Efficient Minimum Cost Seed Selection with Theoretical Guarantees for Competitive Influence Maximization

Wenjing Hong, *Member, IEEE,* Chao Qian, *Member, IEEE,* and Ke Tang, *Senior Member, IEEE*

***Abstract*—Minimum cost seed selection for competitive influence maximization, which selects a set of key users (called seed set) to spread its influence widely into the network at a minimum cost in a competitive social network, is a key algorithmic problem in social influence analysis. Due to its application potential in multiple fields such as market expansion, election campaigns and cultural competition, numerous studies have been emerging recently. Despite these efforts, this problem has not been satisfactorily solved due to that not only finding a (nearly) optimal solution for cost minimization, but also evaluating a seed set is computationally complex. Existing works either trade approximation guarantees for practical efficiency using heuristics, or vice versa due to costly Monte Carlo simulations. In this paper, a competitive reverse influence estimation-based greedy algorithm, which provides bounded approximation guarantees, but offers significantly improved empirical efficiency under the competitive independent cascade model, is proposed. The core of the algorithm is a novel estimation method that improves the efficiency by constructing representative sketches to avoid heavy repeated simulations without compromising its performance guarantees. Experimental results on eight real-world networks with up to 1.13 million users show that compared with state-of-the-art algorithms, our algorithm is the most efficient while keeping the best performance, and can be orders of magnitude faster.**

***Index Terms*—Competitive influence maximization, minimum cost seed selection, reverse influence sampling, social networks.**

I. Introduction

COMPETITIVE Influence Maximization (CIM) is a combinatorial optimization problem that seeks a small set (referred to as seed set) of key users who spread the influence widely into the network via the word-of-mouth effect on competitive social networks [1]. One popular type of CIM problems that aims at seeking the seed set at a minimum cost is also called minimum cost seed selection. It finds numerous real-world applications in market expansion when there are multiple competing products, different political messages or ideas, such as viral marketing [2]–[4], election campaigns [5], rumor blocking [6]–[9], profit maximization [10] and social analysis [11], [12]. Despite its application potential, CIM presents several challenges especially for its computational complexity. It has been proved that not only finding an optimal seed set for cost minimization is a complex combinational optimization problem, but also evaluating the influence of a seed set is NP-hard [13], [14]. As a consequence, research on CIM has witnessed an increasing growth in recent years [1], [14]–[17].

However, existing works either suffer from high computational overheads or are unable to offer any performance guarantees. Specifically, simulation-based greedy algorithms [14], [17]–[20] which adopt greedy algorithms to overcome the combinatorial hardness and Monte Carlo simulations to overcome the influence evaluation source of hardness are the primary methods for solving CIM problems as they can produce near-optimal solutions with theoretical guarantees by modestly extracting problem properties such as submodularity [21]. Although the greedy algorithm may not perform well in general, it is able to provide an approximation guarantee towards the global optimum if the set function is monotone and submodular [22]. Since submodularity and monotonicity often arise in analyzing influence spread in a social network [1], they guarantee why a greedy algorithm finds a nearly optimal solution in this context. However, such method requires a large number of costly Monte Carlo simulations, which can take several days even for small-scale networks, since one set of simulations can only provide an estimation for one specified seed set. Therefore, heuristic algorithms without theoretical guarantees have been proposed to speed up the CIM by simplifying the influence spread process [13], [17], [23]. While these methods might be more scalable, their efficiency often comes at the deterioration of solution quality due to the inaccurate influence estimation caused by the ignorance of some important features of the problem [16], [24], or their quality of solutions may not be robust to network structures partly because of the absence of theoretical guarantees [17]. A few more complicated algorithms such as evolutionary algorithms [25] and reinforcement learning [26] have also been developed, but none of them can handle complex large-scale networks without incurring prohibitive overheads since they require a large number of costly iterations.

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFB1003102, the Natural Science Foundation of China under Grant 61672478, the Program for Guangdong Introducing Innovative and Entrepreneurial Teams under Grant 2017ZT07X386, the Shenzhen Peacock Plan under Grant KQTD2016112514355531, the Guangdong-HongKong-Macao Greater Bay Area Center for Brain Science and Brain-Inspired Intelligence Fund (NO.2019028), and the Fundamental Research Funds for the Central Universities. *Corresponding Author: Chao Qian.*

Wenjing Hong and Ke Tang are with the Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, China (e-mail: hongwj@sustech.edu.cn; tangk3@sustech.edu.cn). Wenjing Hong is also with the School of Management, University of Science and Technology of China, Hefei 230027, China, and the Guangdong-HongKong-Macao Greater Bay Area Center for Brain Science and Brain-Inspired Intelligence, Guangzhou 510515, P. R. China.

Chao Qian is with the National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China (e-mail: qianc@nju.edu.cn).

In this paper, an efficient greedy algorithm able to provide non-trivial approximation guarantees is proposed to solve the CIM problem under the prominent competitive independent cascade model [14]. Compared with state-of-the-art algorithms, its superiority is shown in terms of both the theoretical approximation guarantee for solution quality in the case without a precise evaluation function and the high efficiency for practical use. The core of the algorithm is a novel estimation technique to speed up the influence evaluation by first constructing theoretically grounded sketches under the influence diffusion model based on a Competitive Reverse Influence Sampling (CRIS) strategy, and then estimating the influence based on the constructed sketches. Based on the sketches, the algorithm can avoid the time-consuming overhead of repeated simulations for each specified seed set by employing the same sketches for any influence evaluation. This is inspired by the Reverse Influence Sampling (RIS) [27] which has made significant progress for non-competitive influence maximization [28], [29]. However, existing RIS methods are not applicable to CIM problems because of the ignorance of competitive relationship and benefits among different competitors in the sampling. Hence, the novel CRIS is proposed to solve this problem. In particular, the sketches are finely structured to consider the possible competing situations that may exist among a seed set and other competitors, based on which the influence evaluation can be transformed into a weighted coverage problem. Indeed, we show that a statistic based on CRIS is identical to the expectation of the influence, and moreover, this statistic can be estimated accurately with certain number of samplings using the Chernoff bound [30]. The CRIS-based estimation is then integrated into the generalized greedy algorithm [22] under the guidance of theoretical analysis, and leads to the Competitive Reverse Influence Estimation-based Greedy (CRIEG) algorithm.

Our contributions are summarized as follows.

1) A theoretically grounded influence estimation mechanism is developed for CIM. It is a key to improving the efficiency of algorithms. It also includes a tunable parameter for users to control the balance between running time and precision. Besides, this mechanism extends the powerful RIS technology to a more general case with competitive settings for influence estimation, and thus may be of important interest to CIM.
2) An efficient and provable effective CIM algorithm is proposed based on the above influence estimation. The results show that with probability at least $1 - 1/n$, CRIEG can achieve an approximation ratio within $((1+\eta)/(1-\eta))$-factor of the best existing ratio under the competitive independent cascade model derived in the setting with the precise influence function available [14], where $\eta \in [0, 1)$ indicates the relative estimation error of CRIS estimation. Here, n, m, and l indicate the number of nodes, edges and non-seed nodes of competitors, respectively.
3) Experimental studies are carried out to examine the proposed CRIEG in comparison to four state-of-the-art methods. The results on eight real-world networks show its high efficiency and effectiveness. In particular, on a network with up to 1.13 million nodes, CRIEG is up to five times better in solution quality and up to three orders of magnitude faster.

The remainder of this paper is organized as follows. Section II reviews related work. Section III presents the problem statement and related notations. Section IV details the proposed competitive reverse influence estimation method as well as the CRIEG algorithm. The theoretical and experimental studies are presented in Section V and Section VI, respectively. Section VII finally concludes the paper.

II. Related Work

With the immense application potential in competitive scenarios, CIM has gradually become a research hotspot. Bharathi et al. [20] and Carnes et al. [23] are among the first to study CIM problems. Since then, a number of CIM techniques have been developed. The special case of only two opposite influences has been widely studied [17]–[19], [31], but such assumption breaks down in many scenarios when there are multiple competitors such as when promoting a new production in the mobile phone market [32].

Therefore, existing research efforts have focused on CIM with multiple competitors and devising efficient approximation algorithms. Greedy algorithms are the primarily employed problem-solving mechanism for seed selection with any influence evaluation function σ as they can provide relatively high efficiency as well as a good theoretical guarantee towards the global optimum [1], [21], compared with reinforcement learning [33] or evolutionary algorithms [34]–[36] that require a large number of costly iterations. Specifically, in each iteration, one node v^* is added into the candidate set Q, such that v^* provides the largest ratio of the marginal gain on the influence function σ and the seed cost with respect to the set Q. This selection process is repeated until the termination condition is met. Particularly, it has been widely shown that greedy algorithms can achieve state-of-the-art approximation ratios for CIM problems under various diffusion models. The most relevant work to ours is [14]. The greedy algorithm provides the best approximation ratio of $\ln(f^*/(\epsilon c^*)) + 1$ for CIM under the competitive independent cascade model, where $\epsilon > 0$, and f^* and c^* indicate the influence and cost of the optimal solution, respectively. Nevertheless, such result is derived when the influence is accurately evaluated, and thus CIM is still challenging because of the computational hardness in influence evaluation.

Simulation-based greedy and heuristic algorithms are the two main techniques for processing the challenge posed by the influence evaluation. The former employs Monte Carlo simulations which rely on repeated random sampling to calculate the expected number of influenced non-seed nodes for influence evaluation [14]. Although such methods can achieve high solution quality, they are computationally expensive as they ask for a brand-new estimation which contains many simulations (e.g., 1000 samplings commonly used in practice) for each specified seed set. The latter avoids this expensive overhead by simplifying CIM problems. However, the quality of their

solutions could be worse or less robust to network structures due to the neglect of the nonlinearity or completeness of the competitive diffusion process [13], [24], [37], requirement of priori knowledge of user behavior [16], [33], or excessive dependency on geometric structure [38]. Du et al. [15] propose an efficient greedy algorithm and further show that it can scale well to large-scale networks. However, it is designed for continuous-step diffusion networks and cannot be used in discrete-step models, which are more commonly considered in influence maximization [1]. Thus, it cannot be directly applied to address our problem.

III. Problem Definition and Notations

The problem of minimum cost seed selection for CIM under the competitive independent cascade model is defined on a directed graph $G(V, E)$ and k competitors $I_1, \ldots, I_k$, where $I_j \subseteq V, j \in \{1, \ldots, k\}$, V and E represent the sets of nodes and edges, respectively. A cost $c(v)$ is associated with each node $v \in V$, a propagation probability $w(e)$ is associated with each edge $e(u, v) \in E$ to indicate the probability that node u successfully activates node v through e, and an influence threshold μ is predefined by users. The aim is to determine a small set of nodes (called seeds) $I_0 \subseteq V$ to influence a large number of non-seed nodes that exceeds the threshold μ, at a minimal total cost when all competitors participate in the influence spread, i.e.,

$$\min \sum_{v \in I_0} c(v) \quad s.t.\ \sigma(I_0) \geq \mu, \tag{1}$$

subject to the following competitive influence diffusion process. The influence $\sigma(I_0)$ of competitor I_0 is the expected number of nodes that decided to accept the information from I_0 after the diffusion.

The influence spreads and competes as follows. There are $(k + 1)$ different competitors $I_0, \ldots, I_k$ spreading their influences on graph G, and their nodes are active nodes (called seeds) used to initialize the influence diffusion. Suppose there are a total of m seeds, each seed s_i is associated with a set of competitors $\mathcal{I}_i$ which indicates that one seed may spread multiple influences. The diffusion process begins with these seeds and advances on the graph G in a cascade manner. Specifically, in time 0, all non-seeds are inactive. Each seed begins to spread the influences to its non-seed neighbor nodes, and s_i successfully activates v via $e(s_i, v)$ with the probability $w(e)$. At time t, every node v that successfully received some influences from the nodes that was activated at time $(t - 1)$ becomes active, and intends to spread all influences it received to its inactive neighbor nodes. At the same time, node v selects one competitor to accept based on all influences it received, and changes to the decided state. The acceptance rule is as follows: denote $\mathcal{I}_v$ as the set of competitors that v has received their influences, v becomes decided to any competitor $I \in \mathcal{I}_v$ with equal probability $1/|\mathcal{I}_v|$. The diffusion process ends when no more nodes become active.

