GRAN SASSO
SCIENCE INSTITUTE

SCHOOL OF ADVANCED STUDIES
Scuola Universitaria Superiore

Doctoral thesis

Fairness in Influence Maximization

PhD program in computer science: XXXIV cycle

PhD candidate
Sajjad Ghobadi Babi
Gran Sasso Science Institute

Supervisors
Prof. Gianlorenzo D'Angelo
Gran Sasso Science Institute
Dr. Ruben Becker
Ca' Foscari University of Venice

March 2023

GSSI Gran Sasso Science Institute

Abstract

Online social networks such as Facebook, LinkedIn, and Twitter are an inseparable part of our life. They help us to interact with other people at little cost (and) easily. These networks play an essential role in spreading information, ideas, and knowledge among users. This results in affecting or changing users' opinions about certain topics. When a user of a social network receives a piece of information, she may share it with her friends, and her friends can share it with her friends' friends and so on. In this way, the information may spread to a large number of people. In computer science, these phenomena have been studied under two names information diffusion and social influence. These topics have received a high level of attention by researchers and have many applications in advertisement, news propagation, disease spread, viral marketing, sales promotions and many others. However, social media has been criticized for creating situations that some users or groups of users have a high chance of receiving information or getting most of the attention while others stay disregarded, thus discriminating among users or groups of users. Note that such discrimination or disparity among different groups of a network especially in real world applications related to health, education, and job opportunities, can put minority groups at a big disadvantage. In computational social choice, the notion of *group fairness* was developed in order to address this issue. The study of this notion in the context of information diffusion is the main focus of this thesis. We study several optimization problems with the focus of addressing the groups of a network in a fair way when information is spread in the network.

We first consider the standard maximin criterion for group fairness and study the problem of determining key seed nodes to maximize the minimum probability that groups (or communities) receive information. We define two different variants of this problem that involve probabilistic strategies and analyze the relation between the two problems. We then design approximation algorithms achieving a constant multiplicative factor of $1 - 1/e$ minus an arbitrarily small additive error, while the original deterministic maximin problem was inapproximable. Our experimental study shows that the our methods ex-ante fairness values, i.e., minimum expected probability that an individual (or group) receives the information, dominate over the fairness values achieved by previous approaches. Interestingly and maybe more surprisingly, we observe that even the our methods ex-post fairness values, i.e., fairness values obtained by sampling

single sets according to the probabilistic strategies, frequently outperform the ex-post fairness achieved by other tested methods.

When using the maximin criterion, it is likely that still different groups receive different shares of information. Hence, we turn to study two classes of optimization problems involving notions of group fairness that aim to lessen this unfavourable situation. The goal here is to maximize the overall spread (or spread within a target set) while enforcing strict levels of fairness via constraints (either ex-post or ex-ante). The constraints require the coverage among groups to be similar. The level of fairness hence becomes a user choice rather than a property to be observed upon output. We present several NP-hardness and hardness of approximation results, even in the case that the fairness constraints are violated (multiplicatively and additively). For one of our problems we still design an algorithm with both constant approximation factor and constant fairness violation. For the other problem class, we propose two heuristics that allow the user to choose the tolerated fairness violation. In an extensive experimental study, we show that our algorithms perform well in practice, that is, they achieve the best fairness values while maintaining similar levels of total spread.

Finally, we study optimization problems with the goal of modifying the network structure by adding links in such a way that the minimum community coverage is maximized when information is spread using a purely efficiency oriented seeding strategy. We propose two optimization problems and present NP-hardness and hardness of approximation results for them. For some special cases, we propose efficient algorithms as well as several heuristics for solving one of the problems. In our experimental study, we show that our approach can be very successful in practice.

Acknowledgements

This thesis could not have been completed without the help of my supervisors. I would like to express my deepest gratitude to my supervisors Prof. Gianlorenzo D'Angelo and Dr. Ruben Becker for their continues assistance, invaluable guidance and patience during my Ph.D studies. Their encouragement and willingness to help throughout the research project have made this an inspiring experience for me. I would also like to thank the reviewers: Prof. Robert Bredereck and Prof. Nicola Gatti for their thoughtful comments and suggestions. I thank Prof. Michele Flammini and Prof. Luca Aceto for supporting me before coming to L'Aquila and starting my Ph.D. Last but not least, I deeply thank my family for their unconditional trust and support despite the long distance between us.

