Reachability-Aware Fair Influence Maximization

Wenyue Ma[1(✉)], Maximilian K. Egger[1], Andreas Pavlogiannis[1], Yuchen Li[2], and Panagiotis Karras[1,3]

[1] Department of Computer Science, Aarhus University, Aarhus, Denmark
wenyuema@cs.au.dk

[2] School of Computing and Information Systems, Singapore Management University, Singapore, Singapore

[3] Department of Computer Science, University of Copenhagen, Copenhagen, Denmark

Abstract. How can we ensure that an information dissemination campaign reaches every corner of society and also achieves high overall reach? The problem of maximizing the *spread of influence* over a social network has commonly been considered with an *aggregate* objective. Less attention has been paid to achieving equality of opportunity, reducing information barriers, and ensuring that everyone in the network has a fair chance to be reached. To that end, the *fairness* objective aims to maximize the minimum probability of reaching an individual. To address this inapproximable problem, past research has proposed heuristics, which, however, perform less well when the promotion budget is low and achieve fairness at the expense of overall welfare. In this paper, we propose novel reachability-aware algorithms for the fairness-oriented IM problem. Our experimental study shows that our algorithms outperform past work in challenging real-world problem instances by up to a factor of 4 in terms of the fairness objective and strike a balance between fairness and total welfare, even while no solution is universally superior across data, influence probability models, and propagation models.

1 Introduction

Information dissemination plays a critical role in societal welfare, as in illness prevention [27], welfare distribution [2], and the dissemination of scholarship opportunities [24]. In that regard, the task of *Influence Maximization* (IM) seeks to identify influential initiators within a *budget* that maximize influence spread, a form of *utility*, in a network [5,6,14,16,21,25,26]. However, the maximization of spread is unavoidably biased toward individuals who are easier to reach and already well-informed, hence exacerbates information asymmetries among individuals or groups, which reinforce cultural divides and class stratification [11,12,15]. For instance, on job platforms, those with more connections may find better opportunities [8]. Similarly, marginalized groups may miss crucial information about HIV prevention [27]. When allocating scholarships, organizations aim to benefit deserving individuals. However, some deserving scholars may lack access to information

W. Zhang et al. (Eds.): APWeb-WAIM 2024, LNCS 14963, pp. 342–359, 2024.
https://doi.org/10.1007/978-981-97-7238-4_22

channels that would apprise them of the availability of scholarship opportunities. Such asymmetries introduce a unique form of inequality. Therefore, ensuring fairness [10,17] by bridging information access gaps becomes crucial in promoting an equitable and cohesive society [1,7,8,13,22,27,29].

Most works addressing this information gap problem aim at fairness with respect to predefined groups [7,9,13,22,23,27–29]. However, the clustering of users into groups avoids part of the problem, as some of them may still face disregard within a group. In this paper, we focus on the question of *Fair Influence Maximization* (FairIM), aiming at opportunity equity at the *individual* node level. One way to address individual fairness is the *ex-ante* method [1], which considers how fair the allocation is in advance. Contrariwise, we provide a deterministic strategy for choosing seed nodes aiming at *ex-post* fairness, i.e., considering how fair the *measurable* outcome is *in retrospect*, after the allocation. Fish et al. [8] proposed a heuristic to settle individual fairness, yet did not consider *global welfare*. We aim to ensure that all participants get an equitably high chance to receive some information, announcement, or call, while still gaining high overall information reach, given a *limited budget* of diffusion initiators.

The FairIM problem raises several challenges: First, the problem is NP-hard and hard to approximate to within any constant factor unless P = NP [8]. Second, while fairness is desirable, the total expected reach should not be severely compromised for the sake of fairness. Third, under a limited budget, it becomes challenging to even ensure that the minimum probability of influence rises above zero, particularly in real-world settings where the budget is often low. For example, an agency that seeks to select 1% of potential influencers in a network can face a challenge in ensuring everyone has a fair chance to be reached. Last, its objective function corresponds to *Robust Submodular Observation Selection* (RSOS) [18], hence a conventional submodularity-based approximation guarantee does not apply. Thus, this problem calls for new approaches.

Our Contributions and a Road map. We delve into the problem of *Fair Influence Maximization* (FairIM) and present a group of solutions aptly balance fairness and global welfare under a budget.

- In Sect. 2, we define the FairIM problem and its MaxMin objective at the individual level and state its hardness.
- In Sect. 3, we introduce our heuristic methods solving FairIM. Firstly, we propose a `greedy` baseline. Secondly, we propose `uplift`, a reachability-aware seed selection method that caters to both utility and fairness at low seed budgets. In Sect. 3.2, we enhance `uplift` with a tie-breaking variant, `uplift+`, and a local-search variant, `upliftX`. Thirdly, we propose `super` (Sect. 3.3), a hybrid heuristic method that achieves benefits on both low-budget and high-budget stages.
- In Sect. 4, we present a comprehensive experimental study demonstrating that `uplift` and its variants strike a desirable balance between fairness and global welfare. Remarkably, in challenging cases where the seed budget is low and information propagation is weak, our solutions significantly outperform `myopic` [8], reaching up to a factor of 4 when the budget is 10% of the nodes.

Further, our baseline `greedy` solution performs as well as `saturate` [18], which has a strong theoretical approximation guarantee, and is faster too. In this study, we conduct large-scale experiments on data from four diverse domains and two synthetic datasets under varying edge distribution, encompassing up to 81K entries, well beyond previous works' experimental studies that typically reach sizes of 10K [1,7,8].

2 Problem Statement

In this section, we introduce basic notions and spread models, formally define the problem of *Fair Influence Maximization* (FairIM), and analyze its hardness (Table 1).

Table 1. Notations.

Symbol	Description
$G(V,E)$	Graph G consisting of a edge set $v \in V$ and a node set $e \in E$
G_i	Deterministic graph instance
$\bar{d}$, $\hat{d}$	Average degree and largest degree in G
θ_v	Active threshold of node v in LT model
$p_{v,u}$	Probability on edge (v,u)
S	Candidate seed set
x	$x \in \mathcal{X}$, one possible world in universe $\mathcal{X}$
$I_S(v)$	The probability for node v under seed S
$\sigma(S)$	Expected number of nodes reachable from S
$\sigma_x(S)$	Expected number of nodes reachable from S in possible world x
$\mathbb{I}[v]$	Expectation of v being active, $\mathbb{E}[I_S(v)]$
$\mathbb{I}[V]$	Vector consisting of $\mathbb{I}(v)$, $v \in V$
$\Delta(v\|S)$	Influence increment when adding v to S.