Fig. 1 presents an example with seven nodes and four competitors I_0, I_1, I_2, I_3. Assume that all propagation probabilities in this graph are 1.0. The influence diffusion starts from the seed set $\{s_1, s_2, s_3\}$ and each seed is activated with probability 1.0. At this time, s_1 spreads the influences of I_1, I_2 to y_1, y_2; s_2 spreads the influence of I_0 to y_1, y_4; s_3 spreads the influences of I_1, I_2, I_3 to y_1. Next, y_1, y_2, y_4 become active, and thereafter enter the decided status. y_4 decides from I_0 with probability 1.0; y_1 decides from I_0, I_1, I_2, I_3, each with probability $1/4$; y_2 decides from I_1, I_2, each with probability $1/2$. Afterwards, similarly, y_4 spreads the influence of I_0 to y_3; y_2 spreads the influences of I_1, I_2 to y_3. Then y_3 becomes active, and decides from I_0, I_1, I_2, each with probability $1/3$. So far, the diffusion stops as there has been no potential active nodes. Note that the seeds in $|I_0|$ always accept the information from I_0. Thus, the influence of I_0 can be calculated by $\sigma(I_0) = 1 + 1 + 1/4 + 1/3 = 31/12$.

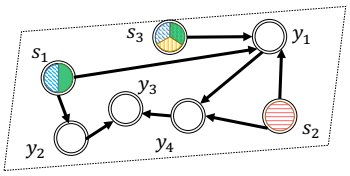

Fig. 1. An example network with seven nodes and four competitors. The red (horizontal line) indicates $I_0 = \{s_2\}$, blue (oblique line) indicates $I_1 = \{s_1, s_3\}$, green (grid) indicates $I_2 = \{s_1, s_3\}$, and yellow (vertical line) indicates $I_3 = \{s_3\}$.

The information spread described above has been proved to be a submodular and monotone set function [14]. This problem has been proved to be NP-hard, and the evaluation of influence spread, i.e., $\sigma(\cdot)$, is #P-hard [14].

IV. The Proposed Approach

As mentioned above, although the CIM problem is computationally complex, the greedy algorithm can produce near-optimal solutions if the precise influence is available, and the simulation-based greedy algorithm is shown to be provable effective. However, the simulation-based greedy algorithm suffers from a key drawback due to the high computational overheads resulting from repeated costly Monte Carlo simulations for each specified seed set. In order to circumvent such difficulty, the CRIEG algorithm is proposed and described in this section. It adopts a sampling first and seed selection second manner to release the expensive computational overheads. The CRIEG involves two key design issues. The first is how to perform the sampling efficiently on competitive social networks, while ensuring a provable effectiveness for the influence evaluation. The second is how to integrate the proposed estimation method into the classic greedy algorithm so that the resultant algorithm can provide non-trivial approximation guarantees. In the following, the CRIS-based estimation method and the CRIEG algorithm are presented for the two issues, respectively.

A. Competitive Reverse Influence Estimation

The key idea of this method is to first construct a set of representative sketches, and then estimate the influence for any seed set based on the same constructed sketches. Thus, it requires the sketches to comprehensively consider the possible competing situations that may exist among a seed set and

other competitors. The novel CRIS strategy is proposed for this purpose. First, a set of sketches named random Competitive Reverse Reachable (CRR) sets is defined and constructed based on an analysis of three situations which represent different competitive relationship and benefits. Second, based on the set of random CRR sets, the influence evaluation is equivalently transformed into a weighted coverage calculation through theoretical derivation. In the following, to explain how the competitive reverse influence estimation works, the CRR sets is introduced first, and then the transformation is described.

Definition 1 *A CRR set contains two parts, a competitive subset B and a competitive-free subset D, which are generated from G by 1) selecting a node u from V uniformly at random; 2) generating a sample graph g from G by removing each edge e in G with probability $1-\omega(e)$; 3) defining the length of the directed path from v to u in g as the arriving time of v, for each $v \in V$; 4) returning $D_g(u)$ as the set of nodes arriving earlier than all nodes in H; 5) returning $B_g(u)$ as the union of the following two sets: one is the set of nodes in H arriving first, and the other is the set of nodes in $\overline{H}$ arriving at the same time, where $H = \bigcup_{j=1}^{k} I_j$ and $\overline{H} = V \backslash H$.*

An intuitive way to understand Definition 1 is as follows. Given a number of random CRR sets, the following three cases are considered. If a certain node $v \in V$ appears frequently in D, then v is likely a good candidate for the most influential node, as it arrives earlier than the competitor. In this case, if a new competitor selects v as a seed node, it is likely to win the competition. This is also the reason why D is called the competitive-free subset. If v appears frequently in B, then it can also be of good potential by competing with competitors, and thus a new competitor involving this node would be competitive. The node activated in this case will select a competitor uniformly at random. Thus, B is named as the competitive subset. Otherwise, the influence spread of v might be trivial as the competitor can be more competitive. The influence spread of a competitor can be derived by combining the above three cases.

Taking the social network in Fig. 1 as an example, the sample graph g contains all nodes and edges, as the propagation probabilities are set to 1.0 in this example. Here $H = \{s_1, s_3\}$. Assume that y_3 is selected, the length of the directed path from nodes y_2, y_4 to y_3 is 1, and the length from nodes s_1, y_1, s_2 to y_3 is 2 in the graph g. As $\{y_2, y_3, y_4\} \cap H = \emptyset$ and $\{s_1, y_1, s_2\} \cap H \neq \emptyset$, it holds that $D_g(y_3) = \{y_2, y_3, y_4\}$ and $B_g(y_3) = \{s_1, y_1, s_2\}$. Similarly, $D_g(y_1) = \{y_1\}$, $B_g(y_1) = \{s_1, s_2, s_3\}$; $D_g(y_2) = \{y_2\}$, $B_g(y_2) = \{s_1\}$; $D_g(y_4) = \{y_1, s_2, y_4\}$, $B_g(y_4) = \{s_1, s_3\}$; $D_g(s_2) = \{s_2\}, B_g(s_2) = \emptyset$. Given I_0, since $I_0 \cap D_g(y_4) \neq \emptyset$, y_4 is activated by I_0 with probability 1; $I_0 \cap B_g(y_1) \neq \emptyset$, and $I_1 \cap B_g(y_1) \neq \emptyset$, $I_2 \cap B_g(y_1) \neq \emptyset$, $I_3 \cap B_g(y_1) \neq \emptyset$, thereby y_1 is activated with probability $1/4$; $I_0 \cap B_g(y_2) = \emptyset$, thereby y_2 is activated with probability 0; $I_0 \cap B_g(y_3) \neq \emptyset$, and $I_1 \cap B_g(y_3) \neq \emptyset$, $I_2 \cap B_g(y_3) \neq \emptyset$, y_3 is activated with probability $1/3$; $I_0 \cap D_g(s_2) \neq \emptyset$, thereby s_2 is activated with probability 1; in addition, as $I_0 \cap H = \emptyset$, the influence spread $\sigma(I_0) = 1 + 1/4 + 0 + 1/3 + 1 + 0 = 31/12$, and it can be seen that the influence computed by the competitive reverse influence estimation equals to that obtained in Section III.

As shown above, to compute the influence spread of a set I_0, its influence on a node u is first computed given a graph g randomly drawn from G; then, a summation over all $u \in \overline{H}$ is taken when computing the influence of I_0 for a given g, from which an expectation over the distribution of g plus $|I_0 \cap H|$ is further taken to get $\sigma(I_0)$. This process is formally stated in the following theorem. In Theorem 3, it will be shown that an empirical average based on a number of randomly generated g can provide a bounded approximation guarantee for the precise influence $\sigma(I_0)$ and this approximation scheme will be adopted in CRIEG.

For an event Ω, let $[\Omega]$ be the indicator function, i.e., if Ω is true, $[\Omega] = 1$; otherwise, $[\Omega] = 0$. Denote $|\cdot|$ as the cardinality of a set.

Theorem 1 *The random competitive and competitive-free subsets are generated based on H, as presented in Definition 1. For any I, $\tilde{I} \subseteq V$, let $Z(I, \tilde{I}) = [I \cap \tilde{I} = \emptyset]$ and $\overline{Z}(I, \tilde{I}) = [I \cap \tilde{I} \neq \emptyset]$. For any new competitor $I_0 \subseteq V$,*

$$\sigma(I_0) = |I_0 \cap H| + \sum_{u \in \overline{H}} \mathbb{E}_g \Bigg(\overline{Z}(I_0, D_g(u)) + \frac{Z(I_0, D_g(u)) \overline{Z}(I_0, B_g(u))}{\sum_{j=1}^{k} \overline{Z}(I_j, B_g(u)) + \overline{Z}(I_0, B_g(u))} \Bigg). \quad (2)$$

Proof Let $S = \bigcup_{j=0}^{k} I_j = I_0 \bigcup H$. Denote $L_g(I_j)$ as the set of nodes that can be firstly reachable by a node $v \in I_j$, $j \in \{0, 1, \ldots, k\}$, $P_g(u)$ as the set of competitors I that can first reach the node u, where $I \in \{I_0, \ldots, I_k\}$, and $A_g(u)$ as the set of nodes v that can first reach the node u, where $v \in S$.

By the definition of σ,

$$\begin{aligned} \sigma(I_0) &= |I_0| + \sum_{u \in V \backslash S} \mathbb{E}_g \left(\frac{[u \in L_g(I_0)]}{\sum_{j=0}^{k} [u \in L_g(I_j)]} \right) \quad (3) \\ &= |I_0| + \sum_{u \in V \backslash S} \mathbb{E}_g \left(\frac{[I_0 \in P_g(u)]}{\sum_{j=0}^{k} [I_j \in P_g(u)]} \right) \\ &= |I_0| + \sum_{u \in V \backslash S} \mathbb{E}_g \left(\frac{[I_0 \cap A_g(u) \neq \emptyset]}{\sum_{j=0}^{k} [I_j \cap A_g(u) \neq \emptyset]} \right), \end{aligned}$$

where $1/(\sum_{j=0}^{k} [u \in L_g(I_j)]) = 0$ if $\sum_{j=0}^{k} [u \in L_g(I_j)] = 0$.

- If $I_0 \cap D_g(u) \neq \emptyset$, then $A_g(u) \subseteq I_0 \cap D_g(u)$. In this case, $I_0 \cap A_g(u) \neq \emptyset$ and $\forall j \in \{1, \ldots, k\}, I_j \cap A_g(u) = \emptyset$.
- Otherwise, $A_g(u) = S \cap B_g(u) = \{I_0 \cup H\} \cap B_g(u)$. In this case, $[I_0 \cap A_g(u) \neq \emptyset] = [I_0 \cap B_g(u) \neq \emptyset]$, and $\forall j \in \{1, \ldots, k\}, [I_j \cap A_g(u) \neq \emptyset] = [I_j \cap B_g(u) \neq \emptyset]$.

