Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

# Union acceptable profit maximization in social networks ☆

Guoyao Rao [a], Yongcai Wang [a], Wenping Chen [a,*], Deying Li [a], Weili Wu [b]

[a] *School of Information, Renmin University of China, Beijing, 100872, China*
[b] *Department of Computer Science, University of Texas at Dallas, Richardson, TX, 75080, USA*

ARTICLE INFO

*Article history:*
Received 27 December 2021
Received in revised form 10 February 2022
Accepted 9 March 2022
Available online 16 March 2022

*Keywords:*
Influence maximization
Union acceptable
Union profit

ABSTRACT

Online social network has deeply changed our lives, such as the style of communication and business, and hence promotes a lot of researches in social influence. The prior works in social influence mainly consider the influence from the view of individuals. However, in many cases, influencing the most of members of an important group such as the board of directors in a company can bring bigger profit than directly influencing the individuals of the company. We call such high profit group which obeys the vote rule as an union, different from existed targeted influence model, we consider such scenarios to make union acceptable and propose the union acceptable profit problem (UAPM) to choose seeds to maximize the union-acceptable profit, i.e., maximize the probability of the union being acceptable. The objective of profit in UAPM is #P-hard, and not submodularity or supmodularity. To solve the problem, we propose an efficient estimation method for the objective and design a heuristic algorithm and further a data-driven $\beta(1-\frac{1}{\epsilon})$-approximation algorithm where $\beta$ is the data-driven parameter which is related to the input data. At last we evaluate the performance of the algorithms we proposed on effectiveness and efficiency by the experiments in real-world social network datasets.

## 1. Introduction

Online social network has deeply changed our lives from almost every aspect such as the communication, information diffusion, business model, marketing and so on. Especially with the outbreak of COVID-19, many offline businesses and marketing speed up to move online and cooperate with online social platforms such as Facebook, Twitter, Weibo and Wechat, the main way for information dissemination and communication in the age of internet. The online social networks can significantly enhance the effect of word-of-mouth which is famous in viral marketing. As the traditional application in viral marketing, many businesses would like to promote their products through social network platforms by choosing few costumers to experience firstly, and then letting them spread the related positive information about their products to attract more latent costumers in social networks. It creates the research of social influence. Since Kemp et al. [2] firstly formulated the influence maximization (IM) problem, many variants, extension and applications have been well studied

☆ This is an enhanced and extended version of a paper [1] presented in The 15th Annual International Conference on Combinatorial Optimization and Applications (COCOA 2021). This work is supported by the National Natural Science Foundation of China (Grant NO. 12071478, 61972404), and partially by NSF 1907472.

* Corresponding author.

*E-mail addresses:* gyr@ruc.edu.cn (G. Rao), ycw@ruc.edu.cn (Y. Wang), chenwenping@ruc.edu.cn (W. Chen), deyingli@ruc.edu.cn, deyingl@hotmail.com (D. Li), weiliwu@utdallas.edu (W. Wu).

https://doi.org/10.1016/j.tcs.2022.03.015
0304-3975/© 2022 Elsevier B.V. All rights reserved.

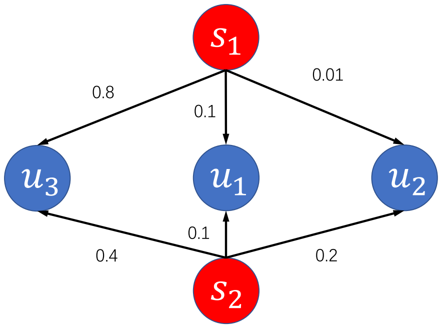


**Fig. 1.** Illustration of distinguishing the union acceptable and targeted influence.

such as [3–12]. Most of the previous works are major in the individual influence, e.g., maximize the statistics number of influenced individuals in whole or targeted part, or maximizing the probability of certain individual to be influenced.

However, in many scenarios, influencing an important group with vote rule, e.g., a member committee obeys the rule of minority obeys the majority, can bring the larger profit than influencing the individual one by one in the entirety. For example, a computer dealer tries to lobby the procurement committee of a company, school or government to adopt their products, and if it can successfully lobby more than half of the committee synchronously, then it can get bigger profit by a group order than the circumstances of individual orders one by one. There is another example of the marketing which aims to families such as group-travel, the key of the marketing to be successful is whether the travel plan can get supported by most of the members of a family synchronously, i.e., whether most of the family member can be simultaneously influenced in the marketing. Supported by the social network, considering the lobby or marketing demands above, letting some agents (e.g., friends or friends' friends, etc.) indirectly influence these target candidates over the influence spread on social networks is developing to be a trend. We note such key ballot entirety like the committee and family above which obeys the rule of the minority is subordinate to the majority. i.e., a given lower bound of proportion of members influenced synchronicity as an **union** which is usual seen as board of directors, group shopping, government lobby and so on. And we consider to use the method of social influence to achieve the goal of maximizing the profit from the union.

The previous works based on the targeted influence [11][12], etc, are closed to solve our problem, in which they can set the group members as targets and further maximize the number of them to be influenced or the pseudos targeted acceptable probability which is the sum of probability to be influenced independently in target. However, such idea may be immature without considering the synchronicity in the influence process and we show it as the following example in Fig. 1. Let the direct edge in the figure stand the direct influence from the starting node to the ending node and the edge weight is the influence probability. We consider to choose a seed from $\{s_1, s_2\}$, and let the union profit to be 100 if at least half of the members get influenced synchronously, if adopting the targeted influence maximization strategy to chose $s_1$ as seed which can lead to expected 0.91 members in the union to be influenced, we have the successful probability to influence the union at least 2 members synchronously is 0.0874 and the expected union profit is 8. However the seed $s_2$ can make the probability larger to 0.124 and larger excepted union profit at 12.4 though the expected number of influenced members is 0.7 less than that with $s_1$.

Different from the native strategies based on the target influence that aims to optimize the number of influenced or other variants, we consider to optimize the chance of target being influenced synchronously. In this paper, we consider such problem that is to make certain proportion of targeted nodes we call as union to be influenced synchronously, in other word, the union to be acceptable as far as possible. The main contributions are as follows:

- We propose the union acceptable profit maximization (UAPM) problem and prove it's NP-hard and the computation problem of objective is #P-hard.
- We prove the UAPM is not either submodularity or supmodularity.
- To solve the #P-hard problem, we propose an efficient algorithm to estimation the objective.
- We design a heuristic algorithm and a data-driven $\beta(1-\alpha)$-approximation algorithm respectively to solve the UAPM problem.
- We conduct various experiments based on real-world databases.

In the rest of this paper, we firstly review some existing related works of social influence in Section 2. In Section 3, we introduce the formulations of the UAPM problem and analyze its hardness and related properties. In section 4, we introduce an estimation method for the target function based on an method of the set sequence sampling. Then in section 5, we firstly propose a heuristic algorithm and then design an data-driven approximation algorithm to solve the UAPM problem. Lastly in section 6, we show various experiments based on real-world datasets and then give the conclusion in section 7.

## 2. Related work

### 2.1. Independent cascade (IC) model

To describe how influence spreads over social network, there are two widely used influence diffusion model, i.e., Independent Cascade (IC) [13], and Linear Threshold (LT) [14]. We specially introduce the classical IC model which we used in this paper as follows.

Let $G(V,E,p)$ be an influence graph which is a directed and weighted network where the weight of each direct edge $p_{uv} \in [0,1]$ is the probability that a node $u$ influences another node $v$, and then the influence spreads from a set of seeds in rounds. Initially, only all seeds are active while other nodes are inactive. In each diffusion round, each node becoming active in the previous round has one chance to influence each one of its inactive out-neighbors following the influence probability. The process terminates as long as no more inactive nodes can be influenced to become active.

There is also an equivalent generating formulation [5] for the diffusion proposed above as follows:

- Firstly, get a G's edge-induced subgraph noted as $g$ by flipping[1] each edge $e_{uv}$ randomly following the probability $p_{uv}$.
- Specially we write $g \sim G$ to mean that $g$ is randomly realized from G by (1). Then secondly, we mark $g_S$ to be the set of nodes which can be reached in the graph $g$ by any seed node in seeds set $S$, and we naturally have $g_S$ are the influenced nodes.

### 2.2. Variants of social influence

Kempe et al. [2] firstly formulate the Influence Maximization problem (IM) into discrete optimization which aims to select $k$ nodes as seeds to maximize the expected number of influenced node through a stochastic diffusion process in social networks, and specially analyze the IM problem on the model of IC and LT. This work attracted and inspired many researches into social influence and brought many variants and extended works. Many researches major in the spread problem based on different and specific scenes, such as time-constrained [3,4], topic-aware [5,6], competition [7,8], rumor-control [9], multi-round [10] and so on. Specially, the researches [11,12] consider the number influence in a group of nodes as a target set, and as we said in the introduction, it's native strategies in our problem. In addition, the research [15] of personalize influence maximization and the works [16,17] of the acceptance probability maximization (APM) problem all are based on the idea to maximize the probability of certain given node to be influenced alone. The previous work in [1] considers to influence several nodes synchronously which is still a special case of our problem, when we consider to set the union to be acceptable when all of the members be influenced synchronously. Then we still don't see any other researches closed to our problem.

