Adaptive Greedy versus Non-Adaptive Greedy for Influence Maximization*

Wei Chen,[1] Binghui Peng,[2] Grant Schoenebeck,[3] Biaoshuai Tao[3]
[1]Microsoft Research, China, [2]Columbia University, [3]University of Michigan, Ann Arbor
weic@microsoft.com, bp2601@columbia.edu, {schoeneb, bstao}@umich.edu

Abstract

We consider the *adaptive influence maximization problem*: given a network and a budget k, iteratively select k seeds in the network to maximize the expected number of adopters. In the *full-adoption feedback model*, after selecting each seed, the seed-picker observes all the resulting adoptions. In the *myopic feedback model*, the seed-picker only observes whether each neighbor of the chosen seed adopts. Motivated by the extreme success of greedy-based algorithms/heuristics for influence maximization, we propose the concept of *greedy adaptivity gap*, which compares the performance of the adaptive greedy algorithm to its non-adaptive counterpart. Our first result shows that, for submodular influence maximization, the adaptive greedy algorithm can perform up to a $(1-1/e)$-fraction worse than the non-adaptive greedy algorithm, and that this ratio is tight. More specifically, on one side we provide examples where the performance of the adaptive greedy algorithm is only a $(1-1/e)$ fraction of the performance of the non-adaptive greedy algorithm in four settings: for both feedback models and both the *independent cascade model* and the *linear threshold model*. On the other side, we prove that in any submodular cascade, the adaptive greedy algorithm always outputs a $(1-1/e)$-approximation to the expected number of adoptions in the optimal non-adaptive seed choice. Our second result shows that, for the general submodular cascade model with full-adoption feedback, the adaptive greedy algorithm can outperform the non-adaptive greedy algorithm by an unbounded factor. Finally, we propose a risk-free variant of the adaptive greedy algorithm that always performs no worse than the non-adaptive greedy algorithm.

1 Introduction

The *influence maximization problem* (InfMax) is an optimization problem that asks which seeds a viral marketing campaign should target (e.g. by giving free products) so that propagation from these seeds influences the most people in a social network. That is, given a graph, a *stochastic diffusion model* defining how each node is infected by its neighbors, and a limited budget k, how to pick k seeds such that the expected number of total infected nodes in this graph at the end of the diffusion is maximized. This problem has significant applications in viral marketing, outbreak detection, rumor controls, etc, and has been extensively studied (cf. Chen, Lakshmanan, and Castillo; Li et al. (2013; 2018)).

For InfMax, most of the existing work has considered *submodular* diffusion models, especially the *independent cascade model* and the *linear threshold model* (Kempe, Kleinberg, and Tardos 2003). Likewise, we also focus on submodular diffusion models. In submodular diffusion models, a vertex v's marginal probability of becoming infected after a new neighbor t is infected given S as the set of v's already infected neighbors is at least the marginal probability that v is infected after t is newly infected given $T \supseteq S$ as the set of v's already infected neighbors (see the paragraph before Theorem 2.4 for more details). Intuitively, this means that the influence of infected nodes are substitutes and never have synergy.

When submodular InfMax is considered, nearly all the known algorithms/heuristics are based on a greedy algorithm that iteratively picks the seed that has the largest marginal influence. Some of them improve the running time of the original greedy algorithm by skipping vertices that are known to be suboptimal (Leskovec et al. 2007; Goyal, Lu, and Lakshmanan 2011a), while the others improve the scalability of the greedy algorithm by using more scalable algorithms to approximate the expected total influence (Borgs et al. 2014; Tang, Xiao, and Shi 2014; Tang, Shi, and Xiao 2015; Cheng et al. 2013; Ohsaka et al. 2014) or computing a score of the seeds that is closely related to the expected total influence (Chen, Wang, and Yang 2009; Chen, Yuan, and Zhang 2010a; 2010b; Goyal, Lu, and Lakshmanan 2011b; Jung, Heo, and Chen 2012; Galhotra, Arora, and Roy 2016; Tang et al. 2018; Schoenebeck and Tao 2019b). Arora, Galhotra, and Ranu (2017) benchmark most of the aforementioned variants of the greedy algorithms.

In this paper, we study the *adaptive influence maximization problem*, where seeds are selected iteratively and feedback is given to the seed-picker after selecting each seed.

*Full version of this paper at: http://arxiv.org/abs/1911.08164.
Grant Schoenebeck and Biaoshuai Tao are pleased to acknowledge the support of National Science Foundation AitF #1535912 and CAREER #1452915.

Two different feedback models have been studied in the past: the *full-adoption feedback model* and the *myopic feedback model* (Golovin and Krause 2011). In the full-adoption feedback model, the seed-picker sees the entire diffusion process of each selected seed, and in the myopic feedback model the seed-picker only sees whether each neighbor of the chosen seed is infected.

Past literature focused on the *adaptivity gap*—the ratio between the performance of the *optimal* adaptive algorithm and the performance of the *optimal* non-adaptive algorithm (Golovin and Krause 2011; Peng and Chen 2019; Chen and Peng 2019). However, even in the non-adaptive setting, InfMax is known to be APX-hard (Kempe, Kleinberg, and Tardos 2003; Schoenebeck and Tao 2019b). As a result, in practice, it is not clear whether the adaptivity gap can measure how much better an adaptive algorithm can do.

In this paper, we define and consider the *greedy adaptivity gap*, which is the ratio between the performance of the adaptive greedy algorithm and the non-adaptive greedy algorithm. We focus on the gap between the greedy algorithms for three reasons. First, as we mentioned, the APX-hardness of InfMax renders the practical implications of the adaptivity gap unclear. Second, as we remarked at the beginning, the greedy algorithm is used almost exclusively in the context of influence maximization. Third, the iterative nature of the original greedy algorithm naturally extends to the adaptive setting.

1.1 Our Results

We show that, for the general submodular diffusion models, with both the full-adoption feedback model and the myopic feedback model, the infimum of the greedy adaptivity gap is exactly $(1-1/e)$. In addition, this result can be extended to the two well-studied submodular diffusion models: the independent cascade model and the linear threshold model. This is proved in two steps.

