

# Fairness in Influence Maximization through Randomization

**Ruben Becker**

**Gianlorenzo D'Angelo**

**Sajjad Ghobadi**

*Gran Sasso Science Institute, L'Aquila, Italy*

**Hugo Gilbert**

*Université Paris-Dauphine, Université PSL, CNRS,*

*LAMSADE, 75016 Paris, France*

RUBEN.BECKER@GSSI.IT

GIANLORENZO.DANGELO@GSSI.IT

SAJJAD.GHOBADI@GSSI.IT

HUGO.GILBERT@DAUPHINE.PSL.EU

## Abstract

The influence maximization paradigm has been used by researchers in various fields in order to study how information spreads in social networks. While previously the attention was mostly on efficiency, more recently fairness issues have been taken into account in this scope. In the present paper, we propose to use randomization as a mean for achieving fairness. While this general idea is not new, it has not been applied in this area.

Similar to previous works like Fish et al. (WWW '19) and Tsang et al. (IJCAI '19), we study the maximin criterion for (group) fairness. In contrast to their work however, we model the problem in such a way that, when choosing the seed sets, probabilistic strategies are possible rather than only deterministic ones. We introduce two different variants of this probabilistic problem, one that entails probabilistic strategies over nodes (node-based problem) and a second one that entails probabilistic strategies over sets of nodes (set-based problem). After analyzing the relation between the two probabilistic problems, we show that, while the original deterministic maximin problem was inapproximable, both probabilistic variants permit approximation algorithms that achieve a constant multiplicative factor of  $1 - 1/e$  minus an additive arbitrarily small error that is due to the simulation of the information spread. For the node-based problem, the approximation is achieved by observing that a polynomial-sized linear program approximates the problem well. For the set-based problem, we show that a multiplicative-weight routine can yield the approximation result.

For an experimental study, we provide implementations of multiplicative-weight routines for both the set-based and the node-based problems and compare the achieved fairness values to existing methods. Maybe non-surprisingly, we show that the ex-ante values, i.e., minimum expected value of an individual (or group) to obtain the information, of the computed probabilistic strategies are significantly larger than the (ex-post) fairness values of previous methods. This indicates that studying fairness via randomization is a worthwhile path to follow. Interestingly and maybe more surprisingly, we observe that even the ex-post fairness values, i.e., fairness values of sets sampled according to the probabilistic strategies computed by our routines, dominate over the fairness achieved by previous methods on many of the instances tested.

## 1. Introduction

The internet has revolutionized the way information spreads through the population. One positive consequence is that important and valuable campaigns can be spread at little cost quite efficiently thanks to news platforms and social media. Examples of such valuable campaigns are HIV prevention (Wilder et al., 2018a; Yadav et al., 2018), public health awareness (Valente & Pumpuang, 2007) or financial inclusion (Banerjee et al., 2013). The information spreading process can be notably optimized by algorithms that identify key people in the network to act as seed users to initiate an efficient spread of the campaign. The well known influence maximization problem formalizes this objective (Kempe et al., 2015): given a network and a probabilistic diffusion model, the task is to find a set of  $k$  seed nodes from which the campaign will start to spread, in order to maximize the expected

number of reached nodes. The problem has received a tremendous amount of attention (Becker et al., 2020; Borgs et al., 2014; Borodin et al., 2017; Budak et al., 2011; Chen & Teng, 2017; Cohen et al., 2014; Tang et al., 2015, 2014).

In the influence maximization problem, the objective is only concerned with the efficiency of the diffusion process, it does not take into account any fairness criteria. In order to underline the need of studying such fairness criteria in this scope, we start with the following motivating example: Consider a simple random graph modeling a network similar to a core-periphery structure (Borgatti & Everett, 2000). The network consists of two communities, the core  $C$  and the periphery  $D$ . The probability of intra-community edges are  $p_C$  and  $p_D$  respectively, while the probability of inter-community edges is  $q$ . For concreteness, assume that  $|C| = 50$ ,  $|D| = 150$  and  $p_C = 0.5$ ,  $p_D = 0.1$  and  $q = 0.1$ . I.e., we obtain a random network consisting of a well-connected rather small core and worse connected larger part of the graph that we refer to as the periphery of the network. Assume now that we use a state-of-the-art algorithm for influence maximization, e.g., the TIM implementation of the greedy algorithm due to Tang et al. (2014), in order to compute a seed set of size  $k$  with large expected coverage in the graph. As we can see in Figure 1 on the left, this can lead to a significant discrepancy in the probability of nodes being reached in the two communities. For concreteness, if  $k = 5$ , an average node in the core is reached with probability larger than 0.25, while the average node in the periphery is reached only with a probability of around 0.05.<sup>1</sup> In the right plot in Figure 1, we observe that the algorithm selects most of the seed nodes in the core. This is clearly because nodes in the core are better connected and thus choosing such nodes as seeds results in better coverage. In summary, we observe that maximizing expected spread without considering any fairness criteria can lead to unfair coverage with respect to communities or groups in the network. Such observations have motivated researchers more recently, to take fairness issues in influence maximization into account. We continue by giving a brief review of the related literature.

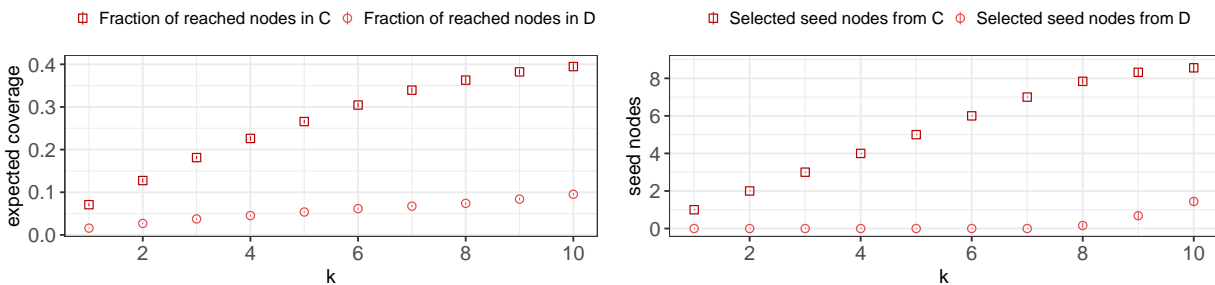


Figure 1: Results for the core-periphery model with a core of 50 nodes and a periphery of 150 nodes. The budget  $k$  is increasing from 1 to 10.

A first sequence of papers has investigated a setting in which several competing players are paying the network's host to influence users in their favor. The goal in these works is to ensure that the host picks seed nodes in a fair way with respect to the different players (Chen et al., 2020; Lu et al., 2013; Yu et al., 2017). Another line of research has investigated the fairness of the diffusion process with respect to the vertices, i.e., the users in the network. Indeed when only efficiency is being optimized, some users, or communities, i.e., groups of users, might get an unfairly low coverage (Ali et al., 2019; Fish et al., 2019; Farnadi et al., 2020; Khajehnejad et al., 2020; Rahmattalabi et al.,

1. In this example, we use the Independent Cascade model in order to model the information spread, edge weights are chosen to be 0.05. We refer the reader to Section 2 for the precise definition of the model used.

2020; Stoica et al., 2020; Tsang et al., 2019). An intuitive criterion to consider here is the maximin criterion. Here, the goal is to choose at most  $k$  seed nodes to maximize the minimum probability of a user being reached. When generalized to groups of users or communities, the goal becomes to maximize the minimum expected fraction of users reached per community. The first problem has been considered by Fish et al. (2019), who showed that the problem is hard to approximate to any constant approximation factor, unless  $P = NP$ . The second problem has been considered by Tsang et al. (2019). Building on previous work by Chekuri et al. (2010) and Udwani et al. (2018), the authors designed an algorithm with an asymptotic approximation ratio of  $1 - 1/e$  provided that the number of communities is not much larger than  $k$ .

In the present paper, we extend these works by studying the impact of randomization on fairness. Our approach is to allow randomized strategies for choosing the seed nodes rather than to restrict to deterministic strategies. Indeed, we introduce two randomized versions of the maximin problem. In our first problem, we consider randomized strategies that pick nodes as seeds independently. In contrast, in the second problem, we allow any probabilistic strategies that choose seed sets of expected size  $k$ , i.e., not restricting to independent distributions. It is easy to envision that such randomized strategies provide certain advantages over deterministic ones. In fact, the use of randomization is a longstanding idea in computational social choice, where it often leads to more tractable results and more expressive solutions via for instance time-sharing mechanisms (Manlove, 2013). It can also be used to incentivize participation (Aziz et al., 2018) or to workaroud impossibility results (Brandl et al., 2016). Lastly and closer to our work, using randomization is frequently used to obtain fairer solutions (Aziz et al., 2013; Bogomolnaia & Moulin, 2001; Katta & Sethuraman, 2006). Indeed, there may be optimization problems for which any deterministic solution is unfair. This was famously illustrated by Machina’s mom example in which a mother should decide which of her two children will receive an indivisible treat (Machina, 1989). In such cases, randomization may help evening things out by considering fairness in expectation, i.e., *ex-ante fairness* rather than *ex-post fairness*. Randomization is both useful for one-shot and for repeated problems. In the former, it provides fairness over opportunities and in the latter it achieves fairness in the long run in a natural way. Lastly, randomization can be used to satisfy the fairness principle of *equal treatment of equals* (Moulin, 1991). Despite being an old research topic, the study of randomized solutions is still a hot topic where many open problems remain to be solved (Aziz, 2019; Brandt, 2019).

## 1.1 Our Contribution

The contributions of this paper are summarized as follows.

- After recalling the necessary technical background related to influence maximization, in Section 2, we introduce the two randomized versions of the maximin problem. In the first one, we consider randomized strategies that pick nodes as seeds independently with some probability such that the expected size of the resulting seed set is bounded by  $k$ , we call this the *node-based problem*. In the second problem, we study a more general feasible set. That is, we consider strategies that consist of probability distributions over seed sets of expected size  $k$ , i.e., not restricting to the special case of distributions that pick nodes independently but allowing for correlation. We then analyze the relation between the two probabilistic problems.
- In Section 3, we quantify the loss in efficiency that can be incurred by following our fairness criteria, i.e., we show bounds on the price of fairness. We continue by proving that both randomized variants of the maximin influence problems are NP-hard. For the node-based problem, we in addition show that, unless  $P = NP$ , there is no algorithm with approximation

ratio better than  $1 - 1/e$ . Thereafter we show that still, in this setting of fairness in influence maximization, randomization leads to a number of advantages.

- In Section 4, we prove that the resulting problems can be approximated to within a factor of  $1 - 1/e$  (plus an additive  $-\varepsilon$  term that is also inherent in the work of Tsang et al. (2019)) even in the case when the number of communities exceeds the number of seed nodes  $k$ . For the node-based problem (up to the additive error term) we thus give a tight approximation result. Furthermore, our work shows that the inapproximability result of Fish et al. (2019) can be circumvented by introducing randomization to the problem. Our algorithms are comparatively simple. For the node-based problem, the feasible set is of dimension  $n$ . After approximating (to within an additive  $\pm\varepsilon$  term) all functions  $\sigma_v$  using concentration bounds, we still face the problem that the resulting optimization problem is not linear. We show however that the non-linear optimization problem is approximated to within a factor of  $1 - 1/e$  by a linear program of the same size. Thus we obtain a polynomial time algorithm with multiplicative approximation ratio  $1 - 1/e$  (plus the additive  $-\varepsilon$  term). For the set-based problem, the situation is different. Here, by introducing a variable for every possible seed set, the problem can be approximated (to within an additive  $\varepsilon$  term) by a linear program. The downside of course is that this program is of dimension  $\Theta(2^n)$ . As the linear program is a covering linear program however, we are able to show that a multiplicative weights routine that is essentially a black-box application of a method by Young (1995) can be used to obtain the described approximation. This method, as a subroutine, requires an algorithm for an oracle problem. We observe that the oracle problem in our case can be solved using standard (weighted) influence maximization and thus can be approximated to within a factor of  $1 - 1/e$  efficiently both in theory and practice. Although the feasible set to the set-based problem is of exponential dimension, the computed solution that is guaranteed to be a multiplicative  $1 - 1/e$  approximation (plus the additive  $-\varepsilon$  term) has only a linear support in  $n$ .
- In Section 5, we evaluate implementations of multiplicative weight routines for both node and set-based problems on random instances, synthetic instances from the work of Tsang et al. (2019), and a wide range of real world networks. We compare both the ex-ante and ex-post performance of our techniques with standard greedy techniques, as well as with the routines proposed by Tsang et al. (2019) and Fish et al. (2019). We observe that our ex-ante values are superior to the ex-post values of all other algorithms and, maybe surprisingly, our experiments indicate that even the ex-post values of our algorithms are competitive or even improve over the ex-post values achieved by the other techniques. We also experimentally evaluate the loss in efficiency, i.e., in total information spread resulting from using our algorithm over a standard IM algorithm that does not consider any fairness criteria. We conclude that our algorithms lead to much fairer solutions while incurring at most a small loss in total spread on all instances tested.

## 1.2 Further Related Work

We review the works that have considered fairness issues in the context of influence maximization in more detail.

The line of research that investigates the fairness of the diffusion process with respect to the vertices (i.e., users) in the network is closest to our setting. Fish et al. (2019), to the best of our knowledge, are the first to study the maximin objective in order to maximize the minimum probability of nodes to be reached by the information spread. They show that this objective leads to an NP-hard optimization problem, and even more, is hard to approximate to within any

constant factor unless  $P = NP$ . Even worse, the authors show that various greedy strategies have asymptotically worst-possible approximation ratios.

In the work of Tsang et al. (2019), the authors introduced the problem of maximizing the spread of a campaign while respecting a *group*-fairness constraint. In their setting, each user of the network belongs to one or several communities and several criteria to guarantee that each community gets its fair share of information are considered. For each of these criteria, maximizing influence while respecting the related fairness constraint can be solved via a multi-objective submodular optimization problem. The authors design an algorithm to tackle such multi-objective submodular optimization problems that provides an asymptotic approximation guarantee of  $1 - 1/e$ . Their work cannot be directly extended to the case where fairness is considered with respect to individuals instead of communities. Indeed, their result requires that  $m = o(k \log^3(k))$  where  $m$  is the number of communities and  $k$  is the seed set cardinality constraint.

The above two works are the most closely related to the present paper. We proceed by reviewing more distant works that still treat fairness issues in influence maximization. Rahmattalabi et al. (2020) further extend the group-fairness approach of Tsang et al. (2019) by following a different path. From the expected fraction of vertices reached in each community, the authors define a utility vector over the entire population of vertices, and then take a welfare optimization approach by optimizing a decision criterion which is a function of this utility vector. Stoica et al. (2020) study how improving the diversity of nodes in the seed set can influence efficiency and fairness of the information diffusion process. In a rather specific setting, where the network is generated using a biased preferential attachment model yielding two unequal communities, the authors show that, under certain conditions, seeding strategies that take into account the diversity of nodes in the seed set are more efficient and equitable. Ali et al. (2019) address fairness of the diffusion process with respect to different communities considering both the number of people influenced and the time step at which they are influenced. After illustrating that both, maximizing the expected number of nodes reached by choosing a seed set of fixed cardinality, and minimizing the number of seeds required to influence a given portion of the network may lead to unfair solutions, the authors propose an objective function which balances two objectives: the expected number of nodes reached which should be maximized, and the maximum disparity in influence between any two communities which should be minimized. More recently, Farnadi et al. (2020) review the different notions of group-fairness criteria used in the influence maximization literature and show how influence maximization problems under these fairness criteria can be expressed as mixed integer linear programs. The authors then provide numerical tests to measure the price of fairness of different fairness criteria as well as the increase in fairness with respect to vanilla influence maximization. Lastly, fair influence maximization was approached by Khajehnejad et al. (2020) based on machine learning techniques. The authors use an adversarial graph embedding approach to choose a seed set which both makes it possible to achieve high influence propagation and fairness between different communities.

An even more distant line of work (Chen et al., 2020; Lu et al., 2013; Yu et al., 2017) consider a setting in which several players compete with similar products on the same networks. These players pay a given budget to the network’s host in order to influence as many users as possible. The question investigated is how to ensure that the host picks solutions in a fair way with respect to the different players. Stated otherwise, the average influence obtained per seed, also called amplification factor, should be similar for all players. We omit further details on this line of research as it is not too closely related to our work.

## 2. Preliminaries

### 2.1 Influence Maximization

We consider the classical influence maximization setting where we are given a directed arc-weighted graph  $G = (V, A, w)$  with  $V$  being the set of  $n$  nodes,  $A$  the set of arcs, and  $w : A \rightarrow [0, 1]$  an arc-weight function. In addition we are given an information diffusion model. A broad variety of models can be used as information diffusion model. Two of the most popular models are the *Independent Cascade* (IC) and *Linear Threshold* (LT) models (Kempe et al., 2015). In both these models, given an initial node set  $S \subseteq V$  called *seed nodes*, a spread of influence from the set  $S$  is defined as a randomly generated sequence of node sets  $(S_t)_{t \in \mathbb{N}}$ , where  $S_0 = S$  and  $S_{t-1} \subseteq S_t$ . These sets represent active users, i.e., we say that a node  $v$  is *active* at time step  $t$  if  $v \in S_t$ . The sequence converges as soon as  $S_{t^*} = S_{t^*+1}$ , for some time step  $t^* \geq 0$  called the time of quiescence. For a set  $S$ , we use the standard notation  $\sigma(S) = \mathbb{E}[|S_{t^*}|]$  to denote the expected number of nodes activated at the time of quiescence when running the process with seed nodes  $S$ , here the expectation is over the random process of information diffusion that depends on the weights  $w$  and moreover on the information diffusion model at hand.

**Information Diffusion Models.** In the *Independent Cascade* (IC) model, the values  $w_a \in [0, 1]$  for  $a \in A$  are probabilities. The sequence of node sets  $(S_t)_{t \in \mathbb{N}}$ , is randomly generated as follows. If  $u$  is active at time step  $t \geq 0$  but was not active at time step  $t - 1$ , i.e.,  $u \in S_t \setminus S_{t-1}$  (with  $S_{-1} = \emptyset$ ), node  $u$  tries to activate each of its neighbors  $v$ , independently, and succeeds with probability  $w_{uv}$ . In case of success,  $v$  becomes active at time step  $t + 1$ , i.e.,  $v \in S_{t+1}$ .