### 2.3. Related algorithm

There were two important properties in related researches, i.e., submodularity and supmodularity [18] which are defined as follows: Let $S$ be a finite set and function $f : 2^S \to \mathbb{R}$, and if for any $A \subseteq B \subseteq S$ and any $x \notin S/B$, we said $f$ is submodularity as long as $f(A \cup \{x\}) - f(A) \geq f(B \cup \{x\}) - f(B)$, and $f$ is supermodularity if the inequality is reversed. The basic IM problem is proved to be NP-hard and the influence computation is #P-hard. With the good property of submodularity for the target function, the basic greedy method can provide a $(1 - 1/e - \epsilon)$-approximation solution [18], where $\epsilon$ is the loss caused by influence estimation since it's hard to get the accurate influence. However, such greedy-based methods cost too much time using the heavy Monte Carlo simulations to estimate the marginal gain of node's influence, and it's hard to apply to the large scale network, although there are many improvements [19–21]. Then Tang et al. [22] and Borgs et al. [23] proposed the reverse influence set (RIS) sampling method to estimate the influence. The idea of RIS is using a revere propagate from a node $v$ to get a random set of nodes that can influence $v$, hence through the number of set covered by the seeds set to estimate the influence which is more efficient than repeatable simulations. Then after transforming the IM problem to the classical set cover problem, there are many RIS-based extensions and improvements such as the IMM [24], SSA and D-SSA [25], OPIM [26].

## 3. Problems formulation

As we said in the introduction, we consider influencing an union with vote rule which is a kernel portion of the whole and then we can get huge profit with the whole driven by the union, for example, a computer dealer can lobby the procurement committee of a company, and then if it can successfully lobby more than half of the committee synchronously, then it can get huger profit by the enterprise order with total number of employees in this company, and if it failed, it can only get few profits from few personal orders of the employees in this company. So, this is the motivation that we

[1] We said flipping an edge $e_{uv}$ is that remove it with the probability $1 - p_{uv}$ from the graph and mark it to be "on" if the edge isn't removed and otherwise be "off".

consider maximizing the probability of union to be acceptable and hence the expected union profit since it holds that the profit when the union is acceptable is significantly larger than the profit when the union isn't acceptable. Then based on the influence spread model as IC and the equivalent generating view, we give our problem defined as follows.

**Definition 1** *(Union acceptable profit maximization problem (UAPM)).* Given the influence graph $G(V, E, p)$ on IC model, an union set $\mathbb{U} \subseteq V$, a set of alternative nodes $\mathbb{S} \subseteq V$, and a budget $k$, the union acceptable profit maximization problem is to choose a set $S^*$ of at most $k$ nodes from $\mathbb{S}$ as seeds maximizing the expected union acceptable profit, i.e.,

$$S^* := \underset{S \subseteq \mathbb{S}, |S| \leq k}{argmax}\ \sigma_\theta$$

and the expected union acceptable profit $\sigma_\theta$ is written as

$$\sigma_\theta = \gamma_1 \cdot Pr_{g \sim G}\{\frac{|g_S \cap |\mathbb{U}|}{|\mathbb{U}|} \geq \theta\} + \gamma_2 \cdot Pr_{g \sim G}\{\frac{|g_S \cap |\mathbb{U}|}{|\mathbb{U}|} < \theta\},$$

where $\gamma_1$, $\gamma_2$ is the profit when the union is acceptable or unacceptable respectively, $\gamma_1 \gg \gamma_2 > 0$, and $Pr_{g \sim G}\{|g_S| \geq \theta\}$, $Pr_{g \sim G}\{|g_S| < \theta\}$ corresponds to the probability of union being acceptable or unacceptable respectively, $\theta \in (0, 1]$ is the threshold corresponding to the lower bound of the proportion of influenced members in the union to promote the union to be acceptable.

By the definition, further we can get the equivalent computation of the expected union acceptable profit shown in Lemma 1.

**Lemma 1.** *We have $\sigma_\theta = \alpha Pr_{g \sim G}\{|g_S \cap |\mathbb{U}| \geq N_\theta\} + \beta$, where $\alpha = \gamma_1 - \gamma_2$, $\beta = \gamma_2$, and $N_\theta = \lceil \theta \cdot |\mathbb{U}| \rceil\}$.*

**Proof.** Since $Pr_{g \sim G}\{|g_s| < \alpha\}$=$1 - Pr_{g \sim G}\{|g_s| < \alpha\}$ and $|g_S \cap \mathbb{U}|$ must be an integer, we can easily have

$$\begin{aligned}\sigma_\theta &= \gamma_1 \cdot Pr_{g \sim G}\{\frac{|g_S \cap |\mathbb{U}|}{|\mathbb{U}|} \geq \theta\} + \gamma_2 \cdot Pr_{g \sim G}\{|\frac{|g_S \cap |\mathbb{U}|}{|\mathbb{U}|} < \theta\} \\ &= \gamma_1 \cdot Pr_{g \sim G}\{\frac{|g_S \cap |\mathbb{U}|}{|\mathbb{U}|} \geq \theta\} + \gamma_2 \cdot (1 - Pr_{g \sim G}\{\frac{|g_S \cap |\mathbb{U}|}{|\mathbb{U}|} \geq \theta\} \\ &= \alpha Pr_{g \sim G}\{\frac{|g_S \cap |\mathbb{U}|}{|\mathbb{U}|} \geq \theta\} + \beta \\ &= \alpha Pr_{g \sim G}\{|g_S \cap |\mathbb{U}| \geq \theta \cdot |\mathbb{U}|\} + \beta \\ &= \alpha Pr_{g \sim G}\{|g_S \cap |\mathbb{U}| \geq N_\theta\} + \beta \quad \square\end{aligned}$$

By the Lemma 1, we can also naturally get three special cases of UAPM problem as follows.

1. When the union has only one member supposed to be node $u$, we have $\sigma_\theta = \alpha Pr_{g \sim G}\{|g_S \cap |\mathbb{U}| \geq \lceil \theta \rceil\} + \beta = \alpha Pr_{g \sim G}\{u \in g_S\} + \beta$, and hence the $\sigma_\theta$ corresponds to the probability of the targeted user to be influenced, i.e., this UAPM problem is equivalent to personal influence maximization problem [15].
2. When $k \geq |\mathbb{U}| \cdot \theta$ and $\mathbb{U} \subseteq \mathbb{S}$, the UAPM problem is easily to be solved by selecting any $k$ nodes in $U$ as the seeds.
3. When $\theta = 1$, we have $\sigma_\theta = \alpha Pr_{g \sim G}\{\mathbb{U} \subseteq g_S\} + \beta$, i.e., this UAPM problem is equivalent to the union-influenced probability maximization problem [15].

Nextly, excluding the special circumstances above, we consider the general situations and analyze the hardness of the general UAPM problem as follows.

**Theorem 1.** *The UAPM problem is NP-hard.*

**Proof.** Consider any instance of the NP-complete Set Cover problem with a set collection $\mathbf{C} = \{c_1, c_2, \ldots, c_m\}$, a set of nodes $T = \{t_1, t_2, \ldots, t_n\}$. We wish to know whether there exist $k$ sets in $\mathbf{C}$ covering all nodes in $T$. We construct special $UAPM$ problems as follows: (1) Create $\mathbb{U}$=$\{u_1, u_2, ..., u_n\}$, where each $u_i (1 \leq i \leq n)$ corresponds to $t_i$ in $T$. (2) Create $\mathbb{S}$=$\{s_1, s_2, ..., s_m\}$, where each $s_j (1 \leq j \leq m)$ corresponds $c_j$ in $\mathbf{C}$. (3) Create $V^*$=$\{v_1, v_2, ..., v_n\}$, and let $V = \mathbb{U} \cup \mathbb{S} \cup V^*$. (3) Create $E$'s edges by connect $s_j$ to $v_i$ with influence probability 1 as long as $t_i \in c_j$ and each $v_i$ to $u_i$ with influence probability $p \in (0, 1)$. It's easy to get that there exist $k$ sets in $\mathbf{C}$ covering all nodes in $T$ if and only if there exists at most $k$ seeds $S_k \subset \mathbb{S}$ with the maximal union acceptable probability $p^* := \sum_{i=N_\theta}^{n} \binom{n}{i} p^i (1-p)^{n-i}$, i.e., $\sigma_\theta(S_k) = \alpha p^* + \beta$. $\square$

**Theorem 2.** *The computation problem for $\sigma_\theta(\cdot)$ is #P-hard.*

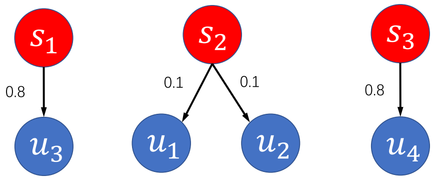


**Fig. 2.** A special case to illustrate of the properties of $\sigma_\theta$.

**Proof.** Consider any instance of the #P-complete s-t connectedness counting problem with $G'(V', E')$ and two vertex $s$ and $t$ in $V'$. We wish to count the number of $G'$'s subgraphs in which $s$ is connected to $t$, and we denote these subgraphs as a set $\mathcal{G}$. We show this problem is equivalent to the following computation problems of $\sigma_\theta(\cdot)$ with $V = V' \cup \mathbb{U}$, $E = E' \cup \{< t, u > | u \in \mathbb{U}\}$, $\mathbb{S} \subseteq V'$, and the constant edge influence probability $p \in (0, 1)$. Given a seed set $S = \{s\}$, we can easily have that $Pr_{g \sim G}\{|g_S \cap |\mathbb{U}| \geq N_\theta\} = Pr_{g' \sim G'}\{t \in g'_{\{s\}}\} \cdot p^* = p^* \sum_{g' \in \mathcal{G}'} Pr(g') = p^* |\mathcal{G}| p^{|E'|}$, where $p^* = \sum_{i=N_\theta}^{|\mathbb{U}|} \binom{n}{i} p^i (1-p)^{|\mathbb{U}|-i}$. Thus we can get the size of $\mathcal{G}$ by the computation of $\frac{\sigma_\theta(S)-\beta}{\alpha p^{|E'|} p^*}$. □

Many problems in social influence have good properties such as submodularity or supmodularity which is very useful in designing approximation algorithms. However, in our problems, these properties are lost.