Definition 1. (Global Welfare). *Given a social graph $G(V,E)$ of directed edges annotated with probability values p reflecting the strength of connections, and an integer k, the problem of Global Welfare, seeks a set S of up to k* seed *nodes that maximizes a spread function $\sigma(S) = \sum_{v \in V} \mathbb{E}[I_S(v)]$.*

We denote the probability that node $v \in V$ is influenced from seed set S as $I_S(v)$. Standard concentration bounds show that this probability can be estimated accurately with relatively few repetitions of a diffusion process [16]. Given t deterministic instances $\{G_i\}$, $1 \leq i \leq t$ of probabilistic graph G, we can estimate $I_S(v)$ as $\frac{\sum_{i=1}^{t} I_S^{G_i}(v)}{t}$, where $I_S^{G_i}(v)$ is an indicator function denoting whether an influence cascade emanating from seed set S by a given spread model on deterministic graph instance G_i reaches node v. We may compute the influence spread function $\sigma(S) = \sum_{v \in V} \mathbb{E}[I_S(v)]$ by Monte Carlo simulations, estimating spread of set of nodes S over t instances [16].

We focus on two popular spread model, the *Independent Cascade* (IC) and *Linear Threshold model* (LT) models [16], both arising from mathematical sociology. By both models, the spread function $\sigma(S)$ is monotonic and submodular, yet evaluating $\sigma(S)$ is #P-hard [3,4].

The *Independent Cascade Model* (IC) introduces an influence probability $p_{u,v} \in [0,1]$ for each edge $e_{u,v}$, representing the likelihood that v is successfully activated by u. When u becomes active at a time step t, it has a chance to independently activate its inactive neighbors v at time $t+1$ with $p_{u,v}$.

By the *Linear Threshold Model* (LT), the sum of incoming edge weights at any node is at most 1, and every node v chooses an activation threshold $\theta_v \in [0,1]$ uniformly at random. The diffusion operates in discrete time steps. In step t, nodes that were active in step $t-1$ remain active, and any node v becomes active if the sum of edge weights from its active incoming neighbors is at least θ_v; equivalently, it chooses at most one incoming edge proportionally to its weight and becomes active if the corresponding neighbor is active [16].

2.1 Fairness in Influence Maximization

We may optimize a *welfare* function to improve access to information *in the aggregate*. However, beyond and in addition to global welfare, we are concerned about individuals who lag behind and become *disadvantaged*. We define the problem of MaxMin Fairness, which caters to those individuals.

Definition 2. (*MaxMin* Fairness). *The MaxMin Fairness problem in graph $G(V,E)$ seeks a set of* seed *nodes S that maximizes* $\min_{v \in V} \mathbb{E}[I_S(v)]$, *with the constraint $|S| \leq k$, where smaller values of k are preferred.*

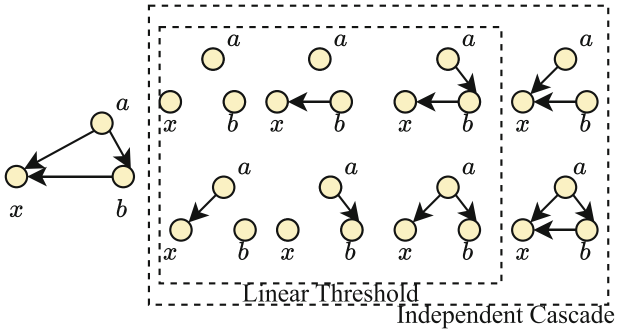


Fig. 1. Example graph and possible worlds.

We aim to lift up the welfare of the least advantaged individual. To that end, an algorithm should focus on reducing the number of disadvantaged individuals, rather than merely reducing some influence gap between arbitrary individuals.

This orientation is well aligned with our goal, as the decrease of such individuals eventually improves fairness. For instance, consider we care about individual nodes a, b, x, where node a is easily reached, and the gap between a and b is small ($I_S(a) \approx I_S(b)$), but reaching x is challenging due to its disadvantaged status. One solution might reduce the gap between a and b to equalize their influence ($I_S(a) = I_S(b)$); however, that would not address the need to raise the influence on x. Therefore, an algorithm should focus on decreasing the number of disadvantaged individuals to eventually eliminate them.

Example. Let $\min_{v \in V} I_S(v)$ be function $f(S)$. Consider a directed graph of three nodes $\{a, b, x\}$ as in Fig. 1, with $A = \{b\}$, $B = \{a, b\}$ as seed sets. Figure 1 shows the possible worlds of different activated edges. By the IC model, we set the probability for each edge to $1/2$, which makes each of the eight worlds equally probable with a probability of $1/8$. The probabilities that $\{a, b, x\}$ receive influence are $\{0, 1, 1/2\}$ under seed set A and $\{1, 1, 3/4\}$ under seed set B, hence $f(A) = \min\{0, 1, 1/2\} = 0$ and $f(B) = \min\{1, 1, 3/4\} = 3/4$. Similarly, by the LT model, each node picks at most one incoming edge, hence there are six possible worlds. We set the probability of edge (a, b) to $1/2$ and that of the other two edges to $1/3$, rendering all six worlds equally probable with probability $1/6$. Following the same configuration as for the IC model above, $f(A) = \min\{0, 1, 1/3\} = 0$ and $f(B) = \min\{1, 1, 2/3\} = 2/3$. Further, it holds that $f(A \cup x) - f(A) < f(B \cup x) - f(B)$ in both models, hence the MaxMin objective is non-submodular[1].

Table 2. Fairness notions for information dissemination.

Method	Objective	Objective Function	Indiv.
Frank-Wolfe [27]	MaxMin g	$\max_{S \subseteq V, \lvert S\rvert \le k} \min_{g_i} \frac{\sigma_{g_i}(S)}{\lvert g_i\rvert}$	✗
	Diversity	$\max_{S \subseteq V, \lvert S\rvert \le k} \begin{cases} \sigma(S), \text{if } \forall i\ \sigma_{g_i}(S) \ge \sigma_{g_i}(k_i) \\ 0, \text{otherwise} \end{cases}$	✗
Mixed Integer Programming (MIP) [7]	Equality	$\frac{\lvert S \cap V_{g_i}\rvert}{k} \approx \frac{\lvert g_i\rvert}{\lvert V\rvert}$	✗
	Equity	$\frac{\sigma_{g_i}(S)}{\sigma(S)} \approx \frac{\lvert g_i\rvert}{\lvert V\rvert}$	✗
	MaxMin g	$\max_{S \subseteq V, \lvert S\rvert \le k} \min_{g_i} \frac{\sigma_{g_i}(S)}{\lvert g_i\rvert}$	✗
	Diversity	$\sigma_{g_i}(S) \ge OPT_g$	✗
`Greedy` [22]	MaxMin g	$\max_{S \subseteq V, \lvert S\rvert \le k} \min_{g_i} \lvert g_i\rvert \sigma_{g_i}(S)$	✗
`Saturate` [29]	MaxAvg g	$\max_{S \subseteq V, \lvert S\rvert \le k} \min_{g_i} \frac{1}{\lvert g_i\rvert} \sigma_{g_i}(S)$	✗
`Random` [1]	Ex-ante	$\max_{S \subseteq V} \min_{g_i} \sigma_{g_i}(S)$	✓
`myopic` [8]	MaxMin v	$\max_{S \subseteq V, \lvert S\rvert \le k} \min_{v \subseteq V} \mathbb{E}(I_S(v))$	✓
`uplift`	MaxMin v	$\max_{S \subseteq V, \lvert S\rvert \le k} \max \min_{v \subseteq V} \mathbb{E}(I_S(v))$	✓