Thus, it holds that $\sigma(I_0) = |I_0| + \sum_{u \in V \backslash S} \mathbb{E}_g(f_1 + f_2)$, where $f_1 = \overline{Z}(I_0, D_g(u))$ and $f_2 = Z(I_0, D_g(u)) \cdot \frac{\overline{Z}(I_0, B_g(u))}{\overline{Z}(I_0, B_g(u)) + \sum_{j=1}^{k} \overline{Z}(I_j, B_g(u))}$. As $V \backslash S = \overline{I_0 \cup H}$, we have

$$\sigma(I_0) = |I_0| + \sum_{u \in \overline{I_0} \cap \overline{H}} \mathbb{E}_g(f_1 + f_2).$$

Considering $u \in \overline{H} \backslash (\overline{I_0} \cap \overline{H}) = I_0 \cap \overline{H}$, we have $I_0 \cap$

$D_g(u) \neq \emptyset$. Thus, it holds that

$$\begin{aligned}\sigma(I_0) &= |I_0| - |I_0 \cap \overline{H}| + \sum_{u \in \overline{H}} \mathbb{E}_g(f_1 + f_2) \\ &= |I_0 \cap H| + \sum_{u \in \overline{H}} \mathbb{E}_g(f_1 + f_2),\end{aligned}$$

i.e., the theorem holds. □

B. The CRIEG Algorithm

To solve the resultant problem using the competitive reverse influence estimation, the generalized greedy algorithm for submodular set cover [22] is employed. It is a natural approach for seed selection with any influence evaluation function $\sigma(\cdot)$. In our concerned problem, the calculation of exact σ is proved to be #P-hard [14], and thus, only an estimation function $\hat{\sigma}(\cdot)$ rather than the exact σ can be obtained. In the CRIEG algorithm, the competitive reverse influence estimation is adopted for the computation of $\hat{\sigma}$. Specifically, it first iteratively generates a number R of random CRR sets with corresponding competitive and competitive-free subsets, then sets Q to an empty set, and after that, the node v^* with the largest ratio of the marginal gain on the influence function $\hat{\sigma}$ and the seed cost is added to Q iteratively. In this process, the influence spread value $\hat{\sigma}(Q)$ is calculated as Eq. (2) by the average of the R samples for any candidate seed set Q. The parameter R is a tunable parameter for users to control the balance between running time and precision. Their relationship will be made clear in the following theoretical analysis in Section V. It will also be shown that by using the novel estimation $\hat{\sigma}$, CRIEG can avoid the time-consuming overhead of repeated simulations for each specified seed set by employing the same CRR sets for any influence evaluation, but still providing approximation guarantees.

The details of CRIEG are presented in Algorithm 1. At a high level, it consists of two phases:

- Competitive reverse influence sampling (lines 1–2);
- Seed node selection (lines 3–7).

Algorithm 1: CRIEG

Input: Graph G, seeds in competitors H, influence threshold μ
Parameter: Sample size R
Output: A seed set Q

1 Generate a number R of random CRR sets from G;
2 $Q \leftarrow \emptyset$;
3 **while** $\hat{\sigma}(Q) < \mu$ **do**
4 /* $\hat{\sigma}(\cdot)$ is calculated as Eq. (2) by the average of the R sets */
5 $v^* \leftarrow \arg\max_{v \in V \setminus Q} \frac{\hat{\sigma}(Q \cup \{v\}) - \hat{\sigma}(Q)}{c(v)}$;
6 $Q \leftarrow Q \cup \{v^*\}$
7 **end**
8 **return** Q

V. Theoretical Studies

In this section, we theoretically analyze the performance of CRIEG. Based on an analysis of the estimation error between σ and $\hat{\sigma}$, the general approximation guarantee of CRIEG is first proved. Next, the relationship between the estimation error and the number R of samples is provided. Combining these two points, the approximation guarantee of CRIEG depending on R is obtained.

A. Approximation Guarantee

We now provide theoretical guarantees of CRIEG in Theorem 2 given that $\hat{\sigma}$ is a good estimation of σ quantified by Eq. (4).

For any $I \subseteq V$ and $s \in V$, denote $p(I, s) = \sigma(I \cup \{s\}) - \sigma(I)$ and $\hat{p}(I, s) = \hat{\sigma}(I \cup \{s\}) - \hat{\sigma}(I)$; that is, $p(I, s)$ and $\hat{p}(I, s)$ denote the true and estimated marginal gain by adding s into I, respectively. Assume

$$|\hat{p}(I, s) - p(I, s)| \leq \eta p(I, s), \tag{4}$$

where η is a real number in $[0, 1)$ that indicates the relative estimation error.

The proof of Theorem 2 employs the standard analysis of the greedy algorithm, as in [14], [22]. The following lemma justifies the rationale of CRIEG by showing that an inclusion of a greedily selected node can improve σ by at least a quantity proportional to the deficiency of the current iteration in terms of σ, with the proportion depending on the error of approximating the influence function by its empirical average.

Let v^j be the node selected by CRIEG in the j-th iteration and $Q^j = \{v^1, \ldots, v^j\}$, and let h be the total number of iterations executed by CRIEG. For any set $I \subseteq V$ we denote $c(I) = \sum_{v \in I} c(v)$.

Lemma 1 *For any $j \in \{1, \ldots, h\}$, we have*

$$\frac{\sigma(Q^j) - \sigma(Q^{j-1})}{c(v^j)} \geq \frac{1 - \eta}{1 + \eta} \frac{\sigma(Q^*) - \sigma(Q^{j-1})}{c(Q^*)}, \tag{5}$$

where

$$Q^* = \arg\min_{I \subseteq V:\ \sigma(I) \geq \mu/(1-\eta)} c(I). \tag{6}$$

To derive approximation guarantees of CRIEG, we still need a lemma to give a general bound on $c(Q)$ with $Q = \{v^1, \ldots, v^h\}$ provided an auxiliary sequence $\{q^j\}_{j=1,\ldots,h}$ satisfying Eqs. (7) and (8) can be built.

Lemma 2 *Let $\{q^j\}_{j=0,\ldots,l}$ be a decreasing sequence of numbers with $q^0 \geq \epsilon \cdot c(Q^*)$, $\epsilon > 0$ and $q^l \leq 0$. If for any $j \in \{1, \ldots, l\}$, it holds that*

$$q^j \leq \left(1 - \frac{1 - \eta}{1 + \eta} \frac{c(v^j)}{c(Q^*)}\right) q^{j-1} \tag{7}$$

and

$$c(v^j) \leq \frac{q^{j-1} - q^j}{\epsilon}, \tag{8}$$

then

$$c(Q) \leq \left(\frac{1 + \eta}{1 - \eta} \ln \frac{q^0}{\epsilon \cdot c(Q^*)} + 1\right) c(Q^*) - \epsilon^{-1} q^l.$$

Now we present our main result in Theorem 2, showing the bi-criteria approximation guarantee of CRIEG. The proof is accomplished by applying Lemma 2. The conditions, i.e., Eqs. (7) and (8), of Lemma 2 can be verified using Lemma 1 and $\sigma(Q^j) - \sigma(Q^{j-1}) \geq \epsilon c(v^j)$. The detailed proofs are provided in Appendix due to space limitations.

Theorem 2 *Assume $\sigma(Q^*) \geq \epsilon \cdot c(Q^*)$, $\epsilon > 0$, $\forall j : \sigma(Q^j) - \sigma(Q^{j-1}) \geq \epsilon c(v^j)$, and Eq. (4) holds. CRIEG returns a solution Q with $\sigma(Q) \geq \mu/(1+\eta)$ and*

$$c(Q) \leq \left(\frac{1+\eta}{1-\eta}\max\left\{\ln\frac{\min\{\sigma(Q^*),\sigma(Q)\}}{\epsilon\cdot c(Q^*)},0\right\}+1\right)c(Q^*) + \min\left\{\frac{\max\{0,\sigma(Q)-\sigma(Q^*)\}}{\epsilon}, c(v^h)\right\}. \quad (9)$$

The existing best approximation ratio for the greedy algorithm with the precise influence function available is $\ln\frac{\sigma(Q^*)}{\epsilon\cdot c(Q^*)}+1$ [14], which is extended in our discussion to the case with the function σ approximated by its empirical average $\hat{\sigma}$, i.e., from $\eta = 0$ to $\eta \in [0,1)$. Thus, the approximation ratio in Theorem 2 is within a $((1+\eta)/(1-\eta))$-factor of the best existing ratio in [14] if ignoring the last term in Eq. (9), which vanishes if $\sigma(Q) \leq \sigma(Q^*)$. For the case $\sigma(Q) > \sigma(Q^*)$, this additional term on the cost is reasonable since the algorithm returns a solution with larger influence. Fortunately, this additional term can be upper bounded by $c(v^h)$ which can be much smaller than $c(Q)$.

B. Competitive Reverse Influence Estimation Error

In this subsection, we aim to show that Eq. (4) holds with a high probability if a sufficient number of random CRR sets is taken. Our basic idea is to show that the function $p(I,s) - \hat{p}(I,s)$ is the expectation of a random variable minus a sample average of the variable, which allows us to apply the Chernoff bound to derive a probabilistic bound on $|p(I,s) - \hat{p}(I,s)|$. This further quantifies the number of random CRR required to meet the criteria of Theorem 2.

For any graph g and any $I \subseteq V$, denote

$$X_g(I) = \sum_{u\in\overline{H}}\left(\overline{Z}(I, D_g(u)) + \frac{Z(I,D_g(u))\overline{Z}(I,B_g(u))}{\sum_{j=1}^{k}\overline{Z}(I_j,B_g(u))+\overline{Z}(I,B_g(u))}\right).$$

According to Eq. (2), we have

$$\sigma(I\cup\{s\})-\sigma(I) = |(I\cup\{s\})\cap H| + \mathbb{E}(X_g(I\cup\{s\})) - |I\cap H| - \mathbb{E}(X_g(I)).$$

For any graph g, $u \in V$ and $I \subset V$, denote

$$X_{g,u}(I) = \overline{Z}(I,D_g(u)) + \frac{Z(I,D_g(u))\overline{Z}(I,B_g(u))}{\sum_{j=1}^{k}\overline{Z}(I_j,B_g(u))+\overline{Z}(I,B_g(u))}.$$

The following lemma will be used to show that the random variables involved in the reformulation of $p(I,s)-\hat{p}(I,s)$ are non-negative and bounded.

Lemma 3 *For any graph g, $u \in V$ and $I \subseteq V$, $X_{g,u}(I\cup\{s\}) - X_{g,u}(I) \in [0,1]$, i.e.,*

$$\overline{Z}(I\cup\{s\},D_g(u)) - \overline{Z}(I,D_g(u)) + \frac{Z(I\cup\{s\},D_g(u))\overline{Z}(I\cup\{s\},B_g(u))}{\sum_{j=1}^{k}\overline{Z}(I_j,B_g(u))+\overline{Z}(I\cup\{s\},B_g(u))} - \frac{Z(I,D_g(u))\overline{Z}(I,B_g(u))}{\sum_{j=1}^{k}\overline{Z}(I_j,B_g(u))+\overline{Z}(I,B_g(u))} \in [0,1].$$

Proof Three cases are considered as follows.

- If $I\cap D_g(u) \neq \emptyset$, it is $X_{g,u}(I\cup\{s\}) - X_{g,u}(I) = 0$.
- If $I\cap D_g(u) = \emptyset$ and $I\cap B_g(u) = \emptyset$, we have $X_{g,u}(I) = 0$ and thus $X_{g,u}(I\cup\{s\}) - X_{g,u}(I) \in [0,1]$.
- If $I\cap D_g(u) = \emptyset$ and $I\cap B_g(u) \neq \emptyset$, we have $(I\cup\{s\})\cap B_g(u) \neq \emptyset$ and thus

$$X_{g,u}(I) = \frac{1}{\sum_{j=1}^{k}\overline{Z}(I_j,B_g(u))+1}.$$

We further distinguish two subcases by considering whether the set $(I\cup\{s\})\cap D_g(u)$ is empty or not.

- If $(I\cup\{s\})\cap D_g(u) \neq \emptyset$, we have

$$X_{g,u}(I\cup\{s\}) - X_{g,u}(I) = 1 - \frac{1}{\sum_{j=1}^{k}\overline{Z}(I_j,B_g(u))+1} \in [0,1].$$

- If $(I\cup\{s\})\cap D_g(u) = \emptyset$, we have

$$X_{g,u}(I\cup\{s\}) = \frac{1}{\sum_{j=1}^{k}\overline{Z}(I_j,B_g(u))+1},$$

leading to $X_{g,u}(I\cup\{s\}) - X_{g,u}(I) = 0$.