**Theorem 3.** *The object function $\sigma_\theta(\cdot)$ is neither submodularity nor supmodularity.*

**Proof.** We show it by a counterexample as shown in Fig. 2 with $\mathbb{U} = \{u_1, u_2, u_3.u_4\}$ and $\mathbb{S} = \{s_1, s_2, s_3\}$. Let $\Delta_{s'}\sigma_\theta(S)$ be the gain of objective $\sigma_\theta$ after adding a seed $s'$ into $S$. Then we can easy compute and have $\sigma_{0.5}(\{s_1\}) = \beta$, $\sigma_{0.5}(\{s_2\}) = \alpha \cdot 0.1^2 + \beta = 0.01\alpha + \beta$, $\sigma_{0.5}(\{s_3, s_2\}) = \sigma_{0.5}(\{s_1, s_2\}) = \alpha \cdot ((1-0.8)0.1^2 + 0.8(1-(1-0.1)^2)) + \beta = 0.16\alpha + \beta$, $\sigma_{0.5}(\{s_1, s_2, s_3\}) = \alpha \cdot ((1-0.8)((1-0.8)0.1^2 + 0.8(1-(1-0.1)^2)) + 0.8(1-(1-0.8)(1-0.1)^2)) + \beta = 0.68\alpha + \beta$, and $\sigma_{0.5}(\{s_1, s_3\}) = \alpha \cdot 0.8^2 + \beta = 0.64\alpha + \beta$. Hence we have $\Delta_{s_1}\sigma_0.5(\{s_2\}) = 0.15\alpha < \Delta_{s_1}\sigma_0.5(\{s_2, s_3\}) = 0.52\alpha$ and $\Delta_{s_2}\sigma_0.5(\{s_1\}) = 0.16\alpha > \Delta_{s_2}\sigma_0.5(\{s_1, s_3\}) = 0.04\alpha$. □

## 4. The algorithm to estimate $\sigma_\theta(\cdot)$

As the computation of probability is #P-hard, it means it's hard to get the truth of the objective. An natural estimation method is regular simulation of Mont Carlo. However, such estimation method runs heavily and specially we need repeat new simulations once the seeds are changed. Inspired by the RIS method of using a set cover to estimate the influence, we adopt the union-reverse influence set-sequences to estimate our objective. We firstly review the definition of union-reverse influence set-sequences which is first introduced in our previous work [1].

**Definition 2** *(Union-reverse influence set-sequences (UIS)).* Given an influence Graph $G$ and an Union $\mathbb{U}$, let $g$ be an edge-induced subgraph of $G$ by flipping edges randomly with the probability, the union-reverse influence set-sequence is a set sequence $(r_{u_1}, r_{u_1}, \ldots, r_{u_{|\mathbb{U}|}})$ where each set $r_{u_i}$ is a subset of alternative seeds $\mathbb{S}$ in which every node can be reversely reached by $\mathbb{U}$'s node $u_i$ over $g$.

The previous work in [1] adopts a native algorithm as shown in Algorithm 1 to sample an UIS set sequence which firstly creates a $G$'s edge-induced subgraph $g$ by flipping all edges advanced with the edge probability, and then does multiple independent reverse BFSs (breadth-first search) over $g$ starting from each different union node $u_i$ in $\mathbb{U}$ to get the corresponding nodes set $r_{u_i}$ reached by $u_i$. However there may be lots of edges which won't be searched at all while we still cost time to flip them during the subgraph inducing, and besides some edges being repeatedly explored by multiple reverse BFSs also wastes time. So we adopt 3 following optimization strategies to avoid the above two terrible situations.

**Algorithm 1:** SG-BFS($G(V, E, p)$, $U$, $S$).

```
Get a $G$'s random subgraph of g ;
for each $u_i \in \mathbb{U}$ do
    Do a BFS search in $g$ sourced from $u_i$ and get the searched set $r_{u_i}$
return $(r_{u_1}, r_{u_2}, \ldots, r_{u_{|\mathbb{U}|}})$
```

**1. Graph pruning**: In fact, the edges not in any path from $\mathbb{U}$ to $\mathbb{S}$ are unnecessary as they have no useful for the exploration to the alternative seeds in the BFS. So we prune all these edges out of the graph and get the edge-induced subgraph $G_{\mathbb{U},\mathbb{S}}$ which will be used for the later exploration.

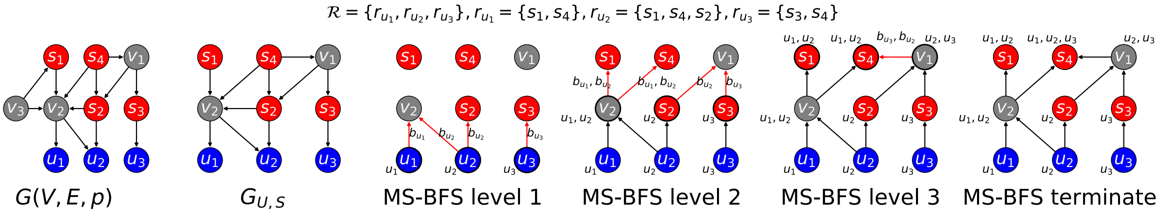


**Fig. 3.** Illustration of generating an UIS set sequence by Algorithm 2.

**2. Multi-source breadth-first search**: Instead of running multiple independent reverse BFSs sourced from different union nodes, all union nodes search jointly and concurrently to share the BFS exploration for the common node, i.e., we adopt a reverse multi-source bread-first search (MS-BFS [27]), in which if multi BFSs sourced from different nodes all will explore a common node, they can share the exploration of this node hence to avoid the repeated exploration.

**3. Flipping the edge on flying**: Instead of flipping all edges in advance, we only flip the edge at the moment that we will explore during the reverse BFS to reduce the unnecessary flipping.

**Algorithm 2:** RRMS-BFS($G(V, E, p), U, S$).

```
$Seen_{u_i} \leftarrow \{u_i\}$ for all $u_i \in \mathbb{U}$;
$Visit = \bigcup_{u_i \in \mathbb{U}} \{(u_i, \{b_{u_i}\})\}$;
while $Visit \neq \emptyset$ do
    $VisitNext \leftarrow \emptyset$;
    for each $v \in \bigcup_{(o, B_o) \in Visit} \{o\}$ do
        $B^*_v = \bigcup_{(v, B'_v) \in Visit} B'_v$;
        for each $v$'s in-neighbor $u$ in $G_{\mathbb{U},\mathbb{S}}$ do
            if edge $e_{uv}$ has not been flipped then
                Flip $e_{uv}$ with probability $p_{uv}$;
            if $e_{uv}$ has been "on" then
                $B_u \leftarrow B^*_v / \{b_o | o \in Seen_u\}$;
                if $B_u \neq \emptyset$ then
                    $VisitNext \leftarrow VisitNext \cup (u, B_u)$;
                    $Seen_u \leftarrow Seen_u \cup \{o | b_o \in B_u\}$;
    $Visit \leftarrow VisitNext$;
$r_{u_i} \leftarrow \emptyset$ for each $u_i \in \mathbb{U}$;
for each $s_i \in S$ do
    for each $u_i \in Seen_{s_i}$ do
        $r_{u_i} \leftarrow r_{u_i} \cup \{s_i\}$;
return $(r_{u_1}, r_{u_2}, \ldots, r_{u_{|\mathbb{U}|}})$
```

Based above 3 ideas, then we get the UIS sampling Algorithm 2 called random reverse multisource BFS in which each edge will be flipped randomly during the union search process. We show an example in the Fig. 3 to show how this algorithm works. In this example, the node $v_3$ in the graph $G(V, E, P)$ doesn't exist in any path starting from the alternative nodes $\{s_1, s_2, s_3, s_4\}$ to the union nodes $\{u_1, u_2, u_3\}$, and hence we prune all $v_3$'s edge out of the graph and get the graph $G_{U,S}$ to avoid it is visited s by $v_2$. Nextly we do random reverse MS-BFS multi-sourced from $\{u_1, u_2, u_3\}$, and get their three initial level of BFSs corresponding to 3 $Visit$ elements as $(u_1, \{b_{u_1}\}), (u_2, \{b_{u_2}\}), (u_3, \{b_{u_3}\})$ respectively. For each union node $u_i$, we get $seen_{u_i} = \{u_i\}$ representing the node $u_i$ has been visited by the source node $u_i$. So all the ingoing-edges of the first exploration nodes are flipped randomly to be "on" and we get second level of BFSs as $(v_2, \{b_{u_1}, b_{u_2}\})$, $(s_2, \{b_{u_2}\}), (s_3, \{b_{u_3}\})$ and update their $Seen$ sets as $Seen_{v_2} = \{u_1, u_2\}, Seen_{s_2} = \{u_2\}, Seen_{s_3} = \{u_3\}$. Except the reverse edges $< v_2, s_2 >$ and $< s_2, s_4 >$, all others ingoing-edges of node in the second exploration are flipped randomly to be "on", and hence we get third level of BFSs as $(s_1, \{b_{u_1}, b_{u_2}\}), (s_4, \{b_{u_1}, b_{u_2}\}), (v_1, \{b_{u_2}, b_{u_3}\})$ and update their $Seen$ sets as $Seen_{s_1} = \{u_1, u_2\}, Seen_{s_4} = \{u_1, u_2\}, Seen_{v_1} = \{u_2, u_3\}$. In the last level of BFSs, as $s_1$ and $s_4$ have no in-going edges and the BFSs $(s_1, \{b_{u_1}, b_{u_2}\}), (s_4, \{b_{u_1}, b_{u_2}\})$ terminates, and for $v_1$, its in-going edge $< s_4, v1 >$ is flipped to be "on", so we get the next level of BFs $(s_4, \{b_{u_3}\})$ and update its $Seen$ set $Seen_{s_4} = \{u_1, u_2, u_3\}$. As the node $s_4$ has no in-going edges, so all of the BFSs terminates. By the statistic of the $Seen$ sets, we get an UIS sequence sample $\mathcal{R} = \{\{s_1, s_4\}, \{s_1, s_4, s_2\}, \{s_3, s_4\}\}$.