In the *Linear Threshold* (LT) model, the values  $w_a \in [0, 1]$  for  $a \in A$  are such that, for each node  $v$ , it holds that  $\sum_{(u,v) \in A} w_{uv} \leq 1$ . The sequence of node sets  $(S_t)_{t \in \mathbb{N}}$ , is randomly generated as follows. At time step  $t + 1$ , every inactive node  $v$  such that  $\sum_{(u,v) \in A, u \in S_t} w_{uv} \geq \theta_v$  becomes active, i.e.,  $v \in S_{t+1}$ , where the thresholds  $\theta_v$  are chosen independently and uniformly at random from the interval  $[0, 1]$  for all nodes  $v \in V$ .

Both the IC and LT models can be generalized to what is known as the *Triggering Model*, see (Kempe et al., 2015, Proofs of Theorem 4.5 and 4.6). For a node  $v \in V$ , let  $N_v$  denote all in-neighbors of  $v$ . In the Triggering model, every node  $v \in V$  independently picks a *triggering set*  $T_v \subseteq N_v$  according to some distribution over subsets of its in-neighbors. For a possible outcome  $L = (T_v)_{v \in V}$  of triggering sets for the nodes in  $V$ , let  $G_L = (V, A_L)$  denote the sub-graph of  $G$  where  $A_L = \{(u, v) | v \in V, u \in T_v\}$ . The graph  $G_L$  is frequently referred to as live-edge graph and the edges  $A_L$  are referred to as live edges. We let  $\rho_L(S)$  be the set of nodes reachable from  $S$  in  $G_L$ . We denote with  $\mathcal{L}$  the random variable that describes this process of generating outcomes or live-edge graphs, and with  $L$  we mean a possible outcome, i.e., value taken by  $\mathcal{L}$ . Then  $\sigma(S) = \mathbb{E}_{\mathcal{L}}[|\rho_{\mathcal{L}}(S)|]$ . The IC model is obtained from the Triggering model, if for each arc  $(u, v)$ , node  $u$  is added to  $T_v$  with probability  $w_{uv}$ . Differently, the LT model is obtained if each node  $v$  picks at most one of its in-neighbors to be in her triggering set, selecting a node  $u$  with probability  $w_{uv}$  and selecting no one with probability  $1 - \sum_{u \in N_v} w_{uv}$ .

**Further Notation.** In what follows, we assume the triggering model to be the underlying model describing the information spread. We define  $\sigma_v(S) := \Pr_{\mathcal{L}}[v \in \rho_{\mathcal{L}}(S)]$  to be the probability that node  $v$  is reached from seed nodes  $S$ . Clearly,  $\sigma(S) = \mathbb{E}_{\mathcal{L}}[|\rho_{\mathcal{L}}(S)|] = \sum_{v \in V} \Pr_{\mathcal{L}}[v \in \rho_{\mathcal{L}}(S)] = \sum_{v \in V} \sigma_v(S)$ . We extend this notation in a natural way, that is, for  $C \subseteq V$ , we denote by  $\sigma_C(S) = \frac{1}{|C|} \cdot \sum_{v \in C} \sigma_v(S)$  the average probability of being reached of nodes in  $C$ . Note that  $\sigma_v(S) = \sigma_{\{v\}}(S)$  for nodes  $v \in V$  and  $\sigma(S) = |V| \cdot \sigma_V(S)$ . Table 1 summarises the notations frequently used in the paper.

Notations	Descriptions
$G = (V, A, w)$	a directed graph with $n$ nodes $V$ , edges $A$ , and $w : A \rightarrow [0, 1]$
$\mathcal{C}$	a set of $m$ communities $C \subseteq V$
$k$	size of the seed set
$G_L$	a live-edge graph
$\rho_L(S)$	set of nodes reachable from seeds $S \subseteq V$ in $G_L$
$\sigma(S)$	expected number of nodes activated from seeds $S \subseteq V$
$\sigma_v(S)$	probability that node $v$ is reached from seeds $S \subseteq V$
$\sigma_C(S)$	average probability of being reached of nodes in $C \in \mathcal{C}$ from seeds $S \subseteq V$
$\tilde{\sigma}(S), \tilde{\sigma}_v(S), \tilde{\sigma}_C(S)$	approximate analogues of $\sigma(S)$ , $\sigma_v(S)$ , and $\sigma_C(S)$ obtained via sampling a fixed number $T$ of live-edge graphs
$\mathbb{1}$	the all-ones vector
$\mathbb{1}_i$	a vector with $i$ 'th entry being one and all the other entries equal to 0
$\mathbb{1}_P$	indicator function that equals 1 if $P$ holds and 0 otherwise
$\mathcal{P}$	set of vectors $p \in [0, 1]^{2^V}$ such that $\mathbb{1}^T p = 1$ and $\sum_{S \subseteq V} p_S  S  \leq k$
$\mathcal{X}$	set of vectors $x \in [0, 1]^n$ such that $\mathbb{1}^T x \leq k$
$\text{opt}_{\mathcal{P}}(G, \mathcal{C}, k)$	optimal objective value of set-based problem
$\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k)$	optimal objective value of node-based problem
$\text{PoF}_{\mathcal{P}}(G, \mathcal{C}, k)$	price of fairness of the set-based problem
$\text{PoF}_{\mathcal{X}}(G, \mathcal{C}, k)$	price of fairness of the node-based problem

Table 1: Notations

For a maximization problem  $\max\{F(x) : x \in R\}$  with feasibility region  $R$ , objective function  $F : R \rightarrow \mathbb{R}_{\geq 0}$  and optimum value  $\text{opt}$ . We say that  $x$  is an  $(\alpha, \beta)$ -approximation for real values  $\alpha \in (0, 1]$  and  $\beta \in [0, \infty)$ , if  $F(x) \geq \alpha \cdot \text{opt} - \beta$ .

**Maximin Optimization.** The standard objective studied in influence maximization is finding a set  $S$  maximizing  $\sigma(S)$  under a cardinality constraint  $|S| \leq k$  for some integer  $k$ . As this objective function does not take into account the fairness of the diffusion process with respect to nodes or communities, Fish et al. (2019) and Tsang et al. (2019), have investigated maximin variants of this objective that can be written as

$$\max_{S \in \binom{V}{k}} \min_{C \in \mathcal{C}} \sigma_C(S),$$

where  $\mathcal{C}$  is a set of  $m \geq 1$  different communities  $\emptyset \neq C \subseteq V$  that may not be disjoint and  $\binom{V}{k}$  denotes the set of subsets of  $V$  of size  $k$ . If each node is its own community, this amounts to finding a set of  $k$  seed nodes maximizing the minimum probability that a node is reached, which is the problem considered by Fish et al. (2019). We note that this is actually one instance of a broader class of optimization problems that ask to maximize a social welfare function, being the  $-\infty$ -mean here. Fish et al. (2019) considered the special case where the diffusion model is the Independent Cascade model and in which all arcs have the same probability of diffusion  $\alpha$ . They proved that the problem of choosing  $k$  seeds  $S$  such as to maximize  $\min_{v \in V} \sigma_v(S)$  is NP-hard to be approximated within a factor better than  $O(\alpha)$  and that minimizing the number of seeds to obtain the optimal solution cannot be approximated within a factor  $O(\ln n)$ . Furthermore, they analysed several natural heuristics which unfortunately exhibit worst-case approximation ratio exponentially small in  $n$ .

## 2.2 Fairness via Randomization

We initiate studying the impact of randomization to increase fairness for influence maximization. We start with a simple example of an influence maximization problem to illustrate the impact of randomization. Let us assume that we are using the IC model. Consider the graph in Figure 2 consisting of two nodes  $u, v$ , each forming their own community, connected in both directions by edges  $(u, v), (v, u)$  with probabilities  $1/2$ . Assume that  $k = 1$ . Then (due to symmetry) the optimal deterministic strategy is to choose any of the two nodes achieving a minimum probability of being reached of  $1/2$  for the non-chosen node. A probabilistic strategy however would be allowed to assign probabilities  $1/2$  to both the sets  $\{u\}$  and  $\{v\}$ . For each of the two nodes, this strategy achieves an expected probability of being reached of  $1/2 + 1/4 = 3/4$ , the  $1/2$  being due to the fact that the node is a seed himself with probability  $1/2$  and the  $1/4$  being due to the probability of being reached (with probability  $1/2$ ) from the other node if she is a seed (happens with probability  $1/2$ ). While the example seems simplistic and artificial, it shows that the probabilistic strategy may in fact achieve a higher degree of fairness. We consider two different ways of introducing randomness, either via distributions over sets or via distributions over nodes.

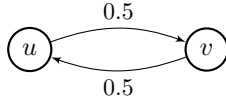


Figure 2: Simple instance showing that randomization allows to increase fairness in influence maximization.

**Probabilistically Choosing Sets.** We relax the maximin problem by allowing for randomized strategies, i.e., feasible solutions in our *set-based probabilistic maximin problem* are not simply sets of size at most  $k$ , but rather distributions over sets. Let  $\mathcal{P}$  be the set of distributions over sets of expected size at most  $k$ , i.e.,  $\mathcal{P} := \{p \in [0, 1]^{2^V} : \mathbf{1}^T p = 1, \sum_{S \subseteq V} p_S |S| \leq k\}$  and let  $S \sim p$  denote the random process of sampling  $S$  according to the distribution  $p$ . We now consider the optimization problem

$$\text{opt}_{\mathcal{P}}(G, \mathcal{C}, k) = \max_{p \in \mathcal{P}} \min_{C \in \mathcal{C}} \mathbb{E}_{S \sim p}[\sigma_C(S)],$$

i.e., we are searching for the probability distribution that maximizes the minimum expected probability of the  $m$  communities to be reached. This notion is frequently referred to as *ex-ante fairness* in the literature (Machina, 1989).

We note that in the conference version (Becker et al., 2021) of this article, we studied the problem where the probability distributions are restricted to be over sets of size *exactly*  $k$ . Here, we explicitly allow sets of size different from  $k$ , the only restriction on the size is in expectation. This new problem constitutes a relaxation of the set-based problem studied in (Becker et al., 2021). We emphasize that all of our results hold for both versions of the problem. The main reason why we further relaxed the studied set-based problem is that this allows us to obtain a clean relationship (see the last paragraph of this section) between the set-based and the node-based problem that we introduce next.

**Probabilistically Choosing Nodes.** An alternative intuitive way of introducing randomness is obtained by considering a maximin problem where feasible solutions are not distributions over sets, but are characterized by probability values for nodes. In this setting, which we call the *node-based probabilistic maximin problem*, we let  $\mathcal{X} := \{x \in [0, 1]^n : \mathbf{1}^T x \leq k\}$  be the feasible set and consider the



process of randomly generating a set  $S$  from  $x$ , denoted by  $S \sim x$ , by letting  $i$  be in  $S$  independently with probability  $x_i$ . In this setting we are thus interested in finding  $x \in \mathcal{X}$  that maximizes the minimum expected coverage from  $S$  of any community, when  $S$  is generated from  $x$  as described and the expectation is over this generation. We write this problem as

$$\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k) = \max_{x \in \mathcal{X}} \min_{C \in \mathcal{C}} \mathbb{E}_{S \sim x}[\sigma_C(S)].$$

**Extending Set Functions to Vectors.** In what follows, we extend set functions to vectors in  $\mathcal{P}$  and  $\mathcal{X}$  in a straightforward way, i.e., for a set function  $f$ , for  $p \in \mathcal{P}$ , we let  $f(p) := \mathbb{E}_{S \sim p}[f(S)]$  and, for  $x \in \mathcal{X}$ , we let  $f(x) := \mathbb{E}_{S \sim x}[f(S)]$ .

**Relationship between Problems.** We first observe that, for  $x \in \mathcal{X}$ , the vector  $p^x$  defined as  $p_S^x := \prod_{i \in S} x_i \prod_{j \in V \setminus S} (1 - x_j)$ , for  $S \subseteq V$ , is in  $\mathcal{P}$  and furthermore  $\sigma_C(x) = \sigma_C(p^x)$  for any  $C \in \mathcal{C}$ . Hence, we obtain the following lemma.

**Lemma 2.1.** *For any  $G, \mathcal{C}$ , and  $k$ , it holds that  $\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k) \leq \text{opt}_{\mathcal{P}}(G, \mathcal{C}, k)$ .*

We proceed by measuring the reverse relation. In fact, the concept of correlation gap can be used in order to upper bound  $\text{opt}_{\mathcal{P}}(G, \mathcal{C}, k)$  in terms of  $\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k)$  incurring only a constant loss.

**Lemma 2.2.** *For any  $G, \mathcal{C}$ , and  $k$ , it holds that  $\text{opt}_{\mathcal{P}}(G, \mathcal{C}, k) \leq \frac{e}{e-1} \cdot \text{opt}_{\mathcal{X}}(G, \mathcal{C}, k)$ .*

*Proof.* Let  $G, \mathcal{C}$ , and  $k$  be arbitrary. For a distribution  $p \in \mathcal{P}$  over  $2^V$ , define the *marginal probabilities*  $y^p$  w.r.t.  $p$  by  $y_i^p := \Pr_{S \sim p}[i \in S] = \sum_{S \subseteq V: i \in S} p_S$ . The correlation gap (Agrawal et al., 2010; Yan, 2011) of  $f : 2^V \rightarrow \mathbb{R}_{\geq 0}$  is defined as

$$\gamma_f := \sup_{p \in [0,1]^{2^V}} \frac{\mathbb{E}_{S \sim p}[f(S)]}{\mathbb{E}_{S \sim y^p}[f(S)]}$$

and it is well-known that the correlation gap of a monotone submodular function is bounded from above by  $\frac{e}{e-1}$ , see Agrawal et al. (2010, Corollary 1.2) or Yan (2011, Theorem 2.1). We may thus conclude that, for all  $C \in \mathcal{C}$ ,  $\gamma_{\sigma_C} \leq \frac{e}{e-1}$ . Now, let  $p \in \mathcal{P}$  be an optimal solution, i.e.,  $\text{opt}_{\mathcal{P}}(G, \mathcal{C}, k) = \min_{C \in \mathcal{C}} \mathbb{E}_{S \sim p}[\sigma_C(S)]$ . We obtain

$$\begin{aligned} \text{opt}_{\mathcal{P}}(G, \mathcal{C}, k) &= \min_{C \in \mathcal{C}} \mathbb{E}_{S \sim p}[\sigma_C(S)] \leq \min_{C \in \mathcal{C}} \left\{ \frac{e}{e-1} \cdot \mathbb{E}_{S \sim y^p}[\sigma_C(S)] \right\} \\ &= \frac{e}{e-1} \cdot \min_{C \in \mathcal{C}} \mathbb{E}_{S \sim y^p}[\sigma_C(S)] \leq \frac{e}{e-1} \cdot \text{opt}_{\mathcal{X}}(G, \mathcal{C}, k), \end{aligned}$$

where the last step uses that  $\sum_{i \in V} y_i^p = \sum_{i \in V} \sum_{S \subseteq V: i \in S} p_S = \sum_{S \subseteq V} p_S \cdot |S| \leq k$  and thus  $y^p \in \mathcal{X}$ .  $\square$

It remains to ask if the bound predicted by the above lemma is tight. We give the following simple example.

**Lemma 2.3.** *There exists a graph  $G$ , community structure  $\mathcal{C}$ , and integer  $k$ , such that  $\text{opt}_{\mathcal{P}}(G, \mathcal{C}, k) \geq \frac{5}{4} \cdot \text{opt}_{\mathcal{X}}(G, \mathcal{C}, k)$  when using the IC model.*

*Proof.* Consider the graph  $G$  consisting of two nodes  $u$  and  $v$  connected back and forth by two edges of weight  $2/3$ . Let  $C$  be the singleton community structure, and  $k = 1$ , i.e., the same instance as in Figure 2 with the difference that the edge weights are  $2/3$ . Then the best node-based solution achieves a value of  $2/3$  (either by choosing one of the two nodes with probability 1 or by choosing both with equal probability  $1/2$ ). The optimal set-based solution that chooses the sets  $\{u\}$  and  $\{v\}$  both with probability  $1/2$  however achieves a value of  $1/2 + 1/2 \cdot 2/3 = 5/6$ .  $\square$

We note that  $e/(e-1) \approx 1.58$ , while  $5/4 = 1.25$ . We consider tightening this gap to be an interesting open problem.

### 3. Price of Fairness and Hardness

#### 3.1 Price of Group Fairness

The price of group fairness is a quantitative loss measuring the decrease in efficiency that is incurred when we restrict ourselves to solutions respecting a group fairness requirement. In the following, we denote the maximizing solutions to the node and general set-based problems by  $F_{\mathcal{X}}(G, \mathcal{C}, k) = \operatorname{argmax}_{x \in \mathcal{X}} \min_{C \in \mathcal{C}} \mathbb{E}_{S \sim x}[\sigma_C(S)]$  and  $F_{\mathcal{P}}(G, \mathcal{C}, k) = \operatorname{argmax}_{p \in \mathcal{P}} \min_{C \in \mathcal{C}} \mathbb{E}_{S \sim p}[\sigma_C(S)]$ , respectively. Then, the respective prices of fairness  $\operatorname{PoF}_{\mathcal{X}}(G, \mathcal{C}, k)$  and  $\operatorname{PoF}_{\mathcal{P}}(G, \mathcal{C}, k)$  incurred by restricting to strategies in  $F_{\mathcal{X}}(G, \mathcal{C}, k)$  and  $F_{\mathcal{P}}(G, \mathcal{C}, k)$  are given by

$$\operatorname{PoF}_{\mathcal{X}}(G, \mathcal{C}, k) = \frac{\max_{S \in \binom{V}{k}} \sigma(S)}{\max_{x \in F_{\mathcal{X}}(G, \mathcal{C}, k)} \sigma(x)} \quad \text{and} \quad \operatorname{PoF}_{\mathcal{P}}(G, \mathcal{C}, k) = \frac{\max_{S \in \binom{V}{k}} \sigma(S)}{\max_{p \in F_{\mathcal{P}}(G, \mathcal{C}, k)} \sigma(p)}.$$

We obtain that for both problems, the price of group fairness can be linear in the graph size.