In the first step, we show that there are InfMax instances where the adaptive greedy algorithm can only produce $(1-1/e)$ fraction of the influence of the solution output by the non-adaptive greedy algorithm. This result is surprising: one would expect that the adaptivity is always helpful, as the feedback provides more information to the seed-picker, which makes the seed-picker refine the seed choices in future iterations. Our result shows that this is not the case, and the feedback, if overly used, can make the seed-picker act in a more myopic way, which is potentially harmful.

In the second step, we show that the adaptive greedy algorithm is always a $(1-1/e)$-approximation of the non-adaptive optimal solution, so its performance is always at least a $(1-1/e)$ fraction of the performance of the non-adaptive greedy algorithm. In particular, combining the two steps, we see that when the adaptive greedy algorithm output only obtains a (nearly) $(1-1/e)$-fraction of the performance of the non-adaptive greedy algorithm, the non-adaptive greedy algorithm is (almost) optimal. This worst-case guarantee indicates that the adaptive greedy algorithm will never be too bad.

As the second result, we show that the supremum of the greedy adaptivity gap is infinity, for the general submodular diffusion model with full-adoption feedback. This indicates that the adaptive greedy algorithm can perform significantly better than its non-adaptive counterpart. We also show, with almost the same proof, that the adaptivity gap in this setting (general submodular model with full-adoption feedback) is also unbounded.

Finally, we propose a risk-free but more conservative variant of the adaptive greedy algorithm, which always performs at least as well as the non-adaptive greedy algorithm. We recommend both the adaptive greedy algorithm and this variant.

1.2 Related Work

The influence maximization problem was initially posed by Domingos and Richardson (2001; 2002). Kempe, Kleinberg, and Tardos (2003) proposed the linear threshold model and the independent cascade model, and show that they are submodular. Whenever a diffusion model is submodular, the greedy algorithm was shown to obtains a $(1-1/e)$-approximation to the optimal number of infections (Nemhauser, Wolsey, and Fisher 1978; Kempe, Kleinberg, and Tardos 2003; 2005; Mossel and Roch 2010).

For adaptive InfMax, Golovin and Krause (2011) showed that InfMax with the independent cascade model and full-adoption feedback is *adaptive submodular*, which implies that the adaptive greedy algorithm obtains a $(1-1/e)$-approximation to the adaptive optimal solution. On the other hand, InfMax for the independent cascade model with myopic feedback, as well as InfMax for the linear threshold model with both feedback models, are not adaptive submodular. In particular, the adaptive greedy algorithm fails to obtain a $(1-1/e)$-approximation for the independent cascade model with myopic feedback (Peng and Chen 2019). Peng and Chen (2019) showed that the adaptivity gap for the independent cascade model with myopic feedback is at most 4 and at least $e/(e-1)$, and they also showed that both the adaptive and non-adaptive greedy algorithms perform a $0.25(1-1/e)$-approximation to the adaptive optimal solution. The adaptivity gap for the independent cascade model with full-adoption feedback, as well as the adaptivity gap for the linear threshold model with both feedback models, are still open problems, although there is some partial progress (Chen and Peng 2019).

Our paper is not the first work studying the adaptive greedy algorithm. Previous work focused on improving the running time of the adaptive greedy algorithm (Han et al. 2018). However, to the best of our knowledge, our work is the first one that compares the adaptive greedy algorithm to its non-adaptive counterpart.

Finally, we remark that there do exist InfMax algorithms that are not based on greedy (Bharathi, Kempe, and Salek 2007; Goldberg and Liu 2013; Angell and Schoenebeck 2016; Schoenebeck and Tao 2017; 2019a; Schoenebeck, Tao, and Yu 2019), but they are typically for non-submodular diffusion models.

2 Preliminary

All graphs in this paper are simple and directed. Given a graph $G=(V,E)$ and a vertex $v\in V$, let $\Gamma(v)$ and $\deg(v)$

be the set of in-neighbors and the in-degree of v respectively.

2.1 Triggering Model

We consider the well-studied *triggering model* (Kempe, Kleinberg, and Tardos 2003), which is commonly used to capture "general" submodular diffusion models.[1]

Definition 2.1 (Kempe, Kleinberg, and Tardos (2003))**.** The *triggering model*, $I_{G,F}$, is defined by a graph $G = (V, E)$ and for each vertex v a distribution $\mathcal{F}_v$ over the subset of its in-neighbors $\{0,1\}^{|\Gamma(v)|}$. Let $F = \{\mathcal{F}_v \mid v \in V\}$.

On an input seed set $S \subseteq V$, $I_{G,F}(S)$ outputs a set of infected vertices as follows:

1. Initially, only vertices in S are infected. Each vertex v samples a subset of its in-neighbors $T_v \subseteq \Gamma(v)$ from $\mathcal{F}_v$ independently. We call T_v the *triggering set* of v.
2. In each subsequent round, a vertex v becomes infected if a vertex in T_v is infected in the previous round.
3. After a round where no additional vertices are infected, the set of infected vertices is the output.

$I_{G,F}$ in Definition 2.1 can be viewed as a random function $I_{G,F} : \{0,1\}^{|V|} \to \{0,1\}^{|V|}$. In addition, if the triggering set T_v is fixed for each vertex v, then $I_{G,F}$ is deterministic. Given v, its triggering set T_v, and an in-neighbor $u \in \Gamma(v)$, we say that the edge (u, v) is *live* if $u \in T_v$, and we say that (u, v) is *blocked* if $u \notin T_v$. It is easy to see that, when the triggering sets for all vertices are sampled, $I_{G,F}(S)$ is the set of all vertices that are reachable from S when removing all blocked edges from the graph.

We define a *realization* of a graph $G = (V, E)$ as a function $\phi : E \to \{\mathtt{L}, \mathtt{B}\}$ such that $\phi(e) = \mathtt{L}$ if $e \in E$ is live and $\phi(e) = \mathtt{B}$ if $e \in E$ is blocked. Let $I^{\phi}_{G,F} : \{0,1\}^{|V|} \to \{0,1\}^{|V|}$ be the deterministic function corresponding to the triggering model $I_{G,F}$ with vertices' triggering sets following realization ϕ. We write $\phi \sim F$ to indicate that a realization ϕ is sampled according to $F = \{\mathcal{F}_v\}$.

The triggering model captures the well-known independent cascade and linear threshold models. In the two definitions below, we define the two models in terms of the triggering model, which is sufficient for this paper. In the full version of this paper, we present the original definitions and give some intuitions for the two models for those readers who are not familiar with them.

