Information Sciences 629 (2023) 204–221

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Maximizing the influence with κ-grouping constraint ☆

Guoyao Rao [a], Deying Li [a,*], Yongcai Wang [a], Wenping Chen [a], Chunlai Zhou [a], Yuqing Zhu [b]

[a] School of Information, Renmin University of China, Beijing, 100872, China
[b] Research Center for Applied Mathematics and Machine Intelligence, Zhejiang Lab, Hangzhou, Zhejiang, China

ARTICLE INFO

Keywords:
Influence maximization
Social networks
Group buying
Viral marketing

ABSTRACT

Recently, a new business model called online group buying is emerging into our daily lives. For example, the online business platforms provide people group-discount coupons which will be issued for at least k buyers grouping for a purchase. With the coupon link widely shared over the social platforms, they hope to promote people into groups to facilitate more purchases. Inspired by aforementioned real-world scenario with grouping constraint of a given minimum number of group members (κ-grouping constraint), in this paper, we analyze and model the diffusion-group behavior, and propose the κ-grouping joining influence maximization (κ-GJIM) problem. Our problem aims to choose budgeted seeds to maximize the number of κ-grouping joiners by social influence, where a κ-grouping joiner is a person who can group with at least $\kappa-1$ ($\kappa \geq 2$) like-mind partners. We prove that this problem is NP-hard. We also prove that the computation of objective is #P-hard and then propose an efficient method to estimate the objective. We show that κ-GJIM is a non-submodular optimization problem, and then design two algorithms to solve it. At last, the experiments based on real-world datasets show that our methods provide good strategies for maximizing the influence with κ-grouping constraint.

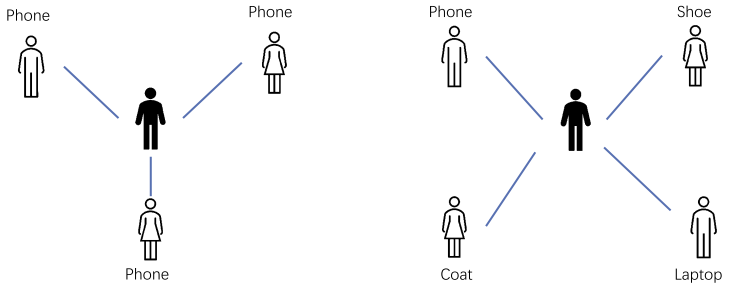
1. Introduction

The Internet has changed our modern life. It brought the online shopping into business and gave birth to many famous online shopping companies such as Amazon and Taobao. The breakout of COVID-19 further pushed more offline businesses to switch to online sales. Meanwhile, many new and creative online business models have been launched. As one of the most successful online business models, the online group buying [9] (also known as collective buying) offers products and services at significantly reduced prices once the buyers form a group with the minimum size requirement to online shop together. With the same item to buy, several individual shoppers who may be friends or possibly strangers can group into an entirety through the internet to collectively bargain with the businesses to get discounts. Since shoppers can benefit by paying less and the businesses can get profits by selling more items in bulk and reducing the cost from unstable transaction and overstock, this new business model has attracted many attentions [15,38,36] of both industry and academia. With the help of internet platform, this win-win business model of online group buying

☆ This work is partly supported by National Natural Science Foundation of China under grant 12071478, 61972404, 62102376, and Renmin University of China.
* Corresponding author.
E-mail addresses: gyr@ruc.edu.cn (G. Rao), deyingli@ruc.edu.cn (D. Li), ycw@ruc.edu.cn (Y. Wang), chenwenping@ruc.edu.cn (W. Chen), czhou@ruc.edu.cn (C. Zhou), yzhu@zhejianglab.com (Y. Zhu).

https://doi.org/10.1016/j.ins.2023.01.139
Received 20 October 2022; Received in revised form 18 January 2023; Accepted 31 January 2023

Available online 3 February 2023



(a) The circle of Bob. (b) The circle of Peter.

Fig. 1. The example to show the difference between the influence with and without κ-grouping constraint.

is on the rise and can be effectively advertised by viral marketing in online social networks, and many online shopping companies in China such as Pinduoduo, Meituan, and Taobao are vying to expand this business model in recent years. Especially Pinduoduo - the most successful social commerce platform [40] in China, has fulfilled 61 billion orders and connected more than 11 million merchants to almost 900 million users globally in 2021. Now Pinduoduo plans to layout this business model to America, and they have launched a shopping application called Temu with a slogan "team up, price down" which became the No. 1 shopping app in the U.S. App Store in September 2022.

A key condition to the development of group buying is the mobile Internet. Today, online social platforms such as Facebook, Twitter, Weibo and Wechat, are rooted in our daily lives. It's usual that we join the information propagation such as opening a shopping link recommended from friends and reposting it to others when we use these social applications. As a product of the Internet, the online social networks are playing an eye-catching role in viral marketing. Business marketing pays attention to taking advantage of the social networks, such as choosing a few people as mouthpieces to spread the positive information through the effect of word-of-mouth. This motivation creates the researches of **influence maximization (IM)**. Based on the two popular cascade diffusion models which describe how the influence spreads over the social networks, Independent Cascade Model (IC) and Linear Threshold Model (LT), Kemp et al. [18] firstly formulate the classical IM problem in discrete optimization to find budgeted seeds such that the number of influenced nodes is maximized when the influence spreads from these seeds through social network. After it, a lot of studies related to social influence have been presented such as [21,7,1,5,3,22,13,29,30].

Due to the lack of researches of group buying over social influence, in this paper, we consider the real-world approaches in grouping buying to model the grouping problem, and further formulate a new influence maximization problem to choose budgeted seeds to maximize the number of the joiners who must group with at least $\kappa - 1$ ($\kappa \geq 2$) like-minded partners among the influenced people. Significantly different from most existing literature of influence maximization, our new problem requires the precision of the influence with grouping constraint of a given minimum number of group members. We show this difference in a vivid example. Fig. 1.(a) and Fig. 1.(b) draw the friendship circles labeled with item intentions for Bob and Peter respectively. With the consideration of viral marketing with grouping constraint of at least three members, Bob is the optimal choice to be a seed since Bob's friends are all like-minded in the same intention item and hence he is more likely to lead to a group with three members. Peter can influence more friends than Bob since he has more friends in traditional influence model, but all of Peter's friends have different intention items and hence are less likely to group together to meet the 3-grouping constraint.

As shown in the example, different from the traditional IM strategies without grouping constraint, intuitively we need to identify the joiners who have high grouping chances to reach the minimum number of like-minded partners. We also need to adopt good strategies of choosing influencers to intrigue this joiner to be grouped successfully as much as possible.

Therefore, based on the motivation above, our key contributions are as follows:

1. By observing the real-world approaches in grouping buying, we propose a labeled-faith graph to model the group to predict all possible groups.

2. Facing the requirement for the minimum number of members in a group, we formulate the κ-grouping joining influence maximization problem (κ-GJIM) to choose a given number of seeds to maximize the number of the joiners who can group with at least $\kappa - 1$ ($\kappa > 2$) like-minded partners among the influenced people. As far as we know, this is the first work to promote people to group via social influence in algorithmic level.

3. We prove that κ-GJIM is NP-hard and the computation problem of its objective is #P-hard. We design a sample method to estimate the objective to make it more conveniently solve the optimal problem. We show that the optimization objective function is non-submodular, and we transfer it to be an optimization problem of the difference of two submodular functions. We also design an algorithm of adaptively greedily selecting based on a weight computation.

4. At last, we conduct several experiments for our methods based on the real-word networks. Our results demonstrate that the proposed methods provide outstanding strategies for the influence problem with grouping constraint such as the viral marketing of group buying.

In the rest of this paper, we firstly review and introduce some existing related works and formulations in section 2. In section 3, we introduce our diffusion grouping model through social influence. In section 4, we formulate the κ-grouping joining influence maximization problem and analyze some related properties. In section 5, we introduce the estimation method for the objective. In

section 6, we discuss the algorithms to solve the problem. Lastly in section 7, we show various experiments based on real-world networks and then give the conclusion in section 8.

2. Related work

In this section, we will briefly introduce some related works in influence maximization from two aspects: the variants in the extension of application scenarios and the optimization algorithms.

Since the work of Kemp et al. [18] firstly formulated the original problem of influence maximization into discrete optimization, many researchers have expanded the problem into more real-world scenarios, such as works in [21,7] with time constraint, [1,5] with the aware of the topic, [3,22] under competition model, [30] with the strategy of multiple rounds of seeds, [13] in the rumor control, [29,28] of influence on a targeted set, [35] with the fairness for groups, [2] for campaigns balance, [27] with matching relationship and so on. Specially, work [41] proposed a group influence maximization problem in social networks which is close to our work. But in their model, there is a premise that there are many definite known groups without considering how the members form into a group. Then they aim to solve how to activate these groups as much as possible and a group is said to be activated if a certain ratio of nodes in this group is activated. However, in reality, it's hard to confirm all the groups in advance since the result of grouping behavior is full of randomness. In our paper, we analyze real-world grouping mechanisms and integrate them into a more realistic model of influence maximization with group constraint.

Like the original problem, almost all variants of IM face the same hardness of the objective computation being #P-hard and the problem being NP-hard to solve. So beside focusing on the extension of application scenarios, many other works also study continuously on improving and optimizing in the aspect of algorithms. Specially, with the good property of submodularity [24] for lots of variants of the IM, the greedy hill-climbing method can guarantee the $(1-1/e)$-approximation. It runs heavily to use the Monte Carlo simulations to estimate the objective. So there are many improvements to reduce the Monte Carlo simulations such as [6,26], and CELF [20], CELF++ [11], but these algorithms are still inefficient in large-scale networks. Tang et al. [33] and Borgs et al. [4] proposed the reverse influence set (RIS) sampling method to estimate the influence for the large scale network. But there is a key problem of how to sample the RIS sets as less as possible to reduce the time complexity. After them, many RIS-based extensions and improvements are leading the researches such as the idea via Martingales (IMM) [32], stop-and-stare (SSA) and dynamic stop-and-stare (D-SSA) [25,14], and the online processing influence maximization (OPIM) [31]. Recently, as far as we have known, the work of [12] is the best one. In this paper, our algorithms are partially inspired by these estimation methods of RIS.

3. Model

In this section, we introduce the model of grouping over the influence spread on a social network $G(V, E)$. In our model, there are two phases: first, by observing realistic approaches of how people group with each other based on the premise that the ones being influenced have willingness to participate the group buying, we model it by constructing a labeled-faith graph which can predict all possible groups; Second, to formulate our final target of promoting people with both the willingness and opportunity to successfully form a group over social viral marketing, we integrate the group model in the first phase with the diffusion model of social influence.

3.1. Grouping on the labeled-faith graph

Our motivation is inspired by observing the real-world approaches of how to group buy in Pinduoduo, a leading social commerce platform which offers customers group coupons with attractive prices. We give a vivid example in Fig. 2 where the group coupons are only applicable to items A and B with the minimum number of group members being at least three. In Fig. 2.(a), people $\{f, g, j\}$ can express their grouping willingness and then wait to match partners automatically recommended by Pinduoduo's backend system. We call such approach **centralization** which is usually seen when intermediary or public platforms (e.g., a mobile application of e-commerce platform, an internet forum, a purchasing agent) recommend like-minded strangers for the registers. By cooperating with Wechat, Pinduoduo also launches groups through Wechat links and moments to inspire people like $\{a, i, e\}$ to group by a bridge from his/her friends like person b. We call such approach as **self-organizing** that people seek like-minded partner via their local social circles of friends. Naturally, we can also mix centralization and self-organizing so people like $\{c, g, d\}$ can find partners partially by the centralization or the self-organizing. From Fig. 2(a) that describes the grouping relationships in the example, we build a labeled graph as an example in Fig. 2(b) and propose a concept as follows:

Definition 1. The **labeled-faith graph** $G^f(V^f, E^f, l)$ is an labeled undirected graph. Each edge in E^f represents that the two ending nodes have **group faith** to group together directly, and each node v in V^f is labeled by $l(v)$ to represent the **item intention** to group.