**Lemma 3.1.** *For any even  $n > 0$ , there is a graph  $G$  with  $n$  nodes and a community structure  $\mathcal{C}$  such that  $\operatorname{PoF}_{\mathcal{X}}(G, \mathcal{C}, 1) = \operatorname{PoF}_{\mathcal{P}}(G, \mathcal{C}, 1) = (n + 2)/4$ , when using the IC model.*

*Proof.* Let  $G$  be composed of two disjoint sets  $J$  and  $I$  of  $n/2$  vertices each. The only edges present in  $G$  are the edges from one specific vertex  $w \in J$  to all other vertices in  $J$ . Let the weight of these edges be 1 and let  $\mathcal{C}$  be the community structure consisting of singletons. Note that for all nodes  $v \in I \cup \{w\}$ , the probability of being reached is equal to the probability of being a seed as these nodes have no incoming edges. In other words, for any strategy  $x \in \mathcal{X}$ , it holds that  $\sigma_v(x) = x_v$ . Similarly, for any  $p \in \mathcal{P}$ , it holds that  $\sigma_v(p) = y_v^p$ , where  $y_v^p := \sum_{S: v \in S} p_S$  are the marginal probabilities with respect to  $p$ . Hence, it follows that the probabilistic solutions that maximize fairness split the budget 1 equally among the nodes in  $I \cup \{w\}$ . More precisely, when defining  $\rho := 1/(\frac{n}{2} + 1)$ , we get  $F_{\mathcal{X}}(G, \mathcal{C}, 1) = \{\rho \cdot \mathbf{1}_{I \cup \{w\}}\}$  and  $F_{\mathcal{P}}(G, \mathcal{C}, 1) = \{p \in \mathcal{P} : y_v^p = \rho \text{ for all } v \in I \cup \{w\}\}$  and in both cases the achieved objective value is  $\operatorname{opt}_{\mathcal{X}}(G, \mathcal{C}, 1) = \operatorname{opt}_{\mathcal{P}}(G, \mathcal{C}, 1) = \rho$ . Furthermore

$$\max_{x \in F_{\mathcal{X}}(G, \mathcal{C}, 1)} \sigma(x) = \max_{p \in F_{\mathcal{P}}(G, \mathcal{C}, 1)} \sigma(p) = n \cdot \rho.$$

The set  $S$  of size 1 that maximizes the expected number of reached nodes however, selects the node  $w$  yielding  $\max_{S \in \binom{V}{1}} \sigma(S) = n/2$ . Hence, we get a price of fairness that is of linear order. More precisely,  $\operatorname{PoF}_{\mathcal{X}}(G, \mathcal{C}, 1) = \operatorname{PoF}_{\mathcal{P}}(G, \mathcal{C}, 1) = (\frac{n}{2} + 1)/2 = (n + 2)/4$ .  $\square$

On the positive side we obtain that the price of group fairness is never larger than  $n/k$ .

**Lemma 3.2.** *For any graph  $G$ , community structure  $\mathcal{C}$  and number  $k$ , it holds that  $\operatorname{PoF}_{\mathcal{X}}(G, \mathcal{C}, k) \leq n/k$  and  $\operatorname{PoF}_{\mathcal{P}}(G, \mathcal{C}, k) \leq n/k$ .*

*Proof.* Note that, for both problems, there exist some optimal solution  $x$  and  $p$ , such that the expected size of the seed set is exactly  $k$ , i.e.,  $\mathbf{1}^T x = k$  and  $\sum_{S \subseteq V} p_S |S| = k$ . Furthermore, the expected size of the spread ( $\sigma(x)$  and  $\sigma(p)$ ) is at least as large as the expected size of the seed set, i.e., for any  $x \in \mathcal{X}$  and  $p \in \mathcal{P}$ , it holds that  $\sigma(x) \geq k$  and  $\sigma(p) \geq k$ . Thus,  $\max_{p \in F_{\mathcal{P}}(G, \mathcal{C}, k)} \sigma(p) \geq k$  and  $\max_{x \in F_{\mathcal{X}}(G, \mathcal{C}, k)} \sigma(x) \geq k$ . Together with  $\sigma(S) \leq n$  for any set  $S$ , we obtain an upper bound of  $n/k$  for the price of (group) fairness.  $\square$

**Pessimistic Price of Fairness.** A more pessimistic point of view leads to a different definition of the price of fairness. Indeed, the reader might have wondered if it is the correct choice to define  $\text{PoF}_{\mathcal{X}}$  and  $\text{PoF}_{\mathcal{P}}$  using the maximum spread over all fair solutions in the denominator. In fact, if we just compute any fair solution, the loss in terms of efficiency that we may incur could be as large as

$$\overline{\text{PoF}}_{\mathcal{X}}(G, \mathcal{C}, k) := \frac{\max_{S \in \binom{V}{k}} \sigma(S)}{\min_{x \in F_{\mathcal{X}}(G, \mathcal{C}, k)} \sigma(x)} \quad \text{and} \quad \overline{\text{PoF}}_{\mathcal{P}}(G, \mathcal{C}, k) := \frac{\max_{S \in \binom{V}{k}} \sigma(S)}{\min_{p \in F_{\mathcal{P}}(G, \mathcal{C}, k)} \sigma(p)},$$

for the node-based problem and the set-based problem, respectively. We call this alternative definition the *pessimistic price of fairness* for a graph  $G$ , community structure  $\mathcal{C}$  and budget  $k$ . We note that clearly  $\overline{\text{PoF}}_{\mathcal{X}}(G, \mathcal{C}, k) \geq \text{PoF}_{\mathcal{X}}(G, \mathcal{C}, k)$  and  $\overline{\text{PoF}}_{\mathcal{P}}(G, \mathcal{C}, k) \geq \text{PoF}_{\mathcal{P}}(G, \mathcal{C}, k)$ . Moreover, we note that Lemma 3.1 holds still for this alternative definition as  $\sigma(x) = \sigma(p) = n \cdot \rho$  for all  $x \in F_{\mathcal{X}}(G, \mathcal{C}, 1)$  and  $p \in F_{\mathcal{P}}(G, \mathcal{C}, 1)$ . In contrast, we observe that Lemma 3.2 does not transfer to the pessimistic notion. In fact, we obtain only the following weaker lemma.

**Lemma 3.3.** *For any graph  $G$ , community structure  $\mathcal{C}$  and number  $k$ , it holds that  $\overline{\text{PoF}}_{\mathcal{X}}(G, \mathcal{C}, k) \leq n$  and  $\overline{\text{PoF}}_{\mathcal{P}}(G, \mathcal{C}, k) \leq n$ .*

*Proof.* Recall that all communities are non-empty. Fix an arbitrary community  $C$ . Now, consider  $\sigma_C(x)$  as a function of  $x_v$  for a node  $v \in C$ . This function is strictly monotonically increasing in  $[0, 1]$ . And thus all fair solutions  $x$  satisfy that the expected size of the seed set is at least 1, i.e.,  $\mathbf{1}^T x \geq 1$ . Similarly, it can be seen that all fair solutions  $p$  to the set-based problem satisfy  $\sum_{S \subseteq V} p_S |S| \geq 1$ . Furthermore, the expected size of the spread  $\sigma(x)$  and  $\sigma(p)$  is larger than the expected size of the seed set, i.e., for any fair  $x$  and  $p$ , it holds that  $\sigma(x) \geq 1$  and  $\sigma(p) \geq 1$ . Thus  $\min_{p \in F_{\mathcal{P}}(G, \mathcal{C}, k)} \sigma(p) \geq 1$  and  $\min_{x \in F_{\mathcal{X}}(G, \mathcal{C}, k)} \sigma(x) \geq 1$ . Together with  $\sigma(S) \leq n$  for any set  $S$ , we obtain an upper bound of  $n$  for the price of (group) fairness.  $\square$

It turns out that the above bound is tight. Consider the following example. The graph  $G$  consists of one isolated node  $v$  and a (to  $v$  unconnected) clique of  $n - 1$  nodes with edge probabilities being 1. Assume furthermore that the community structure  $\mathcal{C}$  is such that the nodes in the clique do not participate in any community, while  $v$  forms its own community. Furthermore, assume that the budget  $k$  is 2. The node-based solution  $x$  that is zero everywhere but for  $x_v = 1$  and the set-based solution  $p$  that is zero everywhere but for  $p_{\{v\}} = 1$  are optimal fair solutions as they achieve an objective value of 1. The deterministic solution  $S = \{u, v\}$ , where  $u$  is an arbitrary node in the clique however satisfies  $\sigma(S) = n$  and thus  $\overline{\text{PoF}}_{\mathcal{X}}(G, \mathcal{C}, 2) \geq n$  and  $\overline{\text{PoF}}_{\mathcal{P}}(G, \mathcal{C}, 2) \geq n$ . Finally, we remark that the above example heavily depends on the fact that it is not necessary to use the whole budget  $k$  in order to obtain an optimal fair node-based or set-based solution.

### 3.2 Hardness

Fish et al. (2019) show that the standard maximin problem as introduced in Section 2 is NP-hard and even inapproximable. In this section, we provide hardness results for our set-based and node-based probabilistic maximin problems that we introduced in Section 2.2. In the first paragraph of this subsection, we prove that both problems are NP-hard. In the second paragraph, we even show that the node-based probabilistic maximin problem cannot be approximated to within  $1 - 1/e + \varepsilon$  for any  $\varepsilon > 0$  unless  $P = NP$ . We note that, although it shows a stronger result, our reduction in the second paragraph is significantly less involved than the NP-hardness result in the first paragraph. The reader may thus wonder why we still present the more involved proof of the weaker NP-hardness of the node-based problem. The reason for this choice is that we think that the first reduction gives further insight into how the hardness is implied from the fairness criteria. Note that the reduction

in the second paragraph uses a single community and thus, in a certain sense, the hardness of approximation is inherent to maximizing influence spread rather than due to any fairness issue.

**NP-Hardness.** The main result of this paragraph is the following theorem.

**Theorem 3.4.** *For a directed arc-weighted graph  $G = (V, E, w)$  it is NP-hard to decide if there is  $p \in \mathcal{P}$  with  $\min_{v \in V} \mathbb{E}_{S \sim p}[\sigma_v(S)] \geq \alpha$  (resp.  $x \in \mathcal{X}$  with  $\min_{v \in V} \mathbb{E}_{S \sim x}[\sigma_v(S)] \geq \alpha$ ) for any  $\alpha \in (0, 1)$  even when using the IC model.*

The proof of the theorem is based on a reduction from the *vertex cover problem*. In the vertex cover problem, we are given a graph  $G = (V, E)$  where  $V$  is a set of  $n$  vertices and  $E$  is a set of  $m$  edges, and an integer  $k$ . The task is to determine if there exists a set  $T = \{v_{i_1}, \dots, v_{i_k}\}$  of  $k$  vertices such that  $\forall e \in E, e \cap T \neq \emptyset$ . We proceed by describing the reduction.

Let  $\alpha \in (0, 1)$  be arbitrary. Given an instance of the vertex cover problem, we create instances of the set-based and node-based probabilistic maximin problem defined as follows, see Figure 3. Let us call the resulting directed graph  $\overline{G} = (U, A)$  and let us assume that the IC model is the underlying diffusion model and that we are considering the singleton community structure. The node set  $U$  contains (1) one vertex  $u_v$  for each vertex  $v \in V$ , (2) one auxiliary vertex  $u_a$ , (3) a vertex  $u_e$  for each edge  $e \in E$ , (4) a set  $I_e = \{u_{e^1}, \dots, u_{e^\lambda}\}$  of  $\lambda := \lceil \frac{mk(k+1)}{\alpha(1-\alpha)^2} \rceil + 1$  vertices for every edge  $e \in E$ . The edge set  $A$  is defined as follows: (1) Each vertex  $u_v$  has an outgoing edge towards  $u_a$  labelled with probability 1, while the vertex  $u_a$  has an outgoing edge with probability  $\alpha$  towards each vertex  $u_v$ . (2) There is an edge labelled with probability 1 from  $u_v$  to  $u_e$  if  $v \in e$ . (3) There are edges from  $u_e$  to all vertices  $u_{e^i}$  for each edge  $e \in E$  labelled with probability  $\alpha$ . We set the budget for both set-based and node-based problems equal to  $k$ . Our aim is now to show that there exists a vertex cover of

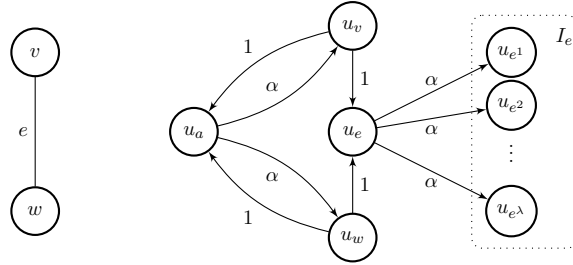


Figure 3: Illustration of the reduction: Scheme in  $\overline{G}$  that is obtained for one edge  $e = \{v, w\}$  in  $G$ . Note that there is just one node  $u_a$  in  $\overline{G}$ , while there is a node  $u_e$  and a set of nodes  $I_e$  for each edge  $e \in E$ .

size  $k$  in  $G$  if and only if there exists  $p \in \mathcal{P}$  (resp.  $x \in \mathcal{X}$ ) such that  $\min_{u \in U} \mathbb{E}_{S \sim p}[\sigma_u(S)] \geq \alpha$  (resp.  $\min_{u \in U} \mathbb{E}_{S \sim x}[\sigma_u(S)] \geq \alpha$ ) in  $\overline{G}$ . The following direction is immediate:

**Lemma 3.5.** *If there exists a vertex cover  $\{v_{i_1}, \dots, v_{i_k}\}$  of size  $k$  in  $G$ , then there exists  $p \in \mathcal{P}$  (resp.  $x \in \mathcal{X}$ ) such that  $\min_{u \in U} \sigma_u(p) \geq \alpha$  (resp.  $\min_{u \in U} \sigma_u(x) \geq \alpha$ ) in  $\overline{G}$ .*

*Proof.* Consider the set-based solution  $p \in \mathcal{P}$  such that  $p_S = 1$  for  $S = \{u_{v_{i_1}}, \dots, u_{v_{i_k}}\}$  and the node-based solution  $x \in \mathcal{X}$  such that  $x_u = 1$  if  $u \in \{u_{v_{i_1}}, \dots, u_{v_{i_k}}\}$ . Then, clearly  $\min_{u \in U} \mathbb{E}_{S \sim p}[\sigma_u(S)] \geq \alpha$  and  $\min_{u \in U} \mathbb{E}_{S \sim x}[\sigma_u(S)] \geq \alpha$ .  $\square$

It remains to argue the reverse direction. We first fix the following two observations.

**Observation 3.6.** 1. *There exists an optimal solution  $p^*$  (resp.  $x^*$ ) such that, for any set  $S \subseteq V$  with  $(\{u_a\} \cup \{u_e : e \in E\}) \cap S \neq \emptyset$ , it holds that  $p_S = 0$  (resp.  $p_S^{x^*} = 0$ , where  $p_S^{x^*} := \prod_{i \in S} x_i^* \prod_{j \in V \setminus S} (1 - x_j^*)$ ).*

2. *For an edge  $e = \{v, w\} \in E$ , let  $u_{e^i} \in I_e$  be a corresponding vertex and let  $S \subseteq U$ . Then (i)  $\sigma_{u_{e^i}}(S) = 1$  if  $u_{e^i} \in S$ , (ii)  $\sigma_{u_{e^i}}(S) = \alpha$  if  $u_{e^i} \notin S$  and  $\{u_v, u_w\} \cap S \neq \emptyset$ , and (iii)  $\sigma_{u_{e^i}}(S) \leq \alpha(1 - (1 - \alpha)^2)$  otherwise.*

*Proof.* 1. Assume that  $p_S > 0$  (resp.  $p_S^x > 0$ ) for some set  $S \subseteq V$  with  $(\{u_a\} \cup \{u_e : e \in E\}) \cap S \neq \emptyset$ . Let  $u$  be a node in the intersection and let  $S'$  be the set  $S$  with  $u$  replaced by one of its in-neighbors, call it  $u'$ . Then, clearly  $p' := p - p_S \mathbf{1}_S + p_S \mathbf{1}_{S'}$  (resp.  $x' := x - x_u \mathbf{1}_u + x_u \mathbf{1}_{u'}$ ) satisfies  $\sigma_u(p') \geq \sigma_u(p)$  (resp.  $\sigma_u(x') \geq \sigma_u(x)$ ) for all vertices  $u \in U$ . It follows that  $p$  (resp.  $x$ ) can be transformed into a distribution (resp. vector) satisfying  $p_S = 0$  (resp.  $p_S^{x^*} = 0$ ) for all  $S \subseteq V$  with  $(\{u_a\} \cup \{u_e : e \in E\}) \cap S \neq \emptyset$ .

2. The first two cases are trivially true. For the third case, note that in that case  $u_{e^i}$  can be reached at most via  $u_a$ . The probability that at least one of  $u_v, u_w$  gets reached from  $u_a$  is at most  $1 - (1 - \alpha)^2$  and thus  $\sigma_{u_{e^i}}(S) \leq \alpha(1 - (1 - \alpha)^2)$ .  $\square$

We are now ready to prove the lemma showing the reverse direction.