Nextly, we will give how the UIS is used to estimate our objective. Given a set sequence $\mathcal{R}$ and sets $r, S$, let $\mathcal{I}_r(S), \mathcal{I}_{\mathcal{R},\theta}$ be the indicator functions as follows:

$$\mathcal{I}_r(S) = \begin{cases} 0 & r \cap S = \emptyset, \\ 1 & r \cap S \neq \emptyset, \end{cases} \quad \mathcal{I}_{\mathcal{R},\theta}(S) = \begin{cases} 1 & \sum_{r \in R} \mathcal{I}_r(S) \geq N_\theta, \\ 0 & \sum_{r \in R} \mathcal{I}_r(S) < N_\theta. \end{cases}$$

Let $\Omega$ be the sample space of the random UIS set, and hence we can get a discrete random variable $\xi_S : \Omega \to \{\alpha, \alpha+\beta\}$ where $\xi_S(R) = \alpha \mathcal{I}_{\mathcal{R},\theta}(S) + \beta$. Nextly we prove that the expectation of $\xi_S$ equals to the objective as follows.

**Theorem 4.** *Given any alternative seeds set $S$, we have $\sigma_\theta(S) = E(\xi_S)$.*

**Proof.** By the Theorem and the definition of UIS, let $\bar{g}$ be the reverse graph of a direct graph $g$ and we have

$$\begin{aligned}
\sigma_\theta(S) &= \alpha Pr_{g\sim G}\{|g_S \cap |\mathbb{U}| \geq N_\theta\} + \beta \\
&= \alpha Pr_{g\sim G}\{\sum_{u\in\mathbb{U}} \mathcal{I}_{\{u\}}(g_S) \geq N_\theta\} + \beta \\
&= \alpha Pr_{g\sim G}\{\sum_{u\in\mathbb{U}} \mathcal{I}_{\{\bar{g}_{\{u\}}\}}(S) \geq N_\theta\} + \beta \\
&= \alpha Pr_{\mathcal{R}}\{\sum_{r_j\in\mathcal{R}} \mathcal{I}_r(S) \geq N_\theta\} + \beta \\
&= \alpha Pr_{\mathcal{R}}\{\mathcal{I}_{\mathcal{R},\theta}(S) = 1\} + \beta \\
&= (\alpha+\beta) Pr_{\mathcal{R}}\{\mathcal{I}_{\mathcal{R},\theta}(S) = 1\} + \beta(1 - Pr_{\mathcal{R}}\{\mathcal{I}_{\mathcal{R},\theta}(S) = 0\}) \\
&= E(\xi_S) \quad \square
\end{aligned}$$

Then we can use the statistic method sampling UIS only once to estimate the expectation of the random variable $\xi_S$ for any given seeds set $S$ instead of Monte Carlo simulations in which it needs re-simulations once the seeds set is changed. Sampling $\lambda$ UIS sequences independently and getting $\Re_\lambda = \{\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_\lambda\}$, we have that $\bar{\sigma}_\theta(S, \Re_\lambda) := \frac{\alpha}{\lambda}\sum_{i\in[\lambda]} \mathcal{I}_{\mathcal{R}_i}(S) + \beta$ is the unbiased estimation for $\sigma_\theta(S)$.

Nextly, we analyze the gap of error between the estimation and truth which is highly related to the sampling number $\lambda$ as shown in Theorem 5, and to prove it, we firstly introduce the Chernoff Bounds [28] as shown in Lemma 2.

**Lemma 2.** *Let $X_1, X_2, \cdots, X_\lambda$ be random variables such that $a \leq X_i \leq b$ for all $i$. Let $X = \sum_{i=1}^{\lambda} X_i$ and $\mu = E(X)$. Then for any $\epsilon \geq 0$,*

$$Pr\{X - \mu \geq \epsilon\mu\} \leq exp(-\frac{2\epsilon^2\mu^2}{\lambda(b-a)^2}), \tag{1}$$

$$Pr\{X - \mu \leq -\epsilon\mu\} \leq exp(-\frac{\epsilon^2\mu^2}{\lambda(b-a)^2}). \tag{2}$$

**Theorem 5.** *If $\lambda \geq \frac{-\alpha^2}{\epsilon^2\beta^2} \cdot min\{\frac{\ln(x)}{2}, \ln(1-\delta-x)\}$, we have $Pr\{|\bar{\sigma}_\theta - \sigma| \leq \epsilon\sigma\} \geq \delta$, where $0 \ll \delta < 1$, $0 < x < 1-\delta$ and $0 < \epsilon \ll 1$.*

**Proof.** Let $X_i = \alpha\mathcal{I}_{\mathcal{R}_i}(S) + \beta$, and then $\beta \leq X_i \leq \alpha+\beta$. We have $X = \sum_{i=1}^{\lambda} X_i = \lambda\bar{\sigma}_\theta$, and $\mu = E(X) = E(\lambda\cdot\xi_S)) = \lambda\cdot\sigma_\theta$. By the Equation (1) in Lemma 2, since $\lambda \geq \frac{-\ln(x)\alpha^2}{2\epsilon^2\beta^2}$, we have

$$Pr\{\lambda\bar{\sigma}_\theta - \lambda\sigma_\theta \geq \epsilon\cdot\lambda\sigma_\theta\} = Pr\{\bar{\sigma}_\theta - \sigma_\theta \geq \epsilon\sigma\} \leq exp(-\frac{2\epsilon^2\lambda\sigma_\theta^2}{\alpha^2}) \leq exp(-\frac{2\epsilon^2\lambda\beta^2}{\alpha^2}) \leq x.$$

Then get $Pr\{\bar{\sigma}_\theta - \sigma_\theta > \epsilon\sigma_\theta\} \leq x$. By the Equation (2) in Lemma 2, since $\lambda \geq \frac{-\ln(1-\delta-x)\alpha^2}{\epsilon^2\beta^2}$, we have

$$Pr\{\lambda\bar{\sigma}_\theta - \lambda\sigma_\theta \leq -\epsilon\cdot\lambda\sigma_\theta\} = Pr\{\bar{\sigma}_\theta - \sigma_\theta \leq -\epsilon\sigma_\theta\} \leq exp(-\frac{\epsilon^2\lambda\sigma_\theta^2}{\alpha^2}) \leq exp(-\frac{\epsilon^2\lambda\beta^2}{\alpha^2}) \leq 1-\delta-x.$$

Then get $Pr\{\bar{\sigma}_\theta - \sigma_\theta < -\epsilon\sigma_\theta\} \leq 1-\delta-x$. So we have $Pr\{|\bar{\sigma}_\theta - \sigma_\theta| \leq \epsilon\sigma\} = 1 - Pr\{|\bar{\sigma}_\theta - \sigma_\theta| > \epsilon\sigma_\theta\} = 1 - (Pr\{\bar{\sigma}_\theta - \sigma_\theta > \epsilon\sigma_\theta\} + Pr\{\bar{\sigma}_\theta - \sigma_\theta < -\epsilon\sigma_\theta\}) \geq 1 - (x + 1 - \delta - x) = \delta$. $\square$

As the running time of sampling is highly related to the samples number, and by the theorem above, we need reduce $\lambda$ as small as possible while guaranteeing the estimation precision $\epsilon$ and confidence $1-\delta$, i.e., we need consider the question to get the lower bound $\lambda_l(\epsilon,\delta)$ of $\{\lambda|\lambda \geq \frac{-\alpha^2}{\epsilon^2\beta^2} \cdot min\{\frac{\ln(x)}{2}, \ln(1-\delta-x)\}, 0 < x < 1-\delta, 0 < \epsilon < 1\}$. Mark function $f(x) := min\{\frac{\ln(x)}{2}, \ln(1-\delta-x)\}$, and then we have $\lambda_l(\epsilon,\delta) = \frac{-\alpha^2}{\epsilon^2\beta^2} f(x^*)$, where $f(x), x \in (0, 1-\delta)$ gets the absolute maximum at $x^*$. Let $g(x) := max\{x, (1-\delta-x)^2\}$, by the monotonicity of function $ln(\cdot)$, we can have $f(x) = \frac{lng(x)}{2}, x \in (0, 1-\delta)$ and then the question of solving absolute maximum of $f(x), x \in (0, 1-\delta)$ corresponds to solve the absolute minimum problem of

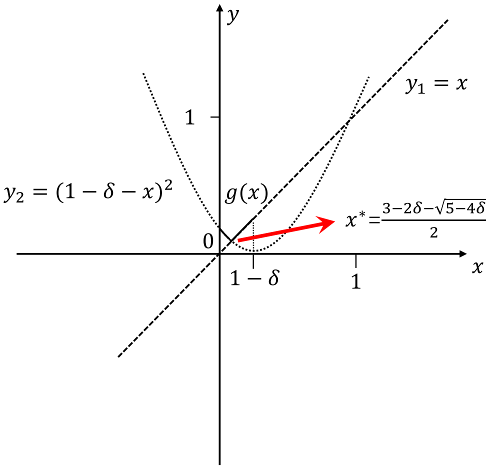


**Fig. 4.** Solve the absolute minimum problem of $g(x), x \in (0, 1-\delta)$ in coordinate system.

$g(x), x \in (0, 1-\delta)$. We can draw the $g(x)$, $x \in (0, \delta)$ into coordinate system to solve its minimum problem. As shown in Fig. 4, we have the $g(x), x \in (0, \delta)$ gets absolute minimum where $y_1 = y_2$, i.e., $x^* = (1-\delta-x^*)^2$. Further, we can compute $x^* = \frac{3-2\delta-\sqrt{5-4\delta}}{2}$, $g(x^*) = x^*$ and we have $f(x^*) = \frac{ln(x^*)}{2} = \frac{ln(3-2\delta-\sqrt{5-4\delta})-ln2}{2}$. So at last, we get $\lambda_l(\epsilon, \delta) = \frac{-\alpha^2(ln(3-2\delta-\sqrt{5-4\delta})-ln2)}{2\epsilon^2\beta^2}$.