Definition 2.2. The *independent cascade model* $\mathtt{ICM}$ is a special case of the triggering model $I_{G,F}$ where $G = (V, E, w)$ is an edge-weighted graph with $w(u, v) \in (0, 1]$ for each $(u, v) \in E$ and $\mathcal{F}_v$ is the distribution such that each $u \in \Gamma(v)$ is included in T_v with probability $w(u, v)$ independently.

Definition 2.3. The *linear threshold model* $\mathtt{LTM}$ is a special case of the triggering model $I_{G,F}$ where $G = (V, E, w)$ is an edge-weighted graph with $w(u, v) > 0$ for each $(u, v) \in E$ and $\sum_{u\in\Gamma(v)} w(u, v) \leq 1$ for each $v \in V$, and $\mathcal{F}_v$ is the distribution defined as follows: order v's in-neighbors $u_1, \ldots, u_T$ arbitrarily, sample a real number r in $[0, 1]$ uniformly, and

$$T_v = \begin{cases} \{u_t\} & \text{if } r \in \left[\sum_{i=1}^{t-1} w(u_i, v), \sum_{i=1}^{t} w(u_i, v)\right) \\ \emptyset & \text{if } r \geq \sum_{i=1}^{T} w(u_i, v) \end{cases}.$$

Intuitively, T_v includes at most one of v's in-neighbors such that each u_t is included with probability $w(u_t, v)$.

Given a triggering model $I_{G,F}$, let $\sigma_{G,F} : \{0,1\}^{|V|} \to \mathbb{R}_{\geq 0}$ be the *global influence function* defined as $\sigma_{G,F}(S) = \mathbb{E}_{\phi\sim F}[|I^{\phi}_{G,F}(S)|]$. We drop the subscripts G, F and write the global influence function as $\sigma(\cdot)$ when there is no ambiguity.

A function f mapping from a set of elements to a non-negative value is *submodular* if $f(A\cup\{v\})-f(A) \geq f(B\cup\{v\}) - f(B)$ for any two sets A, B with $A \subsetneq B$ and any element $v \notin B$.

Theorem 2.4 (Kempe, Kleinberg, and Tardos (2003))**.** *For any triggering model $I_{G,F}$, $\sigma_{G,F}(\cdot)$ is submodular. In particular, $\sigma_{G,F}(\cdot)$ is submodular for both* $\mathtt{ICM}$ *and* $\mathtt{LTM}$*.*

2.2 InfMax and Adaptive InfMax

Definition 2.5. The *influence maximization problem* (InfMax) is an optimization problem which takes inputs $G = (V, E)$, F, and $k \in \mathbb{Z}^+$, and outputs a seed set S that maximizes the expected total number of infections: $S \in \operatorname{argmax}_{S\subseteq V:|S|\leq k} \sigma(S)$.

In the remaining part of this subsection, we define the adaptive version of the influence maximization problem. We will define two different models: the *full-adoption feedback model* and the *myopic feedback model*. Suppose a seed set $S \subseteq V$ is chosen by the seed-picker, and an underlying realization ϕ is given but not known by the seed-picker. Informally, in the full-adoption feedback model, the seed-picker sees all the vertices that are infected by S in all future iterations, i.e., the seed-picker sees $I^{\phi}_{G,F}(S)$. In the myopic feedback model, the seed-picker only sees the states of S's neighbors, i.e., whether each vertex in $\{v \mid \exists s \in S : s \in \Gamma(v)\}$ is infected.

Define a *partial realization* as a function $\varphi : E \to \{\mathtt{L}, \mathtt{B}, \mathtt{U}\}$ such that $\phi(e) = \mathtt{L}$ if e is known to be live, $\phi(e) = \mathtt{B}$ if e is known to be blocked, and $\phi(e) = \mathtt{U}$ if the status of e is not yet known. We say that a partial realization φ *is consistent with* the full realization ϕ, denoted by $\phi \simeq \varphi$, if $\phi(v) = \varphi(v)$ whenever $\varphi(v) \neq \mathtt{U}$. For the ease of notation, for an edge $(u, v) \in E$, we will write $\phi(u, v), \varphi(u, v)$ instead of $\phi((u, v)), \varphi((u, v))$.

Definition 2.6. Given a triggering model $I_{G=(V,E),F}$ with a realization ϕ, the *full-adoption feedback* is a function $\Phi^{\mathtt{f}}_{G,F,\phi}$ mapping a seed set $S \subseteq V$ to a partial realization φ such that

- $\varphi(u, v) = \phi(u, v)$ for each $u \in I^{\phi}_{G,F}(S)$, and
- $\varphi(u, v) = \mathtt{U}$ for each $u \notin I^{\phi}_{G,F}(S)$.

Definition 2.7. Given a triggering model $I_{G=(V,E),F}$ with a realization ϕ, the *myopic feedback* is a function $\Phi^{\mathtt{m}}_{G,F,\phi}$ mapping a seed set $S \subseteq V$ to a partial realization φ such that

[1] A more general way to capture submodular diffusion models is the *general threshold model* (Kempe, Kleinberg, and Tardos 2003) with *submodular local influence functions*. All our results hold under this setting as well. We will discuss this in the full version.

- $\varphi(u,v) = \phi(u,v)$ for each $u \in S$, and
- $\varphi(u,v) = \mathtt{U}$ for each $u \notin S$.

An *adaptive policy* π is a function that maps a seed set S and a partial realization φ to a vertex $v = \pi(S,\varphi)$, which corresponds to the next seed the policy π would choose given φ and S being the set of seeds that has already been chosen. Naturally, we only care about $\pi(S,\varphi)$ when $\varphi = \Phi^{\mathfrak{f}}_{G,F,\phi}(S)$ or $\varphi = \Phi^{\mathfrak{m}}_{G,F,\phi}(S)$, although we define π that specifies an output for any possible inputs S and φ. Notice that we have defined π as a deterministic policy for simplicity, and our results hold for randomized policies. Let Π be the set of all possible adaptive policies.

Notice that an adaptive policy π completely specifies a seeding strategy in an iterative way. Given an adaptive policy π and a realization ϕ, let $\mathcal{S}^{\mathfrak{f}}(\pi,\phi,k)$ be the first k seeds selected according to π with the underlying realization ϕ under the full-adoption feedback model. By on our definition, $\mathcal{S}^{\mathfrak{f}}(\pi,\phi,k)$ can be computed as follows:

1. initialize $S = \emptyset$;
2. update $S = S \cup \{\pi(S, \Phi^{\mathfrak{f}}_{G,F,\phi}(S))\}$ for k iterations;
3. output $\mathcal{S}^{\mathfrak{f}}(\pi,\phi,k) = S$.