**Lemma 3.7.** *If there exists no vertex cover  $\{v_{i_1}, \dots, v_{i_k}\}$  of size  $k$  in  $G$ , then for all optimal solutions  $p^* \in \mathcal{P}$  and  $x^* \in \mathcal{X}$  it holds that  $\min_{u \in U} \sigma_u(x^*) \leq \min_{u \in U} \sigma_u(p^*) < \alpha$  in  $\overline{G}$ .*

*Proof.* Following Observation 3.6, let  $p^*$  be such that, for any set  $S \subseteq V$  with  $(\{u_a\} \cup \{u_e : e \in E\}) \cap S \neq \emptyset$ , it holds that  $p_S^* = 0$ . Our goal is to show that there exists one vertex  $u \in U$  such that  $\sigma_u(p^*) < \alpha$ . Let  $e = \{v, w\} \in E$  be arbitrary and recall that  $I_e$  is of size  $\lambda$ . Thus, there exists  $i \in [\lambda]$  such that  $\Pr_{S \sim p^*}[u_{e^i} \in S] \leq k/\lambda$  as otherwise  $\mathbb{E}_{S \sim p^*}[|S|] > k$ . W.l.o.g. let us assume that  $i = 1$  for all edges  $e \in E$ . Observe that then, for all  $e \in E$ , it holds that

$$\begin{aligned} \sigma_{u_{e^1}}(p^*) &= \Pr_{S \sim p^*}[u_{e^1} \in S] + \mathbb{E}_{S \sim p^*}[\sigma_{u_{e^1}}(S) \mid u_{e^1} \notin S] \cdot \Pr_{S \sim p^*}[u_{e^1} \notin S] \\ &\leq \frac{k}{\lambda} + \mathbb{E}_{S \sim p^*}[\sigma_{u_{e^1}}(S) \mid u_{e^1} \notin S]. \end{aligned} \quad (1)$$

In order to upper bound the above expectation, we define  $\rho_e := \Pr_{S \sim p^*}[S \cap \{u_v, u_w\} = \emptyset]$ . Using Observation 3.6, we get

$$\mathbb{E}_{S \sim p^*}[\sigma_{u_{e^1}}(S) \mid u_{e^1} \notin S] \leq \alpha(1 - (1 - \alpha)^2) \cdot \rho_e + \alpha \cdot (1 - \rho_e) = \alpha - \rho_e \cdot \alpha(1 - \alpha)^2. \quad (2)$$

In order to complete the proof it thus remains to find  $e \in E$  for which  $\rho_e$  can be bounded from below. We deduce this bound from a lower bound on the sum of all  $\rho_e$ 's. We have

$$\begin{aligned} \sum_{e \in E} \rho_e &\geq \sum_{e=(v,w) \in E} \mathbb{E}_{S \sim p^*}[\mathbf{1}_{S \cap \{u_v, u_w\} = \emptyset} \mid |S| \leq k] \cdot \Pr_{S \sim p^*}[|S| \leq k] \\ &\geq \frac{\mathbb{E}_{S \sim p^*}[\sum_{e \in E} \mathbf{1}_{S \cap \{u_v, u_w\} = \emptyset} \mid |S| \leq k]}{k+1} \end{aligned}$$

using that  $\mathbb{E}_{S \sim p^*}[|S|] \leq k$  and hence using Markov's inequality

$$\Pr_{S \sim p^*}[|S| \leq k] = 1 - \Pr_{S \sim p^*}[|S| \geq k+1] \geq 1 - \frac{k}{k+1} = \frac{1}{k+1}.$$

We now use that there exists no vertex cover of size  $k$  in  $G$  and thus for each  $S$  with  $|S| \leq k$ , there exists  $e \in E$  such that  $S \cap \{u_v, u_w\} = \emptyset$ . Hence, we get  $\sum_{e \in E} \rho_e \geq 1/(k+1)$ . This also implies that there exists one edge  $\bar{e} \in E$  for which  $\rho_{\bar{e}} \geq 1/(m(k+1))$ . Plugging this into (2) and this again into (1) gives

$$\sigma_{u_{\bar{e}^1}}(p^*) \leq \frac{k}{\lambda} + \alpha - \frac{\alpha(1-\alpha)^2}{m(k+1)} < \alpha,$$

using that  $\lambda > \frac{mk(k+1)}{\alpha(1-\alpha)^2}$  by definition.  $\square$

The above two lemmata directly prove Theorem 3.4.

**Hardness of Approximation for Node-Based Problem.** We continue by proving an even stronger result for the node-based probabilistic maximin problem.

**Theorem 3.8.** *For any  $\varepsilon > 0$ , the node-based probabilistic maximin problem, cannot be approximated to within a factor  $1 - 1/e + \varepsilon$  unless  $P = NP$ .*

*Proof.* The proof is by reduction from the MAX-COVERAGE problem. An instance of MAX-COVERAGE is given by an integer  $\kappa$  and a collection of subsets  $D = \{S_1, \dots, S_\mu\}$  of a universe of elements  $U = \{e_1, \dots, e_\nu\}$ . The task is to find a subset  $T$  of at most  $\kappa$  subsets from  $D$  such that the number of elements in their union  $|\bigcup_{S \in T} S|$  is maximized. Recall that, for any  $\varepsilon > 0$ , MAX-COVERAGE cannot be approximated to within  $1 - 1/e + \varepsilon$ , unless  $P = NP$  (Feige, 1998, Theorem 5.3).

We construct the following instance of the node-based probabilistic maximin problem. The graph  $G = (V, E)$  has an nodeset  $V$  that consists of two sets of nodes, the first being  $L := \{u^1, \dots, u^\mu\}$  and the second being  $R := \{v^1, \dots, v^\nu\}$ . There is an edge from node  $u^i$  to node  $v^j$ , if and only if  $e_j \in S_i$ . The weight of all these edges is 1. The community structure  $\mathcal{C}$  consists of a single community  $C$  that is equal to  $R$ . Now, w.l.o.g., we can assume that any node-based solution  $x$  satisfies  $x_v = 0$  for all nodes  $v \in R$ . Suppose otherwise, then transferring  $x_v$  to any of  $v$ 's ingoing neighbors can only increase the value of  $\sigma_C(x)$ . There is a one-to-one correspondence between the feasible sets to MAX-COVERAGE and the integral solutions to the node-based probabilistic maximin problem as follows. A feasible set  $T$  in MAX-COVERAGE that covers  $\ell$  elements implies an integral solution to the node-based probabilistic maximin problem with  $x_{u^i} = 1$  for all  $S_i$  in  $T$  and 0 otherwise that achieves a value of  $\sigma_C(x) = \ell/\nu$ . The other direction holds as well.

Now, let  $\varepsilon > 0$  and assume that there exists an algorithm with approximation ratio  $1 - 1/e + \varepsilon$  for the node-based probabilistic maximin problem. Let  $x \in \mathcal{X}$  be the output of that algorithm for the instance constructed above. Notice that the objective function in the node-based probabilistic maximin problem is equal to

$$f(x) = \frac{1}{\nu} \sum_{v \in R} \sigma_v(x) = \frac{1}{\nu} \sum_{v \in R} \left(1 - \prod_{u \in \delta^{in}(v)} (1 - x_u)\right),$$

where  $\delta^{in}(v)$  denote all the ingoing neighbors of node  $v$ . It is clear that  $f$  is  $\varepsilon$ -convex in the sense of Ageev and Sviridenko (2004) and thus Pipeage rounding (Ageev & Sviridenko, 2004) can be used in order to compute a vector  $\mathbf{1}_T$  from  $x$  such that  $f(\mathbf{1}_T) \geq f(x)$ . Denoting with  $x^*$  and  $T^*$  an optimal solution to the node-based probabilistic maximin problem and the MAX-COVERAGE problem, respectively, this implies

$$\frac{|\bigcup_{S \in T} S|}{\nu} = f(\mathbf{1}_T) \geq f(x) \geq \left(1 - \frac{1}{e} + \varepsilon\right) \cdot f(x^*) \geq \left(1 - \frac{1}{e} + \varepsilon\right) \cdot f(\mathbf{1}_{T^*}) = \left(1 - \frac{1}{e} + \varepsilon\right) \cdot \frac{|\bigcup_{S \in T^*} S|}{\nu}.$$

Hence, we obtain an algorithm for MAX-COVERAGE with approximation ratio  $1 - 1/e + \varepsilon$  which is impossible unless  $P = NP$ . This completes the proof.  $\square$

### 4. Approximation Algorithms

In this section, we show that there are algorithms that compute  $(1 - 1/e, \varepsilon)$ -approximations to both the node-based and the set-based probabilistic maximin problems. We start with a standard step that allows us to approximate the functions  $\sigma_C(p)$  and  $\sigma_C(x)$  to within an additive error of  $\varepsilon$  for any  $\varepsilon > 0$ .

**Approximation via Hoeffding’s bound.** The functions  $\sigma_C(p)$  and  $\sigma_C(x)$  involved in the optimization problems at hand are not computable exactly in polynomial time (even for a vector  $p$  of polynomial support). Even worse, they are not relatively approximable using Chernoff bounds as there is no straightforward absolute lower bound on  $\sigma_C(S)$  for sets  $S$  of size  $k$  and communities  $C \in \mathcal{C}$ . Here, we will show that the functions can be absolutely approximated by functions  $\tilde{\sigma}_C(p)$  and  $\tilde{\sigma}_C(x)$  that are obtained by sampling a sufficiently large number of live-edge graphs. Optimal solutions to the resulting maximin problems involving the approximate functions can thus be shown to be  $(1, \varepsilon)$ -approximations to  $\text{opt}_{\mathcal{P}}(G, \mathcal{C}, k)$  and  $\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k)$ , respectively.

Formally, for  $T \in \mathbb{Z}_{>0}$ , we let  $L_1, \dots, L_T$  denote a set of  $T$  live-edge graphs sampled according to the Triggering model (that entails both the IC and LT model). Then, for  $v \in V$  and  $S \in 2^V$ , we define

$$\tilde{\sigma}_v(S) := \frac{1}{T} \sum_{t=1}^T \mathbb{1}_{v \in \rho_{L_t}(S)}.$$

**Lemma 4.1.** *Let  $\delta \in (0, 1/2)$  and  $\varepsilon \in (0, 1)$ . If  $T \geq \varepsilon^{-2} \cdot [n + \log n + \log \delta^{-1}]$ , then, with probability at least  $1 - \delta$ , we have that  $|\tilde{\sigma}_v(S) - \sigma_v(S)| \leq \varepsilon$  holds for all  $v \in V$  and  $S \in 2^V$ .*

*Proof.* We use Hoeffding’s Bound, see for example Theorem 4.12 in the book by Mitzenmacher and Upfal (2017). Fix a node  $v \in V$  and a set  $S \in 2^V$ . Note that the graphs  $L_1, \dots, L_T$  are sampled independently and that  $\mathbb{1}_{v \in \rho_{L_t}(S)} \in [0, 1]$ . Hence,  $\Pr[|\tilde{\sigma}_v(S) - \sigma_v(S)| \geq \varepsilon] \leq 2e^{-2T\varepsilon^2} \leq \delta \cdot 2^{-n}/n$  by the choice of  $T$  and assuming that  $n \geq 2$ . Using a union bound over all  $2^n$  sets  $S \in 2^V$  and all  $n$  nodes  $v \in V$ , we obtain that with probability at least  $1 - \delta$ , we have  $|\tilde{\sigma}_v(S) - \sigma_v(S)| \leq \varepsilon$  for all  $v \in V$  and for all  $S \in 2^V$ .  $\square$

We now observe that the absolute  $\varepsilon$ -approximations  $\tilde{\sigma}_v(S)$  for all nodes  $v \in V$  and sets  $S \in 2^V$  imply also that  $\tilde{\sigma}_v(p) = \mathbb{E}_{S \sim p}[\tilde{\sigma}_v(S)]$  is an absolute  $\varepsilon$ -approximation of  $\sigma_v(p) := \mathbb{E}_{S \sim p}[\sigma_v(S)]$  for any  $p \in \mathcal{P}$ . The same holds true for  $\tilde{\sigma}_v(x) = \mathbb{E}_{S \sim x}[\tilde{\sigma}_v(S)]$  for any  $x \in \mathcal{X}$ .

Furthermore, we get the same result for  $\tilde{\sigma}_C(p) := \frac{1}{|C|} \sum_{v \in C} \tilde{\sigma}_v(p)$  for any  $p \in \mathcal{P}$  and  $C \in \mathcal{C}$  and for  $\tilde{\sigma}_C(x) := \frac{1}{|C|} \sum_{v \in C} \tilde{\sigma}_v(x)$  for any  $x \in \mathcal{X}$  and  $C \in \mathcal{C}$  as these functions are again just averages over other absolute  $\varepsilon$ -approximations. Hence we get the following lemma.

**Lemma 4.2.** *Let  $\delta \in (0, 1/2)$  and  $\varepsilon \in (0, 1)$ . Assume that  $T \geq 4\varepsilon^{-2} \cdot [n + \log n + \log \delta^{-1}]$  and that  $\tilde{\sigma}_C(\cdot)$  is as above. Let  $p \in \mathcal{P}$  and  $x \in \mathcal{X}$  be  $(\alpha, \beta)$ -approximations for  $\max_{p \in \mathcal{P}} \min_{C \in \mathcal{C}} \tilde{\sigma}_C(p)$  and  $\max_{x \in \mathcal{X}} \min_{C \in \mathcal{C}} \tilde{\sigma}_C(x)$ , respectively. Then  $p$  and  $x$  are  $(\alpha, \beta + \varepsilon)$ -approximations of  $\text{opt}_{\mathcal{P}}(G, \mathcal{C}, k)$  and  $\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k)$  with probability at least  $1 - \delta$ , respectively.*

*Proof.* For any  $q \in \mathcal{P}$ , define  $m(q) := \min_{C \in \mathcal{C}} \sigma_C(q)$  and  $\tilde{m}(q) := \min_{C \in \mathcal{C}} \tilde{\sigma}_C(q)$ . Then, according to Lemma 4.1 and the comments preceding this lemma, with probability at least  $1 - \delta$ , it holds that  $\tilde{m}(q) \in [m(q) - \varepsilon/2, m(q) + \varepsilon/2]$ . Let  $p^*$  and  $\tilde{p}^*$  be maximizing solutions for  $m$  and  $\tilde{m}$ , respectively. Then

$$m(p) \geq \tilde{m}(p) - \frac{\varepsilon}{2} \geq \alpha \cdot \tilde{m}(\tilde{p}^*) - \beta - \frac{\varepsilon}{2} \geq \alpha \cdot \tilde{m}(p^*) - (\beta + \frac{\varepsilon}{2}) \geq \alpha \cdot m(p^*) - (\beta + \varepsilon).$$

The proof for the node-based problem is completely analogous.  $\square$

#### 4.1 Probabilistically Choosing Nodes

We start with the node-based problem. It entails to solve the optimization problem  $\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k) := \max_{x \in \mathcal{X}} \min_{C \in \mathcal{C}} \sigma_C(x)$ , where  $\sigma_C(x) = \mathbb{E}_{S \sim x}[\sigma_C(S)]$  for  $C \in \mathcal{C}$  and  $x \in \mathcal{X} := \{x \in [0, 1]^n : \mathbf{1}^T x \leq k\}$ . Recall that  $S \sim x$  denotes the random process of independently letting  $i \in V$  be in  $S$  with probability  $x_i$ . We use Lemma 4.2 in order to approximate  $\sigma_C(\cdot)$  and thus, in what follows, we focus on finding an approximation algorithm for the problem  $\max_{x \in \mathcal{X}} \min_{C \in \mathcal{C}} \tilde{\sigma}_C(x)$ . We note that Theorem II.5 from Chekuri et al. (2010) in combination with a binary search on a threshold can be used in order to get a  $(1 - 1/e, 0)$ -approximation for this problem. In what follows we give a more direct derivation of such an approximation.

**Node-based Problem via LP.** Note that the optimization problem at hand is not linear as, for given  $x$ , the probability to sample  $S \in 2^V$  is equal to  $\prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i)$ . We will now argue however that the problem can be constantly approximated by an LP.

For a live-edge graph  $L$  and a node  $v \in V$ , what is the probability of sampling a set  $S$  that can reach  $v$  in  $L$ , i.e., what is  $q_v(L, x) := \Pr_{S \sim x}[v \in \rho_L(S)]$ ? It is the opposite event of not sampling any node that can reach  $v$  in  $L$ , hence  $q_v(L, x) = 1 - \prod_{i \in V: v \in \rho_L(i)} (1 - x_i)$  and this is approximated by the function  $p_v(L, x) := \min\{1, \sum_{i \in V: v \in \rho_L(i)} x_i\}$  as shown in the following observation.

**Observation 4.3.** *For any live-edge graph  $L$ , node  $v \in V$ , and  $x \in \mathcal{X}$ , it holds that  $q_v(L, x) \in [(1 - \frac{1}{e}) \cdot p_v(L, x), p_v(L, x)]$ .*

*Proof.* We start with the lower bound. For simplicity, we let  $\{i \in V : v \in \rho_L(i)\} =: \{1, \dots, r\} = R$ . Using the geometric-arithmetic mean inequality, we get

$$\begin{aligned} q_v(L, x) &= 1 - \prod_{i \in R} (1 - x_i) \geq 1 - \left(\frac{1}{r} \sum_{i \in R} (1 - x_i)\right)^r = 1 - \left(1 - \frac{1}{r} \sum_{i \in R} x_i\right)^r \\ &\geq \left(1 - \left(1 - \frac{1}{r}\right)^r\right) \cdot \min\left\{1, \sum_{i \in R} x_i\right\} \geq \left(1 - \frac{1}{e}\right) \cdot p_v(L, x), \end{aligned}$$

where the second to last inequality uses that  $f(x) = 1 - (1 - x/r)^r$  is concave on the interval  $[0, 1]$ .

We prove the upper bound by induction on  $r$ . Clearly if  $r = 1$ , by the definition of  $\mathcal{X}$ , we have that  $p_v(L, x) = \min\{1, x_1\} = x_1 = q_v(L, x)$ . Let us show the statement for  $r$ , assuming that it holds for  $r - 1$ . If  $p_v(L, x) = 1$ , the statement is obvious as  $q_v(L, x) \leq 1$ . If  $p_v(L, x) < 1$ , we get

$$q_v(L, x) = 1 - \prod_{i=1}^{r-1} (1 - x_i) + x_r \cdot \prod_{i=1}^{r-1} (1 - x_i) \leq \sum_{i=1}^{r-1} x_i + x_r \cdot \prod_{i=1}^{r-1} (1 - x_i) \leq \sum_{i=1}^r x_i = p_v(L, x),$$

where the first inequality uses the induction hypothesis and  $\min\{1, \sum_{i=1}^{r-1} x_i\} = \sum_{i=1}^{r-1} x_i$  as  $p_v(L, x) < 1$ , while the second inequality uses that  $\prod_{i=1}^{r-1} (1 - x_i) \leq 1$ .  $\square$

We now define  $\lambda_v(x) := \frac{1}{T} \sum_{t=1}^T p_v(L_t, x)$  and analogously  $\lambda_C(x) := \frac{1}{|C|} \sum_{v \in C} \lambda_v(x)$ . As  $p_v(L, x)$  provides an approximation for  $q_v(L, x)$ , we can show that the node-based maximin problem can be approximated by a solution to a maximin problem involving the functions  $\lambda_C(x)$ .