## 5. The algorithms to solve UAPM

By above analysis such like Theorem 5, ignoring the loss of estimation after sampling sufficient number of UIS sequences, we can get a solution of UAPM by solving the maximization problem of $\bar{\sigma}_\theta$ instead of the origin target $\sigma$, i.e., find the nodes set

$$S^\star := \underset{S \subseteq \mathcal{S}, |S| \leq k}{argmax}\ \bar{\sigma}_\theta(S, \Re_\lambda) \tag{3}$$

Further, we can illustrate an new problem from the motivation above as follows:

**Definition 3** *(Set sequences θ-coverage problem (θ-SSCP)).* Given a collection of set sequences $\mathcal{C} = \{C_1, C_2, \ldots, C_n\}$ and a node set $T = \{t_1, t_2, \ldots, t_n\}$, set sequence $\theta$-coverage problem is to find a set of $k$ nodes in $T$ to maximize the number of set sequences $\theta$-covered by it, where we say a set sequence $C_i$ being $\theta$-**covered** by a set of nodes $T'$, that is, in the sequence, there are at least $\theta$ proportion of sets whose intersection with $T'$ is not emptyset.

Reviewing the problem in Equation (3), from the target function $\bar{\sigma}_\theta$ we have $\sum_{i\in[\lambda]} \mathcal{I}_{\mathcal{R}_i,\theta}(S)$ is the count number of set sequences in $\Re_\lambda$ $\theta$-covered by $S$, then we have it equals to a case of $\theta$-SSCP problem above we defined. Further back to our original problem, we can have the following theorem.

**Theorem 6.** *Let $S^\star$ be the optimal solution of the equivalent θ-SSCP of the problem in Equation (3) with set sequences $\Re_\lambda$, and then if $\lambda \geq \lambda_l(\frac{\epsilon}{2-\epsilon}, \delta)$, we have*

$$\sigma_\theta(S^\star) \geq (1-\epsilon)\sigma_\theta(S^*)$$

*with probability at least* $2\delta - 1$.

**Proof.** We have $S^\star$ is also an optimal solution for the problem in Equation (3), and $\bar{\sigma}_\theta(S^\star) \geq \bar{\sigma}_\theta(S^*)$. By the Theorem 5, if $\lambda \geq \lambda_l(\frac{\epsilon}{2-\epsilon}, \delta)$, we have $\bar{\sigma}_\theta(S^*) \geq (1 - \frac{\epsilon}{2-\epsilon})\sigma_\theta(S^*) = \frac{2(1-\epsilon)}{2-\epsilon}\sigma_\theta(S^*)$ and $\sigma_\theta(S^\star) \geq \frac{1}{1+\frac{\epsilon}{2-\epsilon}}\bar{\sigma}_\theta(S^\star) = \frac{2-\epsilon}{2}\bar{\sigma}_\theta(S^\star)$ with the union probability is at least $2\delta - 1$. □

The Theorem 6 tells that we can get a high accuracy and confidence solution for the UAPM by solving an related $\theta$-SSCP problem. So nextly, we consider how to solve the $\theta$-SSCP problem. It seems that our $\theta$-SSCP is little similar with the set coverage problem [12] which is similarly but choosing $k$ nodes to maximize the number of sets covered[2] by them, but

[2] We said a set $A$ covers a set $B$ is that $A \cap B \neq \emptyset$ and a node $a$ covers $B$ is that $a \in B$.

actually they are very different as the set coverage problem has a good property to guarantee to be submodular and our $\theta$-SSCP may be lost.

We show $\theta$-SSCP without submodularity by a special case as follows:

$$\mathcal{R}_1 = (\{v_4\}, \{v_1\}, \{v_3\}, \{v_5\}, \{v_1, v_5\}),$$

$$\mathcal{R}_2 = (\{v_1, v_2\}, \{v_2\}, \{v_4\}, \{v_1\}, \{v_5\}),$$

$$\mathcal{R}_3 = (\{v_1\}, \{v_2\}, \{v_3, v_4\}, \{v_1, v_4\}, \{v_3\})$$

we have $\bar{\sigma}_{0.5}(\{v_1, v_2\}, \Re_\lambda) - \bar{\sigma}_{0.5}(\{v_2\}, \Re_\lambda) = 2 > \bar{\sigma}_{0.5}(\{v_1\}, \Re_\lambda)$-$\bar{\sigma}_{0.5}(\emptyset, \Re_\lambda) = 0$. The general greedy algorithm can't guarantee an approximation solution, and we nextly will still first introduce a heuristic adjusted greedy algorithm and further an approximation algorithm to solve our $\theta$-SSCP problem.

### 5.1. The adjusted greedy algorithm

We still use the greedy-climbing idea of adding a seed with the maximum gain for the object function one by one. However, we often face troubles in the situations when no node can add gain for the current count number of set sequences to be $\theta$-covered. We show the trouble as hard-choice by the special case above: Supposed that we need choice 2 seeds from $\{v_1, v_2, v_3, v_4, v_5\}$, according to the greedy-climbing in rounds, in first round, we meet a problem which seeds we should choice as all of them provide no marginal gain for the objective. If we randomly choice one, e.g. seed $v_3$, we will get a solution to 0.5-cover most 2 set sequences. But if we choice seed $v_1$, we can get the best solution $\{v_1, v_4\}$ to 0.5-cover all 3 set sequences.

So facing such hard-choice trouble, to avoid the bad choice, we adopt the greedy strategy of selecting a node based on a weight we design. In each round, for each sequence, we remove all sets which has been covered by the selected seeds, and we also remove all sets once the sequence is $\theta$-covered. Then although there is no any alternative seed can cover the remain sets to promote any sequence to be $\theta$-covered, i.e., the situation of hard-choice trouble, we can count how many remain sets that the alternative seed can cover in the sequence not $\theta$-covered, i.e., we can compute a sequence remain covering ratio corresponding to the remain sets of each sequence, and the larger the better. Further, we also should consider the sequence covering ratios of the alternative seed over all sequences not only one, that is, the alternative seed with larger covering ratios over all sequence is better. So based on two such heuristic ideas, for each selection round, we compute a weight $CW_v^k := \frac{1}{|\Re_\lambda^k|} \sum_{\mathcal{R}_i \in \Re_\lambda^k} \frac{|\{r_j | v \in r_j, r_j \in \mathcal{R}_i\}|}{|\mathcal{R}_i|}$ which is the average of the sequence covering ratio over all sequences in $\Re_\lambda^k$, where $\Re_\lambda^k$ is the remain sequences after $k-1$ rounds.

So back to the above example, we can get $CW_{v_1}^1$ gets the largest weight of $\frac{2}{15}$ and hence $v_1$ will be selected in first iteration, and we avoid the bad solution misled by hard-choice trouble. We illustrate such adjusted greedy method as shown in Algorithm 3 ($\theta$-SSCP-AG).

**Algorithm 3:** $\theta$-SSCP-AG($\Re_\lambda, \mathbb{S}$).

```
S ← ∅;
ℜ_λ^1 ← ℜ_λ;
for q from 1 to k do
    Get the node $s_q \leftarrow argmax_{s \in T/S} CW_s^k$;
    Let $S \leftarrow S \cup \{s_q\}$;
    Update $\Re_\lambda^{k+1}$ from $\Re_\lambda^k$ as following:
    (1) Remove all sets covered by $s_q$;
    (2) Remove the sequence once the proportion of cumulative removed sets has been no less than $\theta$;
return S as the seed set
```

### 5.2. The approximation algorithm

Considering the similar idea of Sandwich, instead of directly solving target function $\bar{\sigma}_\theta$, we can indirectly solve it by using a tight lower bound $\bar{\sigma}_\theta^l$ and upper bound $\bar{\sigma}_\theta^u$, i.e., $\bar{\sigma}_\theta^l(S) \leq \bar{\sigma}_\theta(S) \leq \bar{\sigma}_\theta^u(S)$ for all $S \in S$. Then if we get a high approximation solution $S^l$ and $S^u$ respectively corresponding to each maximization problem of the two bounds, we can get a data-driven approximation solution for the original problem. Nextly we firstly introduce the upper and lower bounds we designed for $\mathcal{I}_{\mathcal{R},\theta}(S)$ as following Lemma 3.