Define $\mathcal{S}^{\mathfrak{m}}(\pi,\phi,k)$ similarly for the myopic feedback model, where $\Phi^{\mathfrak{m}}_{G,F,\phi}(S)$ instead of $\Phi^{\mathfrak{f}}_{G,F,\phi}(S)$ is used in Step 2 above.

Let $\sigma^{\mathfrak{f}}(\pi,k)$ be the expected number of infected vertices given that k seeds are chosen according to π, i.e., $\sigma^{\mathfrak{f}}(\pi,k) = \mathbb{E}_{\phi\sim F}[|I^{\phi}_{G,F}(\mathcal{S}^{\mathfrak{f}}(\pi,\phi,k))|]$. Define $\sigma^{\mathfrak{m}}(\pi,k)$ similarly for the myopic feedback model.

Definition 2.8. The *adaptive influence maximization problem* (adaptive INFMAX) is an optimization problem which takes as inputs $G = (V,E)$, F, and $k \in \mathbb{Z}^+$, and outputs an adaptive policy π that maximizes the expected total number of infections: $\pi \in \operatorname{argmax}_{\pi\in\Pi} \sigma^{\mathfrak{f}}(\pi,k)$ or $\pi \in \operatorname{argmax}_{\pi\in\Pi} \sigma^{\mathfrak{m}}(\pi,k)$ (depending on the feedback model used).

2.3 Adaptivity Gap and Greedy Adaptivity Gap

The adaptivity gap is defined as the ratio between the performance of the optimal adaptive policy and the performance of the optimal non-adaptive seeding strategy. In this paper, we only consider the adaptivity gap for triggering models.

Definition 2.9. The *adaptivity gap with full-adoption feedback* is

$$\sup_{G,F,k} \frac{\max_{\pi\in\Pi} \sigma^{\mathfrak{f}}(\pi,k)}{\max_{S\subseteq V,|S|\leq k} \sigma(S)}.$$

The *adaptivity gap with myopic feedback* is defined similarly.

The (non-adaptive) *greedy algorithm* iteratively picks a seed that has the maximum marginal gain to the objective function $\sigma(\cdot)$:

1. initialize $S = \emptyset$;
2. update for k iterations $S = S \cup \{s\}$, where $s \in \operatorname{argmax}_{s\in V} \sigma(S \cup \{s\})$ with tie broken in an arbitrarily consistent order;
3. return S.

Let $S^g(k)$ be the set of k seeds output by the (non-adaptive) greedy algorithm.

The *greedy adaptive policy* π^g is defined as $\pi^g(S,\varphi) = s$ such that $s \in \operatorname*{argmax}_{s\in V} \mathop{\mathbb{E}}_{\phi\simeq\varphi}\left[\left|I^{\phi}_{G,F}(S\cup\{s\})\right|\right]$, with tie broken in an arbitrary consistent order.

Definition 2.10. Given a triggering model $I_{G,F}$ and $k \in \mathbb{Z}^+$, the *greedy adaptivity gap with full-adoption feedback* is $\frac{\sigma^{\mathfrak{f}}(\pi^g,k)}{\sigma(S^g(k))}$. The *greedy adaptivity gap with myopic feedback* is defined similarly.

Notice that, unlike the adaptivity gap in Definition 2.9, we leave G, F, k unspecified (instead of taking a supremum over them) when defining the greedy adaptivity gap. This is because we are interested in both supremum and infimum of the ratio $\frac{\sigma^{\mathfrak{f}}(\pi^g,k)}{\sigma(S^g(k))}$. Notice that the infimum of the ratio $\frac{\max_{\pi\in\Pi}\sigma^{\mathfrak{f}}(\pi,k)}{\max_{S\subseteq V,|S|\leq k}\sigma(S)}$ in Definition 2.9 is 1: the optimal adaptive policy is at least as good as the optimal non-adaptive policy, as the non-adaptive policy can be viewed as a special adaptive policy; on the other hand, it is easy to see that there are INFMAX instances such that the optimal adaptive policy is no better than non-adaptive one (for example, a graph containing k vertices but no edges). For this reason, we only care about the supremum of this ratio.

3 Infimum of Greedy Adaptivity Gap

In this section, we show that the infimum of the greedy adaptivity gap for the triggering model is exactly $(1 - 1/e)$, for both the full-adoption feedback model and the myopic feedback model. This implies that the greedy adaptive policy can perform even worse than the conventional non-adaptive greedy algorithm, but it will never be significantly worse. Moreover, we show that this result also holds for both ICM (Definition 2.2) and LTM (Definition 2.3).

Theorem 3.1. *For the full-adoption feedback model,*

$$\inf_{G,F,k:\ I_{G,F}\text{ is ICM}} \frac{\sigma^{\mathfrak{f}}(\pi^g,k)}{\sigma(S^g(k))} = \inf_{G,F,k:\ I_{G,F}\text{ is LTM}} \frac{\sigma^{\mathfrak{f}}(\pi^g,k)}{\sigma(S^g(k))}$$

$$= \inf_{G,F,k} \frac{\sigma^{\mathfrak{f}}(\pi^g,k)}{\sigma(S^g(k))} = 1 - \frac{1}{e}.$$

The same result holds for the myopic feedback model.

In Sect. 3.1, we show by providing examples that the greedy adaptive policy in the worst case will only achieves $(1 - 1/e + \varepsilon)$-approximation of the expected number of infected vertices given by the non-adaptive greedy algorithm, for both ICM and LTM.

In Sect. 3.2, we shows that the greedy adaptive policy has performance at least $(1 - 1/e)$ of the performance of the non-adaptive optimal seeds (Theorem 3.4). Theorem 3.4 provides a lower bound on the greedy adaptivity gap for the triggering model and is also interesting on its own. At the end of Sect. 3.2, we prove Theorem 3.1 by putting the results from Sect. 3.1 and Sect. 3.2 together.