Group Faith: We first claim a concept called group faith which represents that two people can group together directly once they have the same item intentions. In centralization, any two people like f and j who may be far apart in social distance can still build a group faith through a global agent. In self-organizing, two people like a and i can build a group faith through their local neighbors' bridging. However, to get the group faith, comparing to centralization, it's more complex to predict whether two persons u and v can build a group faith by a series of local friends' bridging in self-organizing. We formulate it to be an additional prediction problem of the group faith function $\gamma : V \times V \to \{0, 1\}$, where V is the nodes set. We leave it as an open work which can be modeled into an

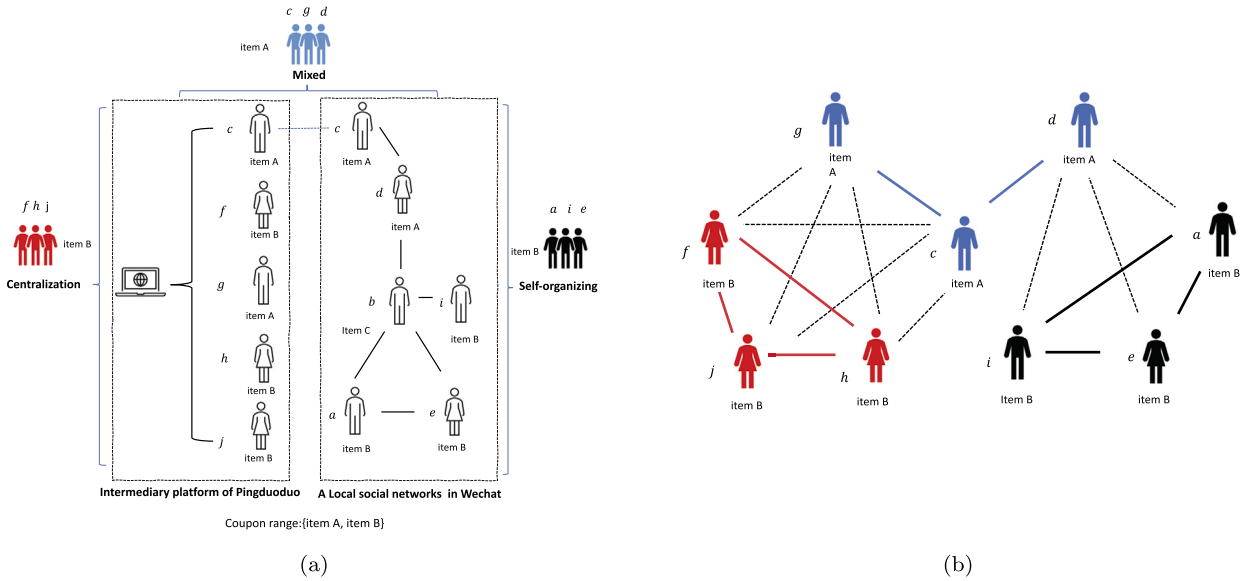


Fig. 2. The example to illustrate the motivation of modeling the labeled-faith graph, where each person has a prediction of item intention, (a) is the observation of grouping approaches in Pinduoduo and (b) is an equivalent view from the graph theory to formulate the grouping approaches.

extra machine learning problem such as learning it from the record of grouping in history. Here, we give a heuristic idea to set the group faith directly based on the social intimacy, i.e., social distance in the social network defined as follows:

$$\gamma^t(u,v) = \begin{cases} 1 & \text{if the number of hops between } u \text{ and } v \text{ is no more than } t \text{ in the social network,} \\ 0 & \text{if the number of hops between } u \text{ and } v \text{ is more than } t \text{ in the social network.} \end{cases}$$

In Fig. 2.(b), we give a vivid sample of building group faith partially by γ^2 (i.e., two nodes having faith are putting into a group because either they are friends or they have a common friends in local social networks in Wechat) and partially by a public platform in Pinduoduo.

Item Intention: Mark a label $l(v)$ for the person v's item intention to group, it can be seen as a signal flag for people to cluster, and more specifically it can be the preferences of the items in the suitable range of the group coupon like the taste preferences in meal, the different projects in playing, the commodities in buying, movies in watching, or some grouping condition limitation such as the generation of age, geographic location and so on. These labels often are obtainable by well studied methods such as the machine learning in recommendation. Note that we have $V^f \subseteq V$, i.e., not every node in V must have a label. Since the group coupon often has a limited range of items, we can remove these people from the labeled-faith graph like person b in Fig. 2.(a) whose item intention may not be in the range.

Grouping on labeled-faith graph: There is a reasonable assumption that people will group and expand the members to meet the minimum number requirement as much as possible by following intuitive strategies: (1) inviting others with the same label but without grouping; (2) merging groups with the same label into one. But we need to identify whether groups can be merged into one and whom people can invite, and the labeled-faith group can solve this issue. For strategy (1), **people can invite the neighbors in the labeled-faith graph**. For strategy (2), **like-minded groups can be merged into one as there exist edges among them in the labeled-faith graph**. So following these strategies on the labeled-faith graph, we can predict all potential groups based on a labeled-faith graph as follows:

Definition 2. For a set T of nodes from a labeled-faith graph $G^f(V^f, E^f, l)$, we define T as a **potential group** in G^f when it satisfies both following conditions:

- All nodes have the same label, i.e., $l_u = l_v$ for any $u, v \in T$.
- The vertex-induced subgraph $G^f(T)$ is a connected graph.

Specially, we call a potential group T a **maximal potential group** in G^f iff the vertex-induced subgraph $G^f(T)$ is a component of G^f.

According to the definition above, from the view of graph theory, we unify the different grouping mechanisms into a general framework of potential groups in the labeled-faith graph, i.e., how people among T finally grouping into a series of groups C by whatever mechanisms is equal to producing a series of potential groups in labeled-faith graph G^f. Nextly we combine it with social influence.

3.2. Integrating grouping with social influence

Let us see how to promote group buying over social viral marketing: Initially, the businesses will choose a few persons as seeds, and these seeds will spread the information of the activities such as providing the link of group purchase coupons. Like the spread of the virus, in the word-of-mouth, people will try to share and repost this message through the social platforms one by one. We mark the information diffusion model as $\mathcal{MI}$, and introduce the widely adopted Independent Cascade (IC) model: Let $G^i(V^i, E^i, p^i)$ denote the influence propagation graph with the influence probability $p^i(\vec{e}) \in [0, 1]$ for each direct edge $\vec{e}$. The influence spreads from a seed set S in rounds. Initially, only the seeds are *active* and other nodes are *inactive*. In each round, every node that becomes *active* in the previous round has one chance to activate its out-neighbors by the influence probability. The process terminates when no *inactive* nodes can be influenced to become *active*. Note that the influence probability in IC model, i.e., the retweet probability between two nodes in diffusion scenes is a preset parameter and can be learned as mentioned by the works in [10] and [39] from real-world data in history.

Once people have willingness to participate, they will try to group and we mark the grouping model as $\mathcal{MG}$. Note that all groups will try their best to achieve the goal of satisfying minimal number constraint by strategies (1) and (2). Let $\mathcal{MG}\circ\mathcal{MI}$ denote the integrate diffusion-group model from $\mathcal{MI}$. Here, we give an example in Fig. 3 to show a possible integrate diffusion-group process with grouping on labeled-faith graph and IC model. In this example, nodes are labeled by the location of NY (New York) and HK (Hong Kong) and we need to group them by the same location with at least 3 nodes in a group. We model its edge of group faith by γ^2. Initially, seeds v_1 and v_3 successfully activate v_4 and v_5 respectively, and based on the faith and intention, we get the first groups $\{v_3, v_5\}$, $\{v_1, v_4\}$. In the next round, we have v_6 being activated by v_4 and v_5, and v_7 being activated by v_5. Based on the faith and intention, v_6, v_7 can group together and we have the groups $\{v_6, v_7\}$, $\{v_3, v_5\}$, $\{v_1, v_4\}$. In the next round, we have v_8, v_{10} being activated by v_6 and v_7 respectively, and v_8 can join the group $\{v_1, v_4\}$ by the invitation of v_4. v_{10} can't join any existing group since there will be no any invitation. Specially, to expand the number of members to be at least 3, group $\{v_6, v_7\}$, $\{v_3, v_5\}$ can be merged into one by the faith between v_6 and v_5. We have final groups $\{v_{10}\}$, $\{v_6, v_7, v_3, v_5\}$, $\{v_8, v_1, v_4\}$. Now, there is no any new node that can be activated, and we get the final successful groups $\{v_6, v_7, v_3, v_5\}$, $\{v_8, v_1, v_4\}$ which satisfy the minimum number limitation of 3.

4. Problem formulation

In this section, based on the model above, we further formalize the problem of κ-grouping joining influence maximization.

Definition 3. **κ-Grouping Joining Influence Maximization (κ-GJIM):** Given a social network $G(V, E)$ with a influence diffusion model $\mathcal{MI}$ and a group model $\mathcal{MG}$, let the κ-grouping joining influence $\sigma_\kappa(S)$ denote the expected number of nodes who can group with at least $\kappa - 1$ ($\kappa \geq 2$) like-mind influenced partners after the stochastic diffusion grouping model $\mathcal{MG}\circ\mathcal{MI}$ from the seeds S. Our κ-Grouping Joining Influence Maximization problem is to find a seed set S^*_κ with size k to maximize the κ-grouping joining influence σ_κ. i.e.,

$$S^*_\kappa := \underset{S\subseteq V, |S|=k}{argmax}\ \sigma_\kappa(S).$$

In this paper, we solve this problem with the group model on the labeled-faith graph we propose and use popular IC model as the influence diffusion model. Kempe et al. [18] gave an equivalent process by the live-edge subgraph to get a diffusion realization of IC, where a live-edge graph g is a subgraph of G^i by randomly preserving each edge $\vec{e}$ from G^i with probability of $p^i(\vec{e})$. Let g_S be the set of nodes that seeds in S can reach on g, then we get g_S as a realization of influence spread from seeds set S. We can further have the objective as following:

Theorem 1. *Let $Pr[g]$ denote the probability distribution for the live-edge subgraph g in G^i and $MG^f_v(g_S)$ be the maximal potential group in the vertex-induced labeled-faith subgraph $G^f(g_S)$ which includes v, we have*

$$\sigma_\kappa(S) = \sum_{g\subseteq G} Pr[g] \sum_{v\in V} \chi\{|MG^f_v(g_S)| \geq \kappa\},$$

where $\chi\{exp\}$ is the indicator function where $\chi\{exp\} = 1$ if exp is true, otherwise $\chi\{exp\} = 0$.

Proof. Let $\sigma_\kappa(v, g, S)$ denote the probability that a node v can join in a potential group with size at least κ over g_S based on the labeled-faith graph G^f. Let $C_{g,S}$ denote all possible outputs of groups based on the diffusion realization of g_S, $Pr[c]$ denote the probability of an output C from $C_{g,S}$, and c_v denote the group in C that v belongs to, then we have $\sigma_\kappa(v, g, S) = \sum_{C\in C_{g,S}} Pr(C)\chi\{|c_v| \geq \kappa\}$. By the group merging strategy on the labeled-faith graph, we can ensure when $|c_v| < \kappa$, c_v must be the maximal potential group in $G^f(g_S)$, otherwise if c is not a maximal potential group in $G^f(g_S)$, we have a node $v, v \notin c$ which has a influenced neighbor u in G^f with same label in c, and we suppose the v's group is c_v, but by the group strategy, c and c_v should be merged into one group to expand the number of members to be κ as much as possible and it's contradictory. So we have $c_v = MG^f_v(g_S)$, and then since $c_v \subseteq MG^f_v(g_S)$, we have $\chi\{|c_v| \geq \kappa\} = \chi\{|MG^f_v(g_S)| \geq \kappa\}$, and hence $\sigma_\kappa(v, g, S) = \chi\{|MG^f_v(g_S)| \geq \kappa\}$. We have

$$\sigma_\kappa(S) = \sum_{g\subseteq G} Pr[g] \sum_{v\in V} \sigma_\kappa(v, g, S) = \sum_{g\subseteq G} Pr[g] \sum_{v\in V} \chi\{|MG^f_v(g_S)| \geq \kappa\}.$$