**Lemma 4.4.** *Let  $x \in \mathcal{X}$  be an optimal solution to  $\max_{x \in \mathcal{X}} \min_{C \in \mathcal{C}} \lambda_C(x)$ , then  $x$  is a  $(1 - 1/e, 0)$ -approximation to  $\max_{x \in \mathcal{X}} \min_{C \in \mathcal{C}} \tilde{\sigma}_C(x)$ .*

*Proof.* Note that  $\tilde{\sigma}_C(x) = \mathbb{E}_{S \sim x}[\tilde{\sigma}_C(S)] = \frac{1}{|C|} \sum_{v \in C} \tilde{\sigma}_v(x)$  for any  $x \in \mathcal{X}$  and that furthermore  $\tilde{\sigma}_v(x) = \frac{1}{T} \sum_{t=1}^T q_v(L_t, x)$ . Hence the definition of  $\lambda_C(x)$  and Observation 4.3 yield the result.  $\square$



Together with Lemma 4.2 and the fact that  $\max_{x \in \mathcal{X}} \min_{C \in \mathcal{C}} \lambda_C(x)$  can be written as a linear program by introducing a threshold variable for the minimum, we get the following result.

**Theorem 4.5.** *Let  $\delta \in (0, 1/2)$  and  $\varepsilon \in (0, 1)$ . There is a polynomial time algorithm that, with probability at least  $1 - \delta$ , computes  $x \in \mathcal{X}$  such that  $\min_{C \in \mathcal{C}} \sigma_C(x) \geq (1 - \frac{1}{e}) \text{opt}_{\mathcal{X}}(G, \mathcal{C}, k) - \varepsilon$ .*

*Proof.* It remains to observe that an optimal solution to  $\max_{x \in \mathcal{X}} \min_{C \in \mathcal{C}} \lambda_C(x)$  can be obtained by solving the following linear program of polynomial size:

$$\begin{aligned} \max \left\{ \tau : \sum_{i=1}^n x_i \leq k, y_{vt} \leq \sum_{i: v \in \rho_{L_t}(i)} x_i \forall v \in V, t \in [T], \right. \\ \left. \sum_{t \in [T]} \sum_{v \in C} y_{vt} \geq T|C| \cdot \tau \forall C \in \mathcal{C}, \right. \\ \left. x \in [0, 1]^n, y_{vt} \in [0, 1] \forall v \in V, t \in [T] \right\}. \end{aligned}$$

By combining Lemma 4.4 and Lemma 4.2, we get that, with probability at least  $1 - \delta$ , the optimal solution to the above LP is a  $(1 - 1/e, \varepsilon)$ -approximation to  $\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k)$ .  $\square$

## 4.2 Probabilistically Choosing Sets

Recall the set-based probabilistic maximin problem  $\text{opt}_{\mathcal{P}}(G, \mathcal{C}, k) := \max_{p \in \mathcal{P}} \min_{C \in \mathcal{C}} \sigma_C(p)$ , where  $\sigma_C(p) = \mathbb{E}_{S \sim p}[\sigma_C(S)]$  for  $C \in \mathcal{C}$  and  $p \in \mathcal{P} := \{p \in [0, 1]^{2^V} : \mathbf{1}^T p = 1, \sum_{S \subseteq V} p_S |S| \leq k\}$ . In the light of Lemma 4.2, we focus on finding approximate solutions to  $\max_{p \in \mathcal{P}} \min_{C \in \mathcal{C}} \tilde{\sigma}_C(p)$ .

Allowing for distributions over sets rather than nodes turns the optimization problem at hand,  $\max_{p \in \mathcal{P}} \min_{C \in \mathcal{C}} \tilde{\sigma}_C(p)$ , into a problem that can be written as a linear program. While the original problem, i.e., the problem of choosing a set maximizing the approximate minimum probability, can be written as an integer linear program using a variable to model a threshold to be maximized. Hence, from an algorithmic point of view, one may think that this makes the problem polynomial time solvable. The caveat is of course that the dimension of  $\mathcal{P}$  is large, namely  $\Theta(2^n)$ , which turns the dimension of the corresponding linear program exponential. In this section, we show that, nevertheless, the problem can be approximated to within a constant factor using a specific kind of linear programming algorithm. The essential observation is that the linear program at hand actually is a covering linear program. We will use a result due to Young (1995) that shows that such linear programs can be solved efficiently independent of their dimension under the condition that a certain oracle problem can be solved efficiently. We proceed by introducing the result of Young.

**Young’s Algorithm.** Young (1995) gives algorithms for solving packing and covering linear programs. A covering problem in the sense of Young is of the following form: Let  $P \subseteq \mathbb{R}^\nu$  be a convex set and let  $f : P \rightarrow \mathbb{R}^\mu$  be a  $\mu$ -dimensional linear function over  $P$ . Assume that  $0 \leq f_j(x) \leq \omega$  for all  $j \in [\mu]$  and  $x \in P$ , where  $\omega$  is the width of  $P$  w.r.t.  $f$ . The *covering problem* consists of computing  $\lambda^* := \max_{x \in P} \min_{j \in [\mu]} f_j(x)$ , when  $f_j(x) \geq 0$  for all  $x \in P$ .

**Theorem 4.6** (Young (1995)). *Let  $\eta \in (0, 1)$  and assume that there is an oracle that, given a non-negative vector  $z \in \mathbb{R}^\mu$  returns  $x \in P$  and  $f(x)$  satisfying  $\sum_{j \in [m]} z_j f_j(x) \geq \alpha \cdot \max_{x \in P} \{\sum_{j \in [m]} z_j f_j(x)\}$  for some constant  $\alpha \leq 1$ , then there is an algorithm that computes  $x \in P$  with  $\min_{j \in [\mu]} f_j(x) \geq \alpha(1 - \eta) \cdot \lambda^*$  in  $O(\omega \eta^{-2} \log \mu / \lambda^*)$  iterations in each of which it does  $O(\mu)$  work and calls the oracle once. The output  $x$  is the arithmetic mean of the vectors returned by the oracle.*

**Set-Based Problem via Young's Algorithm.** Clearly  $\tilde{\sigma}_C$  is a linear function in  $p$ , namely  $\tilde{\sigma}_C(p) = \sum_{S \subseteq V} p_S \tilde{\sigma}_C(S)$  and thus the problem  $\max_{p \in \mathcal{P}} \min_{C \in \mathcal{C}} \tilde{\sigma}_C(p)$  takes exactly the form of a covering problem in the sense of Young with  $\nu = 2^n$ ,  $\mu = m = |\mathcal{C}|$ ,  $P = \mathcal{P}$ , and  $\omega = 1$ . Hence, we can compute a  $(\alpha, 0)$ -approximation for  $\max_{p \in \mathcal{P}} \min_{C \in \mathcal{C}} \tilde{\sigma}_C(p)$ , if we provide an oracle with multiplicative approximation  $\alpha$ . Hence, let us take a closer look at the requirements of Theorem 4.6 in terms of the oracle problem. Given a non-negative vector  $z \in \mathbb{R}^m$ , the oracle is required to return  $p \in \mathcal{P}$  and  $\tilde{\sigma}_C(p)$  for  $C \in \mathcal{C}$  such that  $\sum_{C \in \mathcal{C}} z_C \tilde{\sigma}_C(p) \geq \alpha \cdot \max_{p \in \mathcal{P}} \{\sum_{C \in \mathcal{C}} z_C \tilde{\sigma}_C(p)\}$  for some  $\alpha \leq 1$ . Note that, by linearity of expectation

$$\sum_{C \in \mathcal{C}} z_C \tilde{\sigma}_C(p) = \mathbb{E}_{S \sim p} \left[ \sum_{C \in \mathcal{C}} z_C \cdot \frac{1}{|C|} \sum_{v \in C} \tilde{\sigma}_v(S) \right] = \mathbb{E}_{S \sim p} \left[ \sum_{v \in V} \omega_v \cdot \tilde{\sigma}_v(S) \right],$$

where  $\omega_v := \sum_{C \in \mathcal{C}: v \in C} z_C / |C|$ . Hence the oracle problem that the Young's algorithm has to solve takes the form

$$\text{opt}_{\mathcal{O}}(G, \mathcal{C}, k, \omega) := \max_{p \in \mathcal{P}} \tilde{\sigma}^\omega(p). \quad (3)$$

We obtain the following lemma that shows that there is always an optimal solution of linear support.

**Lemma 4.7.** *It holds that  $\text{opt}_{\mathcal{O}}(G, \mathcal{C}, k, \omega) = \text{opt}_{\mathcal{Q}}(G, \mathcal{C}, k, \omega)$  with*

$$\text{opt}_{\mathcal{Q}}(G, \mathcal{C}, k, \omega) := \max_{q \in \mathcal{Q}} \sum_{i=1}^n q_i \tilde{\sigma}^\omega(S_i^*), \quad (4)$$

where  $\mathcal{Q} := \{q \in [0, 1]^n : \mathbf{1}^T q = 1, \sum_{i=1}^n i \cdot q_i \leq k\}$  and  $S_i^* \in \text{argmax}\{\sigma^\omega(S) : S \in \binom{V}{i}\}$  for  $i \in [n]$ .

*Proof.* Let  $q^*$  be an optimal solution to (4). Define  $p \in \mathcal{P}$  as  $p_S = q_i^*$  if  $S = S_i^*$  and 0 otherwise. Then,  $p$  is a feasible solution to (3) and thus  $\text{opt}_{\mathcal{O}}(G, \mathcal{C}, k, \omega) \geq \text{opt}_{\mathcal{Q}}(G, \mathcal{C}, k, \omega)$ . It remains to prove that  $\text{opt}_{\mathcal{O}}(G, \mathcal{C}, k, \omega) \leq \text{opt}_{\mathcal{Q}}(G, \mathcal{C}, k, \omega)$ . For this sake let  $p^* \in \mathcal{P}$  be an optimal solution to (3). Assume  $p_S^* > 0$  for some set  $S$  of cardinality  $i$ , but  $S \neq S_i^*$ . Consider the solution  $p' := p - p_S \mathbf{1}_S + p_S \mathbf{1}_{S_i^*}$ . Clearly,  $\tilde{\sigma}^\omega(p) \leq \tilde{\sigma}^\omega(p')$  as  $S_i^*$  by definition is a maximizing set of size  $i$ . This modification can be repeated until we obtain a solution  $\bar{p}$  with  $\bar{p}_S = 0$  for all sets  $S$  but the sets  $S_1^*, \dots, S_n^*$ . Clearly  $\bar{p}$  is an optimal solution to (3) as each modification did not decrease the value of  $\tilde{\sigma}^\omega(\cdot)$ . Now consider the vector  $q$  defined by  $q_i = \bar{p}_{S_i^*}$  for  $i \in [n]$ . Then,  $q$  is a feasible solution to (4) and thus  $\text{opt}_{\mathcal{O}}(G, \mathcal{C}, k, \omega) \leq \text{opt}_{\mathcal{Q}}(G, \mathcal{C}, k, \omega)$ .  $\square$

In other words, among the vectors that attain the optimum  $\text{opt}_{\mathcal{O}}(G, \mathcal{C}, k, \omega)$ , there is also one that assigns a positive value to at most  $n$  sets. Namely, to one set  $S_i^* \in \binom{V}{i}$  for each  $i \in [n]$ . We now observe that  $\tilde{\sigma}^\omega(\cdot)$  is a submodular and monotone set function. Hence, for each  $i \in [n]$ , the greedy hill climbing algorithm computes  $(1 - 1/e, 0)$  approximations to  $\max\{\tilde{\sigma}^\omega(S) : S \subseteq V, |S| \leq i\}$ . Let  $S_1, \dots, S_n$  denote these approximate solutions. Note that we can assume  $S_1 \subseteq S_2 \subseteq \dots \subseteq S_n$  as we can compute all the sets via one run of the greedy algorithm with budget  $n$ . Now consider the optimization problem

$$\text{opt}_{\mathcal{Q}}^{\text{gr}}(G, \mathcal{C}, k, \omega) := \max_{q \in \mathcal{Q}} \sum_{i=1}^n q_i \tilde{\sigma}^\omega(S_i) \quad (5)$$

that is identical to (4) up to the replacement of  $S_i^*$  by  $S_i$ . We obtain the following lemma.

**Lemma 4.8.** *The vector  $\mathbf{1}_k$  is an optimal solution to the problem in (5). Consequently,  $\tilde{\sigma}^\omega(S_k) = \text{opt}_{\mathcal{Q}}^{\text{gr}}(G, \mathcal{C}, k, \omega) \geq (1 - 1/e) \cdot \text{opt}_{\mathcal{O}}(G, \mathcal{C}, k, \omega)$ .*

*Proof.* Let  $q \in \mathcal{Q}$  be arbitrary. For  $i \in [n]$ , define  $\alpha_i := \sum_{j=i}^n q_j$  and  $\Delta_i := \sigma^\omega(S_i) - \sigma^\omega(S_{i-1})$  with  $S_0 = \emptyset$ . Recall that  $\mathcal{Q} := \{q \in [0, 1]^n : \mathbf{1}^T q = 1, \sum_{i=1}^n i \cdot q_i \leq k\}$  and notice that the last constraint implies  $\sum_{i=1}^n \alpha_i = \sum_{i=1}^n i \cdot q_i \leq k$ . Now, consider the optimization problem  $\max_{\beta \in [0, 1]^n} \{\sum_{i=1}^n \beta_i \Delta_i : \sum_{i=1}^n \beta_i \leq k\}$ . Note that, by submodularity,  $\Delta_1 \geq \Delta_2 \geq \dots \geq \Delta_n$  and thus the optimum is  $\sum_{i=1}^k \Delta_i$ . This implies that

$$\sum_{i=1}^n q_i \tilde{\sigma}^\omega(S_i) = \sum_{i=1}^n \alpha_i \Delta_i \leq \tilde{\sigma}^\omega(S_k). \quad \square$$

We showed that the set  $S_k$  obtained by greedily maximizing  $\tilde{\sigma}^\omega(\cdot)$  subject to a budget of  $k$  yields a  $(1 - 1/e, 0)$ -approximation to the oracle problem. Hence we get the following theorem.

**Theorem 4.9.** *Let  $\delta \in (0, \frac{1}{2})$  and  $\varepsilon \in (0, 1)$ . There is a polynomial time algorithm that, with probability at least  $1 - \delta$ , computes  $p \in \mathcal{P}$  s.t.  $\min_{C \in \mathcal{C}} \sigma_C(p) \geq (1 - \frac{1}{e}) \text{opt}_{\mathcal{P}}(G, \mathcal{C}, k) - \varepsilon$ . The algorithm calls an oracle for weighted IM in each of its  $O(\varepsilon^{-2} n \log m/k)$  iterations and the support of the output  $p$  is  $O(\varepsilon^{-2} n \log m/k)$ .*

*Proof.* We have argued that, for  $z \in \mathbb{R}^m$ , the greedy hill climbing algorithm can be used on  $\tilde{\sigma}^\omega(\cdot)$ , where  $\omega$  is such that  $\omega_v := \sum_{C \in \mathcal{C}: v \in C} \frac{z_C}{|C|}$  for  $v \in V$ , for obtaining a set  $S$  of cardinality  $k$  that is a  $(1 - 1/e, 0)$ -approximate solution to the problem of maximizing  $\sum_{C \in \mathcal{C}} z_C \tilde{\sigma}_C(p)$  over  $\mathcal{P}$ . Thus, we described an oracle with multiplicative approximation  $1 - 1/e$ . Applying Theorem 4.6 with  $\eta = \varepsilon/2$  thus implies that Young's algorithm returns a solution  $p \in \mathcal{P}$  with  $\min_{C \in \mathcal{C}} \tilde{\sigma}_C(p) \geq \alpha \cdot (1 - \frac{\varepsilon}{2}) \max_{p \in \mathcal{S}} \min_{C \in \mathcal{C}} \tilde{\sigma}_C(p) \geq \alpha \cdot \max_{p \in \mathcal{P}} \min_{C \in \mathcal{C}} \tilde{\sigma}_C(p) - \frac{\varepsilon}{2}$  after  $O(\varepsilon^{-2} \log m/\lambda^*)$  iterations. Observe that for any  $p \in \mathcal{P}$  and  $v \in V$ , it holds that  $\sigma_v(p) \geq \Pr_{S \sim p}[v \in S]$ . Hence,  $\sigma_v(p) \geq k/n$  for all  $v \in V$  and  $\lambda^* \geq k/n$ . Thus the number of iterations is bounded by  $O(\varepsilon^{-2} n \log m/k)$ . As the oracle returns a single set in every iteration it follows that the support of  $p$  is upper bounded in this way as well. Applying Lemma 4.2 with  $\varepsilon/2$  leads that we get a  $(1 - 1/e, \varepsilon)$ -approximation to  $\text{opt}_{\mathcal{P}}(G, \mathcal{C}, k)$  in polynomial time with probability at least  $1 - \delta$ .  $\square$

We now state the multiplicative-weight routine for the set-based problem, see Algorithm 1. We assume a routine  $\text{GREEDY}(\tilde{\sigma}^\omega(\cdot), k)$  that performs  $k$  iterations of the greedy hill climbing algorithm on  $\tilde{\sigma}^\omega(\cdot)$  and returns a  $(1 - 1/e, 0)$ -approximation to the oracle problem.