3.1 Tight Examples

In this subsection, we show that the adaptive greedy algorithm can perform worse than the non-adaptive greedy algorithm by a factor of $(1-1/e+\varepsilon)$, for both ICM and LTM and any $\varepsilon > 0$. This may be surprising, as one would expect that the feedback provided to the seed-picker will refine the seed choices in the future iterations. Here, we provide some intuitions why adaptivity can sometimes hurt. Suppose there are two promising sequences of seed selections, $\{s, u_1, \ldots, u_k\}$ and $\{s, v_1, \ldots, v_k\}$, such that

- s is the best seed which will be chosen first;
- $\{s, u_1, \ldots, u_k\}$ has a better performance;
- the influence of $u_1, \ldots, u_k$ are non-overlapping, the influence of $v_1, \ldots, v_k$ are non-overlapping, but the influence of u_i, v_j overlaps for each i, j; moreover, if u_1 is picked as the second seed, the greedy algorithm, adaptive or not, will continue to pick $u_2, \ldots, u_k$, and if v_1 is picked as the second seed, $v_2, \ldots, v_k$ will be picked next;

Now, suppose there is a vertex t elsewhere which can be infected by both s and v_1, such that

- if t is infected by s, which slightly reduces the marginal influence of v_1, v_1 will be less promising than u_1;
- if t is not infected by s, v_1 is more promising than u_1;
- in average, when there is no feedback, v_1 is still less promising than u_1, even after adding the increment in t's infection probability to v_1's expected marginal influence.

In this case, the non-adaptive greedy algorithm will "go to the right trend" by selecting u_1 as the second seed; the adaptive greedy algorithm, if receiving feedback that t is not infected by s, will "go to the wrong trend" by selecting v_1 next.

As a high-level description of the lesson we learned, both versions of the greedy algorithms are intrinsically myopic, and the feedback received by the adaptive policy may make the seed-picker act in a more myopic way, which could be more hurtful to the final performance.

We will assume in the rest of this section that vertices can have positive integer weights. This can be assumed without loss of generality, as a simple "star gadget" can simulate weighted vertices. See the full version of the paper for details. We will denote by $\overline{w}(v)$ the weight of v.

Lemma 3.2. *For any ε, there exists G, F, k such that $I_{G,F}$ is an ICM and $\frac{\sigma^{\mathfrak{f}}(\pi^g,k)}{\sigma(S^g(k))} \leq 1 - \frac{1}{e} + \varepsilon$, $\frac{\sigma^{\mathrm{m}}(\pi^g,k)}{\sigma(S^g(k))} \leq 1 - \frac{1}{e} + \varepsilon$.*

Proof (sketch). We will construct an InfMax instance $(G = (V, E, w), k + 1)$ with $k + 1$ seeds allowed. Let $W \in \mathbb{Z}^+$ be a sufficiently large perfect square divisible by k^{2k} and whose value is to be decided later. The vertex set V contains the following weighted vertices:

- a vertex s that has weight $2W$;
- a vertex t that has weight $\sqrt{W}/k$;
- $2k$ vertices $u_1, \ldots, u_k, v_1, \ldots, v_k$ that have weight 1;
- $k(k+1)$ vertices $\{w_{ij} \mid i = 1, \ldots, k+1; j = 1, \ldots, k\}$
  - $w_{11}, \ldots, w_{1k}$ have weight $\frac{W}{k}$;
  - $w_{i1}, \ldots, w_{ik}$ have weight $\frac{1}{k}(1 - \frac{1}{k})^{i-1}W + \sqrt{W}$ for each $i = 2, \ldots, k$;
  - $w_{(k+1)1}, \ldots, w_{(k+1)k}$ have weight $(1 - \frac{1}{k})^k W + \frac{\sqrt{W}-k}{k} - (k-1)\sqrt{W}$.

The edge set E is specified as follow:

- create two edges (v_1, t) and (s, t);
- for each $i = 1, \ldots, k$, create $k + 1$ edges $(u_i, w_{1i}), (u_i, w_{2i}), \ldots, (u_i, w_{(k+1)i})$, and create k edges $(v_i, w_{i1}), (v_i, w_{i2}), \ldots, (v_i, w_{ik})$.

For the weights of edges, all the edges have weight 1 except for the edge (s, t) which has weight $2k/\sqrt{W}$.

This construction is based on exactly the intuitions mentioned at the beginning of Sect. 3.1. By a careful analysis, the non-adaptive greedy algorithm will choose $\{s, u_1, \ldots, u_k\}$, with expected influence $(k + 2)W + O(\sqrt{W})$. For the greedy adaptive policy, it will choose $\{s, u_1, \ldots, u_k\}$ if the feedback received from picking s is that t is infected, but this happens with a negligible probability $2k/\sqrt{W}$; otherwise, it will choose $\{s, v_1, \ldots, v_k\}$, with expected influence $(2 + k(1 - (1 - 1/k)^k))W + O(\sqrt{W})$, and this happens with a high probability $1 - 2k/\sqrt{W}$. The lemma concludes by noticing that

$$\lim_{W,k\to\infty} \frac{(2 + k(1 - (1 - 1/k)^k))W + O(\sqrt{W})}{(k + 2)W + O(\sqrt{W})} = 1 - \frac{1}{e}.$$

The detailed analysis is omitted due to the space limit and is available in the full version of this paper. □

Lemma 3.3. *For any ε, there exists G, F, k such that $I_{G,F}$ is an LTM and $\frac{\sigma^{\mathfrak{f}}(\pi^g,k)}{\sigma(S^g(k))} \leq 1 - \frac{1}{e} + \varepsilon$, $\frac{\sigma^{\mathrm{m}}(\pi^g,k)}{\sigma(S^g(k))} \leq 1 - \frac{1}{e} + \varepsilon$.*

Proof. See the full version of this paper. □

3.2 Lower Bound

Theorem 3.4. *For a triggering model $I_{G,F}$, we have both $\sigma^{\mathfrak{f}}(\pi^g, k) \geq \left(1 - \frac{1}{e}\right) \max_{S\subseteq V,|S|\leq k} \sigma(S)$ and $\sigma^{\mathrm{m}}(\pi^g, k) \geq \left(1 - \frac{1}{e}\right) \max_{S\subseteq V,|S|\leq k} \sigma(S)$.*

Proof. The proof is deferred to the full version of this paper. For a high-level idea, let S with $|S| = i$ be the seeds picked by π^g for the first i iterations and S^* be the optimal non-adaptive seed set: $S^* \in \operatorname{argmax}_{|S'|\leq k} \sigma(S')$. Given S as the existing seeds and any feedback (myopic or full-adoption) corresponding to S, we can show that the marginal increment to the expected influence caused by the $(i + 1)$-th seed picked by π^g is at least $1/k$ of the marginal increment to the expected influence caused by S^*. Then, a standard argument showing that the greedy algorithm can achieve a $(1 - 1/e)$-approximation for any submodular monotone optimization problem can be used to prove this theorem. □

Finally, putting Theorem 3.4, Lemma 3.2 and Lemma 3.3 together, Theorem 3.1 can be concluded easily. The proof of Theorem 3.1 is available in the full version.