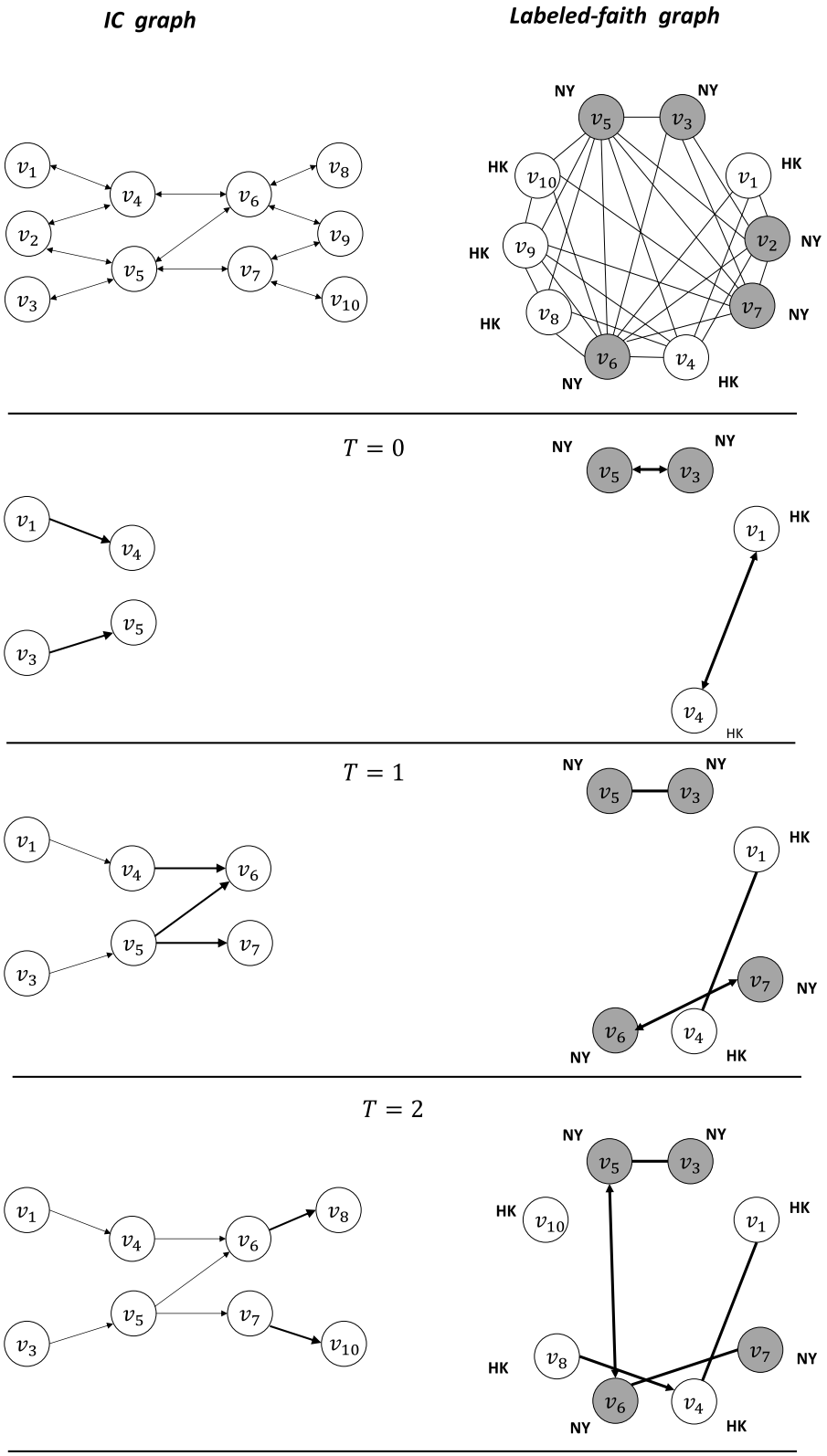


Fig. 3. An instance of the diffusion-group process.

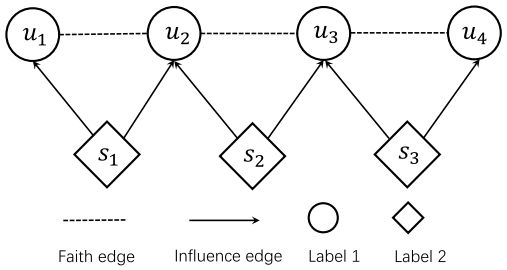


Fig. 4. The example applied to analyzing the properties of modularity.

We proved it. □

If we relax the group size limitation of κ to be 1 (i.e., people can join independently as long as being influenced), it is the traditional IM problem which is NP-hard. However, in our κ-GJIM problem, the κ is at least 2 with grouping constraint, and next we show it is still a hard problem.

Theorem 2. *The κ-GJIM problem is NP-hard.*

Proof. Given an arbitrary instance of the NP-complete Set Cover problem defined by a set collection $\mathcal{S} = \{S_1, S_2, \cdots, S_m\}$ with ground set $U = \{u_1, u_2, \ldots, u_n\}$. We wish to know whether there exists k sets in $\mathcal{S}$ whose union is equal to U. Here we show it's a special case of our κ-GJIM problem by constructing an instance as follows. Let $V = \{v_1^1, \ldots, v_1^\kappa, \ldots, v_n^1, \ldots, v_n^\kappa, s_1, \ldots, s_m\}$, where each v_i^j $(1 \le j \le \kappa, 1 \le i \le n)$ is the j-th copy corresponding to node u_i in U, and each $s_t (1 \le t \le m)$ corresponds to the set S_t in $\mathcal{S}$. Create E^i's edge $< s_t, v_i^j >$ as long as $u_i \in c_t$ and set the influence probability with 1 to make the influence is deterministic. Creat E^f's edge $< v_i^{j_1}, v_i^{j_2} >$ for each couple copies of u_i, and mark each label of v_i^j, s_t respectively with i, $n\kappa + t$. So only same copies can group together. We have if there exists k seeds in G who can influence all nodes $\{v_j^i\}$, i.e., $\sigma_\kappa = \kappa m$, they must be the nodes from $\{s_t\}$ and correspond a solution of sets in $\mathcal{S}$ which can cover U. So we have that any S of k nodes in V has $\sigma_\kappa \ge \kappa m$, iff the Set Cover problem can be solved. So the κ-GJIM problem is NP-hard. □

Theorem 3. *The computation problem of $\sigma_\kappa(.)$ is #P-hard.*

Proof. Given an arbitrary instance of the #P-complete s-t connectedness counting problem defined by a graph $G'(V', E')$ and nodes v_s, v_t. Let $\mathcal{G} = \{g^\eta\}$, $1 \le \eta \le z$, where g^η denotes G's subgraph whose v_s connects to v_t. We wish to count $\mathcal{G}$'s size z. Here we show it's a special case of our computation problem of σ_κ by constructing an instance as follows. Let $V = V' \cup \{v_t^1, \ldots, v_t^\kappa\}$ where v_t^j $(1 \le j \le \kappa)$ is the j-th copy of node t. Create E^f's edge $< v_t^{j_1}, v_t^{j_2} >$ for any two copies of v_t and mark each node in V with different labels and all copies of v_t with same labels. Let $E^i = E' \cup \{< v_t, v_t^j >\}$ with fixed influence probability of p, $(0 < p < 1)$. We have $\sigma_\kappa(\{v_s\}) = \kappa p^\kappa \sum_{g^\eta \in \mathcal{G}} p^{|E'|} = \kappa z p^{|E'|+\kappa}$. So if we have the value of $\sigma_\kappa(\{v_s\})$, then we can get z by $z = \frac{\sigma_\kappa(\{s\})}{\kappa p^{|E'|+\kappa}}$. So the computation problem of $\sigma_\kappa(.)$ is #P-hard. □

The NP-hard problem on set functions can be approximationly solved well under some good properties such as submodularity or supmodularity [24]. Unfortunately, these properties may absent in our problem.

Theorem 4. *The function of κ-grouping joining influence $\sigma_\kappa(.)$ is not neither submodular nor supmodular.*

Proof. We prove it by a counterexample. Let $\Delta_u \sigma_\kappa(S)$ be the gain for the objective after adding any node u into a seeds set S. As shown in the example of Fig. 4, we set the influence to be definite, i.e., each direct edge represents the 100% influence. Then given two seeds set $S_1 = \{s_2\}, S_2 = \{s_1, s_2\}$, $S_1 \subseteq S_2$, when let $\kappa = 2$, we have $\Delta_{s_3}\sigma_2(S_1) = 2$ and $\Delta_{s_3}\sigma_2(S_2) = 1$, i.e., $\Delta_{s_3}\sigma_2(S_2) < \Delta_{s_3}\sigma_2(S_1)$, hence $\sigma_\kappa(.)$ isn't supmodular. But when we let $\kappa = 3$, we have $\Delta_{s_3}\sigma_3(S_1) = 0$ and $\Delta_{s_3}\sigma_3(S_2) = 1$, i.e., $\Delta_{s_3}\sigma_2(S_2) > \Delta_{s_3}\sigma_2(S_1)$, hence $\sigma_\kappa(.)$ isn't submodular. □

Nextly, we will design algorithms to solve this problem. Since computing $\sigma_k(.)$ is a #P-hard problem, in the following section, we consider the estimation for the objective instead of exact computation.

5. Estimation for the objective

Naturally, based on the Monte Carlo simulation of the process in the model $\mathcal{MG} \circ \mathcal{MI}$, we can estimate the objective since it's the expectation of the total number of nodes which group successfully. However, we don't know the exact process of the model $\mathcal{MG}$

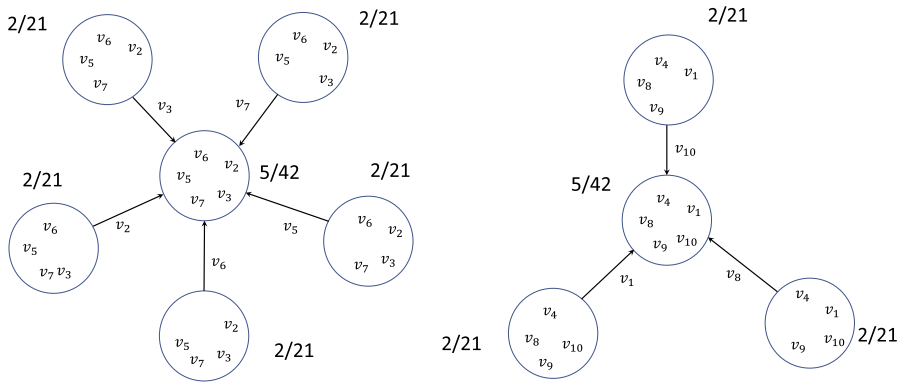


Fig. 5. An instance of graph $\mathcal{G}_4^f$.

even with the group strategies by the labeled-faith graph. Also, like many known problems in social influence based on stochastic model, the simulation is time consuming when the graph is large. Further, it needs to restart the simulations when the set of seeds is changed. Therefore, to estimate the objective fast, we will introduce a method without the need to know the specific process of $\mathcal{MG}$. This method only needs to sample an one-time collection of hypergraphs and then can be used to compute for any seeds set changed without extra sampling.

5.1. κ-joining reverse reachable hypergraph (κ-JRRH)

Let $\mathbf{C}_\kappa[G^f]$ be the family of all potential groups in G^f with size at least κ. Mark $\mathbf{C}_\kappa^v[G^f] := \{c | c \in \mathbf{C}_\kappa[G^f], v \in c\}$, $\mathcal{N}^f[c] := \cup_{v \in c}\{u | < u, v > \in E^f, l(v) = l(u)\}$, $\tau := \sum_{c \in \mathbf{C}_\kappa[G^f]} |c|$ and $Pr[c] := \frac{|c|}{\tau}$. We introduce a following random hypergraph.

Definition 4 *(κ-joining reverse reachable hypergraph (κ-JRRH)).* Choosing a set c from the set family $\mathbf{C}_\kappa[G^f]$ by the probability distribution with $Pr[c]$, sampling a live-edge subgraph g, nextly for each node $u \in c$ and the set $N^f(c)/c$, respectively, getting the set of nodes r_u reversely reachable by u and nodes r reached reversely by $N^f(c)/c$ over g, lastly marking $\omega := (c, g)$ and building the hypergraph $\mathcal{H}_\omega = (r^C, \{r_u | u \in c\})$, we define $\mathcal{H}_\omega$ as the **κ-joining reverse reachable hypergraph (κ-JRRH)**.

Let us review the sample process of κ-JRRH in detail. We need to get the distribution of sets in $\mathbf{C}_\kappa[G^f]$. This problem can be solved through finding all connected subgraphs for a given size cardinality as the works in [17,8,19]. Nextly we introduce how to construct a labeled-weighted directed graph marked as $\mathcal{G}_\kappa^f$ based on the collection of node sets $\mathbf{C}_\kappa[G^f]$ as follows: (1) Each node v_c corresponds to a set c in $\mathbf{C}_\kappa^v[G^f]$ weighted with $w(v_c) := \frac{|c|}{\tau}$; (2) There is a direct edge $< v_{c_1}, v_{c_2} >$ labeled by node u while there is $c_1 \cup \{u\} = c_2$ correspondingly.