Combining the above three cases, the lemma holds. □

Lemma 4 (Chernoff bound [30]) *Let $X_1,\ldots,X_R$ be independent random variables in $[0,b]$ and $\overline{X} = \sum_{i=1}^{R}X_i/R$, then for any $\eta \in [0,1)$, it holds that*

$$\Pr\left\{|\mathbb{E}(\overline{X}) - \overline{X}| \geq \eta\mathbb{E}(\overline{X})\right\} \leq 2\exp\left(-\frac{R\eta^2\mathbb{E}(\overline{X})}{3b}\right).$$

Now we give a probabilistic bound on $|p(I,s) - \hat{p}(I,s)|$.

Theorem 3 *If $I \subseteq V$ and $s \notin I$, then for any $\eta \in [0,1)$, it holds that*

$$\Pr\left\{|p(I,s) - \hat{p}(I,s)| \geq \eta p(I,s)\right\} \leq 2\exp\left(-\frac{R\eta^2}{3(|\overline{H}|+1)}\right). \quad (10)$$

Proof Let $g_1,\ldots,g_R$ be independent samples of graphs. Thus,

$$p(I,s) = \sigma(I\cup\{s\}) - \sigma(I);$$
$$\hat{p}(I,s) = \frac{1}{R}\sum_{i=1}^{R}\left(|(I\cup\{s\})\cap H| + X_{g_i}(I\cup\{s\}) - |I\cap H| - X_{g_i}(I)\right).$$

We define the random variable

$$X_g(I,s) = |(I\cup\{s\})\cap H| + X_g(I\cup\{s\}) - |I\cap H| - X_g(I),$$

where the graph g is generated by step 2) in Definition 1, i.e., by sampling from G by removing each edge e in G with probability $1-\omega(e)$. Thus, $p(I,s) - \hat{p}(I,s)$ can be written as follows:

$$p(I,s) - \hat{p}(I,s) = \mathbb{E}_g\left(X_g(I,s)\right) - \frac{1}{R}\sum_{i=1}^{R}X_{g_i}(I,s).$$

According to Lemma 3, it holds that

$$0 \leq X_g(I,s) \leq 1 + X_g(I \cup \{s\}) - X_g(I) = 1 + \sum_{u \in \overline{H}} (X_{g,u}(I \cup \{s\}) - X_{g,u}(I)) \leq 1 + |\overline{H}|.$$

It then follows from the Chernoff bound in Lemma 4 that

$$\Pr\{|p(I,s) - \hat{p}(I,s)| \geq \eta p(I,s)\} \leq 2\exp\left(-\frac{R\eta^2 p(I,s)}{3(|\overline{H}|+1)}\right). \tag{11}$$

Since $s \notin I$, by Eq. (3), we have

$$\begin{aligned}\sigma(I \cup \{s\}) - \sigma(I) &= |I \cup \{s\}| - |I| \\ &+ \sum_{u \in \overline{S}} \mathbb{E}_g\left(\frac{[u \in L_g(I \cup \{s\})]}{\sum_{j=1}^{k}[u \in L_g(I_j)] + [u \in L_g(I \cup \{s\})]}\right. \\ &\left. - \frac{[u \in L_g(I)]}{\sum_{j=1}^{k}[u \in L_g(I_j)] + [u \in L_g(I)]}\right) \\ &= 1 + \sum_{u \in \overline{S}} \mathbb{E}_g\left(X_{g,u}(I,s)\right), \end{aligned} \tag{12}$$

where

$$X_{g,u}(I,s) = \frac{[u \in L_g(I \cup \{s\})]}{\sum_{j=1}^{k}[u \in L_g(I_j)] + [u \in L_g(I \cup \{s\})]} - \frac{[u \in L_g(I)]}{\sum_{j=1}^{k}[u \in L_g(I_j)] + [u \in L_g(I)]}.$$

Let u be any element in $\overline{H}$.

- If either $u \in L_g(I)$ or $u \notin L_g(I \cup \{s\})$, $X_{g,u}(I,s) = 0$.
- Otherwise, we have

$$X_{g,u}(I,s) = \frac{1}{\sum_{j=1}^{k}[u \in L_g(I_j)] + 1} \geq \frac{1}{1+k}.$$

Combining the above cases together and using Eq. (12), it can be derived that $\sigma(I \cup \{s\}) - \sigma(I) \geq 1$, and thus $p(I,s) \geq 1$.

Plugging $p(I,s) \geq 1$ back into Eq. (11) yields the stated inequality Eq. (10). Thus, the theorem holds. □

Combining Theorems 2 and 3, the approximation guarantee of CRIEG can be obtained, depending on the estimation error, $\sigma(Q)$, $\sigma(Q^*)$ and ϵ.

Theorem 4 *Suppose the number R of random CRR sets satisfies $R \geq 3\eta^{-2}(|\overline{H}|+1)(3\ln(n)+\ln 2)$, with probability at least $1-1/n$, CRIEG returns a solution with $\sigma(Q) \geq \mu/(1+\eta)$ and*

$$c(Q) \leq \left(\frac{1+\eta}{1-\eta}\max\left\{\ln\frac{\min\{\sigma(Q^*),\sigma(Q)\}}{\epsilon \cdot c(Q^*)}, 0\right\}+1\right)c(Q^*) + \min\left\{\frac{\max\{0,\sigma(Q)-\sigma(Q^*)\}}{\epsilon}, c(v^h)\right\}.$$

VI. Empirical Studies

In this section, experimental studies are conducted on eight real-world data sets to examine the actual performance of CRIEG empirically. The proposed CRIEG is compared with state-of-the-art algorithms from the two categories, i.e., the greedy algorithm based on Monte Carlo simulations that also provides performance guarantees [14], and three efficient heuristic methods, i.e., the Linear-combination Single-hop Spread (LSS) based greedy algorithm [14], Greedy++ [16], and the Dominated Competitive Influence Maximization with Cost-Effective Lazy-Forward (DCIM-CELF) [18].

TABLE I
Dataset Statistics

Dataset	#Node	#Edge	90PED
p2p-Gnutella08	6,301	20,777	5.5
eva	7,253	6,726	5.2
CA-AstroPh	18,772	396,220	5.0
p2p-Gnutella24	26,518	65,369	6.1
Slashdot0902	82,168	948,464	4.7
com-dblp	317,080	1,049,866	8.0
Amazon0312	400,727	3,200,440	7.6
com-youtube	1,134,890	2,987,624	6.5

The experiments aim at illustrating the performance from three aspects: 1) its solution quality in terms of the seed set cost when satisfying the influence threshold constraint; 2) its practical efficiency in terms of the computation time; 3) its sensitivity to different network characteristics including propagation probabilities as well as connection ratios.

A. Experimental Settings

The experiments are conducted on a machine with dual Intel Xeon 2.2GHz CPUs and 128GB of RAM. All algorithms tested are implemented in C++ and complied by the GNU C++ Compiler with the version of 4.8.5.

1) Datasets: A set of eight real-world datasets from the famous SNAP and Pajek Datasets [39], [40] is tested. These datasets are selected from various disciplines so that different network structures can be integrated into account. Among them, *p2p-Gnutella08* is collected from the Gnutella peer-to-peer file sharing network from August 8 2002, and *p2p-Gnutella24* is collected similarly. The data set *eva* is collected from a multidisciplinary research project combining information extraction, information visualization, and social network analysis techniques to bring greater transparency to the public disclosure of inter-relationships between corporations. The *CA-AstroPh* is collected from the e-print arXiv and covers scientific collaborations between authors in the Astro Physics category. *Slashdot0902* is collected from Slashdot Zoo social network from February 2009. *com-dblp* is collect from the co-authorship network where two authors are connected if they publish at least one paper together in a computer science bibliography named DBLP. *Amazon0312* is collected from the Amazon product co-purchasing network from March 12 2003. The data set *com-youtube* is collected from the video-sharing social network of YouTube on January 15th, 2007. On the other hand, to test the scalability of the algorithms and examine their performance as the number of nodes increases, the number of nodes is considered and it ranges from 7,253 to 1,134,890. In addition, the connection ratios of the datasets in terms of the 90-Percentile Effective Diameter (referred to as 90PED) [39] are also considered. Table I summarizes the statistics of the tested networks.

2) Problem settings: The number of other competitors is set to two. Thus, there are three competitors in the experiments.

For generating other competitors, we follow the computation in [14], and assign 15 nodes with maximum out-degree to the two competitors in turn. The cost of choosing a seed is uniformly set to 1, i.e., $\forall v \in V, c(v) = 1$. For computing the edge propagation probabilities, the conventional weighted cascade model [41] is adopted. The probability of an edge $e = (u, v)$ is set to $\omega(e) = 1/d(v)$, where $d(v)$ indicates the in-degree of v. In addition, the propagation probabilities randomly generated within $[0, 1]$ are also tested.

3) Algorithms and Parameter Settings: Four state-of-the-art algorithms in terms of solution quality or efficiency are compared in the experiments. The first one is the Monte Carlo simulation-based algorithm (referred to as MC-Greedy) [14]. As mentioned above, such algorithm can produce solutions with non-trivial approximation guarantees. Hence, the MC-Greedy provides a near-optimal reference for the solution quality. The second is the heuristic LSS [14]. To the best of our knowledge, it is the previously best in terms of efficiency under the competitive independent cascade model, and thus it can be used to verify the efficiency of CRIEG. Two recent heuristics, i.e., Greedy++ [16] and DCIM-CELF [18], are also tested under the competitive independent cascade model in the experiments. For the parameters defined for the compared algorithms, the recommended values in their corresponding papers are used if available. In particular, when Monte Carlo simulation is needed, 1,000 independent runs are conducted and the average is returned. In LSS, the number of potential nodes specifies the number of high-degree nodes that are considered only when selecting seeds. As this value is not specified by its authors, it is set to half of the number of nodes to make a trade-off between computational load and solution quality. In CRIEG, the sample size R of CRR sets is set to 10^6 to examine its empirical performance on real-world datasets. The impact of the value of R is also analyzed in the experiments. Each method is repeated 10 times and the average results are reported. The running time of each method in a run is also limited to be within 48 hours.

B. Experimental Results

In this subsection, the performance of CRIEG is examined in terms of efficiency as well as the quality of the obtained seed set. Specifically, we first examine how increasing the scale of the tested graph affects the performance of CRIEG and the compared algorithms, and show the advantages of CRIEG in both efficiency and the quality of the obtained seed set. Second, the effects of propagation probabilities and connection ratios are examined and analyzed for more in-depth understanding. Third, the effect of the sample size R on the performance of CRIEG is examined, demonstrating that it can be more efficient in practice than in theory.

1) Solution Quality and Running Time: The results of the seed set cost and computational running time obtained by MC-Greedy, LSS and CRIEG when using weighted propagation probabilites are shown in Figs. 2–4 and those obtained using random probabilities are shown in Figs. 5–6. The results of Greedy++ and DCIM-CELF are shown in Fig. 7. In particular, as Greedy++ and DCIM-CELF cannot provide influence guarantees since they do not examine whether the influence exceeds predefined thresholds, they are compared by specifying the seed cost in advance.

Generally, it can be observed that CRIEG is always among the best algorithms in seed cost, and exhibits a good scalability to large-scale graphs with high efficiency. As shown in Fig. 2, CRIEG is up to two orders of magnitude faster than MC-Greedy, while still achieving competitive seed set cost. The results in Fig. 4 further show that CRIEG can scale beyond million-sized graphs where MC-Greedy becomes infeasible due to prohibitive computation overheads. For example, when applied to *p2p-Gnutella08*, MC-Greedy needs more than 40 hours to achieve the influence threshold 800, whereas CRIEG needs less than five minutes.