---

**Algorithm 1:** MULTIPLICATIVEWEIGHT( $G, \mathcal{C}, k, \varepsilon, \delta, \eta$ )

---

- 1 Sample  $T \geq 4\varepsilon^{-2} \cdot [n + \log n + \log \delta^{-1}]$  live-edge graphs;
  - 2  $p \leftarrow 0, i \leftarrow 1, \Pi \leftarrow -\infty, \Delta \leftarrow \infty, z_C \leftarrow 1$  and  $s_C \leftarrow 0$  for  $C \in \mathcal{C}$ ;
  - 3 **repeat**
  - 4      $\omega_v \leftarrow \sum_{C \in \mathcal{C}: v \in C} \frac{z_C}{|C|}$  for all  $v \in V$ ;
  - 5      $S \leftarrow \text{GREEDY}(\tilde{\sigma}^\omega(\cdot), k)$ ;
  - 6      $p \leftarrow p + \mathbf{1}_S$ ;
  - 7      $s_C \leftarrow \frac{i-1}{i} \cdot s_C + \frac{1}{i} \cdot \tilde{\sigma}_C(S)$  and  $z_C \leftarrow z_C \cdot (1 - \eta \cdot \tilde{\sigma}_C(S))$ , for all  $C \in \mathcal{C}$ ;
  - 8      $\Pi \leftarrow \min_C \{s_C\}$  and  $\Delta \leftarrow \min \{\Delta, \frac{\tilde{\sigma}^\omega(S)}{\mathbf{1}^T z}\}$ ;
  - 9      $i \leftarrow i + 1$ ;
  - 10 **until**  $\Pi \geq (1 - \eta) \cdot \Delta$ ;
  - 11 **return**  $p/i$
-

## 5. Experiments

We report on an experimental study on the two probabilistic maximin problems. In fact, we provide implementations of multiplicative-weight routines for both the set-based and the node-based problems<sup>2</sup> The routine for the set-based problem is the one described in Section 4.2. We refer to this algorithm as **set-based** in what follows. For the node-based problem, an implementation of the LP-based algorithm from Section 4.1 does not seem promising as it requires solving a large LP. Instead, we propose a heuristic approach that is again based on a multiplicative-weight routine. The essential observation is that the optimization problem  $\max_{x \in \mathcal{X}} \min_{C \in \mathcal{C}} \lambda_C(x)$  from Lemma 4.4 is again a covering LP and thus can be solved using a similar multiplicative-weight routine. In this case however, the oracle problem turns out to be the LP-relaxation of the standard influence maximization problem and thus we are again faced with a linear program of a similar form. This is where our approach becomes heuristic, we propose to again use a greedy algorithm for influence maximization in order to obtain feasible solutions for this LP. While this comes without any guarantee on approximation ratio, it turns out to be very efficient in practice. We refer to this algorithm as **node-based** in the remainder of this section. In our study we use random, artificial, as well as real-world instances. On these instances, we compare our methods, both in terms of fairness and efficiency (i.e., total spread) with a standard implementation of the greedy algorithm for influence maximization and the (very straightforward) methods proposed by Fish et al. (2019) as well as the more involved method due to Tsang et al. (2019). We continue by describing the experimental setting in detail.

### 5.1 Experimental Setting

**Competitors.** The methods of Fish et al. (2019) are simple heuristics. First, they propose to use the greedy algorithm that iteratively picks  $k$  seeds such as to maximize the minimum probability of any node to be reached (note that this is not the same as the greedy algorithm for influence maximization). In our implementation of this algorithm, in order to break ties, we use the node of maximum degree. We refer to this algorithm as **greedy\_maximin**. Second, Fish et al. propose a routine called **myopic** that after choosing the node of maximum degree, iteratively chooses the node that has the minimum probability of being reached as a seed node for  $k - 1$  times. We call this algorithm **myopic\_fish**. As a third heuristic, called **naive-myopic**, they propose to choose the  $k - 1$  nodes of smallest probability all at once instead of in  $k - 1$  iterations. We omit the results of **naive-myopic** as they are much worse than the ones of **myopic\_fish**.

The algorithm of Tsang et al. (2019) is much more involved. They phrase the problem as a multi-objective submodular optimization problem and design an algorithm to tackle such multi-objective submodular optimization problems that provides an asymptotic approximation guarantee of  $1 - 1/e$ . Their algorithm, that improves over previous work by Chekuri et al. (2010) and Udawani (2018), is a Frank-Wolfe style algorithm that simultaneously optimizes the multilinear extensions of the submodular functions that describe the coverage of the respective communities. We stress that their setting is less general than ours as the algorithm only satisfies an approximation guarantee in the case where the number of communities is  $o(k \log^3(k))$ . We use their python implementation as provided while choosing gurobi as solver since the other alternative **md** (their implementation of a mirror-descent) is much less efficient on the instances tested. We refer to the algorithm by Tsang et al as **tsang**.

We also compare our algorithm to the standard greedy algorithm for influence maximization. We use the slightly more involved and very efficient **TIM** implementation (Tang et al., 2014). While

---

2. The code can be downloaded from [https://github.com/sajjad-ghobadi/fair\\_maximin\\_im](https://github.com/sajjad-ghobadi/fair_maximin_im)

there exist even more efficient alternatives to TIM in terms of run-time, the efficiency of TIM is completely sufficient for our purposes. We refer to this method as `tim` in our evaluation.

We also compare our algorithms to the ultimate baseline for randomized strategies that is given by the uniform node-based solution, i.e., every node  $v$  is chosen as a seed with probability  $x_v = k/n$ . We refer to this method as `uf_node_based`.

**Implementation Details.** We implement the multiplicative-weight routines for both the set-based and the node-based problems in C++ and the routines from Fish et al. in python using networkx (Hagberg et al., 2008) for graph related computations. We used the TIM algorithm for influence maximization in order to solve the oracle problems for both multiplicative weight routines. We choose the  $\eta$  parameter of the multiplicative weight routine (see Theorem 4.6) to be 0.1. Our implementations are not optimized for run-time at this point. For example, we do not use any parallelization. While it is easily perceivable that this could lead to big run-time improvements (both for the greedy algorithm for influence maximization and for the multiplicative-weight routines (Young, 2001)), this is out of the scope of the work reported on in this paper.

We use the IC model as diffusion model in all our experiments. For the methods due to Fish et al. (2019) and Tsang et al. (2019) (as also proposed by them), we use a constant number of 100 live-edge graphs for simulating the information spread instead of the number that guarantees  $(1 - \varepsilon)$ -approximations with probability  $1 - \delta$ . As suggested by the confidence intervals in all our plots, this leads to sufficiently small variance on the instances tested.

All experiments were executed on a compute server running Ubuntu 16.04.5 LTS with 24 Intel(R) Xeon(R) CPU E5-2643 3.40GHz cores and a total of 128 GB RAM. The code was executed with python version 3.7.6 and C++ implementations were compiled with g++ 7.5.0. For the random generation of the graphs and the random choices of the live-edge graphs, we do not explicitly set the random seeds used by the random number generator. This does not prevent reproducibility of our results as all the reported results are averages that are robust and independent of the random seeds chosen as indicated by the confidence intervals reported.

**Evaluation Details.** For random and synthetic instances, each datapoint in our plots is the result of averaging over 25 experiments, 5 runs on each of 5 graphs generated according to the respective graph model. For real world instances each datapoint is the result of averaging over 5 runs on each graph. Error-bars in our plots indicate 95-% confidence intervals. For the evaluation of  $\sigma_v(x)$  we choose to approximate the value using a Chernoff bound in a way to obtain an additive  $\varepsilon$ -approximation of the values with probability  $1 - \delta$  and in the reported experiments we choose both parameters as 0.1.

We report both ex-ante and ex-post fairness values for our methods (for short, we use `ea` and `ep` as suffices). These have the following precise meaning. After computing probabilistic strategies  $p$  or  $x$  for the set-based and node-based problems, the ex-ante fairness values correspond to the objective values  $\min_{C \in \mathcal{C}} \mathbb{E}_{S \sim p}[\sigma_C(S)]$  for the set-based and  $\min_{C \in \mathcal{C}} \mathbb{E}_{S \sim x}[\sigma_C(S)]$  for the node-based problem. The ex-post values on the other hand are obtained by sampling a single set  $S$  according to the probabilistic strategy  $p$  or  $x$  and then reporting the value  $\min_{C \in \mathcal{C}} \sigma_C(S)$ . We report also both ex-ante and ex-post values for the method of Tsang et al. (2019), since, at the core, their algorithm works with the multilinear extension and thus also computes a continuous solution  $x \in \mathbb{R}^n$ , i.e., a feasible solution to the node-based problem. Hence for their method we report both the value  $\min_{C \in \mathcal{C}} \mathbb{E}_{S \sim x}[\sigma_C(S)]$  as ex-ante value and a value  $\min_{C \in \mathcal{C}} \sigma_C(S)$  as ex-post value, where  $S$  is computed by swap rounding from  $x$  as described in their paper.

**Instances.** We evaluate the different algorithms on a vast set of instances. We proceed by describing the networks and the community structures that we use in our study.

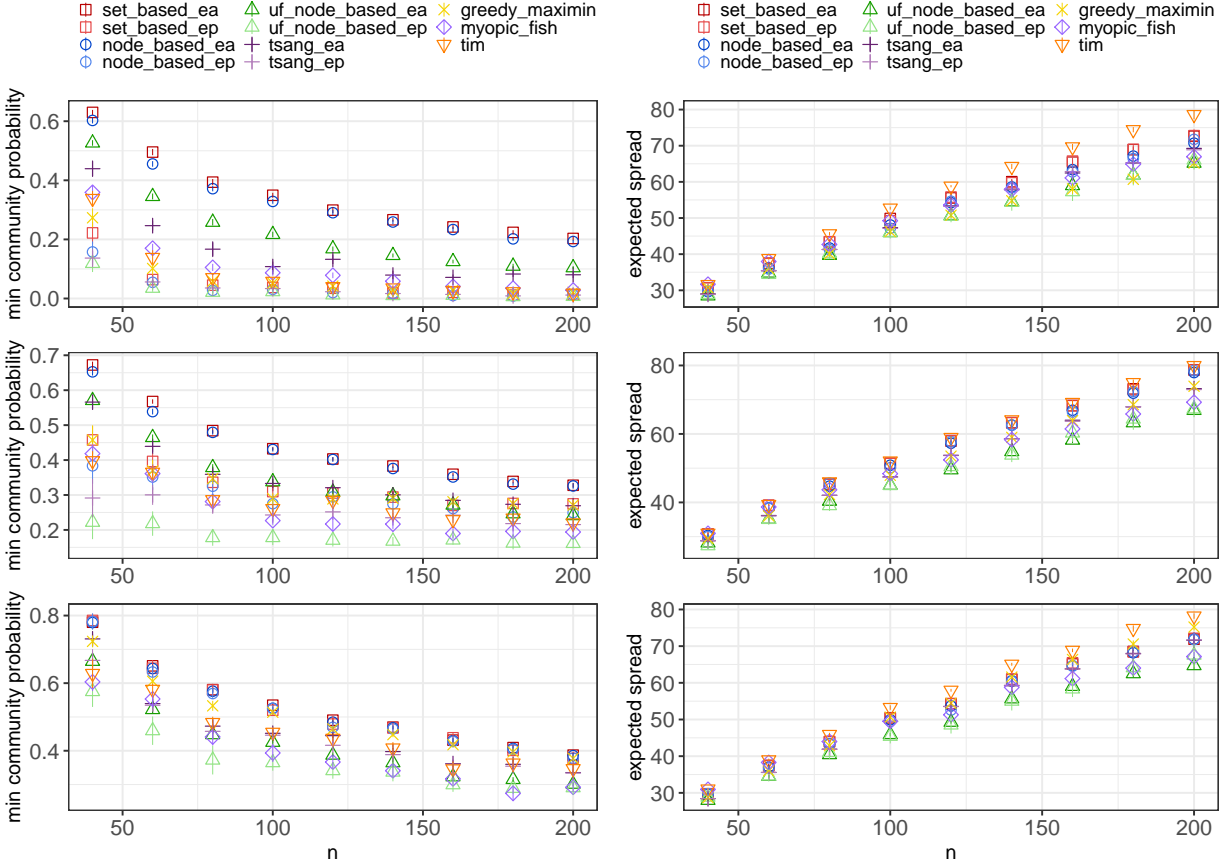


Figure 4: Results for Barabási-Albert instances:  $k = 20$ ,  $n$  increasing from 40 to 200 in steps of 20. The minimum community probability is shown on the left, while expected spread is shown on the right. From top to bottom, we see (1) singleton community structure, (2) BFS community structure, and (3) random imbalanced community structure.

**Networks.** We use the following networks: (1) Random instances generated according to the Barabási-Albert model (Albert & Barabási, 2002) that yields scale-free networks – a property frequently observed in social networks. We use the parameter modeling the preferential attachment to be 2, i.e., every newly introduced node is connected to two previously existing nodes. (2) Random instances generated according to the block-stochastic model that is a natural choice in our setting as the instances come along with a community structure. (3) The publicly available synthetic instances from the work of Tsang et al. (2019). (4) Real world instances from the SNAP database (Leskovec & Krevl, 2014) and a paper by Guimerà et al. (2003), some of which were used also by Fish et al. We describe the real-world instances in detail below in the corresponding paragraph. The number of nodes and edges as well as the information whether the networks are directed or undirected are summarized in Table 2. For most of our experiments and unless stated differently, we choose edge weights uniformly at random in the interval  $[0, 0.4]$  for random instances and the synthetic instances of Tsang et al., and in the interval  $[0, 0.2]$  for the real world instances.

**Community Structure.** Regarding the community structure, in the case of some networks such as the block-stochastic networks, the synthetic networks due to Tsang et al., and some real world

Dataset	Nodes	Edges	Direction
Barabási-Albert	40 – 200	152 – 792	Directed
Block-stochastic ( $q = 0.1$ , $p$ increasing)	200	1920 – 3456	Directed
Block-stochastic ( $p = 0.1$ , $q$ increasing)	200	139 – 2059	Directed
Synthetic networks	500	1576 – 1697	Directed
Arenas	1133	5451	Directed
email-Eu-core	1005	25571	Directed
ca-GrQc	5242	14496	Undirected
ca-HepTh	9877	25998	Undirected
Facebook	4039	88234	Undirected
Irvine	1899	20296	Directed
com-Youtube	3000	29077	Undirected

Table 2: Properties of random, synthetic and real world networks

graphs the community structures are given. On all other networks we use some of the following different ways of constructing the community structure: (1) Singleton communities: every node is his own community. (2) BFS community structure: for a predefined number of communities  $m$ , we iteratively grow communities of size  $n/m$  by breadth first search from a random source node (once there are no more reachable nodes but the community is still not of size  $n/m$ , we pick a new random source, until every node is in one of the  $m$  communities). (3) Random imbalanced community structure: we randomly assign nodes to one of  $m$  communities of fixed sizes. We use different values for the sizes and specify them for each of the experiments.

We note that the BFS community structure results in a rather connected community structure which is realistic for some applications. On the other hand, the random imbalanced community structure, is rather unconnected. Also this setting is realistic for some applications as for example if the groups indicate gender or ethnicity and we assume that people connect independently of these attributes.

## 5.2 Results

**Barabási-Albert Graphs.** For the Barabási-Albert graphs, we explore singleton communities, the BFS community structure with  $m = k$ ,<sup>3</sup> and random imbalanced community structures of sizes  $4n/10, 3n/10, 2n/10, n/10$ . The results are reported in Figure 4. In the left plots in Figure 4, we can see that the ex-ante values of our methods `set_based` and `node_based` dominate over all other ex-ante and ex-post values. Furthermore, we can see that particularly in the last plot, where the community structure is less simplistic, even the ex-post values of our methods dominate over the ones of all competitors. In the right plots, where we show the expected spread, we can see that `tim` outperforms other methods for all values of  $n$ . We note however that the advantage in efficiency of `tim` is not too pronounced, particularly in comparison to the disadvantage it yields in terms of fairness, see for example the plot on the top left.

**Block-Stochastic Graphs.** In order to further explore how the connectivity of the community structure influences the performance of the different approaches, we generate Block Stochastic graphs as follows. We fix the number of nodes to 200, the number of communities to 16 with 6 communities

3. We note that in this case the algorithm of Tsang et al. satisfies its approximation guarantee.

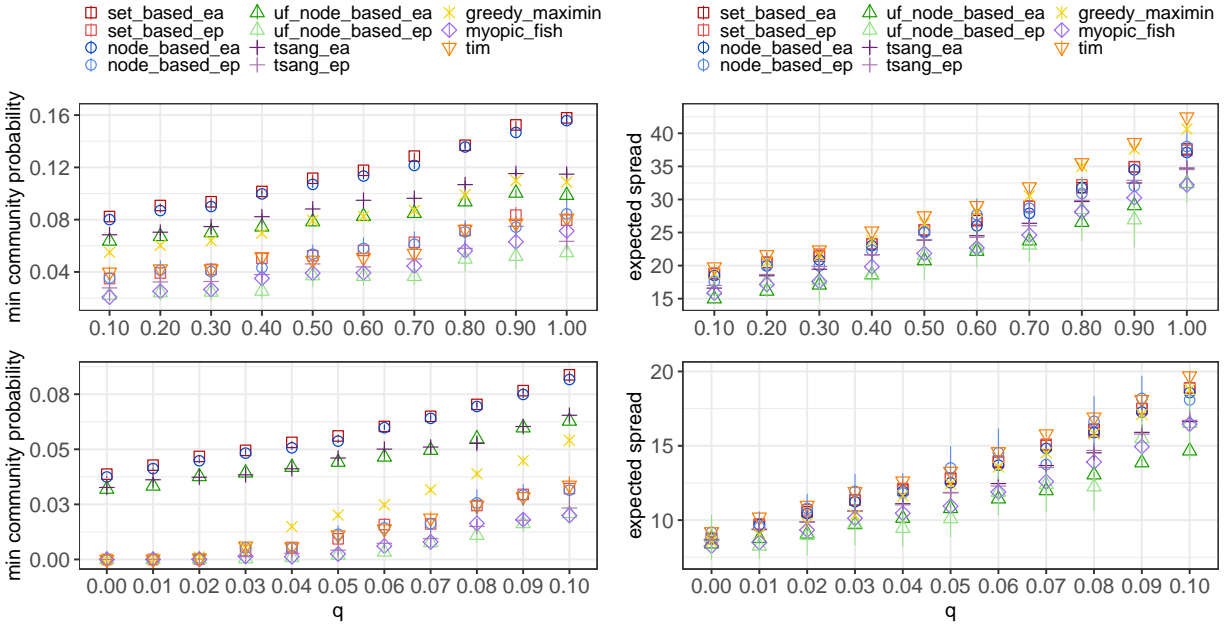


Figure 5: Results for block stochastic graphs: edge weights constant 0.05,  $k = 8$ ,  $n = 200$ . The minimum community probability is shown on the left, while expected spread is shown on the right. From top to bottom, we see (1)  $q = 0.1$  and  $p$  increasing from  $q$  to 1 in steps of 0.1, (2)  $p = 0.1$  and  $q$  increasing from 0 to  $p$  in steps of 0.01.

of size  $n/40$ , 4 communities of size  $2n/40$ , 4 communities of size  $4n/40$  and 2 communities of size  $5n/40$ . We then choose two parameters  $p$  and  $q$ , and create a sequence of instances where the probability of an edge within a community is  $p$  and between communities  $q$ . The larger choices of  $p$  and  $q$  yield very dense graphs and thus instances become trivial. We choose edge weights to be 0.05 in this experiment as for the larger choice the instances become trivial as the minimum community probability becomes very large. The results are reported in Figure 5. In the left plots we can see that again the ex-ante values of `set_based` and `node_based` dominate over all other values. Clearly, by increasing  $p$  and  $q$  in both experiments, the values of all algorithms are increased. In the second experiment, for smaller  $q$ , when communities are better connected within each other than between each other, there is a bigger advantage for ex-ante values over ex-post values. In the right column, for smaller  $p$  and  $q$ , all algorithms are close to each other. Again `tim` dominates over the other algorithms in terms of expected spread, while also the other algorithms – and in particular `set_based` and `node_based` – perform well.