Here in Fig. 5, we show a vivid example constructed from the instance of faithed labeled graph in Fig. 3. Then, review the κ-JRRH sampling process and based on such graph we can easily sample the set c from $\mathbf{C}_\kappa[G^f]$ as sampling a node in $\mathcal{G}_\kappa^f$ following the node weight $w(v_c)$ and get the set $N^f(c)/c$ which is the set of v_c's out-going edge labels. Further, besides sampling a set above, we need to firstly sample a subgraph from G^i by removing each influence edge by the edge probability and then doing multiple reverse breadth-first searches (bfs) sourced from different nodes over the induced subgraph. There is no need to remove all edges in advance since some edges may not be searched. Therefore, we use a flying random reverse search instead, i.e., removing edges randomly while searching. This flying operation can avoid removing the unnecessary extra edge and hence reduce the cost in sampling. We also adopt a more efficient multiple-sourced breadth-first search [34] instead of multi independent reverse breadth-first searches sourced from different nodes. So based on the above optimization strategy, we propose the optimized sampling algorithm as shown in Algorithm 1.

Here in Fig. 6, we show a vivid 4-JRRH sampling process based on the instance in Fig. 3. Firstly, choose a node from the graph in Fig. 5 by the probability of the node weight and they are $\{v_1, v_4, v_8, v_9\}$ and we have $l_{out_c} = \{v_{10}\}$. Nextly we do a multiple-sourced breadth-first search reversely and randomly over G^i from these source nodes and get the initial $Visit$ as $\{(v_1, \{b_1\}), (v_4, \{b_4\}), (v_8, \{b_8\}), (v_9, \{b_9\}), (v_{10}, \{b_{10}\})\}$ corresponding to 5 subprocesses of bfs. For each source node u, we get the $seen_u = \{u\}$ which represents the set of nodes having been visited by the source node u. In the first level of $Visit$, the edge $< v_8, v_6 >$ is flipped to be "off", and the edges $< v_4, v_1 >$, $< v_4, v_2 >$, $< v_9, v_6 >$, $< v_9, v_7 >$, $< v_{10}, v_7 >$ are flipped to be "on". We update $seen_{v_1} = \{v_1, v_4\}$, $seen_{v_6} = \{v_9\}$, $seen_{v_7} = \{v_9, v_{10}\}$ and get the next level of $Visit = \{(v_1, \{b_4\}), (v_2, \{b_4\}), (v_6, \{b_9\}), (v_7, \{b_9\}), (v_7, \{b_{10}\})\}$. In the second level of $Visit$, the edge $< v_6, v_5 >$ is flipped to be "off", and the edges $< v_6, v_4 >$, $< v_7, v_5 >$ are flipped to be "on". We update $seen_{v_4} = \{v_9, v_4\}$, $seen_{v_5} = \{v_9, v_{10}\}$ and get the next level of $Visit = \{(v_4, \{b_9\}), (v_5, \{b_9, b_{10}\})\}$. In the third level, the edge $< v_5, v_2 >$ is flipped to be "off" and the edge $< v_5, v_3 >$ is flipped to be "on". We update $seen_{v_1} = \{v_9, v_1, v_4\}$, $seen_{v_3} = \{v_9, v_{10}\}$ and get the next level of $Visit = \{(v_1, \{b_9\}), (v_2, \{b_9\}), (v_5, \{b_9, b_{10}\})\}$. Since there is no any out-going edge for the nodes v_1, v_2, v_5 in the next level of $Visit$, the search process terminates and we get the corresponding bfs results for each source node and hence the hypergraph as shown in the right part of Fig. 6.

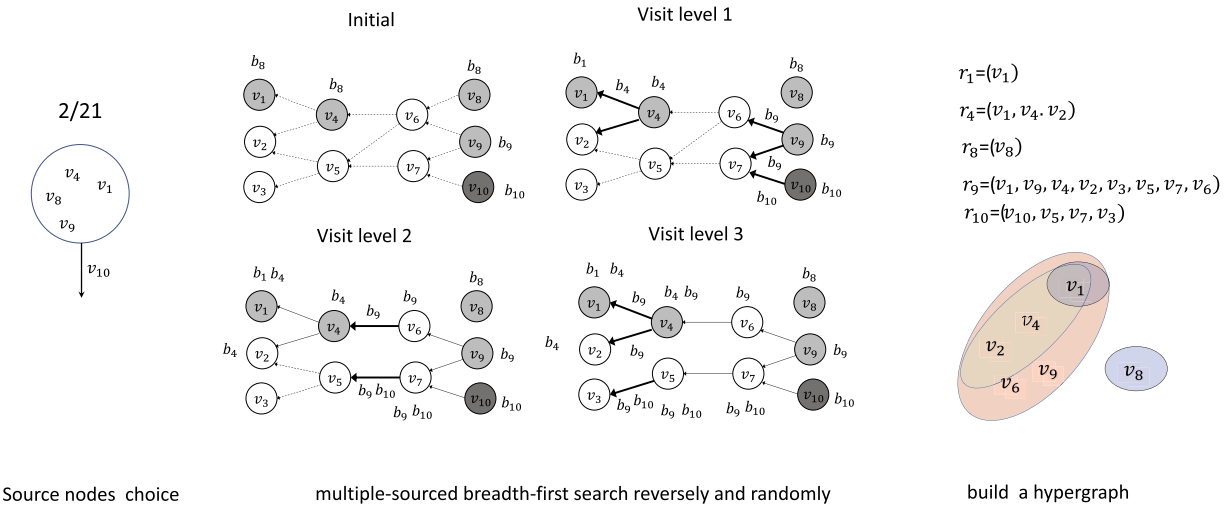


Fig. 6. An instance of Sampling 4-JRRH.

Algorithm 1: Sample κ-JRRH.

Select a node v_c in the graph G_κ^f with the probability of node weight;
Get the node set c that v_c corresponds to;
Get v_c's all out-going edges' labels l_{out_c};
$Seen_u \leftarrow \{u\}$, $r_u \leftarrow \{u\}$ for all $u_i \in c \cup l_{out_c}$;
$Visit = \bigcup_{u \in c \cup l_{out_c}} \{(u, \{b_u\})\}$;
while $Visit \neq \emptyset$ // Each $Visit$ level **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^i **do**
      if *edge* e_{uv} *has not been flipped* **then**
        Flip e_{uv} with probability p_{uv};
        if *successful* **then**
          Mark e_{uv} to be "on";
        else
          Mark e_{uv} to be "off";
      if e_{uv} *has been "on"* **then**
        $B_u \leftarrow B_v^* / \{b_o | o \in Seen_u\}$;
        if $B_u \neq \emptyset$ **then**
          $VisitNext \leftarrow VisitNext \cup (u, B_u)$;
          $Seen_u \leftarrow Seen_u \cup \{o | b_o \in B_u\}$;
          for *each* $o \in \{o | b_o \in B_u\}$ **do**
            $r_o \leftarrow r_o \cup \{u\}$;
  $Visit \leftarrow VisitNext$;
Build a hypergraph $\mathcal{H}(V - \cup_{v \in l_{out_c}} r_v, \{r_u, u \in c\})$;
return $\mathcal{H}$

5.2. Estimation based on κ-JRRH

Mark the sample space $\mathbf{\Omega}^* := \{(c, g) | c \in \mathbf{C}_\kappa[G^f], g \in G\}$, and we can compute the union probability $P[\omega] = Pr[c]Pr[g] = \frac{Pr[g]|c|}{\tau}$, where $\omega \in \mathbf{\Omega}^*$ is the tuple generated in the random process of κ-JRRH. Then we have another sample space $\mathbf{\Omega} := \{\mathcal{H}_\omega | \omega \in \mathbf{\Omega}^*\}$, and hence get a bernoulli variable ξ_S: $\mathbf{\Omega} \to \{0, 1\}$ where $\xi_S(\mathcal{H}) = \chi\{S \subseteq V_\mathcal{H}\} \cdot \chi\{deg_\mathcal{H}(S) = |E_\mathcal{H}|\}$, $deg_\mathcal{H}(S) := |D_\mathcal{H}(S)|$ is the number of all hyperedges $D_\mathcal{H}(S) \subseteq E_\mathcal{H}$ that intersect S. Nextly as shown in the Theorem 5, we can get an equivalent computation for our objective by computing the expectation of the bernoulli variable ξ_S. To prove this theorem, we need firstly prove following three lemmas:

Lemma 1. *We have* $E(\xi_S) = \frac{1}{\tau} \sum_{c \in \mathbf{C}_\kappa[G^f], g \in G} P[g]|c|(\prod_{u \in c} \chi\{S \cap \bar{g}_{\{u\}} \neq \emptyset\}) \cdot \chi\{S \cap \bar{g}_{Nf(c)/c} = \emptyset\}$, *where* $\bar{g}$ *denotes the reverse of a direct graph* g.

Proof. By the definition of the ξ_S, we have

$$E(\xi_S) = \sum_{\mathcal{H}\in\mathbf{\Omega}} Pr(\mathcal{H})\xi_S(\mathcal{H}) = \sum_{\omega\in\Omega^*} Pr[\omega]\xi_S(\mathcal{H}_\omega) = \sum_{\omega\in\Omega^*} \frac{Pr[g]|c|}{\tau}\xi_S(\mathcal{H}_\omega)$$
$$= \sum_{\omega\in\Omega^*} \frac{Pr[g]|c|}{\tau}\chi\{S\subseteq V_{\mathcal{H}_\omega}\}\cdot\chi\{deg_{\mathcal{H}_\omega}(S) = |E_{\mathcal{H}_\omega}|\}$$

Nextly we prove

$$\chi\{S\subseteq V_{\mathcal{H}_\omega}\}\cdot\chi\{deg_{\mathcal{H}_\omega}(S) = |E_{\mathcal{H}_\omega}|\} = (\prod_{u\in c}\chi\{S\cap\bar{g}_{\{u\}}\neq\emptyset\})\cdot\chi\{S\cap\bar{g}_{N^f(c)/c} = \emptyset\}.$$

By the construction of the hypergraph, we have $r = \bar{g}_{N^f(c)/c}$, and $V_{\mathcal{H}_\omega} = r^C$. It's obviously that $\chi\{S\subseteq r^C\} = \chi\{S\cap r = \emptyset\}$. So when $\chi\{S\cap r = \emptyset\} = 0$, we also have $\chi\{S\subseteq r^C\} = 0$ and the left must equal to the right with value of 0. When $\chi\{S\cap r = \emptyset\} = 1$, we also have $\chi\{S\subseteq r^C\} = 1$. Since $E_{\mathcal{H}_\omega} = \{r_u|u\in c\}$ and $r_u = \bar{g}_{\{u\}}$, we can easily have $\prod_{u\in c}\chi\{S\cap\bar{g}_{\{u\}}\neq\emptyset\} = \chi\{\sum_{u\in c}\chi\{S\cap r_u\neq\emptyset\} = |c|\}$. Then we have $\prod_{u\in c}\chi\{S\cap\bar{g}_{\{u\}}\neq\emptyset\} = \chi\{\sum_{u\in c}\chi\{S\cap(r_u)\neq\emptyset\} = |c|\} = \chi\{deg_{\mathcal{H}_\omega}(S) = |E_{\mathcal{H}_\omega}|\}$. So we have the left always equals to the right. □

Lemma 2. *We have $|MG_v^f(g_S)|\geq\kappa$ iff exists a potential group c ($v\in c$) in G^f with size at least κ satisfying two following conditions:*

(1): *Each node in c can reach reversely by at least one seed node over g, and the equivalent logical equation is $\prod_{u\in c}\chi\{S\cap\bar{g}_{\{u\}}\neq\emptyset\} = 1$.*
(2): *For each node u in c, all neighbors of u in G^f with the same label as u not belonging to c can't reach reversely by any seed node over g. The equivalent logical equation is $\chi\{S\cap\bar{g}_{N^f(c)/c} = \emptyset\} = 1$.*

Further, we have the lemma as the following logical equation:

$$\chi\{|MG_v^f(g_S)|\geq\kappa\} = \bigvee_{c\in\mathbf{C}_\kappa^v[G^f]}(\prod_{u\in c}\chi\{S\cap\bar{g}_{\{u\}}\neq\emptyset\})\cdot\chi\{S\cap\bar{g}_{N^f(c)/c} = \emptyset\}.$$

Proof. ⇒: If $|MG_v^f(g_S)|\geq\kappa$, we have a potential group $c' := MG_v^f(g_S)$ in G^f which naturally satisfies condition (1). For condition (2), if it doesn't satisfy it, i.e., there is a node in c' whose neighbor u in G^f with same label as u not belonging to c' can reach a seed node reversely over g, then the group $c'\cup\{u\}$ is also a potential group in $G^f(g_S)$ and it's contradictory since c' is a maximal potential group.

