subsets for bridge ends. The experimental reports over two real-world social networks demonstrate that the SCBG algorithm outperforms the two heuristics: MaxDegree and Proximity.

In Sect. 4.3, we proposed two models to capture competitive influence diffusion process, namely the RPIC-M and RPLT-M models. In these two models, two kinds of cascades propagate: protector and rumor. These two models extends the seminal IC and LT models [93] to the case of two-cascade influence diffusion. Furthermore, the following three features are also included in these models: a time deadline, random time delay between information exchange, and personal interests regarding the acceptance of information.

Under these two models, we study the RC-DMP problem: given a directed graph $G = (V, E)$, a rumor seed set A_r, and two positive integers k and T, our aim is to find a protector seed set A_p (with $|A_p| \leq k$) to minimize the expected number of infected nodes by the end of time deadline T. We prove that the problem under the two models is both NP-hard. Moreover, we demonstrate that the objective functions under the two different models are both monotone and submodular. Therefore, we are able to apply the seminal result in [141] and use the greedy algorithm as a constant-factor approximation algorithm with performance guarantee ratio of $1 - \frac{1}{e}$.

About future directions, we mention several clues. First, the greedy approximation algorithm is inefficient and time-consuming as it lacks of a way to efficiently compute the objective functions for our problem. To overcome such inefficiency, we hope to find more efficient algorithms to compute the objective function under the two proposed models.

Second, we have noticed that under most situations, the spread of influence and the meeting events occurred among individuals are in continuous time. Thus developing continuous-time diffusion models for our problem is promising.

Third, more real-world factors such as personal interests, different influence diffusion speed, deadline, etc., could be incorporated into current diffusion models.

Last but not the least, in society, influence diffuses in different mechanisms, as well as in different contexts, that is, there exist various models in reality. Therefore, it is interesting to look into our problem under other influence diffusion models.