[1] A set function $f : 2^V \to \mathbb{R}$ is *submodular* iff $\forall x \in V \setminus A \subseteq B \subseteq V$, $f(A \cup x) - f(A) \ge f(B \cup x) - f(B)$.

Table 2 presents some representative notions of fairness and methods used in existing works on FairIM. Such works mainly focus on *group-oriented* fairness [7,13,22,27,29]. A group or community g_i can receive a partial budget of $k_i < k$ seed nodes, while $\sigma_{g_i}(S)$ denotes the expected spread by seed S within group g_i. Individual fairness is a special case of group fairness with $|g_i| = |V|$. However, group fairness seed selection strategies rely on a budget allocated to each group [7,27], which cannot transfer into individual fairness under a low budget. Therefore, it is imperative to develop individual fairness-oriented algorithms, especially for low budgets.

2.2 Hardness

By reduction from the Set Cover problem, FairIM is NP-hard [8]. Further, when the probability of transmission among nodes is $p < \frac{\sqrt{5}-1}{2}$, the MaxMin objective cannot be approximated better than $O(p)$ [8, Theorem 4.1]. Since Set Cover is $O(\ln n)$-inapproximable, we can only approximate the optimal set of k seeds using an additional $O(\ln n)$-factor seed budget.

3 Achieving Influence Fairness

As FairIM is an inapproximable [8] robust *Submodular Observation Selection Problem* [18], we address it heuristically. We seek a seed set S that maximizes the minimum probability to influence a node: $S = \arg\max_{S \subseteq V, |S| < k} \min_{v \in V} \mathbb{E}[I_S(v)]$. In Sect. 3.1 we describe a `greedy` baseline. In Sect. 3.2 we present our novel `uplift` algorithm and its variants. Additionally, we propose the `super` framework in Sect. 3.3.

Algorithm 1: `Greedy`

```
Function Greedy(G, k, t, ε):
    S = ∅                                              // initialize S as empty
    while S == ∅ or |S| ≤ k do
        p_min = 0, count = +∞        // initialize temporary minimum and counter
        for v ∈ S \ V do
            𝕀[V] = MC(G, S ∪ {v}, t)
            ξ = min_{v∈V} 𝕀[v]                // minimum node influence probability
            T = {u ∈ V | 𝕀[u] ∈ [ξ, ξ + ε]}, η = |T|
            if p_min == ξ and η ≤ count then
                v_node = v, count = η     // for nodes yielding same minimum influence
                                             probability, choose one with smallest set size
            if p_min > ξ then
                v_node = v, p_min = ξ, count = η
        S = S ∪ {v_node}
    return S
Function MC(G, S, t):
    for v ∈ S, C[v] = t; for v ∈ V \ S, C[v] = 0         // initialize each entry in C
    for i ≤ t, i++ do
        Q = Queue(S) while Q ≠ ∅ do
            v = Q.pop()
            for u ∈ N_v do
                p'_{v,u} ~ U(0, 1) if p'_{v,u} < p_{v,u} then
                    Q.push(u), C[u]++               // update counter if node activated
    for v ∈ V, 𝕀[v] = C[v]/t
    return 𝕀[V]                                         // return expectation vector
```

3.1 Greedy Baseline

We present a `greedy` algorithm that selects the seed set S with a view towards individual fairness. Algorithm 1 presents the pseudocode. The key variable is ξ, standing for the minimum influence probability obtained when adding a candidate node v into seed set S; in each step, we add the candidate node yielding with the highest ξ value. Still, some seed candidates may yield the same ξ value, while the greedy selection does not consider how many influenced nodes have the minimum influence probability ξ. The growth of ξ in the first few steps will be small; at the start, we may get $\xi = 0$. To address this predicament, we select, in each iteration (Lines 9–12), the candidate node, among those having the same highest ξ value, that yields the *least* target nodes of minimum influence probability ξ within a tolerance threshold ϵ, as defined in Line 8.

Influence Spread Simulation. The evaluation of influence spread, $\sigma(S)$, is #P-hard, hence so is the calculation of the expected influence received by a node v under S, $\mathbb{E}(I_S(v)) = \mathbb{I}[v]$. Still, Monte Carlo (MC) simulation returns a solution with a constant bounded ratio of approximation [20]. Lines 14– 24 employ MC simulation to count activations of v over t rounds by the IC model; we apply a similar simulation for the LT model.

3.2 UpLift Approach.

While the `greedy` algorithm is efficacious, it neglects network reachability properties. Here, we propose our `uplift` algorithm, which is grounded in and attends to the reachability of *disadvantaged* nodes. It works with a *target* node set V_t of the most disadvantaged nodes and finds a reverse-reachable node set V_r, from which we can reach nodes in V_t in one hop.

Algorithm 2: `uplift`

```
1  Function uplift (G, k, t, ϵ):
2      V_t = V, S = ∅
3      while |S| < k do
4          C_r = Reachability(G, V_t, S)
5          v = arg max_{u∈V\S} C_r[u]
             // Tie-uplift applies Eq. (1)
6          𝕀[V] = MC(G, S ∪ {v}, t)
7          V_t ← {u ∈ V | 𝕀[u] ∈ [ξ, ξ+ϵ]}
8      return S
9  Function Reachability(G, V_t, S):
10     for v ∈ V \ S, C[v] = 0
11     for v ∈ V_t do
12         C[v] + +            // update counters
13         for u ∈ N_v \ S      // neighbors
14          do
15             C[u] + +
16     return C
```

Algorithm 3: `super`

```
1  Function super(G, k, t, ϵ):
2      S = {v_d̂}        // highest-degree node
3      𝕀[V] = MC(G, S, t)
4      while |S| < k do
5          v^M = arg min_{v_i∈V\S} 𝕀[V]
             // node chosen by myopic
6          ξ = min_{v∈V} 𝕀[v]
7          V_t = {u ∈ V | 𝕀[u] ∈ [ξ, ξ + ϵ]}
8          C_r = Reachability(G, V_t, S)
9          v^R = arg max_{v_i∈V\S} C_r(v_i)
             // node chosen by uplift
10         if Δ(v^M|S) ≥ Δ(v^R|S) then
11             v = v^M
12         else v = v^R
13         𝕀[V] = MC(G, S ∪ {v}, t)
14     return S
```

Naïve UpLift. Algorithm 2 illustrates `uplift`. In each iteration, we identify the set V_t of target nodes having influence probability within a small tolerance threshold ϵ of the minimum (Line 7). To cater to these *disadvantaged* nodes (initially, all nodes in V), we add to the seed set S the candidate seed node v that reaches the most target nodes $V_r[v]$, $v \in V$, as in Line 5.