**Lemma 3.** *For any $S \subseteq \mathbb{S}$, we have $L\mathcal{I}_{\mathcal{R},\theta}(S) \leq \mathcal{I}_{\mathcal{R},\theta}(S) \leq U\mathcal{I}_{\mathcal{R},\theta}(S)$, where $U\mathcal{I}_{\mathcal{R},\theta}(S) := \frac{\sum_{r \in R} \mathcal{I}_r(S))}{|R|\theta}$, $L\mathcal{I}_{\mathcal{R},\theta}(S) := \mathcal{I}_{r_\theta}(S)$ and $r_\theta := \{s | s \in \mathbb{S}, \mathcal{I}_{\mathcal{R},\theta}(\{s\}) = 1\}$.*

**Proof.** It's obviously that $\mathcal{I}_{\mathcal{R},\theta}(S) \leq U\mathcal{I}_{\mathcal{R},\theta}(S)$. When $\mathcal{I}_{\mathcal{R},\theta}(S) = 0$, we must have $L\mathcal{I}_{\mathcal{R},\theta}(S) = 0$, otherwise there is a seed $s$ in $S$, s.t., $\mathcal{I}_{\mathcal{R},\theta}(\{s\}) = 1$ and hence $L\mathcal{I}_{\mathcal{R},\theta}(S) = 1$ contradictorily. So $L\mathcal{I}_{\mathcal{R},\theta}(S) \leq \mathcal{I}_{R_i}(S)$. □

Then we have two bounds for $\bar{\sigma}_\theta$ as shown in Theorem 8 which can be naturally inferred by Lemma 3.

**Theorem 7.** *For all $S \subseteq S$, we have $\eta^l_\theta(S, \Re_\lambda) \leq \bar{\sigma}_\theta(S, \Re_\lambda) \leq \eta^u_\theta(S, \Re_\lambda)$, where $\eta^l_\theta(S, \Re_\lambda) := \frac{\alpha}{\lambda} \sum_{i\in[\lambda]} L\mathcal{I}_{\mathcal{R}_i}(S) + \beta$ and $\eta^u_\theta(S, \Re_\lambda) := \frac{\alpha}{\lambda} \sum_{i\in[\lambda]} U\mathcal{I}_{\mathcal{R}_i}(S) + \beta$ are nonnegative, monotonic increasing, submodular bounds.*

**Proof.** By the Lemma 3, it's easy to have $\eta^l_\theta(S, \Re_\lambda) \leq \bar{\sigma}_\theta(S, \Re_\lambda) \leq \eta^u_\theta(S, \Re_\lambda)$, and by the definition, we have these properties for $\eta^l_\theta$ and $\eta^u_\theta$ are consistent with the indicator function $\mathcal{I}_r(\cdot)$. It's easy to have $\mathcal{I}_r(\cdot)$ is nonnegative, monotonic increasing and submodular. □

Then the general greedy algorithm can provide $(1 - 1/e)$-approximation solutions $S^l, S^u$ for the maximum problem of $\eta^l_\theta$ and $\eta^u_\theta$ respectively. Then further we can get the SA-algorithm in Algorithm 4 can provide a solution for our $\theta$-SSCP problem and hence a solution for our original problem with a data-driven approximation guaranteed as Theorem 8.

**Algorithm 4:** $\theta$-SSCP-SA$(G_{U,S}, k, \epsilon, \delta)$.

```
Sample $\lambda = \lambda_l(\frac{\epsilon}{2-\epsilon}, \delta)$ UIS sequences by Algorithm 1;
Using general greedy algorithm to get a solution $S^l$ for $\eta^l_\theta$;
Using general greedy algorithm to get a solution $S^u$ for $\eta^u_\theta$;
Let $S^+ = argmax\{\bar{\sigma}_\theta(S^l), \bar{\sigma}_\theta(S^u)\}$;
return $S^+$
```

**Theorem 8.** *The solution $S^+$ given by Algorithm 4 can guarantee that*

$$\sigma_\theta(S^+) \geq \beta(1 - \frac{1}{e})\sigma_\theta(S^*)$$

*with probability at least $2\delta - 1$, where*

$$\beta = (1 - \epsilon) \cdot max\{\frac{\bar{\sigma}_\theta(S^u)}{\eta^u_\theta(S^u)}, \frac{\eta^l_\theta(S^l)}{\bar{\sigma}_\theta(S^\star)}\}.$$

**Proof.** We have

$$\bar{\sigma}_\theta(S^u) = \frac{\bar{\sigma}_\theta(S^u)}{\eta^u_\theta(S^u)} \cdot \eta^u_\theta(S^u) \geq \frac{\bar{\sigma}_\theta(S^u)}{\eta^u_\theta(S^u)} \cdot (1 - \frac{1}{e}) \cdot \eta^u_\theta(S^\star) \geq \frac{\bar{\sigma}_\theta(S^u)}{\eta^u_\theta(S^u)} \cdot (1 - \frac{1}{e}) \cdot \bar{\sigma}_\theta(S^\star),$$

$$\bar{\sigma}_\theta(S^l) = \eta^l_\theta(S^l) \geq (1 - \frac{1}{e}) \cdot \eta^l_\theta(S^\star) \geq (1 - \frac{1}{e}) \cdot \frac{\eta^l_\theta(S^l)}{\bar{\sigma}_\theta(S^\star)}\bar{\sigma}_\theta(S^\star),$$

$$\bar{\sigma}_\theta(S^+) \geq (1 - 1/e) \cdot max\{\frac{\bar{\sigma}_\theta(S^u)}{\eta^u_\theta(S^u)}, \frac{\eta^l_\theta(S^l)}{\bar{\sigma}_\theta(S^\star)}\} \cdot \bar{\sigma}_\theta(S^\star) \geq (1 - 1/e) \cdot max\{\frac{\bar{\sigma}_\theta(S^u)}{\eta^u_\theta(S^u)}, \frac{\eta^l_\theta(S^l)}{\bar{\sigma}_\theta(S^\star)}\} \cdot \bar{\sigma}_\theta(S^*).$$

By the Theorem 5, if $\lambda \geq \lambda_l(\frac{\epsilon}{2-\epsilon}, \delta)$, we have $\bar{\sigma}_\theta(S^*) \geq (1 - \frac{\epsilon}{2-\epsilon})\sigma(S^*) = \frac{2(1-\epsilon)}{2-\epsilon}\sigma(S^*)$ and $\sigma(S^+) \geq \frac{1}{1+\frac{\epsilon}{2-\epsilon}}\bar{\sigma}_\theta(S^+) = \frac{2-\epsilon}{2}\bar{\sigma}_\theta(S^+r)$ with the union probability is at least $2\delta - 1$. □

Note that we can't get $\beta$ exactly because of $\frac{\eta^l_\theta(S^l)}{\bar{\sigma}_\theta(S^\star)}$ since the optimum solution of $\bar{\sigma}_\theta(S^\star)$ is hard to get, but we can directly compute the value of $\frac{\bar{\sigma}_\theta(S^u)}{\eta^u_\theta(S^u)}$, which is a lower bound for the $\beta$, and hence we still can definitely guarantee and estimate the theoretical accuracy for the result. We simply introduce the physical motivations and explanations for the sandwich algorithm based on the lower bound and upper bound we design. The lower bound is aimed to select the seeds from the node like $s_1$ who has high probability to simultaneously influence at least $\theta$ proportion of nodes in the union. The upper bound is aimed to select the seeds from the node like $s_2$ who can highly influence the nodes in the union as more as possible. So in the network, if there are more the nodes like $s_1$, the best solution may be closed to the one provided by lower bound, and if there are more nodes like $s_2$ not $s_1$, the best solution may be closed to the one provided by upper bound.

**Table 1**
Datasets.

| Dataset | #Nodes | #Edges |
|---|---|---|
| BlogCatalog | 10K | 333K |
| Flickr | 80K | 5.9M |
| DBLP | 203K | 382K |
| Twitter | 580K | 717K |

## 6. Experiments

We have conduct an experimental study to evaluate the performance of our proposed methods over 4 real-world datasets[3] (BlogcCatalog, Flickr, DBLP, Twitter) as shown in Table 1. All codes of the experiments are written in c++ and all experiments run in a linux server with a 12 cores, 24 threads, 3.6 Hz, CPU and 64G memory.

### 6.1. Experiment setup

**Influence probability**: As the general setting in IC model, we set the influence probability i.e. the weight of each direct edge $< u, v >$, $p_{uv} := \frac{1}{d_{in}(v)}$ where $d_{in}(v)$ is the in-degree of node $v$.

**Union choice**: For the node choice of the union, we first exclude the nodes with zero in-degree as they can't be influenced. Since the node's influence probability is related to the in-degree, so to avoid low union influence probability of the union, we specially choice nodes for each union from the remaining nodes according to the in-degree such that is the node with lower in-degree has larger probability to be chosen.

**Influence evaluation**: We use 10000 times of Menton Carlo simulations and count the proportion of union being acceptable to get the evaluation of union acceptable probability for given seeds.

**Alternative seeds**: We exclude the nodes with zero out-degree as they can't influence anyone, we also exclude the union nodes and their one hop in-going neighbors as its easy to make choice by selecting nodes from them. So we set the alternative seeds $\mathbb{S} = V/(\mathbb{U} \bigcup (\cup_{u \in \mathbb{U}} N_u^{in}) \cup \{v | d_{ou}(v) = 0, v \in V\})$, where $d_{in}(v)$ is the out-degree of node $v$ and $N_u^{in}$ are u's one hop in-going neighbors.

**Global Parameters setting**: By default, we set the parameters as follows. Let estimation error $\epsilon = 0.1$, and confidence $\delta = 0.99$, which can guarantee a high accuracy and confidence for the algorithm we designed. Set the profit when the union is acceptable or not as $c_1 = 100$, $c_2 = 1$ respectively. Set that the lower bound of the proportion of influenced members in the union to promote the union to be acceptable is 0.5, i.e., at least half of the members in the union to be influenced.

We compare our algorithms with some baseline algorithms as follows.