⇐: If there exists a potential group c' ($v\in c'$) in G^f with size at least κ satisfying both the conditions, then since $S\cap\bar{g}_u\neq\emptyset$ for each node $u\in c'$ and we have $c'\subseteq g_S$, i.e., c' is also a potential group in $G^f(g_S)$ and then $|MG_v^f(g_S)|\geq\kappa$. □

Lemma 3. *There is at most one potential group c ($v\in c$) in G^f with size at least κ satisfying both the conditions in Lemma 2, i.e., we have the lemma as the following logical equation:*

$$\bigvee_{c\in\mathbf{C}_\kappa^v[G^f]}(\prod_{u\in c}\chi\{S\cap\bar{g}_{\{u\}}\neq\emptyset\})\cdot\chi\{S\cap\bar{g}_{N^f(c)/c} = \emptyset\} = \sum_{c\in\mathbf{C}_\kappa^v[G^f]}(\prod_{u\in c}\chi\{S\cap\bar{g}_{\{u\}}\neq\emptyset\})\cdot\chi\{S\cap\bar{g}_{N^f(c)/c} = \emptyset\}.$$

Proof. Given any two potential groups c_1, c_2 in $\mathbf{C}_\kappa^v[G^f]$ satisfying both the conditions in Lemma 2, since c_1 and c_2 both are connected to the node v in G^f with same label as v, then $c_1\cup c_2$ also forms a potential group in G^f. If $c_1\neq c_2$, we have $c_1 - c_2\neq\emptyset$ or $c_2 - c_1\neq\emptyset$. Let's suppose $c_1 - c_2\neq\emptyset$, then $c_1 - c_2$ is also connected with c_2 in G^f, i.e., there must be a node $u_1(u_1\in c_1 - c_2)$ who has a neighbor u_2 ($u_2\in c_2$), hence $u_1\in N^f(\{u_2\})/c_2$ and $u_1\in c_1$. Since $S\cap\bar{g}_{N^f(c2)/c2} = \emptyset$ by condition (2) for c_2, $S\cap\bar{g}_{\{u_1\}} = \emptyset$ which is contradictorily by condition (1) for c_1. So c_1 must be equal to c_2. □

Theorem 5. $\sigma_\kappa(S) = \tau E(\xi_S)$.

Proof. By the Lemma 1, 2, 3, we have

$$\sigma_\kappa(S) = \sum_{g\in G} Pr[g]\sum_{v\in V}\chi\{|MG_v^f(g_S)|\geq\kappa\}$$
$$= \sum_{g\in G} Pr[g]\sum_{v\in V}\sum_{c\in\mathbf{C}_\kappa^v[G^f]}(\prod_{u\in c}\chi\{S\cap\bar{g}_{\{u\}}\neq\emptyset\})\cdot\chi\{S\cap\bar{g}_{N^f(c)/c} = \emptyset\}$$
$$\sum_{g\in G} Pr[g]\sum_{v\in V}\sum_{c\in\mathbf{C}_\kappa[G^f]}\chi\{v\in c\}(\prod_{u\in c}\chi\{S\cap\bar{g}_{\{u\}}\neq\emptyset\})\cdot\chi\{S\cap\bar{g}_{N^f(c)/c} = \emptyset\}$$
$$\sum_{g\in G} Pr[g]\sum_{c\in\mathbf{C}_\kappa[G^f]}((\prod_{u\in c}\chi\{S\cap\bar{g}_{\{u\}}\neq\emptyset\})\cdot\chi\{S\cap\bar{g}_{N^f(c)/c} = \emptyset\})\sum_{v\in V^l}\chi\{v\in c\}$$
$$\sum_{g\in G}\sum_{c\in\mathbf{C}_\kappa[G^f]} Pr[g]|c|((\prod_{u\in c}\chi\{S\cap\bar{g}_{\{u\}}\neq\emptyset\})\cdot\chi\{S\cap\bar{g}_{N^f(c)/c} = \emptyset\})$$

$$=\tau E(\xi_S)$$

We proved it. □

Let $\mathbb{H}=\{\mathcal{H}_1,\mathcal{H}_2,\cdots,\mathcal{H}_\lambda\}$ be the set of κ-JRRH sampled λ times independently. Mark $F_{\mathbb{H}}(S):=\sum_i^\lambda \xi_S(\mathcal{H}_i)$, then $\frac{F_{\mathbb{H}}(S)}{|\mathbb{H}|}$ is an unbiased estimation of the expectation of ξ_S. Hence $\bar{\sigma}_\kappa(S):=\tau\frac{F_{\mathbb{H}}(S)}{|\mathbb{H}|}$ is an estimation for $\sigma_\kappa(S)$. Nextly we will analyze the error gap between the incomputable σ_κ and computable $\bar{\sigma}_\kappa$ in the following theorem. To prove it, we first introduce the Chernoff Bounds [23] as follows.

Lemma 4. *Let $X=\sum_{i=1}^\lambda X_i$ and $\mu=E(X)$ where $X_1,X_2,\cdots,X_\lambda$ are independent bernoulli random variables. For any $\epsilon\in(0,1)$, we have*

$$Pr\{X-\mu\geq\epsilon\mu\}\leq e^{-\frac{\epsilon^2\mu}{2+\epsilon}}, \tag{1}$$

$$Pr\{X-\mu\leq-\epsilon\mu\}\leq e^{-\frac{\epsilon^2\mu}{2}}. \tag{2}$$

Theorem 6. *If $\lambda\geq\frac{-\tau}{\epsilon^2\kappa}\cdot min\{\ln x^{2+\epsilon},\ln(1-\delta-x)^2\}$, we have $Pr\{|\bar{\sigma}_\kappa-\sigma_\kappa|\leq\epsilon\sigma_\kappa\}\geq\delta$, where $0\ll\delta<1$, $0<x<1-\delta$ and $0<\epsilon\ll1$.*

Proof. Let $X_i=\xi_S(\mathcal{H}_i)$. We have $X=\sum_{i=1}^\lambda X_i=F_{\mathbb{H}}(S)=\frac{\lambda\bar{\sigma}_\kappa(S)}{\tau}$, and $\mu=E(X)=\lambda\cdot E(\xi_S)=\frac{\lambda\sigma_\kappa(S)}{\tau}$. By Equation (1) in Lemma 4, since $\lambda\geq\frac{-\tau(2+\epsilon)\ln x}{\epsilon^2\kappa}$, we can ensure the lower bound of objective is κ since it is the same with all $\sigma_\kappa(\cdot)<\kappa$. By default, we add extra κ full connected nodes into G^f with the same label, and all of them can be influenced by any node in V definitely. So we replace the objective by $\sigma_\kappa(\cdot):=\sigma_\kappa(\cdot)+\kappa$ to ensure the lower bound of objective to be κ, such operation won't change the order of objective's value for all sets. We ensure that $\sigma_k\geq\kappa$, and then have

$$Pr\{\frac{\lambda\bar{\sigma}_\kappa(S)}{\tau}-\frac{\lambda\sigma_\kappa(S)}{\tau}\geq\epsilon\cdot\frac{\lambda\sigma_\kappa(S)}{\tau}\}=Pr\{\bar{\sigma}_\kappa-\sigma_\kappa\geq\epsilon\sigma_\kappa\}\leq e^{-\frac{\epsilon^2\lambda\sigma_\kappa}{\tau(2+\epsilon)}}\leq e^{-\frac{\epsilon^2\lambda\kappa}{\tau(2+\epsilon)}}\leq x.$$

By Equation (2) in Lemma 4, since $\lambda\geq\frac{-\ln(1-\delta-x)\tau}{\epsilon^2\kappa}$, we have

$$Pr\{\frac{\lambda\bar{\sigma}_\kappa(S)}{\tau}-\frac{\lambda\sigma_\kappa(S)}{\tau}\leq-\epsilon\cdot\frac{\lambda\sigma_\kappa(S)}{\tau}\}=Pr\{\bar{\sigma}_\kappa-\sigma_\kappa\leq-\epsilon\sigma_\kappa\}\leq e^{-\frac{\epsilon^2\lambda\sigma_\kappa}{2\tau)}}\leq e^{-\frac{\epsilon^2\lambda\kappa}{2\tau}}\leq1-\delta-x.$$

Therefore, $Pr\{|\bar{\sigma}_\kappa-\sigma_\kappa|\leq\epsilon\sigma_\kappa\}=1-Pr\{|\bar{\sigma}_\kappa-\sigma_\kappa|>\epsilon\sigma_\kappa\}=1-(Pr\{\bar{\sigma}_\kappa-\sigma_\kappa>\epsilon\sigma_\kappa\}+Pr\{\bar{\sigma}_\kappa-\sigma_\kappa<-\epsilon\sigma_\kappa\})\geq1-(x+1-\delta-x)=\delta$. □

By the Theorem 6, if we sample enough κ-JRRH samples, then we can get an accurate estimation for the objective with high confidence. Further, to reduce the cost of sampling as possible, we discuss the lower bound $\lambda_l(\epsilon,\delta)$ of the samples number λ which can ensure the ratio of error ϵ and the confidence δ in Theorem 6, i.e., solving the problem

$$x_*:=argmax\{f_1(x)|x\in(0,1-\delta)\},f_1(x):=min\{\ln x^{2+\epsilon},\ln(1-\delta-x)^2\}.$$

Since logarithmic functions are monotonically increasing, we have $x_*=argmax\{f_2(x)|x\in(0,1-\delta)\}$, where $f_2(x):=min\{x^{2+\epsilon},(1-\delta-x)^2\}$. We draw $f_2(x)$ into coordinate system as shown in Fig. 7 and ${x_*}^{2+\epsilon}=(1-\delta-x_*)^2$ which is hard to be solved analytically, but we can solve a suboptimal $x_\star$ where ${x_\star}^2=(1-\delta-x_\star)^2$ and we have $x_\star=\frac{1-\delta}{2}$. Hence we can get an analytical suboptimal lower bound $\lambda_l^\star(\epsilon,\delta)=\frac{-\tau}{\epsilon^2\kappa}f_1(x_\star)=\frac{2\tau(\ln2-\ln(1-\delta))}{\epsilon^2\kappa}$. Now, we have solved the problem of estimation for the objective. Nextly we design algorithms to solve the optimal problem of selecting seeds.

6. Algorithms

Instead of solving the problem directly, by the estimation method above, we have $\sigma_\kappa\approx\bar{\sigma}_\kappa$ when the total count λ of samples $\mathbb{H}$ is large enough. Therefore we consider solving the maximization problem of $\bar{\sigma}_\kappa$ and further the maximization problem of $F_{\mathbb{H}}(\cdot)$, i.e.,

$$S^\star:=argmax_{S\in V,|S|=k}F_{\mathbb{H}}(S). \tag{3}$$

In Theorem 7, we give the gap between a solution S^+ provided by solving problem (3) and the optimal S^* for the original problem.

Theorem 7. *When $\lambda\geq\lambda_l(\frac{\epsilon}{2},\delta)$, we have $\sigma_\kappa(S^+)\geq(1-\epsilon)\frac{F_{\mathbb{H}}(S^+)}{F_{\mathbb{H}}(S^\star)}\sigma_\kappa(S^*)$ with a probability at least $2\delta-1$.*

Proof. Since $S^\star$ is an optimal solution for problem (3), we have $\bar{\sigma}_\kappa(S^\star)\geq\bar{\sigma}_\kappa(S^*)$ and hence $\frac{F_{\mathbb{H}}(S^+)}{F_{\mathbb{H}}(S^\star)}\bar{\sigma}_\kappa(S^\star)=\bar{\sigma}_\kappa(S^+)\geq\frac{F_{\mathbb{H}}(S^+)}{F_{\mathbb{H}}(S^\star)}\bar{\sigma}_\kappa(S^*)$. By Theorem 6, if $\lambda\geq\lambda_l(\frac{\epsilon}{2-\epsilon},\delta)$, we have $\bar{\sigma}_\kappa(S^*)\geq(1-\frac{\epsilon}{2-\epsilon})\sigma_\kappa(S^*)=\frac{2(1-\epsilon)}{2-\epsilon}\sigma_\kappa(S^*)$ and $\sigma(S^+)\geq\frac{1}{1+\frac{\epsilon}{2-\epsilon}}\bar{\sigma}_\kappa(S^+)=\frac{2-\epsilon}{2}\bar{\sigma}_\kappa(S^+)$ with the union probability is at least $2\delta-1$. We proved it. □

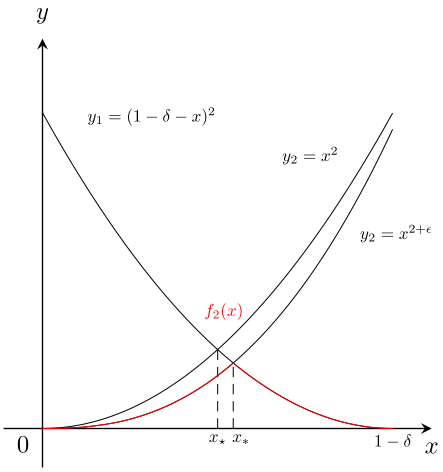


Fig. 7. The $f_2(x)$ in coordinate system.