For each target node $v_t \in V_t$, we increase the counter of their one-hop incoming neighbors, as function `Reachability` in Lines 8–16 shows, hence count the appearances of each node v_r as a neighbor of a target node v_t (Line 15).

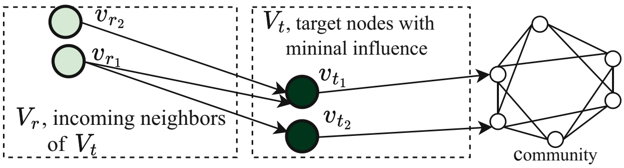


Fig. 2. Diffusion from V_r to V_t in `uplift`

Figure 2 shows an example with two disadvantaged nodes, v_{t_1} and $v_{t_2} \in V_t$. If we select a seed node in the community, then every node in it is easily activated. However, nodes in V_t are hard to activate, as they are hard to reach by the community. Target node v_{t_1} has incoming edge $e(v_{r_1}, v_{t_1})$ with neighbor v_{r_1}, while v_{t_2} has incoming neighbors v_{r_1}, v_{r_2} from edges $e(v_{r_1}, v_{t_2}), e(v_{r_2}, v_{t_2})$. Thus v_{r_1} and $v_{r_2} \in V_r$ are two reverse-reachable nodes for V_t. Among them, only v_{r_1} can reach both target nodes, hence we select that node as the next seed node.

Tie UpLift, `uplift+`. Naïve `uplift` breaks ties in its selection randomly. To improve upon it, we introduce a tie-breaking variant `uplift+` (Algorithm 2, Line 5). Among candidate seed nodes having equal reachability counts, we pick the one of *lowest* own influence probability, as in Eq. (1). This way we advance the most disadvantaged nodes in terms of both reaching them and turning them to seeds.

$$\underset{w \in \{v \in V \setminus S | V_r[v] = \max\limits_{u \in V \setminus S} V_r[u]\}}{\arg\min} I(w) \tag{1}$$

Local Search UpLift, `upliftX`. To further enhance the outcome of `uplift`, we add a *local search* component to it, as Algorithm 4 illustrates. After we obtain a k size set with `uplift`, we apply local refinements on the seed set to enhance the influence function $\sigma(\cdot)$. We repeatedly remove the node yielding the smallest influence loss and add a candidate node with the highest marginal gain $\Delta(v|S)$. This swapping proceeds until the node we add is the same as the node removed, whereupon the loop stops.

Algorithm 4: `upliftX`

Function `upliftX` (G, k, S, t, ϵ):
  v = last seed selected by `uplift`
  do
    $v' = \arg\min_{u \in S} \Delta(u|S \setminus u)$, $S = S \setminus v'$ `// node of smallest loss`
    $v = \arg\max_{u \in V \setminus S} \Delta(u|S)$, $S = S \cup v$ `// node of largest gain`
  while $v' \neq v$
  return S

3.3 Super Approach

We now define a method that combines the benefits of the `uplift` methods with those of the `myopic` solution [8], since, as we find, `uplift` performs well at challenging low budgets while `myopic` does well at larger budgets. As both `uplift` and `myopic` invoke MC simulation by `icExp` in each round, we can as well select a candidate seed by the `myopic` strategy, which simply picks the most disadvantaged node as seed. The `super` strategy, shown in Algorithm 3, chooses, among these two approaches, the one that yields the best result regarding marginal gain in terms of the MaxMin objective.

We choose the first node as the highest-degree node. Thereafter, we estimate influence probabilities by `icExp`, and choose target nodes having the minimum probability within a threshold (Line 7). We identify the chosen candidate by `myopic` in Line 5 and by `uplift` in Line 9, and compare their marginal gains $\Delta(v^M|S) = \texttt{icExp}(G, S \cup \{v^M\}, t)$ and $\Delta(v^R|S) = \texttt{icExp}(G, S \cup \{v^R\}, t)$ to pick the candidate with larger marginal gain among those two. To further narrow down the search among reachable set, we expand the `super` algorithm to a `super*` variant, which uses the same tie-breaking rule as in `uplift+`.

Complexity. While `saturate` [18] achieves bicriterion guarantees in $\mathcal{O}(|V|^{\log\log|V|})$ time, `greedy` is the most time-consuming algorithm, having quadratic complexity $\mathcal{O}(c \cdot |V|^2)$. The complexity of `uplift` is $\mathcal{O}(kt\bar{D}|V|)$, where $\bar{D}$ is the average degree and t the number of MC iterations; `uplift+` is faster as it narrows down the search space. Local search `upliftX`, on the other hand, is sensitive to the seed selection by `uplift`. The time complexity of `super` is the sum of `myopic` and `uplift`, hence $\mathcal{O}(kt\bar{D}|V|)$, rendering it still linear in $|V|$.

4 Experimental Study

Experiment Setting. *Algorithms.* We conduct a thorough experimental study juxtaposing the following algorithms: `random` [1], which selects a random $S, |S| = k$; `myopic` [8], which iteratively picks the vertex of smallest influence probability; `saturate` [18], which uses a binary search procedure maintaining a search interval and provides a fairness guarantee [29]; `greedy` (Algorithm 1); `uplift` (Algorithm 2); `uplift+`, the tie-resolving variant of `uplift`; `super` (Algorithm 3), which combines `myopic` [8] and `uplift`; `super*`, the tie-resolving variant of `super`; and `upliftX`, which enhances upon `uplift` by local search (Algorithm 4).