On the other hand, CRIEG not only consistently outperforms LSS in terms of the solution quality (i.e., seed cost), but also scales better to large-scale networks in terms of the running time. As shown in Fig. 4 and Fig. 6, although CRIEG needs more time on small-scale networks than LSS, its advantages emerge with the increasing network size, which is partly because of the tradeoff mechanism between running time and solution quality. Specifically, LSS adopts a fast but coarse method for estimating the influence when selecting the next seed in each iteration, thus its speed can be fast in this period. However, in order to achieve a better tradeoff, LSS also adopts the costly Monte Carlo simulations for the influence validation for each selected candidate set. While this allows LSS to provide a good tradeoff between solution quality and running time on small-scale networks, its performance may deteriorate rapidly as the network size increases, since more nodes are then needed to satisfy the influence constraint due to the coarse estimation. In contrast, CRIEG needs fewer nodes to satisfy the influence constraint thanks to the more precise estimation. Take a look at the *com-youtube*, which has 1.13 million nodes and 2.99 million edges, LSS needs more than 30 hours to return a solution with the influence exceeding 10,000, whereas CRIEG returns a better solution within 15 seconds.

As compared to Greedy++ and DCIM-CELF, the results in Fig. 7 indicate that CRIEG always achieves larger influence for the same seed cost. Besides, unlike the inefficiency of the two algorithms when faced with large-scale networks due to prohibitive computation overheads, CRIEG can scale beyond million-sized networks, as shown in Figs. 3–6. Thus, it is shown that CRIEG is not only provable effective, but also presents an empirical efficiency.

In the following, to gain an in-depth understanding of the performance, the effects of different network sizes are further analyzed in detail. For the relatively small-scale graphs, i.e., *p2p-Gnutella08* and *eva*, where MC-Greedy is feasible to run, the seed cost results in Fig. 2 show that CRIEG and MC-Greedy consistently achieve similar results. These results are consistent with the theoretical analysis in Section IV-A that the competitive reverse influence estimation can precisely measure the expectation of competitive influence. Besides, it can be found that the running time of MC-Greedy is much higher than that of CRIEG, implying that most of the repeated samplings for estimating the expected spread of each potential node sets of MC-Greedy might be wasting. Hence, by sharing the results

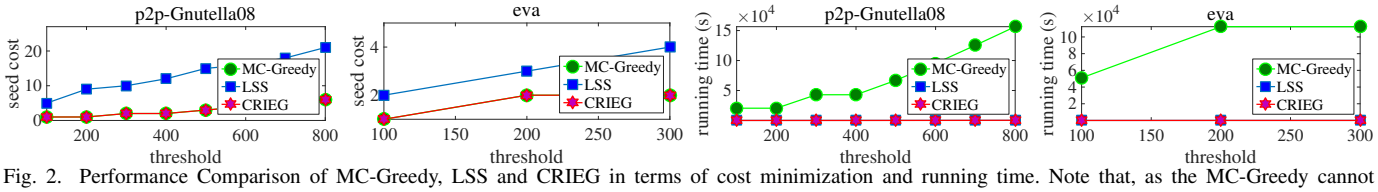

Fig. 2. Performance Comparison of MC-Greedy, LSS and CRIEG in terms of cost minimization and running time. Note that, as the MC-Greedy cannot produce a solution that satisfies the influence threshold within the time allowed, i.e., 48 hours, on large-scale networks, only the cases where it finds the solution are presented.

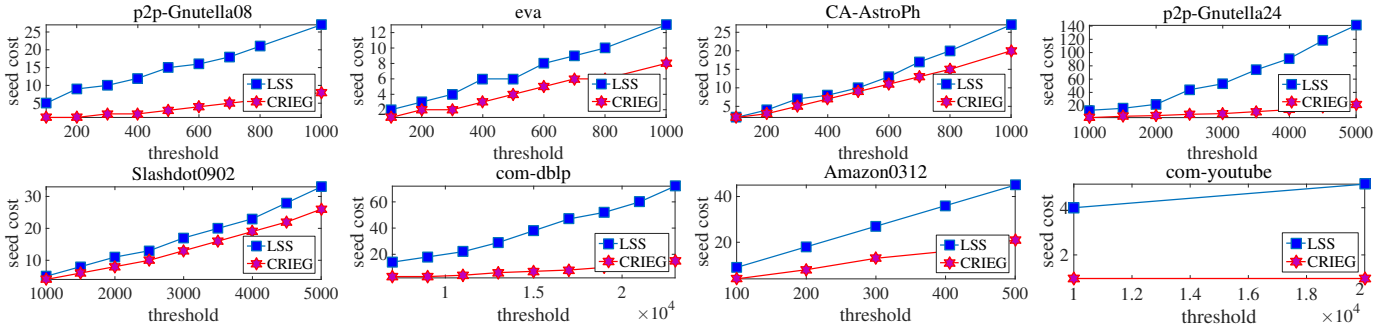

Fig. 3. Seed cost of LSS and CRIEG. MC-Greedy is omitted here since it cannot find a solution due to prohibitive computation overheads. The thresholds examined are also restricted for the same reason and thus there are fewer points on the outcome curves of networks with larger scales.

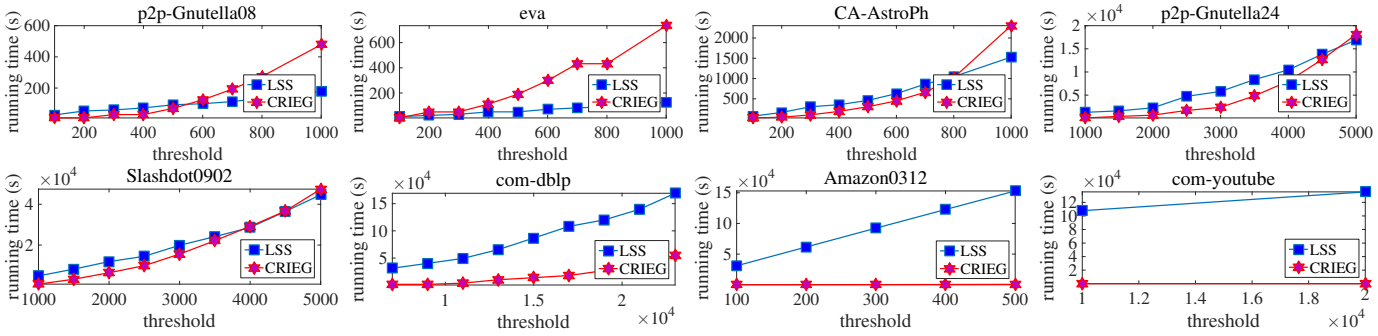

Fig. 4. Running time of LSS and CRIEG. MC-Greedy is omitted here since it cannot find a solution due to prohibitive computation overheads. The thresholds examined are also restricted for the same reason and thus there are fewer points on the outcome curves of networks with larger scales.

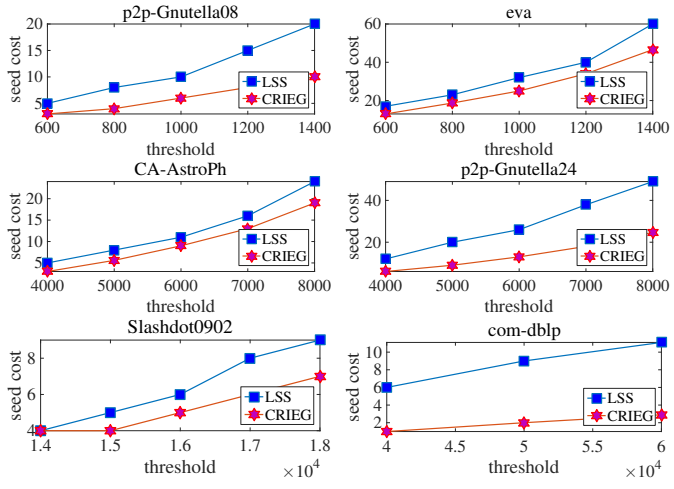

Fig. 5. Seed cost of LSS and CRIEG when using random propagation probabilities. MC-Greedy is omitted here since it cannot find a solution due to prohibitive computation overheads. For the same reason, two large-scale datasets are omitted here, and the thresholds examined are restricted and thus there are fewer points on the outcome curves of networks with larger scales.

of Monte Carlo simulations, the reverse sampling technology can be more efficient, as already demonstrated in the special case when the number of competitors is one [1], [27]. This indicates that the competitive reverse influence estimation can be a good option for estimating competitive influence. On the other hand, although LSS performs best in running time on the small-scale graphs, its seed cost is worse than both CRIEG and MC-Greedy, verifying that the single-hop linear-combination estimation of LSS compromises the accuracy of competitive influence severely.

For the moderate sized graphs, i.e., *CA-AstroPh*, *p2p-Gnutella24*, *Slashdot0902*, as MC-Greedy incurs prohibitive computation overheads and cannot be run out within 48 hours, it is omitted from Figs. 3–4. It can be observed that LSS still performs worse than CRIEG in terms of the solution quality, consistent with LSS's main idea of trading performance guarantees for practical efficiency. However, unlike the fast running on the small-scale graphs, the efficiency of LSS deteriorates rapidly as the scale of graphs increases. In fact, LSS becomes slightly worse than CRIEG when applied to *p2p-Gnutella24*

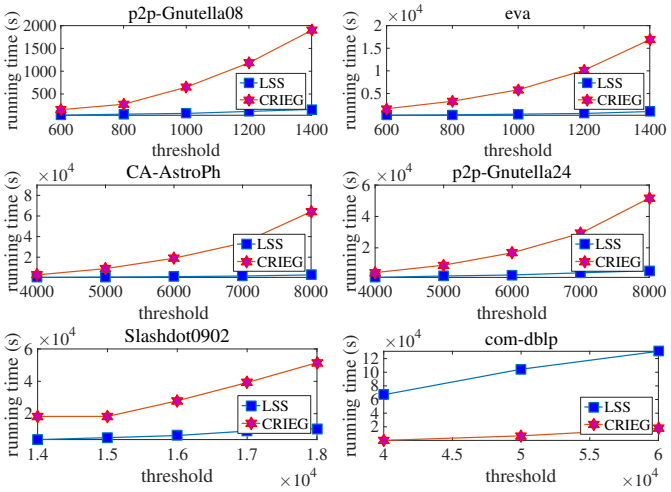

Fig. 6. Running time of LSS and CRIEG when using random propagation probabilities. MC-Greedy is omitted here since it cannot find a solution due to prohibitive computation overheads. For the same reason, two large-scale datasets are omitted here, and the thresholds examined are restricted and thus there are fewer points on the outcome curves of networks with larger scales.

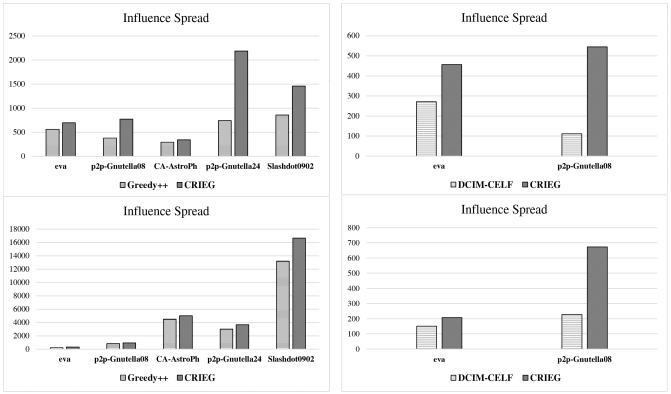

Fig. 7. Influence spread of Greedy++ (left), DCIM-CELF (right) and CRIEG. Due to prohibitive computation overheads, the number of seed size is set to 5 for Greedy++ and 3 for DCIM-CELF. The top are the results using the weighted cascade model and the bottom are results using random probabilities.

and *Slashdot0902* with a relatively large threshold. Yet, this is not surprising, as LSS uses Monte Carlo simulations as the stopping criterion for each iteration to alleviate the inaccurate estimation of the heuristic mechanism. Since Monte Carlo simulation is time consuming, especially when the scale of a graph increases, LSS requires more time in each iteration, leading to a deterioration in efficiency. For graphs with larger scale, this issue becomes more severe, and further erodes the efficiency of LSS.