4 Supremum of Greedy Adaptivity Gap

In this section, we show that, for the full-adoption feedback model, both the adaptivity gap and the supremum of the greedy adaptivity gap are unbounded. As a result, in some cases, the adaptive version of the greedy algorithm can perform significantly better than its non-adaptive counterpart.

Theorem 4.1. *The greedy adaptivity gap with full-adoption feedback is unbounded: there exists a triggering model $I_{G,F}$ and k such that $\frac{\sigma^{\mathfrak{f}}(\pi^g,k)}{\sigma(S^g(k))} = 2^{\Omega(\log\log|V|/\log\log\log|V|)}$.*

Theorem 4.2. *The adaptivity gap for the general triggering model with full-adoption feedback is infinity.*

In Sect. 4.1, we consider a variant of INFMAX such that the seeds can only be chosen among a prescribed vertex set $\overline{V} \subseteq V$, where $\overline{V}$ is specified as an input to the INFMAX instance. We show that, under this setting with LTM, both the adaptivity gap and the supremum of the greedy adaptivity gap with the full-adoption feedback model are unbounded (Lemma 4.5). Since it is common in practice that only a subset of nodes in a network is visible or accessible to the seed-picker, Lemma 4.5 is also interesting on its own. In Sect. 4.2, we show that how Lemma 4.5 can be used to prove Theorem 4.1 and Theorem 4.2. Notice that Theorem 4.1 and Theorem 4.2 hold for the standard INFMAX setting without a prescribed set of seed candidates, but we do not know if they hold for LTM (instead, they are for the more general triggering model).

We first present the following lemma revealing a special additive property for LTM, which will be used later.

Lemma 4.3. *Suppose $I_{G,F}$ is LTM. If $U_1, U_2 \subseteq V$ with $U_1 \cap U_2 = \emptyset$ satisfy that there is no path from any vertices in U_1 to any vertices in U_2 and vice versa, then $\sigma(U_1) + \sigma(U_2) = \sigma(U_1 \cup U_2)$.*

Proof. See the full version of this paper. □

4.1 On LTM with Prescribed Seed Candidates

Definition 4.4. The *influence maximization problem with prescribed seed candidates* is an optimization problem which takes as inputs $G = (V, E)$, F, $k \in \mathbb{Z}^+$, and $\overline{V} \subseteq V$, and outputs a seed set $S \subseteq \overline{V}$ that maximizes the expected total number of infections: $S \in \operatorname{argmax}_{S\subseteq\overline{V}:|S|\leq k}\sigma(S)$. The *adaptive influence maximization problem with prescribed seed candidates* has the same definition as it is in Definition 2.8, with the exception that the range of the function π is now $\overline{V}$, and Π is the set of all such policies.

Lemma 4.5. *For* INFMAX *with prescribed seed candidates with LTM and full-adoption feedback, the adaptivity gap is infinity, and the greedy adaptivity gap is $2^{\Omega(\log|V|/\log\log|V|)}$.*

Proof. For $d, W \in \mathbb{Z}^+$ with d being sufficiently large and $W \gg d^{d+1}$, we construct the following (adaptive) INFMAX instance with prescribed seed candidates:

- the edge-weighted graph $G = (V, E, w)$ consists of an $(d + 1)$-level directed full d-ary tree with the root node being the sink (i.e., an in-arborescence) and W vertices each of which is connected *from* the root node of the tree; the weight of each edge in the tree is $1/d$, and the weight of each edge connecting from the root to those W vertices is 1;
- the number of seeds is given by $k = 2(\frac{d+1}{2})^d$;
- the prescribed set for seed candidates $\overline{V}$ is the set of all the leaves in the tree.

Since the leaves are not reachable from one to another, Lemma 4.3 indicates that choosing any k vertices among $\overline{V}$, i.e., the leaves, infects the same number of vertices in expectation. It is easy to see that a single seed among the leaves will infect the root node with probability $1/d^d$, and those W vertices will be infected with probability 1 if the root of the tree is infected. Thus, for any seed set $S \subseteq \overline{V}$, by assuming all vertices in the tree are infected (in the sake of finding an upper bound for $\sigma(S)$), we have $\sigma(S) \leq \frac{1}{d^d} \cdot |S| \cdot W + \sum_{i=0}^{d} d^i < \frac{|S|W}{d^d} + d^{d+1}$. This gives an upper bound for the performance of both the non-adaptive greedy algorithm and the non-adaptive optimal seed set.

Now, we consider the greedy adaptive policy. If the root is not infected, there always exists a path from a certain leaf to the root such that all vertices on the path are not infected. This is because, if all children of an internal node w are infected, w will be infected with probability 1 (as $f_w = d \times \frac{1}{d} = 1$ which will always be no smaller than θ_w). In other words, if an internal node is uninfected, at least one of its children is uninfected. It is easy to see that, before the root is infected, the greedy adaptive policy will always choose a leaf such that all vertices on the path between the leaf and the root are uninfected.

Next, we evaluate the expected number of seeds required to infect the root, under the greedy adaptive policy. Suppose the tree only has two levels (i.e., a star). The number of seeds among the leaves required to infect the root is a random variable with uniform distribution on $\{1, \ldots, d\}$, with expectation $\frac{d+1}{2}$. By induction on the number of levels of the tree, with a d-level tree as it is in our case, the expected number of seeds required to infect the root is $(\frac{d+1}{2})^d$, which equals to $\frac{k}{2}$. By Markov's inequality, the k seeds chosen according to the greedy adaptive policy will infect the root with probability at least 1/2. Therefore, $\sigma^{\mathfrak{f}}(\pi^g, k) \geq \frac{1}{2}W$, and the optimal adaptive policy can only be better: $\max_{\pi\in\Pi}\sigma^{\mathfrak{f}}(\pi, k) \geq \sigma^{\mathfrak{f}}(\pi^g, k) \geq \frac{1}{2}W$.