Table 3. Dataset details

(a) Dataset

	Small		Large			
	SPA [1]		**bitcoin** [19]		**social network**	
	spa0	**spa10**	**alpha**	**otc**	**facebook** [19]	**twitter** [19]
$\|V\|$	500	500	3k	5k	4k	81k
$\|E\|$	1.6k	1.6k	24k	35k	88k	1.7M
Directed	✓	✓	✓	✓	✗	✓
$\bar{D}$	6.7	6.6	7.5	7.3	43.69	43.49
maximal d	47	41	510	795	1045	3758

(b) Edge Distribution

$p_{v,u}$	**edge distribution**
xp	$p_{v,u} \sim (1/8)$
tp	$p_{v,u} \sim (1/4, 1/16, 1/64)$
htp	$p_{v,u} \sim (1/2, 1/16, 1/64)$
hp	$p_{v,u} \sim (1/2, 1/4, 1/8)$
idp	$p_{v,u} \sim (1/d_{in}[u])$
np	$p_{v,u} \sim U(0, 1)$

Datasets. To ensure our methods are practical, we experiment using four real-world networks [19] and the publicly available synthetic instances from the work of Becker et al. [1]. Besides, we use graph data sets of size up to 81K, as Table 3a shows, while previous works' experimental studies have limited themselves to size of up to 10K [1,7,8]. We tested those graphs on six edge influence probability distributions, as in Table 3b: *fixed* (xp), *trivalency* (tp), *half-trivalency* (htp), *higher* (hp), *in-degree* (idp), and *uniform* marked as np.

Implementation Details. We run experiments on a 14-core Intel Core i9 10940X machine @3.3GHz with 256GB RAM. The C++ code[2] is compiled by gcc 9.4 with o3 optimization.

Parameters. We run each trial 10 times, on both the IC and LT models. By default, we use 10,000 Monte Carlo (`MC`) runs with spa data and 4,000 runs in larger data sets; 1,000 `MC` runs during `uplift` reachable node selection, and 100 `MC` runs with `saturate` node selection phase on spa data. We use $\epsilon = 0.02$ on spa data and $\epsilon = 0.01$ on larger data sets. We tune the budget range based on the data; on smaller networks we set the maximum budget to 40% of the total number of nodes, while on larger data we set the maximum budget to 0.1%–3% of the nodes. Notably, previous works have set their budget parameters as follows: in [8], the budget is up to 13% of the network with a default $k = 100$; in [27], the default budget is set to $k = 15$ while a test considers a network of 60–70 individual youth; in [1], the budget is up to 10% of the data and by default $k = 20$.

[2] https://anonymous.4open.science/r/fairness-2D9D/.

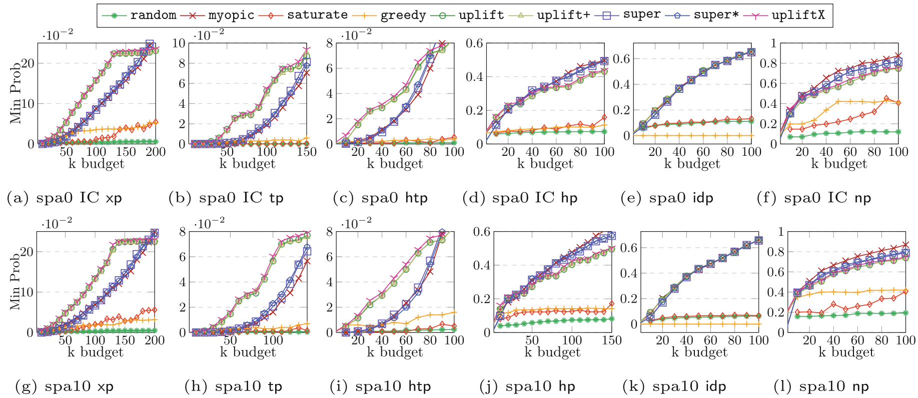


Fig. 3. Fairness objective on spa0 and spa10 data, IC model, $\epsilon = 0.02$.

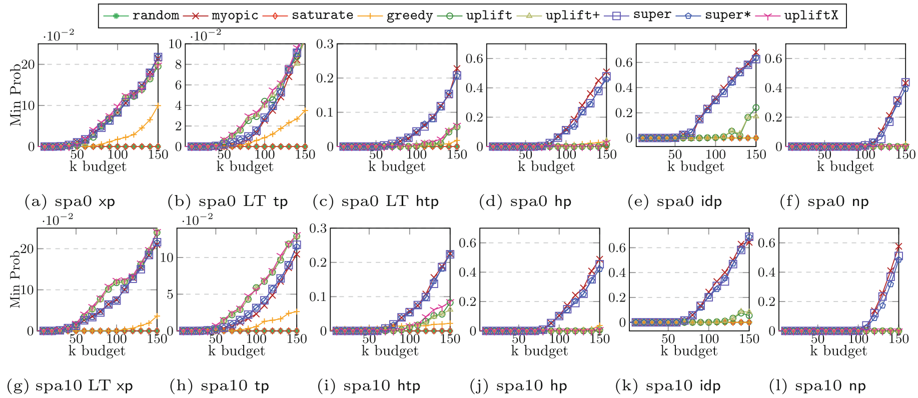
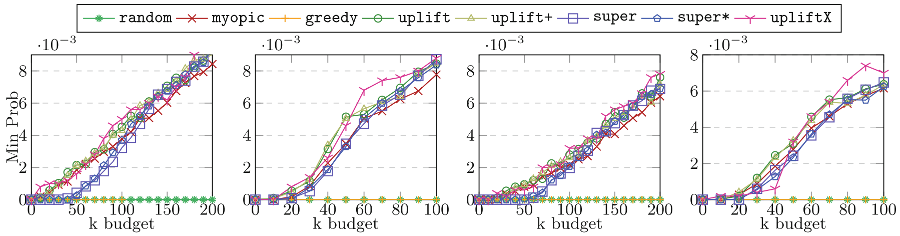


Fig. 4. Fairness objective on spa0 and spa10 data, LT model, $\epsilon = 0.02$.

Minimum Probability. Figure 3 shows our results on minimum influence probability with the IC model on spa data. With fixed probability distribution `xp` (Figs. 3a, 3g), `uplift` variants outperform `myopic` until the budget gets too high compared to the total size. `upliftX` always performs slightly better than `uplift`, while `uplift` dominates other methods at low budget values. With higher edge probabilities, the inflection point drops from around $40\%|V|$ with `xp` to $30\%|V|$ with `tp` (Figs. 3b, 3h), to $20\%|V|$ with `htp` (Figs. 3c, 3i). `uplift` maintains its advantage when the budget is 4% of graph nodes with `hp` (Figs. 3d, 3j). With uniform probability distribution `np` (Figs. 3f, 3l) and 1/in-degree (`idp`) probability (Fig. 3e, 3k), there are less significant gaps.