**Instances of Tsang et al.** Next we evaluate the algorithms on the instances used by Tsang et al. (2019). These are synthetic networks introduced by Wilder et al. (2018b) in order to analyze the effects of health interventions. Each of the 500 nodes in these networks has some attributes (region, ethnicity, age, gender, status) and more similar nodes are more likely to share an edge. The attributes induce communities and we test, as proposed by Tsang et al. (2019), all algorithms w.r.t. group fairness of the communities induced by some of those attributes. The results are reported in Figure 6. Again the ex-ante fairness values of our methods dominate over all other algorithms as can be seen in the left column. In the first experiment (communities induced by gender), the ex-post values of `set_based`, `node_based`, `tsang`, and `uf_node_based` are all almost identical to



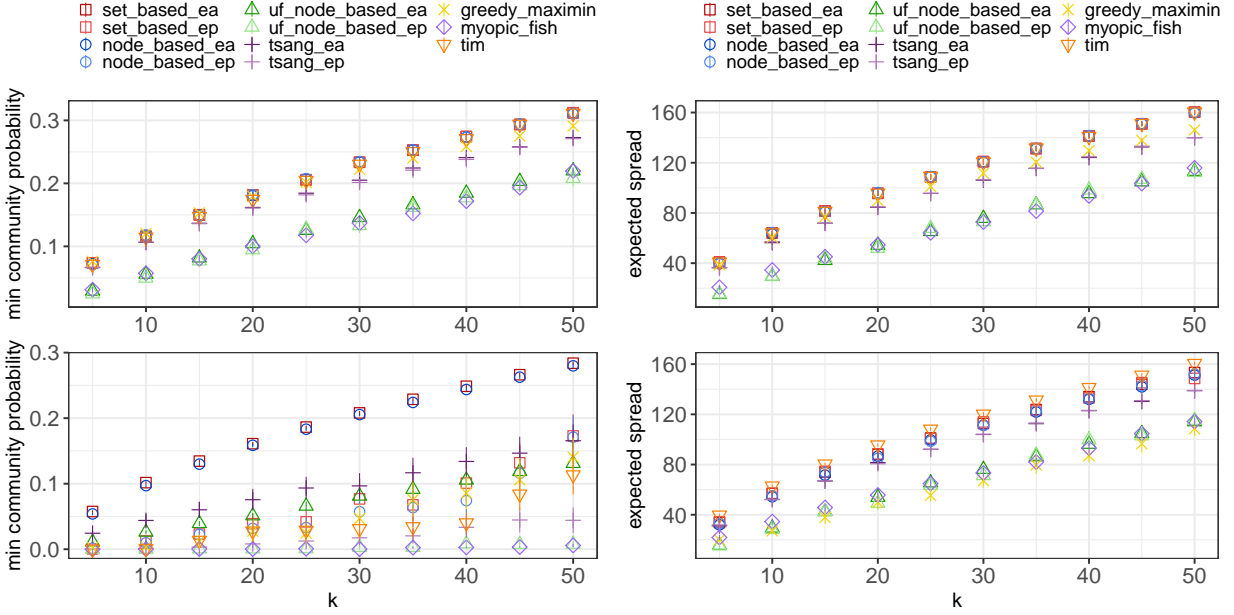


Figure 6: Results for the instances of Tsang et al. (2019):  $k$  increasing from 5 to 50 in steps of 5. The minimum community probability is shown on the left, while expected spread is shown on the right. From top to bottom, we see (1) community structure induced by attribute gender, (2) community structure induced by attributes region, gender and ethnicity.

their respective ex-ante values. In the second experiment (communities induced by three attributes, namely region, gender, and ethnicity) we obtain a much more complex community structure. Here, our algorithms `set_based` and `node_based` perform best not only in the ex-ante values, but also in terms of ex-post values for most values of  $k$ . Even more, in the right plots, we can see that the achieved values in expected spread by `tim` and our methods are very close to each other.

**Real World Instances.** We proceed by describing the used real world instances.

**Arenas (Guimerà et al., 2003)** This dataset represents an email communication network at the University Rovira i Virgili (Spain). Each user is represented by a node and there is a directed edge from a node  $u$  to a node  $v$  if  $u$  sent at least one email to  $v$ .

**email-Eu-core (Leskovec et al., 2007)** Also this dataset is an email network, this time from a large European research institution. Each member of the research institution belongs to one of 42 departments, which predefines a community structure. Nodes and edges have the same interpretation as in **Arenas**.

**ca-GrQc and ca-HepTh (Leskovec et al., 2007)** These datasets are co-authorship networks for two different categories of arXiv (General Relativity and Quantum Cosmology and High Energy Physics - Theory). The nodes in the networks correspond to authors and there is an undirected edge between two nodes if the authors co-authored at least one arXiv paper in this category.

**Facebook (Leskovec & McAuley, 2012)** This dataset represents a part of the Facebook network, where nodes are users and edges indicate friendships.

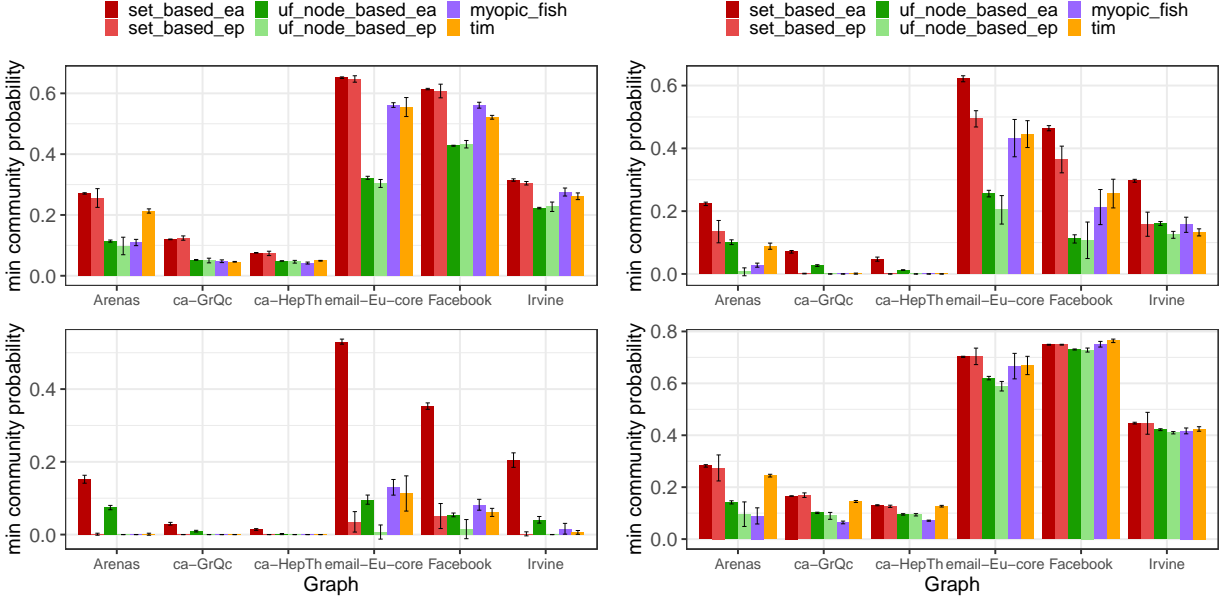


Figure 7: Results for the instances used by Fish et al. for  $k = 100$ . BFS community structure with (top left) 10 communities, (top right)  $n/10$  communities, (bottom left)  $n/2$  communities. (bottom right) Random imbalanced community structure with 16 communities.

**Irvine (Opsahl & Panzarasa, 2009)** This dataset is a network created from an online community at the University of California, Irvine. Nodes here represent students and each directed edge represents that at least one online message was sent among the students.

**com-Youtube (Yang & Leskovec, 2015)** This dataset consists of a snapshot of the social network included in Youtube. The network has 1134890 nodes and 2987624 edges. Nodes correspond to users, edges to friendships between users. Also this network contains a predefined community structure that is given by the so-called Youtube groups. On Youtube, users can open groups that others can join. As the complete network is very large, we use a connected sub-network. We first remove all nodes that do not belong to any community. We then obtain the sub-network as the induced graph among the first 3000 nodes that are seen by a BFS from a random source node while removing singleton communities. The resulting network contains 3000 nodes (some may not belong to any community), 29077 edges. The number of communities is 1575.

If networks are undirected, we interpret the edges as existent in both directions. In order to obtain non-trivial results, i.e., achieve non-zero minimum probabilities in the experiments (especially for the ex-post values), for each network (other than **com-Youtube**) we considered the largest weakly connected component. We exclude **greedy** and the method of Tsang et al. from the further experiments as they are not efficient enough to deal with instances of this size. We also restrict to the set-based method from our two methods as the results of our two methods are very similar. The results are reported in the following figures.

**Evaluation on Networks used by Fish et al.** We start with the networks that were also used in the study of Fish et al. (2019). We experiment with different community structures, both the BFS community structures with different community sizes and the random imbalanced community

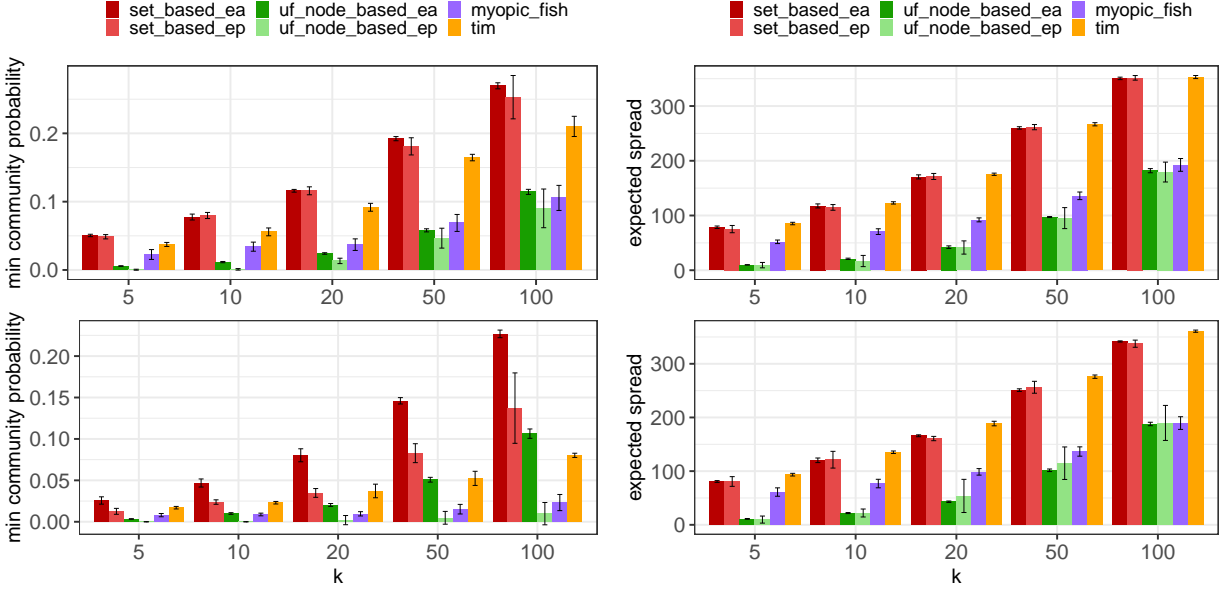


Figure 8: Results for the `Arenas` network for increasing  $k = 5, 10, 20, 50, 100$ . The minimum community probability is shown on the left, while expected spread is shown on the right. BFS community structure with (1) 10 communities, (2)  $n/10$  communities.

structure. The results can be found in Figure 7. We omit the results for the BFS community structure with only 2 communities as they are very similar to the case of 10 communities. We observe that in all cases the ex-ante value of our algorithm is dominating over all other values. In some cases, the values achieved by TIM are comparable but these are instances where all algorithms perform very close to each other and the minimum community probabilities are rather high anyways. Furthermore, on several instances, e.g., in the case of 10 BFS communities, the ex-post values of our algorithm are significantly better than the ex-post values of all other methods.

**Fairness vs. Efficiency on Email-Networks.** We proceed by focusing on the email networks `Arenas` and `email-Eu-core` and comparing the fairness achieved by the different algorithms with the efficiency, i.e., expected spread, see Figures 8 and 9. We again use the BFS community structures with different community sizes. In the case of the `email-Eu-core` dataset, we evaluate the different algorithms on the community structure induced by the departments as well. We evaluate the algorithms for increasing values of  $k = 5, 10, 20, 50, 100$ . We observe that `set_based` performs best in terms of fairness among all ex-ante as well as ex-post values both on the `Arenas` dataset as well as on the `email-Eu-core` dataset. In terms of efficiency, we observe that on the `Arenas` dataset, `tim` and `set_based` perform similarly good and much better than the other competitors. For the `email-Eu-core` dataset, we see that in terms of efficiency all algorithms (even `uf_node_based` for small values of  $k$ ) perform almost identical.

**Fairness vs. Efficiency on Co-Authorship Networks.** We turn to the two co-authorship datasets `ca-GrQc` and `ca-HepTh` and evaluate the fairness and efficiency achieved by the different algorithms. The results are depicted in Figure 10. Focusing on the fairness values first, we observe that in a setting with  $n/10$  communities, no algorithm achieves a positive ex-post value. Instead the two randomized algorithms `set_based` and `uf_node_based` do achieve a significantly non-zero

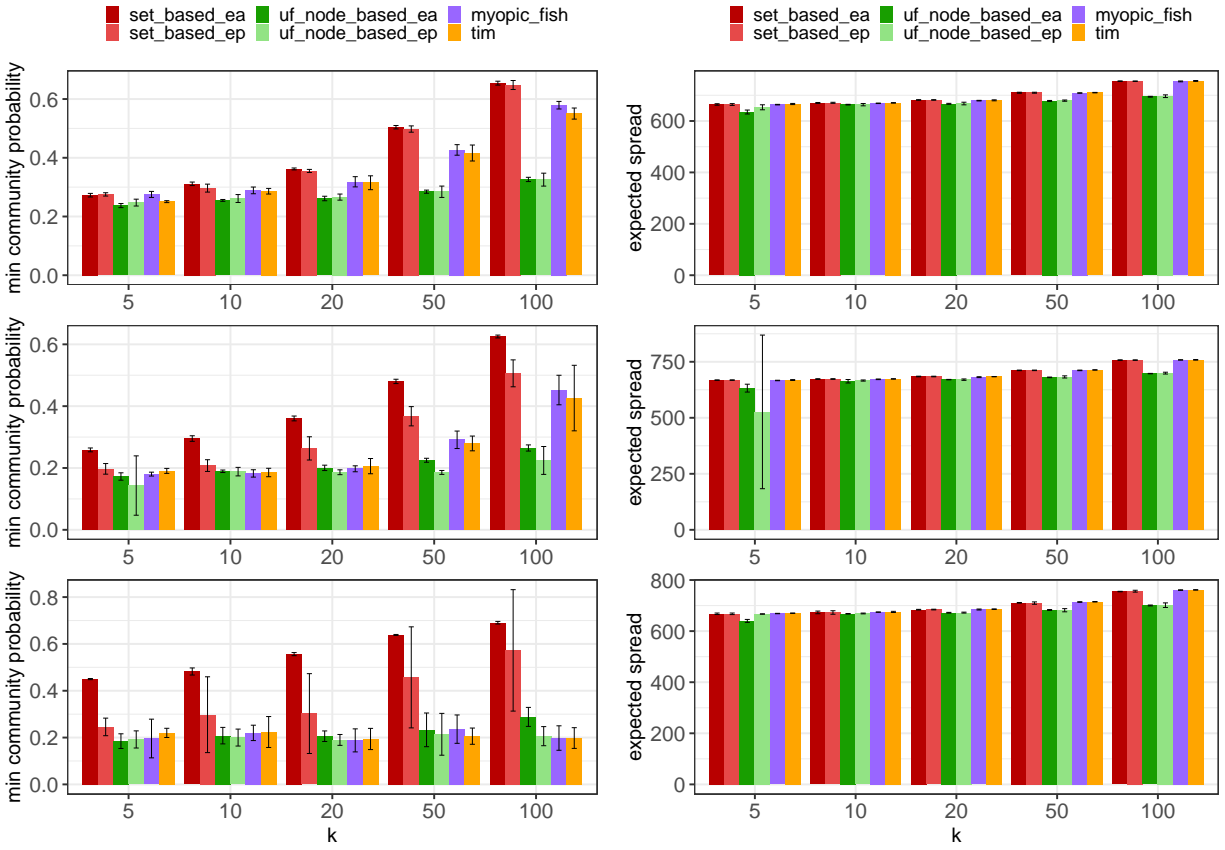


Figure 9: Results for `email-Eu-core` network for increasing  $k = 5, 10, 20, 50, 100$ . The minimum community probability is shown on the left, while expected spread is shown on the right. (1) BFS community structure with 10 communities, (2) BFS community structure with  $n/10$  communities, (3) community structure induced by departments.

ex-ante value, the results of `set_based` being more than twice as high compared to the values of `uf_node_based`. For 10 communities, we again end up in a setting where both the ex-ante and ex-post values of `set_based` dominate over all other algorithms. The discrepancy between the ex-post value achieved by `set_based` and the other algorithms appears to become more and more pronounced with increasing values of  $k$ . In terms of efficiency, we observe that again `tim` and `set_based` perform the best, while there is a bigger advantage for `tim` in this case than with most other instances tested. Note however that `set_based` does achieve significantly better fairness values as compared to `tim` in all settings where it falls behind `tim` in terms of efficiency.