By Theorem 7, if we sample enough κ-JRRTH samples, by solving the maximization problem of $F_{\mathbb{H}}(S)$, we can get a high quality solution with high confidence for the original problem. So nextly we consider how to solve the maximization problem of $F_{\mathbb{H}}(S)$. Unfortunately, we can prove this problem is still NP-hard.

Theorem 8. *The maximization problem for $F_{\mathbb{H}}(\cdot)$ in Equation (3) is NP-hard.*

Proof. We can easily prove it by a special case with $\mathbb{H}^*$ in which for each hypergraph $\mathcal{H}$ the hyperedges are repeated to be $e_{\mathcal{H}}$ and $V_{\mathcal{H}} = V$. We have $F_{\mathbb{H}^*}(S) = \sum_{\mathcal{H}\in\mathbb{H}^*} \chi\{S \cap e_{\mathcal{H}} \neq \emptyset\}$ and it equals to the problem of to find a set of S with k nodes that maximizes the number of sets hit by S in $\{e_{\mathcal{H}} | \mathcal{H} \in \mathbb{H}^*\}$, which is a variant of NP-hard set cover problem. □

The importance of submodularity in machine learning and data mining applications has been demonstrated in many literatures, but we still can't guarantee that $F_{\mathbb{H}}(S)$ is submodular since we can give a simple counterexample with $\mathbb{H} = \{\mathcal{H}_1, \mathcal{H}_2\}$ where $\mathcal{H}_1 = \{\{1\}, \{2,3\},\}$, $\mathcal{H}_2 = \{\{1\}, \{3\}\}$. For non-submodularity maximization problem, denoting it by a difference of two submodular functions is a widely used strategy. So nextly, we firstly discuss how to differentiate the objective into a difference of two submodularities, and secondly we propose a heuristic algorithm based on the greedy hill-climbing strategy.

6.1. Difference of two submodularities

In this subsection, we will convert the objective function to a difference of two submodular functions. We firstly construct a following set function based on a given hypergraph $\mathcal{H}$.

$$g_{\mathcal{H}}(S) = \begin{cases} \deg_{\mathcal{H}}(S) & \text{if } \deg_{\mathcal{H}}(S) < |E_{\mathcal{H}}|, \\ \deg_{\mathcal{H}}(S) - 1 & \text{if } \deg_{\mathcal{H}}(S) = |E_{\mathcal{H}}|. \end{cases}$$

Nextly we prove that the set functions of $g_{\mathcal{H}}(\cdot)$ and $\deg_{\mathcal{H}}(S)(\cdot)$ have following properties:

Lemma 5. *$g_{\mathcal{H}}(\cdot)$ and $deg_{\mathcal{H}}(\cdot)$ are non-negative, non-decreasing and submodular.*

Proof. It's obviously that $g_{\mathcal{H}}(\cdot)$ and $deg_{\mathcal{H}}(\cdot)$ are non-negative and non-decreasing. Nextly we prove the submodularity. Let $\Delta_v \deg_{\mathcal{H}}(S)$ and $\Delta_v g_{\mathcal{H}}(S)$ correspond to the marginal gains of adding a node v into S for $\deg_{\mathcal{H}}(\cdot)$ and $g_{\mathcal{H}}(\cdot)$ respectively. Considering any sets $S_1 \subseteq S_2 \in 2^V$ and node $v \in V/S_1$, since $D_{\mathcal{H}}(S_1) \subseteq D_{\mathcal{H}}(S_2)$, we have $\Delta_v \deg_{\mathcal{H}}(S_1) = |D_{\mathcal{H}}(\{u\})/D_{\mathcal{H}}(S_1)| \geq \Delta_v \deg_{\mathcal{H}}(S_2) = |D_{\mathcal{H}}(\{u\})/D_{\mathcal{H}}(S_2)|$. So $\deg_{\mathcal{H}}(\cdot)$ is submodular. We discuss $\Delta_v g_{\mathcal{H}}(S_1)$ and $\Delta_v g_{\mathcal{H}}(S_2)$ as follows:

- When $\deg_{\mathcal{H}}(S_1) < |E_{\mathcal{H}}|$ and $\deg_{\mathcal{H}}(S_1 \cup \{v\}) < |E_{\mathcal{H}}|$, we have $\Delta_v g_{\mathcal{H}}(S_1) = \Delta_v \deg_{\mathcal{H}}(S_1)$, and $\Delta_v g_{\mathcal{H}}(S_2) \in \{\Delta_v \deg_{\mathcal{H}}(S_2), \Delta_v \deg_{\mathcal{H}}(S_2) - 1, 0\}$. So $\Delta_v g_{\mathcal{H}}(S_2) \leq \Delta_v \deg_{\mathcal{H}}(S_2)$ and hence $\Delta_v g_{\mathcal{H}}(S_2) \leq \Delta_v g_{\mathcal{H}}(S_1)$.
- When $\deg_{\mathcal{H}}(S_1) < |E_{\mathcal{H}}|$ and $\deg_{\mathcal{H}}(S_1 \cup \{v\}) = |E_{\mathcal{H}}|$, we have $\Delta_v g_{\mathcal{H}}(S_1) = \Delta_v \deg_{\mathcal{H}}(S_1) - 1 \geq 0$, and $\Delta_v g_{\mathcal{H}}(S_2) \in \{\Delta_v \deg_{\mathcal{H}}(S_2) - 1, 0\}$. Since $\Delta_v \deg_{\mathcal{H}}(S_2) - 1 \leq \Delta_v \deg_{\mathcal{H}}(S_1) - 1$, hence $\Delta_v g_{\mathcal{H}}(S_2) \leq \Delta_v g_{\mathcal{H}}(S_1)$.
- When $\deg_{\mathcal{H}}(S_1) = |E_{\mathcal{H}}|$, we have $\Delta_v g_{\mathcal{H}}(S_2) = \Delta_v g_{\mathcal{H}}(S_1) = 0$.

Therefore, we always have $\Delta_v g_{\mathcal{H}}(S_2) \leq \Delta_v g_{\mathcal{H}}(S_1)$. $g_{\mathcal{H}}(S)(\cdot)$ is submodular too. □

Let $\bar{\mathcal{H}}$ be the new hypergraph by adding an extra hyperedge $V_{\mathcal{H}}^C$ into $E_{\mathcal{H}}$. Then according to Lemma 5, we can construct following two non-negative, non-decreasing and submodular set functions:

$$\hat{F}_{\mathbb{H}}(S) := \sum_{\mathcal{H}\in\mathbb{H}} (deg_{\mathcal{H}}(S) + g_{\bar{\mathcal{H}}}(S)),\ \bar{F}_{\mathbb{H}}(S) := \sum_{\mathcal{H}\in\mathbb{H}} (g_{\mathcal{H}}(S) + \deg_{\bar{\mathcal{H}}}(S)).$$

Until now, we finally find a difference of submodularities for $F_{\mathbb{H}}$. To prove this theorem, mark the indicator set function $I_{\mathcal{H}}(S) := \chi\{\deg_{\mathcal{H}}(S) = |E_{\mathcal{H}}(S)|\}$ and we introduce Lemma 6, 7.

Lemma 6. $I_{\mathcal{H}}(\cdot) = \deg_{\mathcal{H}}(\cdot) - g_{\mathcal{H}}(\cdot)$.

Proof. We discuss the left and right values in the equation as follows: (1) when $I_{\mathcal{H}}(S) = 1$, we have that $g_{\mathcal{H}}(S) = \deg_{\mathcal{H}}(S) - 1$ and $\deg_{\mathcal{H}}(S) - g_{\mathcal{H}}(S) = 1$; (2) when $I_{\mathcal{H}}(S) = 0$, i.e., $\deg_{\mathcal{H}}(S) < E_{\mathcal{H}}(S)$, we have that $g_{\mathcal{H}}(S) = \deg_{\mathcal{H}}(S)$, $\deg_{\mathcal{H}}(S) - g_{\mathcal{H}}(S) = 0$. □

Lemma 7. $\xi_S(\mathcal{H}) = I_{\mathcal{H}}(S) - I_{\bar{\mathcal{H}}}(S)$.

Proof. We discuss the values of left and right in the equation as follows:

- When $\xi_S(\mathcal{H}) = 1$, we have $\deg_{\mathcal{H}}(S) = |E_{\mathcal{H}}(S)|$ and $S \subseteq V_{\mathcal{H}}$. Hence $S \cap V_{\mathcal{H}}^C = \emptyset$ and $\deg_{\bar{\mathcal{H}}}(S) = |E_{\bar{\mathcal{H}}}| - 1$. So $I_{\mathcal{H}}(S) = 1$ and $I_{\bar{\mathcal{H}}}(S) = 0$. i.e., $\xi_S(\mathcal{H}) = I_{\mathcal{H}}(S) - I_{\bar{\mathcal{H}}}(S) = 1$.
- When $\xi_S(\mathcal{H}) = 0$, we have $\deg_{\mathcal{H}}(S) < |E_{\mathcal{H}}(S)|$ or $S \nsubseteq V_{\mathcal{H}}$. If $\deg_{\mathcal{H}}(S) < |E_{\mathcal{H}}(S)|$, we have that $\deg_{\bar{\mathcal{H}}}(S) \le |\deg_{\mathcal{H}}(S)| + 1 < |E_{\bar{\mathcal{H}}}|$ and $I_{\mathcal{H}}(S) = I_{\bar{\mathcal{H}}}(S) = 0$. i.e., $\xi_S(\mathcal{H}) = I_{\mathcal{H}}(S) - I_{\bar{\mathcal{H}}}(S) = 0$. Else if $\deg_{\mathcal{H}}(S) = |E_{\mathcal{H}}(S)|$, we have that $S \nsubseteq V_{\mathcal{H}}$, hence $S \cap V_{\mathcal{H}}^C \neq \emptyset$ and $\deg_{\bar{\mathcal{H}}}(S) = |E_{\bar{\mathcal{H}}}|$. Then we have that $I_{\mathcal{H}}(S) = I_{\bar{\mathcal{H}}}(S) = 1$ which means $\xi_S(\mathcal{H}) = I_{\mathcal{H}}(S) - I_{\bar{\mathcal{H}}}(S) = 0$.