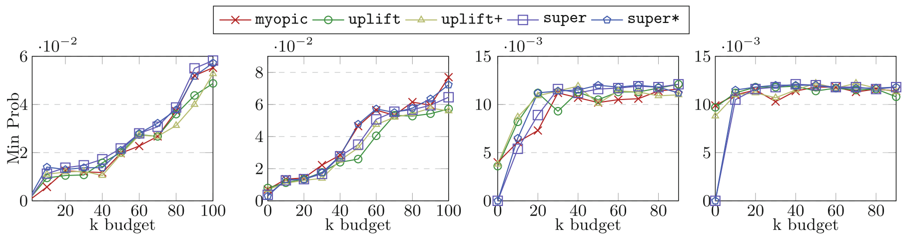
Figure 4 presents the corresponding results with the LT model and budget from 20% to 30% of $|V|$. Notably, with the LT model, raising the disadvantaged nodes is more difficult. In Figs. 4a–4b and 4g–4h, `uplift` dominates others in the `xp` and the `tp` model, while `upliftX` performs best. As edge probability rises in Figs. 4c–4f and 4i–4l, `myopic` and `super` succeed to lift disadvantaged nodes up, leading to the conclusion that there is no one-size-fits-all solution.



(a) alpha IC tp (b) alpha IC htp (c) otc IC tp (d) otc IC htp

Fig. 5. Fairness objective on bitcoin alpha and otc data, IC model, $\epsilon = 0.01$.

Figure 5 shows our results on the two bitcoin data sets, alpha and otc, with the IC model and budget up to $5\%|V|$ nodes. Here, our `uplift`-based algorithms always dominate `myopic` in `tp` (Figs. 5a and 5c), while `upliftX` has a distinctive advantage. The plot is similar with `htp` distribution (Figs. 5b and 5d), though the inflection point comes at around $1.5\%|V|$ vertex.

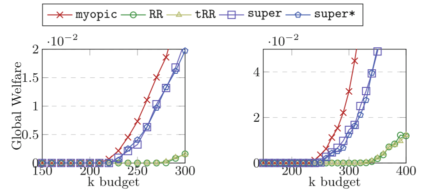


(a) facebook IC tp (b) facebook IC htp (c) twitter IC tp (d) twitter IC htp

Fig. 6. Fairness objective on facebook and twitter data, IC model, $\epsilon = 0.01$.

We also assess performance on an undirected graph, facebook, and a larger dataset, twitter, with the IC model. Figure 6 shows our results, with budget up to $2.4\%|V|$ on facebook and $0.1\%|V|$ on twitter. On facebook dataset, `super` dominates `uplift` and `myopic` with `tp` distribution (Fig. 6a), while `myopic` outperforms others with `htp` (Fig. 6b). No method is universally superior. On twitter, `myopic` achieves lower minimal probability than others with `tp` (Fig. 6c). `uplift` presents a rapid growth and reaches plateau within the first twenty seed nodes. With the `htp` distribution, all methods perform similarly.

Figure 7 shows results with the LT model on facebook; `myopic` now stands out. We also tried LT on twitter data, yet it was hard to raise the objective above zero with a small budget, illustrating result dependence on the spread model.



(a) facebook LT tp (b) facebook LT htp

Fig. 7. Utility on Facebook, LT, $\epsilon = 0.01$.

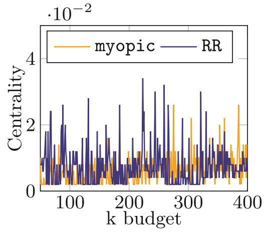
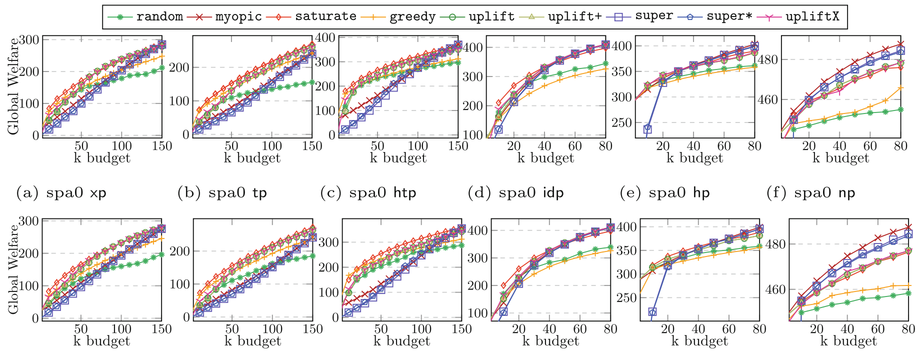


Fig. 8. Choices of `myopic` and `RR` vs. node centrality

To investigate why `uplift` performs better in certain situations, we study how its selections relate to *degree* centrality. As Fig. 8 shows, after selecting the node of largest degree, `myopic` prefers low-degree nodes, which tend to be weakly connected. On the other hand, `uplift` selects nodes of evenly distributed centrality, as it considers their influence on disadvantaged neighbors. Thereby, `uplift` outperforms `myopic` on low budgets.

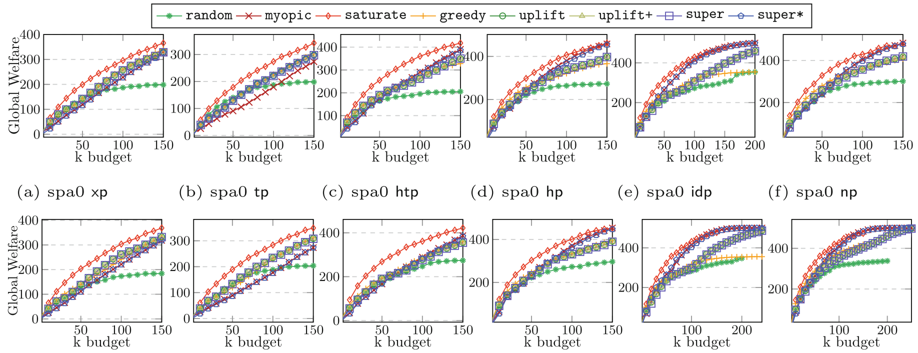


(a) spa0 xp (b) spa0 tp (c) spa0 htp (d) spa0 idp (e) spa0 hp (f) spa0 np

(g) spa10 xp (h) spa10 tp (i) spa10 htp (j) spa10 idp (k) spa10 hp (l) spa10 np

Fig. 9. Global welfare on spa0 and spa10 data, IC model, $\epsilon = 0.02$.