For the large-scale graphs, i.e., *com-dblp*, *Amazon0312* and *com-youtube*, CRIEG consistently achieves better performance in terms of both the solution quality and running time than LSS. Further taking into account MC-Greedy, which is omitted due to prohibitive computation overheads, these results suggest that CRIEG scales best to the large-scale graphs. Note that the graph *com-youtube*, which has 1.13 million nodes and 2.99 million edges, is the largest dataset ever used for the competitive independent cascade model in the literature. The results on *com-youtube* show that our algorithm can efficiently handle graphs with millions of nodes and edges, and also verify the efficiency of the RIS mechanism in a competitive environment.

2) Sensitivity analyses with respect to network characteristics: In addition to network sizes, propagation probabilities and connection ratios are two important aspects of networks which affect the intensity and depth of a diffusion process. In the following, their effects are analyzed respectively.

For propagation probabilities, two models have been examined, i.e., the weighted cascade model that assigns the probability of an edge (u, v) to $1/d(v)$, where $d(v)$ indicates the in-degree of v, and one that generates probabilities within $[0, 1]$ uniformly at random. It is worth noting that the latter is more likely to have a higher probability of activating a node when examined on real-world networks, since they are more likely to consist of nodes with large in-degree due to complex interactions. Indeed, the results in Figs. 3–7 suggest that larger probabilities are more likely to lead to a larger influence given seed sets with the same seed cost, which might result in more running time. For example, as shown in Fig. 4 and Fig. 6, when examined on *p2p-Gnutella08* with seed cost 8, the running time of CRIEG is about 479s under the weighted cascade model, but more than 1000s when using random propagation probabilities. On the other hand, when compared with LSS, as CRIEG relies on CRR sets sampled from networks and LSS adopts a heuristic, the running time of CRIEG can be more affected by the increasing of propagation probabilities than LSS. In spite of that, the advantages of CRIEG still emerge with the increasing network size under the random propagation probabilities, as that shown under the weighted cascade model. It is shown by *com-dblp* in Fig. 4 and Fig. 6 that CRIEG consistently outperforms LSS under the two probability models in terms of the solution quality.

For connection ratios, the 90PED which measures the 90-th percentile of undirected shortest path length distribution (sampled over 1,000 random nodes) [39] is adopted. The 90PED values of the examined graphs are summarized in Table I. In general, the results suggest that this network statistic can have an important effect on the running time of CRIEG as well as the compared LSS. As for CRIEG, a larger 90PED is more likely to result in sampling deeper diffusion paths, thus leading to more running time. As for LSS, although its diffusion paths are almost unaffected since it simplifies the diffusion process by only considering a few hops of the diffusion paths, its running time is also likely to increase rapidly with the increase of 90PED. This is because its estimate would become farther from the precise influence and thus may need more iterations to select more seeds. Thus, it is more likely that the running time of LSS would exceed that of CRIEG on graphs with larger 90PED. On the other hand, CRIEG consistently outperforms LSS on seed cost and the improvement is likely to increase with 90PED.

3) Varying sample size R: The number of CRR sets R is a tunable parameter in achieving a balance between efficiency and accuracy for CRIEG. According to the theoretical analysis in Section V, CRIEG requires the number R of CRR sets to be $3\eta^{-2}(|\overline{H}| + 1)(3\ln(n) + \ln 2)$, as shown in Theorem 4.

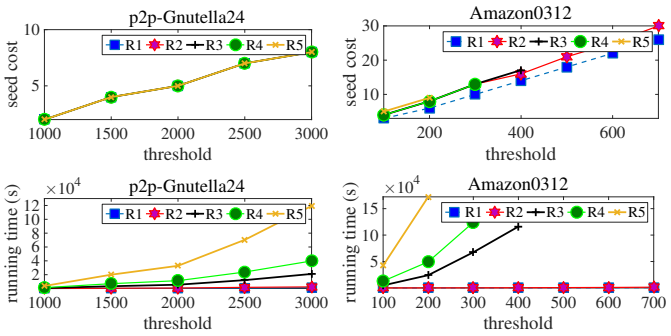

Fig. 8. Seed cost and running time for varying R. The dashed line indicates that the points on it fail to meet the corresponding threshold constraint on the horizontal axis. Besides, some curves are truncated here since the corresponding algorithm cannot find a solution due to prohibitive computation overheads.

Since this is a theoretical upper bound with worst-case quality guarantees, the value of R that makes CRIEG an empirically good balance between efficiency and accuracy is studied in this part.

Two datasets with different scales are used in this experiment, i.e., *p2p-Gnutella24* and *Amazon0312*. Five values of R are tested, i.e., $R1 = 5.0 \times 10^5$, $R2 = 1.0 \times 10^6$, and the theoretical sample size $R3$ with $\eta = 0.7$, $R4$ with $\eta = 0.5$, and $R5$ with $\eta = 0.3$. Fig. 8 shows the results of seed cost and running time. The results indicate that when decreasing the value of η, the influence of the obtained seed set can be more guaranteed towards the influence threshold, which might result in slightly larger seed costs, but at the same time, more running time is needed, which is consistent with the theoretical analyses. Besides, it can also be observed that CRIEG generally achieves similar results to the theoretical case, while when $R1$ is used, it performs slightly worse on *Amazon0312*. This implies that CRIEG can be more efficient in practice than in theory. Actually, $R2$ is adopted for the experiments in Section VI.

VII. Conclusion

In this paper, the CRIEG algorithm is proposed for the minimum cost seed selection problem for CIM under the competitive independent cascade model. CRIEG can efficiently handle networks with size up to millions of users, but also provide bounded performance guarantees. It is shown that with probability at least $1 - 1/n$, CRIEG can achieve an approximation ratio within $((1+\eta)/(1-\eta))$-factor of the best existing ratio under the competitive independent cascade model derived in the setting with the precise influence function available [14], where $\eta \in [0, 1)$ indicates the relative estimation error. The core of CRIEG is the competitive reverse influence estimation mechanism, which borrows the idea from the powerful RIS technology but overcomes its unavailability in competitive settings. The experimental studies on various real-world networks show that compared with the state-of-the-art algorithms, CRIEG can be orders of magnitude faster while consistently offering the best performance.

There are several future directions. First, the competitive reverse influence estimation is derived from the competitive independent cascade model, and it is interesting to examine it for other diffusion models. Second, it is relevant to study the parallelization of the algorithm so that it can be better used in real-world applications. Third, it is expected to extend CRIEG to other formulations of CIM, e.g., combining with novel context features of social networks such as topic, location and time. Fourth, as CRIEG is based on the greedy algorithm, it cannot ensure global optimality since it always makes a locally-optimal choice at a given point. Furthermore, it can be sensitive when addressing noisy influence evaluation function in general [42], [43]. Thus, it is expected to combine CRIEG with some more advanced search mechanisms such as evolutionary algorithms [44], [45] for better performance and noise tolerance.

Appendix

Proof of Lemma 1 According to Eq. (4), $|\hat{p}(Q^{j-1}, v^j) - p(Q^{j-1}, v^j)| \leq \eta p(Q^{j-1}, v^j)$, leading to

$$(1-\eta)p(Q^{j-1}, v^j) \leq \hat{p}(Q^{j-1}, v^j) \leq (1+\eta)p(Q^{j-1}, v^j). \tag{13}$$

By the greedy choice, it holds that

$$\frac{\sigma(Q^j) - \sigma(Q^{j-1})}{c(v^j)} = \frac{\sigma(Q^{j-1} \cup \{v^j\}) - \sigma(Q^{j-1})}{c(v^j)}$$
$$= \frac{p(Q^{j-1}, v^j)}{c(v^j)} \geq \frac{1}{1+\eta}\frac{\hat{p}(Q^{j-1}, v^j)}{c(v^j)} \geq \frac{1}{1+\eta}\frac{\hat{p}(Q^{j-1}, \bar{v}^j)}{c(\bar{v}^j)}$$
$$\geq \frac{1-\eta}{1+\eta}\frac{p(Q^{j-1}, \bar{v}^j)}{c(\bar{v}^j)},$$

where $\bar{v}^j = \arg\max_s \frac{p(Q^{j-1}, s)}{c(s)}$. Let $Q^* = \{x^1, \ldots, x^L\}$. Then,

$$\frac{\sigma(Q^j) - \sigma(Q^{j-1})}{c(v^j)} \geq \frac{1-\eta}{1+\eta}\max_{1\leq i\leq L}\frac{p(Q^{j-1}, x^i)}{c(x^i)}$$
$$\geq \frac{1-\eta}{1+\eta}\frac{\sum_{i=1}^{L} p(Q^{j-1}, x^i)}{c(Q^*)}$$
$$\geq \frac{1-\eta}{1+\eta}\frac{\sigma(Q^* \cup Q^{j-1}) - \sigma(Q^{j-1})}{c(Q^*)}$$
$$\geq \frac{1-\eta}{1+\eta}\frac{\sigma(Q^*) - \sigma(Q^{j-1})}{c(Q^*)},$$

where the third inequality holds by

$$\sum_{i=1}^{L}\left(\sigma(Q^{j-1} \cup x^i) - \sigma(Q^{j-1})\right)$$
$$\geq \sum_{i=1}^{L}\left(\sigma(Q^{j-1} \cup A_i) - \sigma(Q^{j-1} \cup A_{i-1})\right)$$
$$= \sigma(Q^{j-1} \cup Q^*) - \sigma(Q^{j-1})$$

due to the submodularity of σ and $A_i = \{x^1, \ldots, x^i\}$, and the fourth inequality holds due to the monotonicity of σ. Thus, the lemma holds. □

Proof of Lemma 2 An integer r can be found such that

$$q^{r+1} < \epsilon \cdot c(Q^*) \leq q^r.$$

Let $q' = \epsilon \cdot c(Q^*) - q^{r+1}, q'' = q^r - \epsilon \cdot c(Q^*)$,

$$c' = \frac{q' c(v^{r+1})}{q^r - q^{r+1}} \quad \text{and} \quad c'' = \frac{q'' c(v^{r+1})}{q^r - q^{r+1}}. \tag{14}$$

Thus, we have

$$\frac{q^r - q^{r+1}}{c(v^{r+1})} = \frac{q' + q''}{c' + c''} = \frac{q'}{c'} = \frac{q''}{c''}. \tag{15}$$

By Eqs. (7) and (15), it holds that

$$\frac{q^r - \epsilon \cdot c(Q^*)}{c''} = \frac{q''}{c''} = \frac{q^r - q^{r+1}}{c(v^{r+1})} \geq \frac{1-\eta}{1+\eta}\frac{q^r}{c(Q^*)}.$$

Thus, we have

$$\begin{aligned}
\epsilon \cdot c(Q^*) &\leq \left(1 - \frac{1-\eta}{1+\eta}\frac{c''}{c(Q^*)}\right) q^r \\
&\leq \left(1 - \frac{1-\eta}{1+\eta}\frac{c''}{c(Q^*)}\right) q^0 \prod_{i=1}^{r}\left(1 - \frac{1-\eta}{1+\eta}\frac{c(v^i)}{c(Q^*)}\right) \\
&\leq q^0 \exp\left(-\frac{1-\eta}{1+\eta}\frac{c''}{c(Q^*)}\right) \exp\left(-\frac{1-\eta}{(1+\eta)c(Q^*)}\sum_{i=1}^{r} c(v_i)\right) \\
&= q^0 \exp\left(-\frac{1-\eta}{1+\eta}\frac{c'' + \sum_{i=1}^{r} c(v^i)}{c(Q^*)}\right),
\end{aligned} \tag{16}$$

where the second inequation is by Eq. (7) and the third is by $e^x \geq (1+x)$. Eq. (16) implies that

$$c'' + \sum_{i=1}^{r} c(v^i) \leq \frac{(1+\eta)c(Q^*)}{1-\eta} \ln \frac{q^0}{\epsilon \cdot c(Q^*)}. \tag{17}$$