Putting together, both the adaptivity gap and the supremum of the greedy adaptivity gap is at least $\frac{\frac{1}{2}W}{\frac{kW}{d^d}+d^{d+1}} = \frac{\frac{1}{2}W}{\frac{1}{2^{d-1}}(1+\frac{1}{d})^d W + d^{d+1}.} = \Omega(2^d)$, if setting $W = d^{d+10} \gg d^{d+1}$. The lemma concludes by noticing $d = \Omega(\frac{\log|V|}{\log\log|V|})$. □

4.2 Proof of Theorem 4.1, 4.2

To prove Theorem 4.1 and Theorem 4.2, we construct an INFMAX instance with a special triggering model $I_{G,F}$ which is a combination of ICM and LTM.

Definition 4.6. The *mixture of ICM and LTM* is a triggering model $I_{G,F}$ where $G = (V, E, w)$ is an edge-weighted graph with $w(u, v) \in (0, 1]$ for each $(u, v) \in E$ and each vertex v is labelled either **IC** or **LT** such that T_v is sampled according to $\mathcal{F}_v$ described in Definition 2.2 if v is labelled **IC** and T_v is sampled according to $\mathcal{F}_v$ described in Definition 2.3 if v is labelled **LT**. In addition, each vertex v labelled L satisfies $\sum_{u \in \Gamma(v)} w(u, v) \leq 1$.

To prove Theorems 4.1 and 4.2, we want to leverage the tree construction of Lemma 4.5. To do so, we make M copies of the tree construction in Lemma 4.5 for a large M and a separate set of vertices A connecting to the leaves of these trees such that each vertex in A is connected to exactly one leaf in each tree. In particular, we have $|A| = L^M$, where $L = d^d$ is the number of leaves in each tree. All the leaves in each of the M copies of the tree are labelled **IC**, and the remaining vertices are labelled **LT**. The weight of each edge from a vertex u in A to a leaf node is 1, guaranteeing the activation of the leaf when u is selected as a seed. Intuitively, each vertex in A simulates the action that a leaf is chosen as a seed in each tree and M seeds are chosen simultaneously. By choosing M large enough, we can show that both the adaptive and non-adaptive greedy algorithm will only select vertices in A as the first k seeds, and both the adaptivity gap and the supremum of the greedy adaptivity gap are infinite, as they are in Lemma 4.5. The full proof of Theorems 4.1 and 4.2 is in the full version of this paper.

5 A Variant of Greedy Adaptive Policy

Although we have seen that the adaptive version of the greedy algorithm can perform worse than its non-adaptive counterpart, in general, we would still recommend the use of it as long as it is feasible, as it can also perform significantly better than the non-adaptive greedy algorithm (Theorem 4.1) while never being too bad (Theorem 3.4). As we remarked, the adaptivity may be harmful because exploiting the feedback may make the seed-picker too myopic. In this section, we propose a less aggressive risk-free version of the greedy adaptive policy, π^{g-}, in that it balances between the exploitation of the feedback and the focus on the average in the conventional non-adaptive greedy algorithm.

First, we apply the non-adaptive greedy algorithm with $|V|$ seeds to obtain an order $\mathcal{L}$ on all vertices. Then for any $S \subseteq V$ and any partial realization φ, $\pi^{g-}(S, \varphi)$ is defined to be the first vertex v in $\mathcal{L}$ that is not known to be infected. Formally, v is the first vertex in $\mathcal{L}$ that are not reachable from S when removing all edges e with $\varphi(e) \in \{\mathtt{B}, \mathtt{U}\}$. This finishes the description of the policy.

This adaptive policy is always no worse than the non-adaptive greedy algorithm, as it is easy to see that those seeds chosen by π^g are either seeded or infected by previously selected seeds in π^{g-}.

However, π^{g-} can sometimes be conservative. It is possible that π^{g-} has the same performance as the non-adaptive greedy algorithm, but π^g is much better. Especially, when there is no path between any two vertices among the first k vertices in $\mathcal{L}$, π^{g-} will make the same choice as the non-adaptive greedy algorithm. The INFMAX instance in Sect. 4.2 is an example of this.

We have seen that π^{g-} sometimes performs better than π^g (e.g., in those instances constructed in the proofs of Lemma 3.2 and Lemma 3.3) and sometimes performs worse than the π^g (e.g., in the instance constructed in Sect. 4.2). Therefore, given a *particular* INFMAX instance, for deciding which of π^{g-} and π^g to be used (we should never consider the non-adaptive greedy algorithm if adaptivity is available, as it is always weakly worse than π^{g-}), we recommend a comparison of the two policies by simulations. Notice that the seed-picker can randomly sample a realization ϕ and simulate the feedback the policy will receive. Thus, given $I_{G,F}$, both π^{g-} and π^g can be estimated by taking an average over the numbers of infected vertices in a large number of simulations.

6 Conclusion and Open Problems

We have seen that the infimum of the greedy adaptivity gap is exactly $(1 - 1/e)$ for ICM, LTM, and general triggering models with both the full-adoption feedback model and the myopic feedback model. We have also seen that the supremum of this gap is infinity for the full-adoption feedback model. One natural open problem is to find the supremum of the greedy adaptivity gap for the myopic feedback model. Another natural open problem is to find the supremum of the greedy adaptivity gap for the more specific ICM and LTM.

The greedy adaptivity gap studied in this paper is closely related to the adaptivity gap studied in the past. Since the non-adaptive greedy algorithm is always a $(1 - 1/e)$-approximation of the non-adaptive optimal solution, a constant adaptivity gap implies a constant greedy adaptivity gap. For example, the adaptivity gap for ICM with myopic feedback is at most 4 (Peng and Chen 2019), so the greedy adaptivity gap in the same setting is at most $\frac{4}{1-1/e}$. In addition, the greedy adaptive policy is known to achieve a $(1 - 1/e)$-approximation to the adaptive optimal solution for ICM with full-adoption feedback (Golovin and Krause 2011), so the adaptivity gap and the greedy adaptivity gap could either be both constant or both unbounded for ICM with full-adoption feedback model, but it remains open which case is true. The adaptivity gap for ICM with full-adoption feedback, as well as the adaptivity gap for LTM with both feedback models, are all important open problems. We believe these problems can be studied together with the greedy adaptivity gap.