Global Welfare. To better understand the ramifications of using fairness-oriented algorithms on the tradeoff between individual fairness and total welfare, we also test performance on the aggregate influence objective, i.e., on the original *Influence Maximizing* (IM) problem. Figure 9 presents the results of our study on the sum of influence probabilities using the IC model on the spa dataset. With a fixed probability distribution `xp` (Fig. 9a, 9g), as with minimum probability, `uplift` variants outperform `myopic` until the budget gets too high. While `saturate` may outperform others in global welfare, it performs poorly in `MaxMin Faireness` (Fig. 3, Fig. 4) and, as we will see, in runtime (Fig. 14a). Additionally, `greedy` outperforms `uplift` variants in the first $1\%|V|$ and `myopic` in the first 10%$|V|$. With higher edge probabilities, the inflection point drops, following the pattern observed with minimum probability. The inflection point is around 40%$|V|$ with xp, and decreases to 30%$|V|$ with tp (Fig. 9b, 9h), to 20%$|V|$ with htp (Fig. 9c, 9i) and idp (Fig. 9d, 9j). With uniform probability distribution np (Fig. 9f, 9l), myopic outperform others.

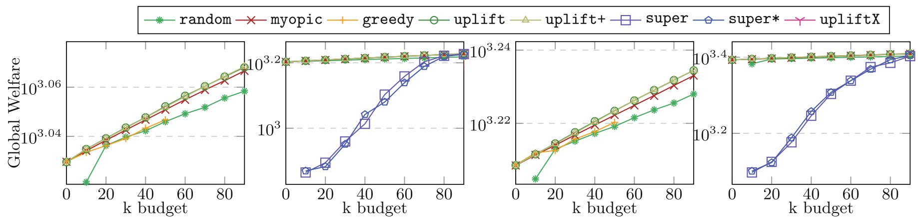


(a) spa0 xp (b) spa0 tp (c) spa0 htp (d) spa0 hp (e) spa0 idp (f) spa0 np

(g) spa10 xp (h) spa10 tp (i) spa10 htp (j) spa10 hp (k) spa10 idp (l) spa10 np

Fig. 10. Global welfare on spa0 and spa10 data, LT model, $\epsilon = 0.02$.

Figure 10 shows the corresponding results with the LT Model and budget in 30%$|V|$ – 40%$|V|$. Here, saturate maintains an advantage with every edge distribution in lower budgets. With the xp distribution (Fig. 10a, 10g), super* and myopic slightly underperform others in the 30%$|V|$ range of budget. However, with the tp distribution (Fig. 10b, 10h), greedy and uplift outperform myopic. In denser distributions, htp, hp, idp and np (Figs. 10c–10f and 10i–10l), the inflection point drops to 10%$|V|$. upliftX did not finish within our 7-h time limit in Fig. 10. However, in Fig. 10a, 10g, the pattern of upliftX follows uplift variants.

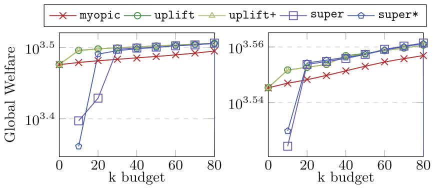


(a) alpha IC tp (b) alpha IC htp (c) otc IC tp (d) otc IC htp

Fig. 11. Global welfare on bitcoin alpha and otc data, IC model, $\epsilon = 0.01$.

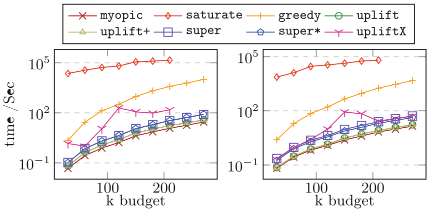
Figure 11 shows our results on the two bitcoin datasets, alpha and otc, with the IC model and a budget of up to 3%$|V|$ nodes. Here, uplift variants always outperform others. Figures 11a and 11c zoom in the performance of `uplift` variants, `myopic`, and `greedy`. In this closer view, we see that `uplift` variants outperform both `myopic` and `greedy` in this measure.

Figure 12 shows results with IC model on facebook. `uplift`-based algorithms and `myopic` only outperform others in the first 20 nodes. Thereafter, `uplift` variants and `super` outperform `myopic` in this undirected graph.



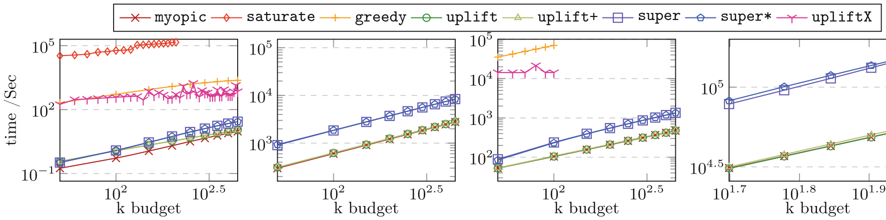
(a) facebook IC tp (b) facebook IC htp

Fig. 12. Utility Sum, $\epsilon = 0.01$.



(a) spa10 IC (b) spa10 LT

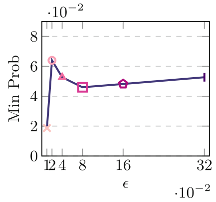
Fig. 13. Scalability, tp, $\epsilon = 0.01$



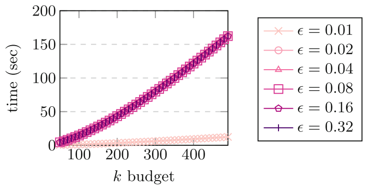
(a) spa10 IC tp (b) facebook IC tp (c) bc-alpha IC thp (d) twitter IC tp

Fig. 14. Runtime, IC model, $\epsilon = 0.01$.

Runtime. We also evaluate runtime scalability. Figure 13 presents the scalability of selecting $60\%|V|$ as candidate nodes on data sets of different sizes with both IC (13a) and LT (13b) models. Unsurprisingly, `uplift` and `myopic` outperform others. Figure 14 confirms that `uplift` spends slightly more time due to searching reachable sets in the early stages compared to `myopic`. The running time of `super` is slower than `myopic` and `uplift` but still linear. Figure 14a shows results on spa10 data; `saturate` is most time-consuming, as it performs greedy calls by binary search, while `uplift` variants and `myopic` run in linear time. Other results in Fig. 14 show that, on large data, such as facebook, bitcoin, and twitter, there is not a distinguishable difference between `uplift` and `myopic`, in agreement with our complexity analysis in Sect. 3. On the other hand, the runtime of `upliftX` is sensitive to the seed selection by `uplift`.