Since $\frac{q'}{c'} = \frac{q^r - q^{r+1}}{c(v^{r+1})} \geq \epsilon$ due to Eq. (8), it can be known from the definition of q' that

$$c' \leq \frac{q'}{\epsilon} = \frac{\epsilon \cdot c(Q^*) - q^{r+1}}{\epsilon}. \tag{18}$$

According to Eq. (8), it holds that

$$\sum_{i=r+2}^{l} c(v^i) \leq \frac{1}{\epsilon}\sum_{i=r+2}^{l}\left(q^{i-1} - q^i\right) = \frac{1}{\epsilon}\left(q^{r+1} - q^l\right). \tag{19}$$

Combining Eqs. (14), (17)–(19) together, we have

$$\begin{aligned}
c(Q) &= \sum_{i=1}^{l} c(v^i) = \sum_{i=1}^{r} c(v^i) + c(v^{r+1}) + \sum_{i=r+2}^{l} c(v^i) \\
&= \sum_{i=1}^{r} c(v^i) + c'' + c' + \sum_{i=r+2}^{l} c(v^i) \\
&\leq \frac{(1+\eta)c(Q^*)}{1-\eta} \ln \frac{q^0}{\epsilon \cdot c(Q^*)} + \frac{\epsilon \cdot c(Q^*) - q^{r+1}}{\epsilon} + \frac{q^{r+1} - q^l}{\epsilon} \\
&= \frac{(1+\eta)c(Q^*)}{1-\eta} \ln \frac{q^0}{\epsilon \cdot c(Q^*)} + c(Q^*) - \epsilon^{-1} q^l,
\end{aligned}$$

i.e., the lemma holds. □

Proof of Theorem 2 If $c(Q) < c(Q^*)$, it is clear that Eq. (9) holds. In the following, we consider the case when $c(Q) \geq c(Q^*)$. As $\forall j : \sigma(Q^j) - \sigma(Q^{j-1}) \geq \epsilon c(v^j)$, $\sigma(Q) \geq \epsilon c(Q)$, and thus $\sigma(Q) \geq \epsilon c(Q^*)$. Therefore, $(\min\{\sigma(Q^*), \sigma(Q)\})/(\epsilon c(Q^*)) \geq 1$, implying that $\ln((\min\{\sigma(Q^*), \sigma(Q)\})/(\epsilon c(Q^*))) \geq 0$. In this case, the Eq. (9) to be proved becomes

$$\begin{aligned}
c(Q) \leq &\left(\frac{1+\eta}{1-\eta} \ln \frac{\min\{\sigma(Q^*), \sigma(Q)\}}{\epsilon \cdot c(Q^*)} + 1\right) c(Q^*) \\
&+ \min\left\{\frac{\max\{0, \sigma(Q) - \sigma(Q^*)\}}{\epsilon}, c(v^h)\right\}.
\end{aligned} \tag{20}$$

We first use Lemma 2 to show

$$c(Q) \leq \left(\frac{1+\eta}{1-\eta} \ln \frac{\min\{\sigma(Q^*), \sigma(Q)\}}{\epsilon \cdot c(Q^*)} + 1\right) c(Q^*) + c(v^h). \tag{21}$$

It can be assumed that $c(Q^{h-1}) \geq c(Q^*)$ since the above inequality is trivial otherwise. According to Eq. (13), we have

$$\begin{aligned}
\sigma(Q^{h-1}) &= \sigma(Q^{h-2} \cup \{v^{h-1}\}) - \sigma(Q^{h-2}) + \sigma(Q^{h-2}) \\
&= \sum_{i=1}^{h-1} p(Q^{i-1}, v^i) + \sigma(Q^0) \leq \frac{1}{(1-\eta)} \sum_{i=1}^{h-1} \hat{p}(Q^{i-1}, v^i) \\
&= \frac{1}{(1-\eta)} \hat{\sigma}(Q^{h-1}) \leq \frac{1}{(1-\eta)} \mu \leq \sigma(Q^*),
\end{aligned}$$

where the last two inequalities hold by the implementation of Algorithm 1 and the definition of Q^* in Lemma 1. We define

$$q^j = \sigma(Q^{h-1}) - \sigma(Q^j), \quad j \in \{0, \ldots, h-1\}.$$

Note that the parameter l in Lemma 2 equals to $(h-1)$ here. According to Lemma 1 and $\sigma(Q^*) \geq \sigma(Q^{h-1})$, we have

$$\begin{aligned}
\frac{q^{j-1} - q^j}{c(v^j)} &= \frac{\sigma(Q^j) - \sigma(Q^{j-1})}{c(v^j)} \geq \frac{1-\eta}{1+\eta}\frac{\sigma(Q^*) - \sigma(Q^{j-1})}{c(Q^*)} \\
&\geq \frac{1-\eta}{1+\eta}\frac{\sigma(Q^{h-1}) - \sigma(Q^{j-1})}{c(Q^*)} = \frac{1-\eta}{1+\eta}\frac{q^{j-1}}{c(Q^*)},
\end{aligned} \tag{22}$$

implying that Eq. (7) holds. Furthermore, it is clear that

$$q^{j-1} - q^j = \sigma(Q^j) - \sigma(Q^{j-1}) \geq \epsilon c(v^j), \tag{23}$$

implying that Eq. (8) holds. According to Eq. (23), we have

$$q^0 = \sigma(Q^{h-1}) = \sum_{j=1}^{h-1}\left(q^{j-1} - q^j\right) \geq \epsilon c(Q^*) \quad \text{and} \quad q^{h-1} = 0,$$

where the inequality holds by $c(Q^{h-1}) \geq c(Q^*)$. Thus, we can apply Lemma 2 with $q^0 \leq \min\{\sigma(Q^*), \sigma(Q)\}$ to derive

$$c(Q^{h-1}) \leq \left(\frac{1+\eta}{1-\eta} \ln \frac{\min\{\sigma(Q^*), \sigma(Q)\}}{\epsilon \cdot c(Q^*)} + 1\right) c(Q^*),$$

from which we can immediately derive Eq. (21).

We next use Lemma 2 to prove

$$\begin{aligned}
c(Q) \leq &\left(\frac{1+\eta}{1-\eta} \ln \frac{\min\{\sigma(Q^*), \sigma(Q)\}}{\epsilon \cdot c(Q^*)} + 1\right) c(Q^*) \\
&+ \frac{\max\{0, \sigma(Q) - \sigma(Q^*)\}}{\epsilon}
\end{aligned} \tag{24}$$

by considering two cases.

We first consider the case $\sigma(Q^*) \geq \sigma(Q)$. In this case, we define

$$q^j = \sigma(Q) - \sigma(Q^j), \quad j \in \{0, \ldots, h\}.$$

According to Lemma 1 and $\sigma(Q^*) \geq \sigma(Q)$, we have

$$\begin{aligned}
\frac{q^{j-1} - q^j}{c(v^j)} &= \frac{\sigma(Q^j) - \sigma(Q^{j-1})}{c(v^j)} \geq \frac{1-\eta}{1+\eta}\frac{\sigma(Q^*) - \sigma(Q^{j-1})}{c(Q^*)} \\
&\geq \frac{1-\eta}{1+\eta}\frac{\sigma(Q) - \sigma(Q^{j-1})}{c(Q^*)} = \frac{1-\eta}{1+\eta}\frac{q^{j-1}}{c(Q^*)},
\end{aligned}$$

implying that Eq. (7) holds. Similar to the analysis of Eq. (23), we can derive that

$$q^{j-1} - q^j = \sigma(Q^j) - \sigma(Q^{j-1}) \geq \epsilon c(v^j), \tag{25}$$

implying that Eq. (8) holds. Furthermore, according to Eq. (25) and $c(Q) \geq c(Q^*)$, we have

$$q^0 = \sigma(Q) = \sum_{j=1}^{h} \left(q^{j-1} - q^j\right) \geq \epsilon c(Q^*) \quad \text{and} \quad q^h = 0.$$

Note that the parameter l in Lemma 2 equals to h here. Thus, we can apply Lemma 2 to derive

$$c(Q) \leq \left(\frac{1+\eta}{1-\eta} \ln \frac{q^0}{\epsilon \cdot c(Q^*)} + 1\right) c(Q^*). \tag{26}$$

We now consider the case $\sigma(Q^*) < \sigma(Q)$. In this case, we define

$$q^j = \sigma(Q^*) - \sigma(Q^j), \quad j \in \{0, \ldots, h\}.$$

According to Lemma 1, we derive

$$\begin{aligned} \frac{q^{j-1} - q^j}{c(v^j)} &= \frac{\sigma(Q^j) - \sigma(Q^{j-1})}{c(v^j)} \\ &\geq \frac{1-\eta}{1+\eta} \frac{\sigma(Q^*) - \sigma(Q^{j-1})}{c(Q^*)} = \frac{1-\eta}{1+\eta} \frac{q^{j-1}}{c(Q^*)}, \end{aligned}$$

implying that Eq. (7) holds. It is easy to see that as Eq. (23), Eq. (8) still holds. Furthermore, $\sigma(Q^*) \geq \epsilon c(Q^*)$ and $\sigma(Q^*) < \sigma(Q)$ imply that

$$q^0 = \sigma(Q^*) \geq \epsilon c(Q^*) \quad \text{and} \quad q^h = \sigma(Q^*) - \sigma(Q) \leq 0.$$

Thus, Lemma 2 can be applied to derive

$$c(Q) \leq \left(\frac{1+\eta}{1-\eta} \ln \frac{q^0}{\epsilon \cdot c(Q^*)} + 1\right) c(Q^*) - \frac{q^h}{\epsilon}. \tag{27}$$

Eq. (24) then follows from Eqs. (26) and (27).

The bounds Eqs. (21) and (24) can be written compactly as Eq. (20), and thus Eq. (9) holds. By Eq. (13), it holds that

$$\begin{aligned} \sigma(Q) &= \sigma(Q^{h-1} \cup \{v^h\}) - \sigma(Q^{h-1}) + \sigma(Q^{h-1}) \\ &= \sum_{i=1}^{h} p(Q^{i-1}, v^i) + \sigma(Q^0) \\ &\geq \frac{1}{1+\eta} \sum_{i=1}^{h} \hat{p}(Q^{i-1}, v^i) = \frac{\hat{\sigma}(Q)}{1+\eta} \geq \frac{\mu}{1+\eta}. \end{aligned}$$

Thus, the theorem holds. □

Proof of Theorem 4 According to Theorem 2, if Eq. (4) holds for the evaluation during the running of CRIEG, then the greedy algorithm achieves the approximation guarantee shown in Eq. (9). According to Theorem 3 and $R \geq 3\eta^{-2}(|\overline{H}| + 1)(3\ln(n) + \ln 2)$, the probability for one seed set to violate is at most $1/(n^3)$. During the running of CRIEG, we estimate the influence spread for at most nh seed sets, where h indicates the size of the output solution of the algorithm. By the union bound, we have the probability of at most $1/n$ for some estimate to violate the error bound. Thus, the theorem holds. □

REFERENCES

[1] Y. Li, J. Fan, Y. Wang, and K.-L. Tan, "Influence maximization on social graphs: A survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 10, pp. 1852–1872, 2018.

[2] H. Hotelling, "Stability in competition," in *The Collected Economics Articles of Harold Hotelling*. Springer, 1990, pp. 50–63.

[3] Y. Yu, J. Jia, D. Li, and Y. Zhu, "Fair multi-influence maximization in competitive social networks," in *Proceedings of the 12th International Conference on the Wireless Algorithms, Systems, and Applications*, Guilin, China, 2017, pp. 253–265.