Then we have that the value of left always equals to the left. □

Theorem 9. *$F_{\mathbb{H}} = \hat{F}_{\mathbb{H}} - \bar{F}_{\mathbb{H}}$, where $\hat{F}_{\mathbb{H}}$ and $\bar{F}_{\mathbb{H}}$ are non-negative, non-decreasing and submodular.*

Proof. By Lemma 6, 7, we have

$$\begin{aligned}
F_{\mathbb{H}}(S) &= \sum_{\mathcal{H}\in\mathbb{H}} \xi_{\mathcal{H}}(S) \\
&= \sum_{\mathcal{H}\in\mathbb{H}} I_{\mathcal{H}}(S) - I_{\bar{\mathcal{H}}}(S) \\
&= \sum_{\mathcal{H}\in\mathbb{H}} \deg_{\mathcal{H}}(S) - g_{\mathcal{H}}(S) - (\deg_{\bar{\mathcal{H}}}(S) - g_{\bar{\mathcal{H}}}(S)) \\
&= \sum_{\mathcal{H}\in\mathbb{H}} \deg_{\mathcal{H}}(S) + g_{\bar{\mathcal{H}}}(S) - (\deg_{\bar{\mathcal{H}}}(S) + g_{\mathcal{H}}(S)) \\
&= \sum_{\mathcal{H}\in\mathbb{H}} \deg_{\mathcal{H}}(S) + g_{\bar{\mathcal{H}}}(S) - \sum_{\mathcal{H}\in\mathbb{H}} \deg_{\bar{\mathcal{H}}}(S) + g_{\mathcal{H}}(S) \\
&= \hat{F}_{\mathbb{H}}(S) - \bar{F}_{\mathbb{H}}(S)
\end{aligned}$$

We proved it. □

Based on Theorem 9, we have that the maximization problem of $F_{\mathbb{H}}$ equals to minimizing $\bar{F}_{\mathbb{H}}(S) - \hat{F}_{\mathbb{H}}(S)$. To solve such problem, there are several alternative approaches such as the convex-concave procedure [37] on the Lovász extensions of f and g. Besides, Iyer et al. proposed three algorithms [16] for approximate minimization of the difference between two submodular functions, i.e., the following optimization problem: $min_{T\subseteq N}[f(T) - g(T)]$, given two submodular set functions f and g, and the ground set N. Here we adopt their Modular-Modular procedure (MMP) but change it lightly to adapt the constraint of knapsack $|T| = k$ as shown in Algorithm 2. The idea of Modular-Modular procedure algorithm is to iteratively solve a minimization problem of the difference

Algorithm 2: k-knapsack-Modular-Modular (k-ModMod) procedure.

```
T^0 = ∅; t ← 0;
while not converged (i.e., (T^{t+1} = T^t)) do
    Choose a permutation π^t whose chain contains the set T^t;
    T^{t+1} := argmin_{|T|=k, T∈N} [m^f_{T^t}(T) − h^g_{T^t,π^t}(T)];
    t ← t + 1;
return T^t
```

by replacing f by its modular upper bound $m_{T^t}^f$ and g by its modular lower upper bound h_{T^t,π^t}^g respectively, and the adapted

Table 1
The datasets.

Network	#nodes	#edges	direct	#vertex in $\mathcal{G}_3^f$	#vertex in $\mathcal{G}_{10}^f$
Facebook	4.0K	0.8M	False	9.8K	0.8K
Twitter	81.3K	17.6M	True	2.1M	10K
Gplus	1.0M	136.7M	True	10.2M	1.1M

k-knapsack-Modular-Modular procedure algorithm is guaranteed to monotonically decrease the objective at every iteration and converge to a local minima. Since in each iteration, minimizing a modular set function with the constant constraint of set size can be done in $O(|N|)$, the iteration is extremely easy.

6.2. Heuristic algorithm

Here, we consider the general greedy hill-climbing algorithm to solve the problem by choosing seeds with maximal marginal gain for the objective one by one. However, such short-sighted strategy may lead to a bad result since there is often a problematic situation where nodes provide same marginal gains such as zero but different long-term delayed gain for objective $F_{\mathbb{H}}$. To show this problem, we still use the simple example with $\mathbb{H} = \{\mathcal{H}_1, \mathcal{H}_2\}$ where $\mathcal{H}_1 = \{\{1\}, \{2,3\}\}$, $\mathcal{H}_2 = \{\{1\}, \{2\}\}$. Suppose that we need to choose two seeds. By the greedy hill-climbing strategy, in the first step, all nodes provide zero marginal gain for the objective. If we choose node 3, we will get a bad solution $\{1,3\}$, but if we choose node 2 or 1 in the first step, it will lead to the best solution $\{1,2\}$. So based on this idea, we need to design a weight to distinguish each node's long-term delayed gain for the objective. Note that $\xi_S(\mathcal{H})$ is computed by $\chi\{S \subseteq V_{\mathcal{H}}\} \cdot \chi\{deg_{\mathcal{H}}(S) = |E_{\mathcal{H}}|\}$. Therefore, in order to add a node to provide more gain for the objective in future, we need v to be included in: 1) as many sets of $V_{\mathcal{H}}$ as possible; 2) as many subsets of $E_{\mathcal{H}}$ that have no intersection with the previous selected seeds S as possible. Based on such heuristic ideas, we distinguish node v's long-term delayed gain by computing the weight $W_S(v) := \sum_{\mathcal{H}\in\mathbb{H}} \chi\{v \in V_{\mathcal{H}}\} \cdot \frac{(deg_{\mathcal{H}}(\{v\}\cup\{S\}) - deg_{\mathcal{H}}(\{S\})}{|E_{\mathcal{H}}|}$. By combining the immediate marginal gain and long-term delayed gain for the objective, we employ the adaptive greedy hill-climbing algorithm as shown in Algorithm 3 with the time complexity of $O(k|V||\mathbb{H}|)$. Back to the example above, by Algorithm 3, in the first step we have node 1 and 2 with the weight of long-term delayed gain 1 and

Algorithm 3: AG($\mathbb{H}, k$).

```
S ← ∅;
for q from 1 to k do
    // immediate marginal gain firstly.
    Get all nodes S_q ← argmax_{s∈V} F_ℍ(S ∪ {v}) − F_ℍ(S);
    // long-term delayed gain secondly.
    Get a node s_q ← argmax_{v∈S_q} W_S(v);
    Let S ← S ∪ {s_q};
return S as the seed set
```

node 3 with the lower weight of long-term delayed gain 0.5, thereby this algorithm will lead to the best result.

7. Experiments

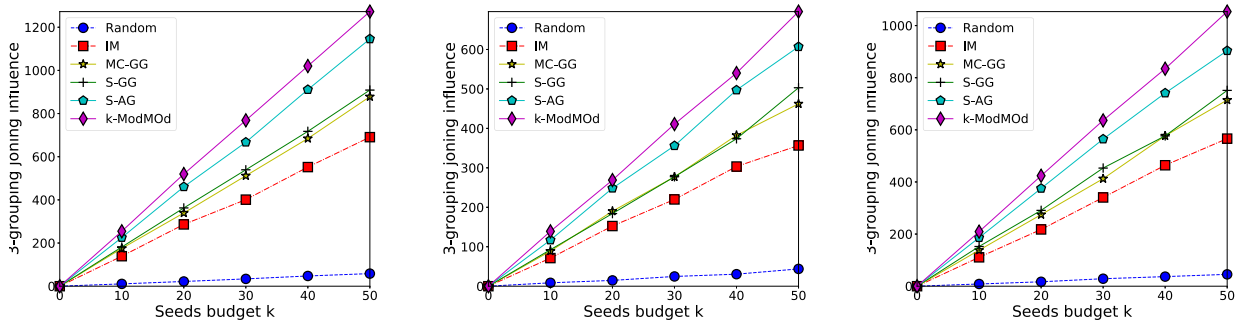
In this section, based on 3 real-world labeled datasets[1] (facebook, twitter, gplus) shown in Table 1, we conduct extensive experiments to evaluate the performance of the methods we proposed above for both time efficiency and effectiveness. All of our codes are written in c++ with coding optimization, and run on a linux server with 12 cores, 24 threads, 2.4 GHz CPU and 64 GB RAM.

7.1. Experiment preparation

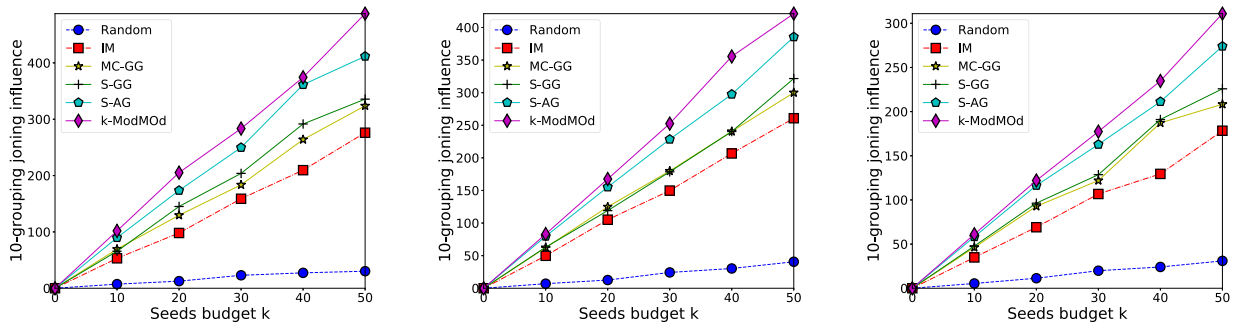
To get the influence graph of the IC model we used for information diffusion, for the directed graph datasets of twitter and gplus, we set an influence edge $\vec{e}_{uv}$ if user v follows user u, and for the undirected graph datasets of facebook, we set two influence edges $\vec{e}_{uv}$ and $\vec{e}_{vu}$ if user v and u are friends. As the general setting in prior works for IC model, for each influence edge $\vec{e}_{uv}$, we set its influence probability p^i_{uv} to $\frac{1}{deg_{in}(v)}$ where $deg_{in}(v)$ is the in-going degree of v in G^i. We randomly label nodes uniformly from 100 labels and use $\gamma^2(u,v)$ to build the faith edge for two nodes u and v, i.e., u and v have faith into one group iff the number of hops between them is no more than 2 in the social network. Then we get the corresponding graph $\mathcal{G}_\kappa^f$ in which each vertex corresponds to a possible group satisfying the requirement size of variable κ from 3 and 10. We compare our algorithms with following baselines:

- **Random**: Get k seeds randomly from the nodes set V by 100 times and then chose the best one.

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



(a) Facebook, $\kappa = 3$. (b) Twitter, $\kappa = 3$. (c) Gplus, $\kappa = 3$.



(d) Facebook, $\kappa = 10$. (e) Twitter, $\kappa = 10$. (f) Gplus, $\kappa = 10$.

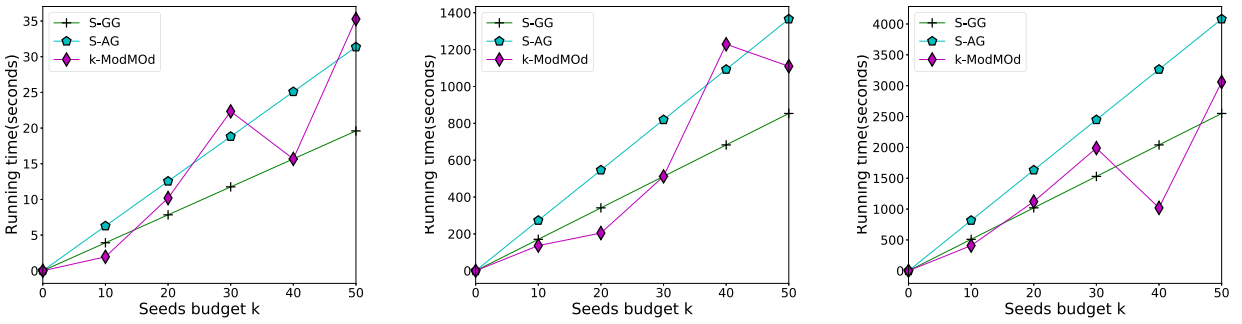
Fig. 8. Expected 3(10)-grouping joining influence achieved by all comparison algorithms on the three real-world datasets.

- **IM**: Choose k seeds provided by the algorithm [32] which aims to solve the influence maximization problem.
- **MC-GG**: Use the mean of many Monte Carlo simulations of dynamic diffusion-group to estimate expectation (i.e., the objective) for given seeds and hence select seeds one by one through the general greedy hill-climbing algorithm.
- **S-GG**: Firstly sample enough κ-JRRHs, and secondly select seeds one by one as the general greedily climbing algorithm by estimating the objective for given seeds through the computation based on these κ-JRRHs.
- **S-AG**: Sample enough κ-JRRHs to estimate the objective but select seeds one by one through the adaptive greedily climbing, i.e., Algorithm 3 we proposed.
- **k-ModMod**: Use Algorithm 2 we proposed by a local descending searching algorithm for the maximization problem of the difference of two submodular functions.

7.2. Effectiveness

In this part, by running these comparison algorithms, we will compare and analyze the effectiveness for κ-grouping joining influence, i.e., the quality of seeds the comparison algorithms provide. Notice that in these comparison algorithms, we need firstly sample certain number of enough κ-JRRHs in algorithm **S-GG**, **S-AG** and **k-ModMod**, and then following the theory result in Theorem 7, here we fix the samples number to be computable $\lambda_l^\star(0.05, 0.99)$ by setting $\epsilon = 0.1$, $\delta = 0.99$ to ensure a high confidence and accuracy for the objective estimation and hence a solution with high quality. For algorithm **MC-GG** of Monte Carlo simulations, we set the number of simulations to be 10000 which is large enough to provide a high estimation precision. To evaluate and distinguish the effectiveness of the solutions provided by different algorithms, we count the mean of total number of members over all groups that meet the condition with size at least κ in 10000 Monte Carlo simulations. We compare the algorithms with variable seeds budget k from the set $\{10, 20, 30, 40, 50\}$.