- **Target-IM**: The algorithm proposed [12] to solve the targeted influence in $\mathbb{U}$ to maximize the expected number of members influenced in union.
- $\theta$**-SSCP-GG**: The general greedy algorithm to solve the $\theta$-SSCP problem.
- $\theta$**-SSCP-AG**: The adjusted greedy algorithm we proposed to solve the $\theta$-SSCP problem.
- $\theta$**-SSCP-SA**: The Sandwich-based algorithm we proposed to solve the $\theta$-SSCP problem.
- **Random**: The basic baseline algorithms by choosing seeds randomly.

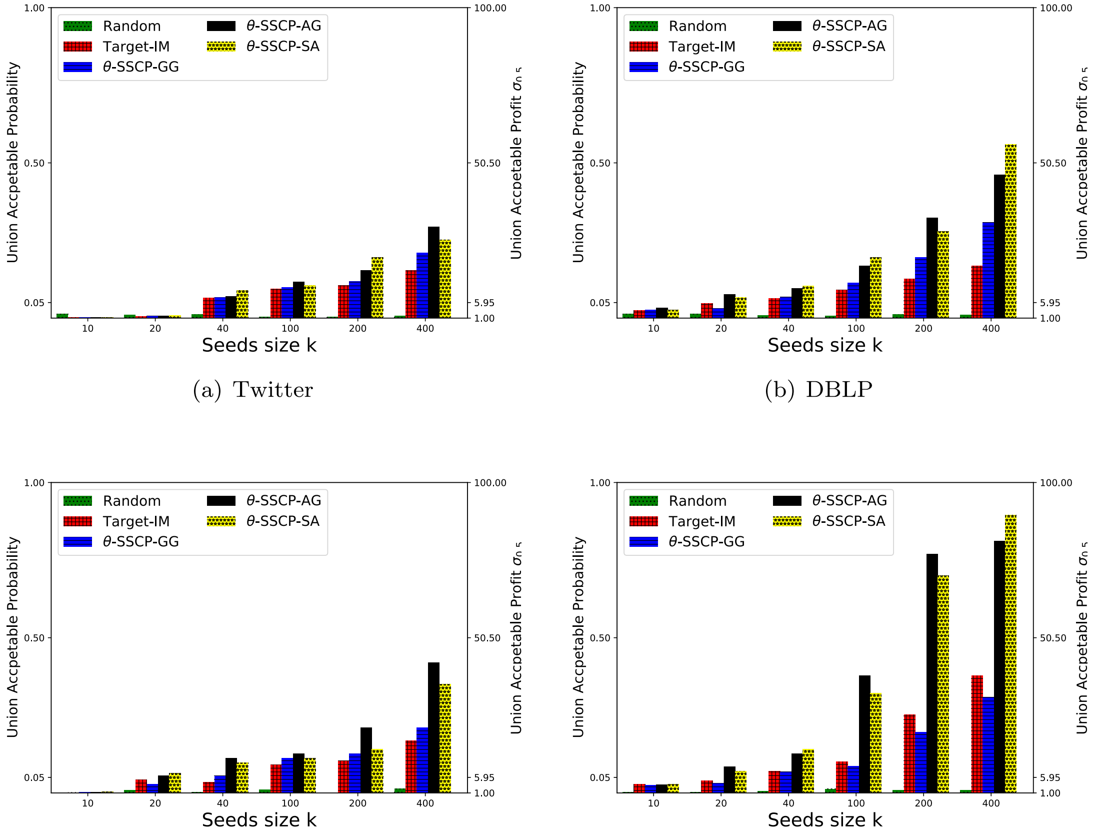
We also compare the optimization UIS sampling algorithm **RRMS-BFS** based on random multisource-BFS with the basic sampling algorithm **SG-BFS** based on repeated single-source-BFS.

### 6.2. Experiment result

**Union acceptable profit:** Firstly, for each dataset, we choose 50 nodes to compose an union to evaluate the performance of the different algorithms by running each algorithm 10 times. We vary the budgets k by 10,20,40,100,200,400 by setting half of the budgets are less than the union size and half are more than the union size. At last, we report the best result of the 10 running instances for each algorithm. As shown in Fig. 5, the algorithms we proposed are significantly better than others and specially the improved algorithms $\theta$-SSCP-SA and $\theta$-SSCP-AG have great improvement compared with the general $\theta$-SSCP-GG and T arget-IM (at least 20% in DBLP, 10% in Twitter and Flickr, 50% in BlogCatalog). In all datasets, the $\theta$-SSCP-SA and $\theta$-SSCP-AG has the similar performance except that $\theta$-SSCP-AG keeps improving almost 10% more than $\theta$-SSCP-SA in Flickr.

**Lower bound of budget cost:** Nextly we evaluate the lower bound of the budget cost (i.e., the necessary number of seeds) for certain profit requirement in each dataset. We set 6 unions with different sizes of 2, 10, 50, 100, 200 and vary the budgets k from 1 to 500 by a step of 10 to compare the lower bound of the budget satisfying the expect profit at least $0.5c_1 + c_2$, where the profit $c_1 = 100, c_2 = 1$, under different algorithms, by running each 10 times. Special for the

[3] http://networkrepository.com.



(a) Twitter

(b) DBLP

(c) Flickr

(d) BlogCatalog

**Fig. 5.** The performance comparisons achieved by different algorithms in different datasets with an union of size 50.

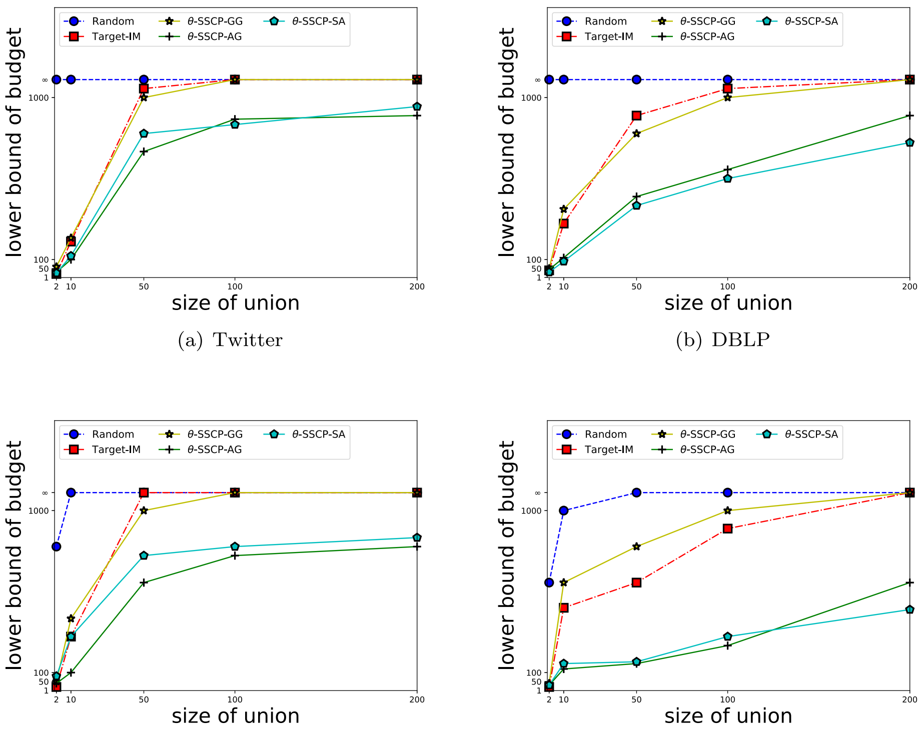
**Table 2**
$\theta \sim \frac{\alpha^2}{\beta^2}$.

| $\alpha^2/\beta^2$ | 10 | $10^2$ | $10^3$ | $10^4$ | $10^5$ |
|---|---|---|---|---|---|
| $\lambda_l(\frac{\epsilon}{2-\epsilon}, \delta)$ | 1.2E+02 | 1.2E+03 | 4.61E+04 | 4.61E+05 | 4.61E+06 |

situation that all of the budgets $k (k \leq 1000)$, we can't achieve such influence, we record its lower bound as $\infty$. As shown in Fig. 6, to achieve the targeted profit, we have the budget cost comparison: Random>Target-IM≈θ-SSCP-GG≫θ-SSCP-AG≈θ-SSCP-SA.

**Union-acceptable probability maximization**: Note that the objective $\sigma_\theta$ can be seen as a linear enlarge measurement of the union-acceptable probability by Lemma 1. In another view, solve the problem to chose seeds maximize the union-acceptable probability is equivalent to the UAPM under two parameters $c_1$ and $c_2$. However, in our designed algorithms, the accuracy of solution for UAPM is highly related to USI sampling number $\lambda_l$ which is positive linear correlation to $\frac{\alpha^2}{\beta^2}$ and hence is related the setting of $c_1, c_2$, since $\alpha = c_1 - c_2$, $\beta = c_2$. So, nextly we experiment how setting $c_1, c_2$ exactly influences the accuracy of our algorithms in terms of to solving the union-acceptable probability maximization problem. We set $c_1, c_2$ by varying $\frac{\alpha^2}{\beta^2}$ as shown in Table 2, and we fix the union size to be 50 and the seeds set size to be 400. Nextly, we analyze how our adjusted greedy and SA algorithms performance. As shown in Fig. 7, in all datasets, it shows that higher value of $\frac{\alpha^2}{\beta^2}$ will improve the accuracy of the union-influenced probability and especially in the case of $\frac{\alpha^2}{\beta^2} \leq 10^3$. When $\frac{\alpha^2}{\beta^2} \geq 10^3$, the improvement is not significant.