References

Angell, R., and Schoenebeck, G. 2016. Don't be greedy: Leveraging community structure to find high quality seed sets for influence maximization. *WINE*.

Arora, A.; Galhotra, S.; and Ranu, S. 2017. Debunking the myths of influence maximization. In *Proceedings of the 2017 ACM International Conference on Management of Data-SIGMOD'17*.

Bharathi, S.; Kempe, D.; and Salek, M. 2007. Competitive influence maximization in social networks. In *WINE*.

Borgs, C.; Brautbar, M.; Chayes, J.; and Lucier, B. 2014. Maximizing social influence in nearly optimal time. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, 946–957. SIAM.

Chen, W., and Peng, B. 2019. On adaptivity gaps of influence maximization under the independent cascade model with full adoption feedback. Technical report, arXiv:1907.01707.

Chen, W.; Lakshmanan, L. V.; and Castillo, C. 2013. Information and influence propagation in social networks. *Synthesis Lectures on Data Management* 5(4):1–177.

Chen, W.; Wang, Y.; and Yang, S. 2009. Efficient influence maximization in social networks. In *ACM SIGKDD*, 199–208. ACM.

Chen, W.; Yuan, Y.; and Zhang, L. 2010a. Scalable influence maximization in social networks under the linear threshold model. In *Data Mining (ICDM), 2010 IEEE 10th International Conference on*, 88–97. IEEE.

Chen, W.; Yuan, Y.; and Zhang, L. 2010b. Scalable influence maximization in social networks under the linear threshold model. In *2010 IEEE International Conference on Data Mining*, 88–97. IEEE.

Cheng, S.; Shen, H.; Huang, J.; Zhang, G.; and Cheng, X. 2013. Staticgreedy: solving the scalability-accuracy dilemma in influence maximization. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, 509–518. ACM.

Domingos, P., and Richardson, M. 2001. Mining the network value of customers. In *ACM SIGKDD*.

Galhotra, S.; Arora, A.; and Roy, S. 2016. Holistic influence maximization: Combining scalability and efficiency with opinion-aware models. In *Conference on Management of Data*, 743–758. ACM.

Goldberg, S., and Liu, Z. 2013. The diffusion of networking technologies. In *SODA*.

Golovin, D., and Krause, A. 2011. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *Journal of AI Research* 42:427–486.

Goyal, A.; Lu, W.; and Lakshmanan, L. V. 2011a. Celf++: optimizing the greedy algorithm for influence maximization in social networks. In *Proceedings of the 20th international conference WWW*, 47–48. ACM.

Goyal, A.; Lu, W.; and Lakshmanan, L. V. 2011b. Simpath: An efficient algorithm for influence maximization under the linear threshold model. In *Data Mining (ICDM), 2011 IEEE 11th International Conference on*, 211–220. IEEE.

Han, K.; Huang, K.; Xiao, X.; Tang, J.; Sun, A.; and Tang, X. 2018. Efficient algorithms for adaptive influence maximization. *Proceedings of the VLDB Endowment* 11(9):1029–1040.

Jung, K.; Heo, W.; and Chen, W. 2012. Irie: Scalable and robust influence maximization in social networks. In *Data Mining (ICDM), 2012 IEEE 12th International Conference on*, 918–923. IEEE.

Kempe, D.; Kleinberg, J. M.; and Tardos, É. 2003. Maximizing the spread of influence through a social network. In *ACM SIGKDD*, 137–146.

Kempe, D.; Kleinberg, J. M.; and Tardos, É. 2005. Influential nodes in a diffusion model for social networks. In *ICALP*, 1127–1138.

Leskovec, J.; Krause, A.; Guestrin, C.; Faloutsos, C.; VanBriesen, J.; and Glance, N. 2007. Cost-effective outbreak detection in networks. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, 420–429. ACM.

Li, Y.; Fan, J.; Wang, Y.; and Tan, K.-L. 2018. Influence maximization on social graphs: A survey. *IEEE Trans. on Knowledge and Data Engineering* 30(10):1852–1872.

Mossel, E., and Roch, S. 2010. Submodularity of influence in social networks: From local to global. *SIAM J. Comput.* 39(6):2176–2188.

Nemhauser, G. L.; Wolsey, L. A.; and Fisher, M. L. 1978. An analysis of approximations for maximizing submodular set functions. *Mathematical Programming* 14(1):265–294.

Ohsaka, N.; Akiba, T.; Yoshida, Y.; and Kawarabayashi, K.-i. 2014. Fast and accurate influence maximization on large networks with pruned monte-carlo simulations. In *AAAI*, 138–144.

Peng, B., and Chen, W. 2019. Adaptive influence maximization with myopic feedback. Technical report, arXiv:1905.11663.

Richardson, M., and Domingos, P. 2002. Mining knowledge-sharing sites for viral marketing. In *ACM SIGKDD*, 61–70.

Schoenebeck, G., and Tao, B. 2017. Beyond worst-case (in) approximability of nonsubmodular influence maximization. In *International Conference on Web and Internet Economics*, 368–382. Springer.

Schoenebeck, G., and Tao, B. 2019a. Beyond worst-case (in)approximability of nonsubmodular influence maximization. *ACM Trans. Comput. Theory* 11(3):12:1–12:56.

Schoenebeck, G., and Tao, B. 2019b. Influence maximization on undirected graphs: Towards closing the $(1-1/e)$ gap. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, 423–453. ACM.

Schoenebeck, G.; Tao, B.; and Yu, F.-Y. 2019. Think globally, act locally: On the optimal seeding for nonsubmodular influence maximization. In *RANDOM 2019: International Conference on Randomization and Computation*.

Tang, J.; Tang, X.; Xiao, X.; and Yuan, J. 2018. Online processing algorithms for influence maximization. In *Proceedings of the 2018 International Conference on Management of Data*, 991–1005. ACM.

Tang, Y.; Shi, Y.; and Xiao, X. 2015. Influence maximization in near-linear time: A martingale approach. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, 1539–1554. ACM.

Tang, Y.; Xiao, X.; and Shi, Y. 2014. Influence maximization: Near-optimal time complexity meets practical efficiency. In *SIGMOD international conference on Management of data*, 75–86. ACM.