In Fig. 8, we firstly plot the expected 3-grouping joining influence on all three networks with the minimum number requirement for partners into a group to be 2, i.e., $\kappa = 3$ and then plot the expected 10-grouping joining influence on all three networks with $\kappa = 10$. Totally, we have that the expected grouping joining influences with the seeds provided by different algorithms satisfy following comparisons: **k-ModMod**>**S-AG**>**S-GG**≈**MC-GG**>**IM**>**Random**. **k-ModMod** performs better than **S-AG** in most situations but not significantly in some situations such as in Facebook with $\kappa = 10$, $k = 40$, and Gplus with $\kappa = 10$, $k = 10, 20$. Specifically, the expected grouping joining influences obtained by the general greedily climbing algorithm are very close whenever the objective is estimated by the κ-JRRHs or the Monte Carlo simulations. It proves that the estimation based on the κ-JRRHs we propose is not weaker than the traditional Monte Carlo in terms of precision. We also notice that the adaptive greedily climbing algorithm improves the grouping joining influence significantly in all datasets and it indeed has an advantage to avoid some short-sighted choices in the



(a) Facebook, $\kappa = 3$. (b) Twitter, $\kappa = 3$. (c) Gplus, $\kappa = 3$.

Fig. 9. The running time of selecting seeds achieved by three κ-JRRHs based algorithms.

greedily climbing. It is no accident that **IM** algorithm is just better than the **Random** but worse significantly than the method we proposed since **IM** algorithm aims to solve the breath of influence spread without considering the grouping constraint. It may cause that many nodes can be influenced but can't find enough like-mind partners.

7.3. Time efficiency

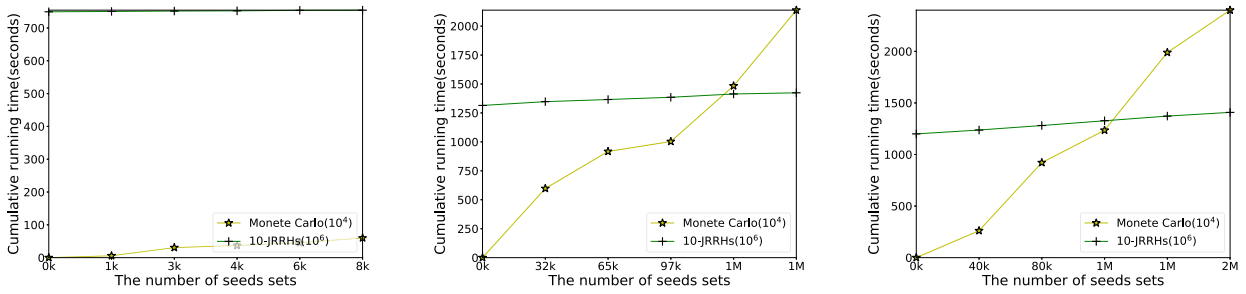
In this part, we will analyze the time efficiency for our algorithms and estimation method based on κ-JRRH respectively. Notice that these algorithms **k-ModMod**, **S-GG** and **S-AG** are based on the objective estimation of sampling certain number of κ-JRRHs, hence there are two parts to decide the time efficiency. The first part is the process of sampling κ-JRRHs and it's related to the number of samples λ and the expected time cost t_s in one random sampling process, i.e., the computational complexity in κ-JRRHs sampling is $O(\lambda \cdot t_s)$. The second part is the process of selecting seeds. Especially in algorithm **k-ModMod**, since the time cost in each iteration is $O(|V|)$, the total computational complexity of selecting seeds is $O(t_i \cdot |V|)$ where t_i is the number of converge iterations of searching. Therefore, in this part of simulation, we firstly compare the time efficiency in seeds selecting of these three κ-JRRH based algorithms under the same samples. Notice that in the estimation method that κ-JRRH uses, different from Monte Carlo in which we need repeat simulations once seeds changed, we just need to collect them once and then directly use them to compute the estimation for any given seeds. So in order to compare the time efficiency of these two objective estimation methods, we accumulate the time cost in both sampling and estimation computation while adding a lot of random seeds sets constantly.

Fig. 9 plots the running time of the selected seeds in three κ-JRRH based algorithms **k-ModMod**, **S-GG** and **S-AG** under the same samples with $\lambda_l^\star(0.05, 0.99)$ and $\kappa = 3$. There is no doubt that the running time of both greedy climbing algorithms varies nearly linearly with the budget k, and the adaptive one we propose costs nearly more 50% time than the general one since there is an extra step of weight computation in each loop. For **K-ModMod**, we find that it's not positively correlated with the budget k, such as the situation when $k = 40$ in *Gplus* and $k = 50$ in *Twitter*. The reason is that the algorithm is not a strict incremental algorithm in $O(k)$. Although in each iteration, there runs a linearly complexity algorithm of minimizing a modular function with the knapsack constraint of set size k, the running time may not be positively correlated with the budget k. Specially we notice that our algorithms seem to run slowly on *Twitter* and *Gplus*, it's because the number of samples is significantly enlarged in larger scale networks under our setting of $\epsilon = 0.1$, $\delta = 0.99$. Actually, it's a conservative setting to ensure the estimation precision and we can reduce the number of samples to an acceptable range to avoid such intensive computation.

Fig. 10 plots the cumulative running time to estimate the objective for given seeds sets achieved by Monte Carlo and κ-JRRHs respectively. We compare these two methods by setting the number of Monte Carlo simulations to be 10k for each estimation of given seeds set and the number of sampling 10-JRRHs to be 10M. In *Twitter* and *Gplus*, the time cost of estimation in Monte Carlo increases significantly comparing with the computation of κ-JRRHs. Overall from the figure, we can see that the time cost of computation based on κ-JRRHs is negligible comparing to the heaviest part of κ-JRRHs sampling. Specifically, on large scale networks like *Twitter* and *Gplus*, in the long term the value of objective will be frequently needed in most algorithms such as the greedy hill-climbing. Comparing to Monte Carlo simulations, our method of sampling κ-JRRHs can significantly reduce the total cost in objective estimation and make seeds selecting algorithms more feasible on larger scale networks.

8. Conclusions

In this paper, we investigate the marketing for the business model of group buying over social influence. We formulate it into the problem of choosing budgeted seeds to maximize the influence with grouping constraint based on a diffusion-group model we propose. We summarize the properties of this problem with the strict theoretical analysis and propose a method of sampling hypergraphs to estimate the objective since its computation is #P-hard. Based on the estimation method we propose, we design two algorithms: the first one is a local descending search by transforming the objective function to a difference of two submodular functions, and the second one is an adaptive greedy hill-climbing to avoid short-sighted problem in general hill-climbing. At last, extensive experiments conducted on real-world datasets show that our methods perform well.



(a) Facebook, $\kappa = 10$. (b) Twitter, $\kappa = 10$. (c) Gplus, $\kappa = 10$.

Fig. 10. The cumulative running time to estimation the objective for given seeds sets achieved by Monte Carlo and κ-JRRHs respectively.

CRediT authorship contribution statement

Guoyao Rao: Conceptualization, Methodology, Software, Writing – original draft. **Deying Li:** Funding acquisition, Supervision, Writing – review & editing. **Yongcai Wang:** Writing – review & editing. **Wenping Chen:** Writing – review & editing. **Chunlai Zhou:** Funding acquisition, Writing – review & editing. **Yuqing Zhu:** Writing – review & editing.

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.

Data availability

Data will be made available on request.

References

[1] N. Barbieri, F. Bonchi, G. Manco, Topic-aware social influence propagation models, Knowl. Inf. Syst. 37 (2013) 555–584.
[2] R. Becker, F. Coro, G. D'Angelo, H. Gilbert, Balancing spreads of influence in a social network, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2020, pp. 3–10.
[3] 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.
[4] 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.
[5] S. Chen, J. Fan, G. Li, J. Feng, K.l. Tan, J. Tang, Online topic-aware influence maximization, Proc. VLDB Endow. 8 (2015) 666–677.
[6] W. Chen, Y. Wang, S. Yang, Efficient influence maximization in social networks, in: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2009, pp. 199–208.
[7] E. Cohen, D. Delling, T. Pajor, R.F. Werneck, Timed influence: computation and maximization, arXiv preprint, arXiv:1410.6976, 2014.
[8] K. Elbassioni, A polynomial delay algorithm for generating connected induced subgraphs of a given cardinality, arXiv preprint, arXiv:1411.2262, 2014.
[9] I.E. Erdoğmus, M. Cicek, Online group buying: what is there for the consumers?, Proc., Soc. Behav. Sci. 24 (2011) 308–316.
[10] A. Goyal, F. Bonchi, L.V. Lakshmanan, Learning influence probabilities in social networks, in: Proceedings of the Third ACM International Conference on Web Search and Data Mining, 2010, pp. 241–250.
[11] 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.
[12] Q. Guo, S. Wang, Z. Wei, M. Chen, Influence maximization revisited: efficient reverse reachable set generation with bound tightened, in: Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data, 2020, pp. 2167–2181.
[13] 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.
[14] K. Huang, S. Wang, G. Bevilacqua, X. Xiao, L.V. Lakshmanan, Revisiting the stop-and-stare algorithms for influence maximization, Proc. VLDB Endow. 10 (2017) 913–924.
[15] T. Ito, H. Ochi, T. Shintani, A group-buy protocol based on coalition formation for agent-mediated e-commerce, Int. J. Comput. Inf. Sci. 3 (2002) 11–20.
[16] R. Iyer, J. Bilmes, Algorithms for approximate minimization of the difference between submodular functions, with applications, arXiv preprint, arXiv:1207.0560, 2012.
[17] S. Karakashian, B.Y. Choueiry, S.G. Hartke, An algorithm for generating all connected subgraphs with k vertices of a graph, Lincoln, NE, 2013.
[18] 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.
[19] C. Komusiewicz, F. Sommer, Enumerating connected induced subgraphs: improved delay and experimental comparison, Discrete Appl. Math. 303 (2021) 262–282.
[20] 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.
[21] 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.

[22] W. Lu, W. Chen, L.V. Lakshmanan, From competition to complementarity: comparative influence diffusion and maximization, Proc. VLDB Endow. 9 (2015) 60–71.
[23] R. Motwani, P. Raghavan, Randomized Algorithms, Cambridge University Press, 1995.
[24] G.L. Nemhauser, L.A. Wolsey, M.L. Fisher, An analysis of approximations for maximizing submodular set functions—i, Math. Program. 14 (1978) 265–294.
[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] 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.
[27] G. Rao, Y. Wang, W. Chen, D. Li, W. Wu, Matching influence maximization in social networks, Theor. Comput. Sci. 857 (2021) 71–86.
[28] G. Rao, Y. Wang, W. Chen, D. Li, W. Wu, Union acceptable profit maximization in social networks, Theor. Comput. Sci. 917 (2022) 107–121.
[29] 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.
[30] 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.
[31] 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.
[32] 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.
[33] 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.
[34] 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 (2014) 449–460.
[35] A. Tsang, B. Wilder, E. Rice, M. Tambe, Y. Zick, Group-fairness in influence maximization, in: Proceedings of the 28th International Joint Conference on Artificial Intelligence, AAAI Press, 2019, pp. 5997–6005.
[36] E.S.T. Wang, N.P.Y. Chou, Consumer characteristics, social influence, and system factors on online group-buying repurchasing intention, J. Electron. Commer. Res. 15 (2014) 119.
[37] A.L. Yuille, A. Rangarajan, The concave-convex procedure (cccp), in: Advances in Neural Information Processing Systems, 2002, pp. 1033–1040.
[38] Z. Zhang, Z. Zhang, F. Wang, R. Law, D. Li, Factors influencing the effectiveness of online group buying in the restaurant industry, Int. J. Contemp. Hosp. Manag. 35 (2013) 237–245.
[39] Z. Zhang, W. Zhao, J. Yang, C. Paris, S. Nepal, Learning influence probabilities and modelling influence diffusion in Twitter, in: Companion Proceedings of the 2019 World Wide Web Conference, 2019, pp. 1087–1094.
[40] W. Zhao, A. Wang, Y. Chen, How to maintain the sustainable development of a business platform: a case study of pinduoduo social commerce platform in China, Sustainability 11 (2019) 6337.
[41] J. Zhu, S. Ghosh, W. Wu, Group influence maximization problem in social networks, IEEE Trans. Comput. Soc. Syst. 6 (2019) 1156–1164.