**Fairness vs. Efficiency on com-Youtube Network.** We conclude with the `com-Youtube` network and evaluate the different algorithms in terms of fairness and efficiency, see Figure 11. In this network we choose edge weights uniformly at random in the interval  $[0, 0.1]$ . In the left plot, we observe that the ex-ante fairness values achieved by `set_based` are significantly better than the values of all other algorithms, especially with increasing values of  $k$ . The ex-post values of all algorithms are much smaller and close to each other. In terms of efficiency, we observe that the results of all algorithms are very similar. Note that `set_based` performs almost the same as `tim` in the expected spread while being significantly better than `tim` in terms of fairness.

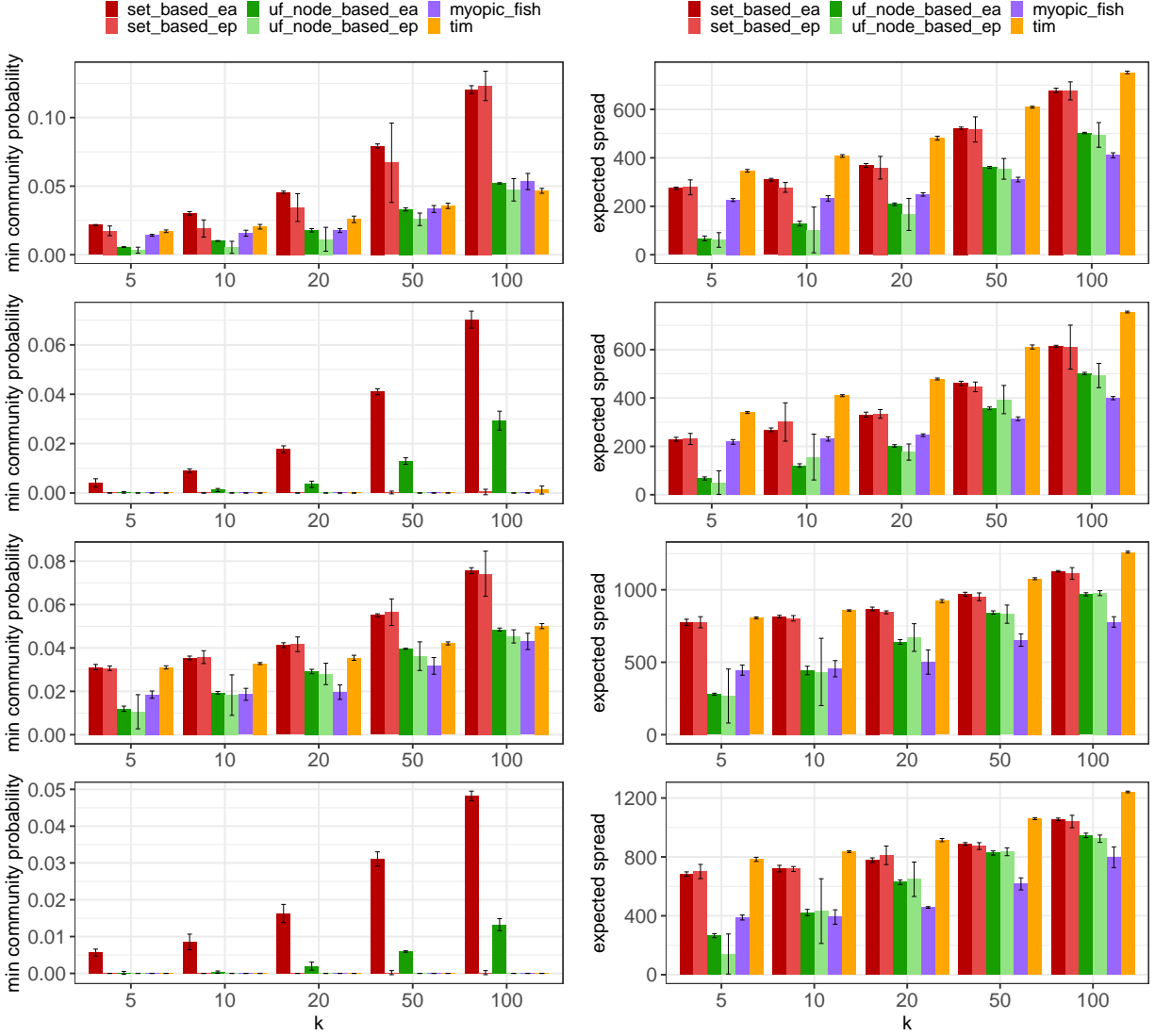


Figure 10: Results for co-authorship networks for increasing  $k = 5, 10, 20, 50, 100$ . The minimum community probability is shown on the left, while expected spread is shown on the right. BFS community structure (1) `ca-GrQc` with 10 communities, (2) `ca-GrQc` with  $n/10$  communities, (3) `ca-HepTh` with 10 communities (4) `ca-HepTh` with  $n/10$  communities.

## 6. Conclusion

In this paper, we studied the problem of determining key seed nodes for influence maximization in social networks in an efficient *and* fair manner. Notably, we have designed approximation algorithms achieving a constant multiplicative factor of  $1 - 1/e$  (plus an additive arbitrarily small error) for the objective of maximizing the maximin influence received by a community. For one of the two variants of our problem, we have shown that this approximation factor is best possible (up to the arbitrarily small error term). We achieved our algorithmic result by using randomized strategies, thus enlarging the solution set and enabling us to find fairer solutions ex-ante. Our detailed experimental study

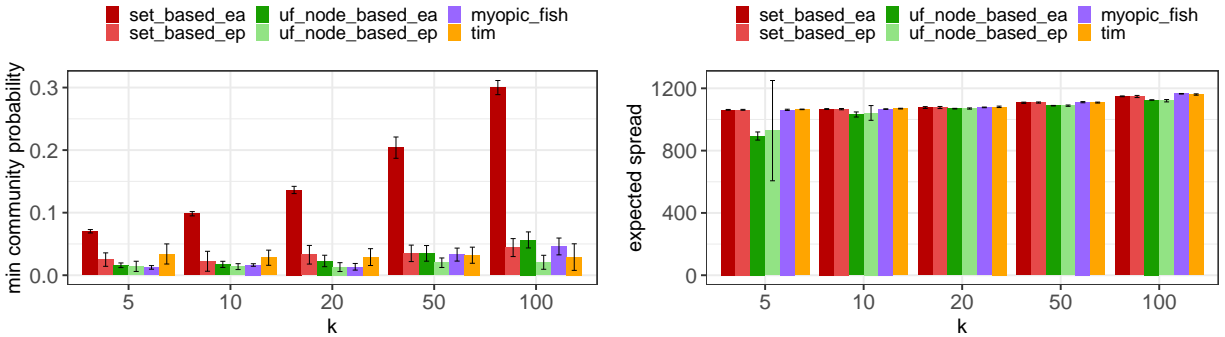


Figure 11: Results for the `com-Youtube` network for increasing  $k = 5, 10, 20, 50, 100$ . The minimum community probability is shown on the left, while expected spread is shown on the right.

confirms the increase in ex-ante fairness achieved over previous methods (Fish et al., 2019; Tsang et al., 2019), indicating that randomness as source of fairness in influence maximization is very promising to be further explored. Interestingly, our study shows that even the ex-post fairness achieved by our methods frequently outperforms the fairness achieved by other tested methods. While our theoretical results predict that the price of fairness in the studied setting can be very bad, our experimental evaluation instead indicates that the loss in efficiency when using our methods is very limited.

Several directions are conceivable as future work. Improving our approximation guarantees for the set-based problem or providing a matching approximation hardness result seems a challenging direction of exploration. Tightening the result on the gap between node-based and set-based problem is certainly an interesting question. Further engineering the implementations of our methods could yield further gains in efficiency, particularly in terms of running time. Lastly, we believe that the idea of using randomization to increase the fairness of solutions for influence maximization may be used for other fairness criteria as, e.g., the group rational criterion of Tsang et al. (2019).

## Acknowledgments

An earlier version of this article has been published in the Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI 2021) with the same title (Becker et al., 2021). This work was partially supported by the Italian MIUR PRIN 2017 Project “ALGADIMAR” Algorithms, Games, and Digital Markets.

## References

- Ageev, A. A., & Sviridenko, M. (2004). Pipage rounding: A new method of constructing algorithms with proven performance guarantee. *J. Comb. Optim.*, 8(3), 307–328.
- Agrawal, S., Ding, Y., Saberi, A., & Ye, Y. (2010). Correlation robust stochastic optimization. In *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2010, Austin, Texas, USA, January 17-19, 2010*, pp. 1087–1096.
- Albert, R., & Barabási, A.-L. (2002). Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74(1), 47–97.
- Ali, J., Babaei, M., Chakraborty, A., Mirzasoleiman, B., Gummadi, K. P., & Singla, A. (2019). On the fairness of time-critical influence maximization in social networks. *arXiv preprint*

*arXiv:1905.06618.*

- Aziz, H. (2019). A probabilistic approach to voting, allocation, matching, and coalition formation. In *The Future of Economic Design*, pp. 45–50. Springer.
- Aziz, H., Brandt, F., & Stursberg, P. (2013). On popular random assignments. In *International Symposium on Algorithmic Game Theory*, pp. 183–194. Springer.
- Aziz, H., Luo, P., & Rizkallah, C. (2018). Rank maximal equal contribution: A probabilistic social choice function. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Banerjee, A., Chandrasekhar, A. G., Duflo, E., & Jackson, M. O. (2013). The diffusion of microfinance. *Science*, *341*(6144), 1236498.
- Becker, R., Corò, F., D’Angelo, G., & Gilbert, H. (2020). Balancing spreads of influence in a social network. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pp. 3–10.
- Becker, R., D’Angelo, G., Ghobadi, S., & Gilbert, H. (2021). Fairness in influence maximization through randomization. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pp. 14684–14692.
- Bogomolnaia, A., & Moulin, H. (2001). A new solution to the random assignment problem. *Journal of Economic theory*, *100*(2), 295–328.
- Borgatti, S. P., & Everett, M. G. (2000). Models of core/periphery structures. *Soc. Networks*, *21*(4), 375–395.
- Borgs, C., Brautbar, M., Chayes, J. T., & Lucier, B. (2014). Maximizing social influence in nearly optimal time. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*, pp. 946–957.
- Borodin, A., Braverman, M., Lucier, B., & Oren, J. (2017). Strategyproof mechanisms for competitive influence in networks. *Algorithmica*, *78*(2), 425–452.
- Brandl, F., Brandt, F., & Seedig, H. G. (2016). Consistent probabilistic social choice. *Econometrica*, *84*(5), 1839–1880.
- Brandt, F. (2019). Collective choice lotteries. In *The Future of Economic Design*, pp. 51–56. Springer.
- Budak, C., Agrawal, D., & El Abbadi, A. (2011). Limiting the spread of misinformation in social networks. In *Proceedings of the 20th International Conference on World Wide Web, WWW 2011, Hyderabad, India, March 28 - April 1, 2011*, pp. 665–674.
- Chekuri, C., Vondrák, J., & Zenklus, R. (2010). Dependent randomized rounding via exchange properties of combinatorial structures. In *51th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2010, October 23-26, 2010, Las Vegas, Nevada, USA*, pp. 575–584.
- Chen, W., & Teng, S.-H. (2017). Interplay between social influence and network centrality: a comparative study on shapley centrality and single-node-influence centrality. In *Proceedings of the 26th international conference on world wide web (WWW)*, pp. 967–976.
- Chen, Y., Wang, W., Feng, J., Lu, Y., & Gong, X. (2020). Maximizing multiple influences and fair seed allocation on multilayer social networks. *Plos one*, *15*(3), e0229201.

- Cohen, E., Delling, D., Pajor, T., & Werneck, R. F. (2014). Sketch-based influence maximization and computation: Scaling up with guarantees. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, CIKM 2014, Shanghai, China, November 3-7, 2014*, pp. 629–638.
- Farnadi, G., Babaki, B., & Gendreau, M. (2020). A unifying framework for fairness-aware influence maximization. In *Companion Proceedings of the Web Conference 2020*, pp. 714–722.
- Feige, U. (1998). A threshold of  $\ln n$  for approximating set cover. *J. ACM*, 45(4), 634–652.
- Fish, B., Bashardoust, A., Boyd, D., Friedler, S., Scheidegger, C., & Venkatasubramanian, S. (2019). Gaps in information access in social networks?. In *The World Wide Web Conference*, pp. 480–490. ACM.
- Guimerà, R., Danon, L., Díaz-Guilera, A., Giralt, F., & Arenas, A. (2003). Self-similar community structure in a network of human interactions. *Physical review E*, 68(6), 065103.
- Hagberg, A. A., Schult, D. A., & Swart, P. J. (2008). Exploring network structure, dynamics, and function using networkx. In Varoquaux, G., Vaught, T., & Millman, J. (Eds.), *Proceedings of the 7th Python in Science Conference*, pp. 11 – 15, Pasadena, CA USA.
- Katta, A.-K., & Sethuraman, J. (2006). A solution to the random assignment problem on the full preference domain. *Journal of Economic theory*, 131(1), 231–250.
- Kempe, D., Kleinberg, J. M., & Tardos, É. (2015). Maximizing the spread of influence through a social network. *Theory of Computing*, 11, 105–147.
- Khajehnejad, M., Rezaei, A. A., Babaei, M., Hoffmann, J., Jalili, M., & Weller, A. (2020). Adversarial graph embeddings for fair influence maximization over social networks. *arXiv preprint arXiv:2005.04074*.
- Leskovec, J., Kleinberg, J., & Faloutsos, C. (2007). Graph evolution: Densification and shrinking diameters. *ACM transactions on Knowledge Discovery from Data (TKDD)*, 1(1), 2–es.
- Leskovec, J., & Krevl, A. (2014). SNAP Datasets: Stanford large network dataset collection. <http://snap.stanford.edu/data>.
- Leskovec, J., & Mcauley, J. J. (2012). Learning to discover social circles in ego networks. In *Advances in neural information processing systems*, pp. 539–547.
- Lu, W., Bonchi, F., Goyal, A., & Lakshmanan, L. V. (2013). The bang for the buck: fair competitive viral marketing from the host perspective. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 928–936.
- Machina, M. J. (1989). Dynamic consistency and non-expected utility models of choice under uncertainty. *Journal of Economic Literature*, 27(4), 1622–1668.
- Manlove, D. (2013). *Algorithmics of matching under preferences*, Vol. 2. World Scientific.
- Mitzenmacher, M., & Upfal, E. (2017). *Probability and Computing: Randomization and Probabilistic Techniques in Algorithms and Data Analysis* (2nd edition). Cambridge University Press, USA.
- Moulin, H. (1991). *Axioms of Cooperative Decision Making*. Econometric Society Monographs. Cambridge University Press.
- Opsahl, T., & Panzarasa, P. (2009). Clustering in weighted networks. *Social networks*, 31(2), 155–163.
- Rahmattalabi, A., Jabbari, S., Lakkaraju, H., Vayanos, P., Rice, E., & Tambe, M. (2020). Fair influence maximization: A welfare optimization approach. *arXiv preprint arXiv:2006.07906*.



- Stoica, A.-A., Han, J. X., & Chaintreau, A. (2020). Seeding network influence in biased networks and the benefits of diversity. In *WWW2020*.
- Tang, Y., Shi, Y., & 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, Melbourne, Victoria, Australia, May 31 - June 4, 2015*, pp. 1539–1554.
- Tang, Y., Xiao, X., & Shi, Y. (2014). Influence maximization: near-optimal time complexity meets practical efficiency. In *International Conference on Management of Data, SIGMOD 2014, Snowbird, UT, USA, June 22-27, 2014*, pp. 75–86.
- Tsang, A., Wilder, B., Rice, E., Tambe, M., & Zick, Y. (2019). Group-fairness in influence maximization. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pp. 5997–6005.
- Udwani, R. (2018). Multi-objective maximization of monotone submodular functions with cardinality constraint. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*, pp. 9513–9524.
- Valente, T. W., & Pumpuang, P. (2007). Identifying opinion leaders to promote behavior change. *Health education & behavior*, 34(6), 881–896.
- Wilder, B., Onasch-Vera, L., Hudson, J., Luna, J., Wilson, N., Petering, R., Woo, D., Tambe, M., & Rice, E. (2018a). End-to-end influence maximization in the field. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pp. 1414–1422. International Foundation for Autonomous Agents and Multiagent Systems.
- Wilder, B., Ou, H., de la Haye, K., & Tambe, M. (2018b). Optimizing network structure for preventative health. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018, Stockholm, Sweden, July 10-15, 2018*, pp. 841–849.
- Yadav, A., Wilder, B., Rice, E., Petering, R., Craddock, J., Yoshioka-Maxwell, A., Hemler, M., Onasch-Vera, L., Tambe, M., & Woo, D. (2018). Bridging the gap between theory and practice in influence maximization: Raising awareness about hiv among homeless youth.. In *IJCAI*, pp. 5399–5403.
- Yan, Q. (2011). Mechanism design via correlation gap. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011, San Francisco, California, USA, January 23-25, 2011*, pp. 710–719.
- Yang, J., & Leskovec, J. (2015). Defining and evaluating network communities based on ground-truth. *Knowledge and Information Systems*, 42(1), 181–213.
- Young, N. E. (1995). Randomized rounding without solving the linear program. In *Proceedings of the Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, 22-24 January 1995. San Francisco, California, USA*, pp. 170–178.
- Young, N. E. (2001). Sequential and parallel algorithms for mixed packing and covering. In *42nd Annual Symposium on Foundations of Computer Science, FOCS 2001, 14-17 October 2001, Las Vegas, Nevada, USA*, pp. 538–546.
- Yu, Y., Jia, J., Li, D., & Zhu, Y. (2017). Fair multi-influence maximization in competitive social networks. In *International Conference on Wireless Algorithms, Systems, and Applications*, pp. 253–265. Springer.