**UIS sampling cost:** All algorithms we proposed are based on the UIS sequences, so the running time cost of these algorithms is determined mainly by the UIS sampling algorithm and the sample number $\lambda_l$ which is constant. To evaluate the performance of the improved sampling algorithms RRMS-BFS and basic SG-BFS, we record the total running time to sample 1 million UIS sequences by these two different methods. As shown in Fig. 8, the RRMS-BFS reduced time by 10 times compared to SG-BFS.



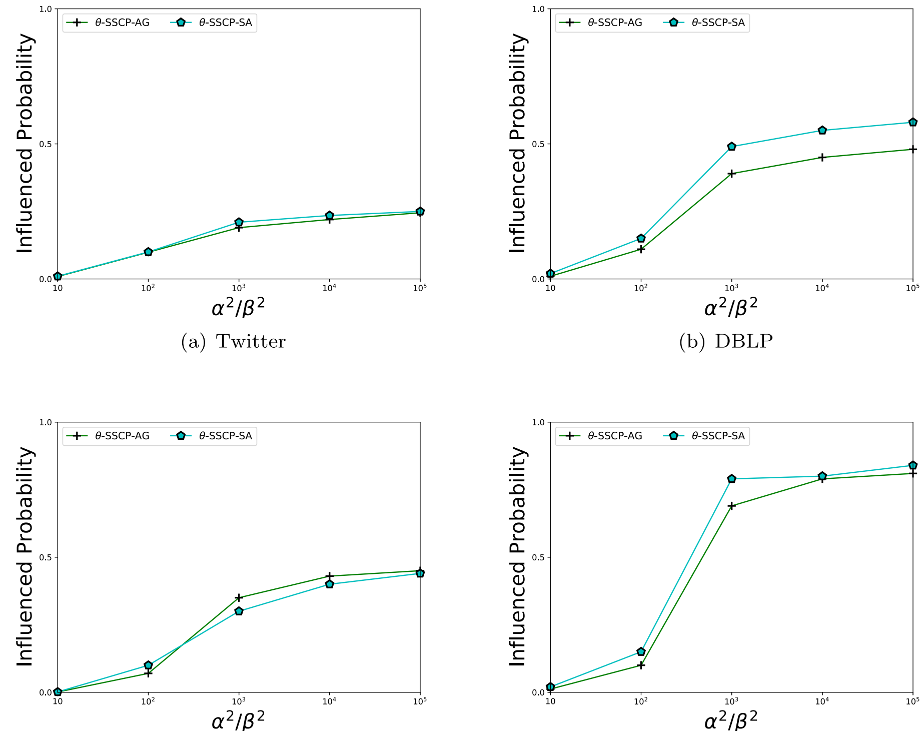
(a) Twitter

(b) DBLP

(c) Flickr

(d) BlogCatalog

**Fig. 6.** The comparisons for lower bound of budget with union accept profit at least $0.5c_1 + c_2$ achieved by different algorithms in different datasets for different sizes of unions.



(a) Twitter

(b) DBLP

(c) Flickr

(d) BlogCatalog

**Fig. 7.** The comparisons for our designed algorithms by set $c_1, c_2$ to vary $\theta \sim \frac{\alpha^2}{\beta^2}$ with $k = 400$, $|\mathbb{U}| = 50$.

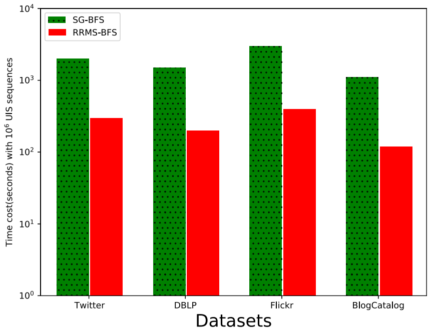


**Fig. 8.** The time cost comparisons for two sample algorithms in different datasets with the sampling number of 1 million.

## 7. Conclusions

In this paper, we propose the union acceptable profit maximization (UAPM) problem in which the goal is to find a seed set with budget size to maximize the excepted profit when union is acceptable. We show the UAPM is NP-hard and the computation of target function is #P-hard. Without the property of submodularity, we propose several algorithms (a heuristic algorithm and a $\beta(1-\frac{1}{e})$-approximation algorithm) based on the union reverse influence set-sequences and propose a random multi source breadth-first search method to optimize the process of sampling the UIS sequence. To analyze and evaluate proposed methods, a lot of experiments have been conducted on real-world datasets. The results show that the methods we proposed perform well.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## References

[1] G. Rao, Y. Wang, W. Chen, D. Li, W. Wu, Maximize the probability of union-influenced in social networks, in: International Conference on Combinatorial Optimization and Applications, Springer, 2021, pp. 288–301.

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

[3] B. Liu, G. Cong, D. Xu, Y. Zeng, Time constrained influence maximization in social networks, in: 2012 IEEE 12th International Conference on Data Mining, IEEE, 2012, pp. 439–448.

[4] E. Cohen, D. Delling, T. Pajor, R.F. Werneck, Timed influence: computation and maximization, preprint, arXiv:1410.6976, 2014.

[5] N. Barbieri, F. Bonchi, G. Manco, Topic-aware social influence propagation models, Knowl. Inf. Syst. 37 (3) (2013) 555–584.

[6] S. Chen, J. Fan, G. Li, J. Feng, K.-l. Tan, J. Tang, Online topic-aware influence maximization, Proc. VLDB Endow. 8 (6) (2015) 666–677.

[7] S. Bharathi, D. Kempe, M. Salek, Competitive influence maximization in social networks, in: International Workshop on Web and Internet Economics, Springer, 2007, pp. 306–311.

[8] W. Lu, W. Chen, L.V. Lakshmanan, From competition to complementarity: comparative influence diffusion and maximization, Proc. VLDB Endow. 9 (2) (2015) 60–71.

[9] X. He, G. Song, W. Chen, Q. Jiang, Influence blocking maximization in social networks under the competitive linear threshold model, in: Proceedings of the 2012 Siam International Conference on Data Mining, SIAM, 2012, pp. 463–474.

[10] L. Sun, W. Huang, P.S. Yu, W. Chen, Multi-round influence maximization, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, ACM, 2018, pp. 2249–2258.

[11] G. Li, S. Chen, J. Feng, K.-l. Tan, W.-s. Li, Efficient location-aware influence maximization, in: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, ACM, 2014, pp. 87–98.

[12] C. Song, W. Hsu, M.L. Lee, Targeted influence maximization in social networks, in: Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, ACM, 2016, pp. 1683–1692.

[13] J. Goldenberg, B. Libai, E. Muller, Talk of the network: a complex systems look at the underlying process of word-of-mouth, Mark. Lett. 12 (3) (2001) 211–223.

[14] M. Granovetter, Threshold models of collective behavior, Am. J. Sociol. 83 (6) (1978) 1420–1443.

[15] J. Guo, P. Zhang, C. Zhou, Y. Cao, L. Guo, Personalized influence maximization on social networks, in: Proceedings of the 22nd ACM International Conference on Information & Knowledge Management, 2013, pp. 199–208.

[16] D.-N. Yang, H.-J. Hung, W.-C. Lee, W. Chen, Maximizing acceptance probability for active friending in online social networks, in: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2013, pp. 713–721.

[17] H. Chen, W. Xu, X. Zhai, Y. Bi, A. Wang, D.-Z. Du, How could a boy influence a girl?, in: 2014 10th International Conference on Mobile Ad-Hoc and Sensor Networks, IEEE, 2014, pp. 279–287.

[18] G.L. Nemhauser, L.A. Wolsey, M.L. Fisher, An analysis of approximations for maximizing submodular set functions—I, Math. Program. 14 (1) (1978) 265–294.

[19] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, N. Glance, Cost-effective outbreak detection in networks, in: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2007, pp. 420–429.

[20] A. Goyal, W. Lu, L.V. Lakshmanan, Celf++: optimizing the greedy algorithm for influence maximization in social networks, in: Proceedings of the 20th International Conference Companion on World Wide Web, ACM, 2011, pp. 47–48.

[21] N. Ohsaka, T. Akiba, Y. Yoshida, K.-i. Kawarabayashi, Fast and accurate influence maximization on large networks with pruned Monte-Carlo simulations, in: Twenty-Eighth AAAI Conference on Artificial Intelligence, 2014.
[22] Y. Tang, X. Xiao, Y. Shi, Influence maximization: near-optimal time complexity meets practical efficiency, in: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, ACM, 2014, pp. 75–86.
[23] C. Borgs, M. Brautbar, J. Chayes, B. Lucier, Maximizing social influence in nearly optimal time, in: Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SIAM, 2014, pp. 946–957.
[24] Y. Tang, Y. Shi, X. Xiao, Influence maximization in near-linear time: a martingale approach, in: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, ACM, 2015, pp. 1539–1554.
[25] H.T. Nguyen, M.T. Thai, T.N. Dinh, Stop-and-stare: optimal sampling algorithms for viral marketing in billion-scale networks, in: Proceedings of the 2016 International Conference on Management of Data, ACM, 2016, pp. 695–710.
[26] J. Tang, X. Tang, X. Xiao, J. Yuan, Online processing algorithms for influence maximization, in: Proceedings of the 2018 International Conference on Management of Data, 2018, pp. 991–1005.
[27] M. Then, M. Kaufmann, F. Chirigati, T.-A. Hoang-Vu, K. Pham, A. Kemper, T. Neumann, H.T. Vo, The more the merrier: efficient multi-source graph traversal, Proc. VLDB Endow. 8 (4) (2014) 449–460.
[28] R. Motwani, P. Raghavan, Randomized Algorithms, Cambridge University Press, 1995.