[4] F. Zhou, R. J. Jiao, and B. Lei, "Bilevel game-theoretic optimization for product adoption maximization incorporating social network effects," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 8, pp. 1047–1060, 2016.

[5] M. Vergeer and L. Hermans, "Campaigning on Twitter: Microblogging and online social networking as campaign tools in the 2010 general elections in the Netherlands," *Journal of Computer-Mediated Communication*, vol. 18, no. 4, pp. 399–419, 2013.

[6] J. Kostka, Y. A. Oswald, and R. Wattenhofer, "Word of mouth: Rumor dissemination in social networks," in *Structural Information and Communication Complexity*, Berlin, Germany, 2008, pp. 185–196.

[7] X. He, G. Song, W. Chen, and Q. Jiang, "Influence blocking maximization in social networks under the competitive linear threshold model," in *Proceedings of the 12th SIAM International Conference on Data Mining*, Anaheim, CA, 2012, pp. 463–474.

[8] L. Fan, Z. Lu, W. Wu, B. Thuraisingham, H. Ma, and Y. Bi, "Least cost rumor blocking in social networks," in *Proceedings of the 33rd International Conference on Distributed Computing Systems*, Philadelphia, PA, 2013, pp. 540–549.

[9] S. Li, Y. Zhu, D. Li, D. Kim, and H. Huang, "Rumor restriction in online social networks," in *Proceedings of the 32nd IEEE International Performance Computing and Communications Conference*, San Diego, CA, 2013, pp. 1–10.

[10] H. Zhang, H. Zhang, A. Kuhnle, and M. T. Thai, "Profit maximization for multiple products in online social networks," in *Proceedings of the 35th Annual IEEE International Conference on Computer Communications*, San Francisco, CA, 2016, pp. 1–9.

[11] M. Brede, M. Stella, and A. C. Kalloniatis, "Competitive influence maximization and enhancement of synchronization in populations of non-identical Kuramoto oscillators," *Scientific Reports*, vol. 8, no. 702, 2018.

[12] W. Li, Q. Bai, M. Zhang, and T. D. Nguyen, "Modelling multiple influences diffusion in on-line social networks," in *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*. Stockholm, Sweden: International Foundation for Autonomous Agents and Multiagent Systems, 2018, pp. 1053–1061.

[13] G. Gao, M. Xiao, J. Wu, H. Huang, and G. Chen, "Minimum cost seed selection for multiple influences diffusion in communities," in *Proceedings of the 15th IEEE International Conference on Mobile Ad Hoc and Sensor Systems*, Chengdu, China, 2018, pp. 263–271.

[14] Y. Zhu, D. Li, and Z. Zhang, "Minimum cost seed set for competitive social influence," in *Proceedings of the 35th Annual IEEE International Conference on Computer Communications*, San Francisco, CA, 2016, pp. 1–9.

[15] N. Du, Y. Liang, M.-F. Balcan, M. Gomez-Rodriguez, H. Zha, and L. Song, "Scalable influence maximization for multiple products in continuous-time diffusion networks," *Journal of Machine Learning Research*, vol. 18, no. 2, pp. 1–45, 2017.

[16] C. V. Pham, N. V. Nguyen, T. X. Le, and H. X. Hoang, "Competitive influence maximization on online social networks: A deterministic modeling approach," in *Proceedings of the 2019 IEEE-RIVF International Conference on Computing and Communication Technologies*, 2019.

[17] J. Zhao, Q. Liu, L. Wang, and X. Wang, "Competitiveness maximization on complex networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 7, pp. 1054–1064, 2018.

[18] H. Li, L. Pan, and P. Wu, "Dominated competitive influence maximization with time-critical and time-delayed diffusion in social networks," *Journal of Computational Science*, vol. 28, pp. 318–327, 2018.

[19] W. Chen, A. Collins, R. Cummings, T. Ke, Z. Liu, D. Rincon, X. Sun, Y. Wang, W. Wei, and Y. Yuan, "Influence maximization in social networks when negative opinions may emerge and propagate," in *Proceedings of the 2011 SIAM International Conference on Data Mining*, Mesa, AZ, 2011, pp. 379–390.

[20] S. Bharathi, D. Kempe, and M. Salek, "Competitive influence maximization in social networks," in *Proceedings of the 3rd International Workshop on Internet and Network Economics*, San Diego, CA, 2007, pp. 306–311.

[21] W. Chen, L. V. Lakshmanan, and C. Castillo, "Information and influence propagation in social networks," *Synthesis Lectures on Data Management*, vol. 5, no. 4, pp. 1–177, 2013.

[22] L. A. Wolsey, "An analysis of the greedy algorithm for the submodular set covering problem," *Combinatorica*, vol. 2, no. 4, pp. 385–393, 1982.
[23] T. Carnes, C. Nagarajan, S. M. Wild, and A. Van Zuylen, "Maximizing influence in a competitive social network: a follower's perspective," in *Proceedings of the 9th International Conference on Electronic Commerce*, Minneapolis, MN, 2007, pp. 351–360.
[24] C. V. Pham, H. V. Duong, B. Q. Bui, and M. T. Thai, "Budgeted competitive influence maximization on online social networks," in *Computational Data and Social Networks*, X. Chen, A. Sen, W. W. Li, and M. T. Thai, Eds. Cham: Springer International Publishing, 2018, pp. 13–24.
[25] S. Wang, J. Liu, and Y. Jin, "Finding influential nodes in multiplex networks using a memetic algorithm," *IEEE Transactions on Cybernetics*, 2019.
[26] K. Ali, C. Wang, and Y. Chen, "Boosting reinforcement learning in competitive influence maximization with transfer learning," in *Proceedings of the 2018 IEEE/WIC/ACM International Conference on Web Intelligence*, 2018, pp. 395–400.
[27] C. Borgs, M. Brautbar, J. Chayes, and B. Lucier, "Maximizing social influence in nearly optimal time," in *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms*, Portland, OR, 2014, pp. 946–957.
[28] Y. Tang, X. Xiao, and Y. Shi, "Influence maximization: Near-optimal time complexity meets practical efficiency," in *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*. Snowbird, UT: ACM, 2014, pp. 75–86.
[29] H. T. Nguyen, M. T. Thai, and 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*. San Francisco, CA: ACM, 2016, pp. 695–710.
[30] B. Doerr, *Probabilistic Tools for the Analysis of Randomized Optimization Heuristics*. Cham: Springer International Publishing, 2020, pp. 1–87.
[31] G. Tong, W. Wu, L. Guo, D. Li, C. Liu, B. Liu, and D.-Z. Du, "An efficient randomized algorithm for rumor blocking in online social networks," *IEEE Transactions on Network Science and Engineering*, 2018.
[32] H. Li, S. S. Bhowmick, J. Cui, Y. Gao, and J. Ma, "GetReal: Towards realistic selection of influence maximization strategies in competitive networks," in *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, Melbourne, Australia, 2015, pp. 1525–1537.
[33] S.-C. Lin, S.-D. Lin, and M.-S. Chen, "A learning-based framework to handle multi-round multi-party influence maximization on social networks," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Sydney, Australia, 2015, pp. 695–704.
[34] M. Gong, J. Yan, B. Shen, L. Ma, and Q. Cai, "Influence maximization in social networks based on discrete particle swarm optimization," *Information Sciences*, vol. 367-368, pp. 600–614, 2016.
[35] Y. Zhou, J. Hao, and F. Glover, "Memetic search for identifying critical nodes in sparse graphs," *IEEE Transactions on Cybernetics*, vol. 49, no. 10, pp. 3699–3712, 2019.
[36] L. Wang, Z. Yu, F. Xiong, D. Yang, S. Pan, and Z. Yan, "Influence spread in geo-social networks: A multiobjective optimization perspective," *IEEE Transactions on Cybernetics*, 2019.
[37] Y. Zhu, D. Li, H. Guo, and R. Pamula, "New competitive influence propagation models in social networks," in *Proceedings of the 10th International Conference on Mobile Ad-hoc and Sensor Networks*, 2014, pp. 257–262.
[38] K. Kandhway and J. Kuri, "Using node centrality and optimal control to maximize information diffusion in social networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 7, pp. 1099–1110, 2017.
[39] J. Leskovec and A. Krevl, "SNAP Datasets: Stanford large network dataset collection," http://snap.stanford.edu/data, 2014.
[40] V. Batagelj and A. Mrvar, "Pajek Datasets," http://vlado.fmf.uni-lj.si/pub/networks/data, 2006.
[41] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Washington, DC, 2003, pp. 137–146.
[42] C. Qian, "Distributed Pareto optimization for large-scale noisy subset selection," *IEEE Transactions on Evolutionary Computation*, 2019, doi: 10.1109/TEVC.2019.2929555.
[43] C. Qian, J.-C. Shi, Y. Yu, K. Tang, and Z.-H. Zhou, "Subset selection under noise," in *Advances in Neural Information Processing Systems 30*, 2017, pp. 3563–3573.
[44] X. Zhou, H. Wang, W. Peng, B. Ding, and R. Wang, "Solving multi-scenario cardinality constrained optimization problems via multi-objective evolutionary algorithms," *SCIENCE CHINA Information Sciences*, vol. 62, no. 9, pp. 192 104:1–192 104:18, 2019.
[45] P. Xu, X. Liu, H. Cao, and Z. Zhang, "An efficient energy aware virtual network migration based on genetic algorithm," *Frontiers of Computer Science*, vol. 13, no. 2, pp. 440–442, 2019.

Wenjing Hong received the B.Eng. and Ph.D. degrees in computer science and technology from the University of Science and Technology of China, China, in 2012 and 2018, respectively. She is currently a postdoc in the Department of Computer Science and Engineering, Southern University of Science and Technology, and in the School of Management, University of Science and Technology of China, China. Her research interests include evolutionary computation, evolutionary learning and network analysis.

Chao Qian received the B.Sc. and Ph.D. degrees in the Department of Computer Science and Technology from Nanjing University, China, in 2009 and 2015, respectively. From 2015 to 2019, he was an associate researcher in the School of Computer Science and Technology, University of Science and Technology of China, China. He is currently an associate professor in the School of Artificial Intelligence, Nanjing University, China. His research interests are mainly in evolutionary computation and machine learning, particularly, the theoretical foundation of evolutionary algorithms and its application with theoretical guarantees in machine learning. He has published one book *Evolutionary Learning: Advances in Theories and Algorithms* and over 20 first-author papers in leading international journals and conference proceedings, including *Artificial Intelligence*, *Evolutionary Computation*, *IEEE Transactions on Evolutionary Computation*, *Algorithmica*, NIPS, IJCAI, AAAI, etc. He won the ACM GECCO 2011 Best Paper Award (Theory Track), the IDEAL 2016 Best Paper Award, and the 2017 Outstanding Doctoral Dissertation Award of CAAI. He has served as chair of the IEEE Computational Intelligence Society Task Force "Theoretical Foundations of Bio-inspired Computation", and an Young Associate Editor of *Frontiers of Computer Science*. He has also been selected to the Young Elite Scientists Sponsorship Program by CAST.

Ke Tang (M'07-SM'13) received the B.Eng. degree from the Huazhong University of Science and Technology, Wuhan, China, in 2002 and the Ph.D. degree from Nanyang Technological University, Singapore, in 2007. From 2007 to 2017, he was with the School of Computer Science and Technology, University of Science and Technology of China, China, first as an Associate Professor from 2007 to 2011 and later as a Professor from 2011 to 2017. He is currently a Professor with the Department of Computer Science and Engineering, Southern University of Science and Technology, China. He has over 7000 Google Scholar citations with an H-index of 41. He has published over 70 journal papers and over 80 conference papers. His current research interests include evolutionary computation, machine learning, and their applications. Dr. Tang was a recipient of the Royal Society Newton Advanced Fellowship in 2015 and the 2018 IEEE Computational Intelligence Society Outstanding Early Career Award. He is an Associate Editor of the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION and served as a member of Editorial Boards for a few other journals.