(a) spa0 IC tp, $k = 100$



(b) spa0 IC tp

Fig. 15. Effect of ϵ on spa0 `tp` in IC model

Tolerance ϵ. Lastly, we delve into the behavior vs. the tolerance variable ϵ, examining different ϵ values on spa0 with `tp` probability distribution and $k = 100$. Figure 15a shows the results. We obtain a peak at $\epsilon = 0.02$. Figure 15b shows that runtime stays stable for $\epsilon \geq 0.02$. Such ϵ values yield many selected nodes beyond the disadvantaged ones and bring no improvement in results while expanding the search space. We obtained similar peaks with other data.

5 Conclusion

We proposed a reachability-aware framework for fair influence maximization. Our experimental study demonstrates that our strategies strike an attractive balance between individual fairness and total expected welfare, especially in low budget settings, and outperform past work in challenging real-world problem instances by up to a factor of 4 in terms of fairness, even while no algorithm is universally superior across data, spread models, and probability distributions. When aiming to achieve fair influence, one should carefully consider which method to use, taking into consideration the desirable tradeoff between fairness and total welfare, as well as the data features. Our algorithms perform best in networks where people are influenced only by trusting groups, especially with the IC model, and present an advantage in overall welfare. Our hybrid solution, `super`, has an advantage on the LT model in terms of both individual fairness and global welfare. In the future, we aim to delve into adaptive seed selection and explore more efficient and effective sampling approaches.

References

1. Becker, R., D'Angelo, G., Ghobadi, S., Gilbert, H.: Fairness in influence maximization through randomization. J. Artif. Intell. Res. **73**, 1251–1283 (2022)
2. Brown, A.: Equality of Welfare, pp. 28–49. Palgrave Macmillan UK, London (2009)
3. Chen, W., Wang, C., Wang, Y.: Scalable influence maximization for prevalent viral marketing in large-scale social networks. In: KDD, pp. 1029–1038 (2010)
4. Chen, W., Yuan, Y., Zhang, L.: Scalable influence maximization in social networks under the linear threshold model. In: ICDM, pp. 88–97 (2010)
5. Domingos, P.M., Richardson, M.: Mining the network value of customers. In: KDD, pp. 57–66 (2001)

6. Eshghi, S., Maghsudi, S., Restocchi, V., Stein, S., Tassiulas, L.: Efficient influence maximization under network uncertainty. In: IEEE Conference on Computer Communications INFOCOM Workshops, pp. 365–371 (2019)
7. Farnadi, G., Babaki, B., Gendreau, M.: A unifying framework for fairness-aware influence maximization. In: WebConf, pp. 714–722 (2020)
8. Fish, B., Bashardoust, A., danah boyd, Friedler, S.A., Scheidegger, C., Venkatasubramanian, S.: Gaps in information access in social networks? In: WebConf, pp. 480–490 (2019)
9. Fu, X., Bhatt, R.R., Basu, S., Pavan, A.: Multi-objective submodular optimization with approximate oracles and influence maximization. In: 2021 IEEE International Conference on Big Data (Big Data), pp. 328–334 (2021)
10. Fu, Z., et al.: Fairness-aware explainable recommendation over knowledge graphs. In: SIGIR, pp. 69–78 (2020)
11. Garimella, K., Gionis, A., Parotsidis, N., Tatti, N.: Balancing information exposure in social networks. In: NeurIPS, pp. 4663–4671 (2017)
12. Garimella, K., Morales, G.D.F., Gionis, A., Mathioudakis, M.: Political discourse on social media: echo chambers, gatekeepers, and the price of bipartisanship. In: WebConf, pp. 913–922. ACM (2018)
13. Gershtein, S., Milo, T., Youngmann, B.: Multi-objective influence maximization. In: EDBT, pp. 145–156 (2021)
14. Huang, K., Wang, S., Bevilacqua, G.S., Xiao, X., Lakshmanan, L.V.S.: Revisiting the stop-and-stare algorithms for influence maximization. PVLDB **10**(9), 273–285 (2017)
15. Jagadish, H.V., Bonchi, F., Eliassi-Rad, T., Getoor, L., Gummadi, K.P., Stoyanovich, J.: The responsibility challenge for data. In: SIGMOD, pp. 412–414 (2019)
16. Kempe, D., Kleinberg, J.M., Tardos, É.: Maximizing the spread of influence through a social network. Theory Comput. **11**, 105–147 (2015)
17. Khajehnejad, A., Khajehnejad, M., Babaei, M., Gummadi, K.P., Weller, A., Mirzasoleiman, B.: Crosswalk: fairness-enhanced node representation learning. In: AAAI, pp. 11963–11970 (2022)
18. Krause, A., McMahan, H.B., Guestrin, C., Gupta, A.: Robust submodular observation selection. J. Mach. Learn. Res. **9**(93), 2761–2801 (2008)
19. Leskovec, J., Lang, K.J., Dasgupta, A., Mahoney, M.W.: Community structure in large networks. Internet Math. **6**(1), 29–123 (2009)
20. Li, Y., Fan, J., Wang, Y., Tan, K.: Influence maximization on social graphs: a survey. IEEE Trans. Knowl. Data Eng. **30**(10), 1852–1872 (2018)
21. Nguyen, H.T., Thai, M.T., Dinh, T.N.: Stop-and-stare: optimal sampling algorithms for viral marketing in billion-scale networks. In: SIGMOD, pp. 695–710 (2016)
22. Rahmattalabi, A., et al.: Fair influence maximization: a welfare optimization approach. In: AAAI, pp. 11630–11638 (2021)
23. Rahmattalabi, A., et al.: Exploring algorithmic fairness in robust graph covering problems. In: NeurIPS, pp. 15750–15761 (2019)
24. Roeschley, A., Buchanan, S.A., Burke, M., Graf, A., Zavalina, O.L.: Considering individual and community contexts within information pedagogy, scholarship, and practice. In: ASIST. Proc. Assoc. Inf. Sci. Technol., vol. 57 (2020)
25. Tang, Y., Shi, Y., Xiao, X.: Influence maximization in near-linear time: a martingale approach. In: SIGMOD, pp. 1539–1554 (2015)
26. Tang, Y., Xiao, X., Shi, Y.: Influence maximization: near-optimal time complexity meets practical efficiency. In: SIGMOD, pp. 75–86 (2014)

27. Tsang, A., Wilder, B., Rice, E., Tambe, M., Zick, Y.: Group-fairness in influence maximization. In: IJCAI, pp. 5997–6005 (2019)
28. Udwani, R.: Multi-objective maximization of monotone submodular functions with cardinality constraint. In: NeurIPS, pp. 9513–9524 (2018)
29. Wang, Y., Li, Y., Bonchi, F., Wang, Y.: Balancing utility and fairness in submodular maximization. In: EDBT, pp. 1–14 (2024)