Contents

List of Figures

List of Tables

Bibliographic Notes

This thesis is based on the following papers.

The results of Chapter 3 are published in the following two papers. Note that the paper [15] is the journal version of paper [14].

[15] Ruben Becker, Gianlorenzo D'Angelo, Sajjad Ghobadi, and Hugo Gilbert. Fairness in influence maximization through randomization. *Journal of Artificial Intelligence Research.*, 73:1251–1283, 2022.

[14] Ruben Becker, Gianlorenzo D'Angelo, Sajjad Ghobadi, and Hugo Gilbert. 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*, pages 14684–14692, 2021.

The following paper contains most of the results presented in Chapter 4.

- Ruben Becker, Gianlorenzo D'Angelo, and Sajjad Ghobadi. On the Cost of Demographic Parity in Influence Maximization. In *37th AAAI Conference on Artificial Intelligence* (AAAI 2023). To appear.

Moreover, the results of Chapter 5 are published in the following paper.

- Ruben Becker, Gianlorenzo D'Angelo, and Sajjad Ghobadi. Improving Fairness in Information Exposure by Adding Links. In *37th AAAI Conference on Artificial Intelligence* (AAAI 2023). To appear.

Chapter 1

Introduction

Access to news or important information like job-related information may have a big impact on our life, because people make important decisions based on the information they receive or have access to. Social media platforms such as Facebook, LinkedIn, and Twitter, have provided a situation where users can easily access information and share their opinions about certain topics online. With the rapid growth of these platforms, online social networks have received a high level of attention. These social network platforms are very effective tools in bringing people together, sharing, exchanging, and spreading information or ideas to influence a large population in a short period of time.

The internet and especially social media have revolutionized the way information spreads through the population. On social media platforms users can pass information on to users they have a connection to, and subsequently these users are also connected to other users, and so on. Thus, information can be spread at little cost quite efficiently thanks to news platforms and social media. Consider the following examples that deal with the spread of information in social networks. Suppose that a company develops an online application for an online social network and would like to market it with the hope that it will be used by many individuals in the network. The idea is to select a small number of influential users in the network to use the application (by providing it for free or by paying them). These users will recommend the application to their friends, and their friends would recommend it to their friends' friends and so on, and thus through the "word-of-mouth" effect many individuals will end up using the application [27]. As another example consider Facebook, where a user John writes a post about an event that is happening in town. John's friends can see this post and by commenting on or sharing this post, their friends can see the information about this event and so on. In this way, the information about John's post will propagate through the network.

The basic algorithmic question here is which initial individuals to target such as to maximize the overall spread of influence in the network. More precisely, given a directed graph where nodes represent the users of a social network, and edges show the relationship between them and a probabilistic model on how information propagates through it, the main addressed question has been the following: Which *seed set* (influential users) of a limited size to target such that the expected number of nodes that obtain the information is maximized, when the information spreads from the chosen seed set? This problem, called *influence maximization*, was first introduced by Domingos and Richardson [35, 64] and formulated by Kempe et al. [46] as a discrete optimization problem. The problem has received a tremendous amount of attention [13, 18, 19, 22, 26, 30, 67] and has many applications in domains such as viral marketing [64], social recommendations, propagation of information related to jobs, financial inclusion [10], and public health programs [78, 79].

Efficient algorithmic solutions to the influence maximization clearly have the potential of being exploited with a malicious intention and there is reasons to believe that such malicious acts have already had a big impact on the world recently. Notice that about two-thirds of American adults get news on social media [1], and hence are comparatively vulnerable to fake news that are spread with the intention not to inform but to manipulate. Many believe this to have had a decisive impact on the 2016 presidential elections in the US [2] [61, 63]. Another consequence on society related to efficient algorithmic solutions for this problem stems from the fact that networks are not homogeneous but instead composed of different individuals forming groups or communities. Such groups can be defined based on the common attributes of their members like race, age, and gender. It is possible that some groups are well-connected and some are poorly connected. Thus, network structure can cause that any influence maximization algorithm may focus on well-connected users because choosing such nodes as seeds maximizes the overall spread (coverage). This implies that algorithms discriminate among different users or groups of users (called communities) (i.e., some users or communities may be covered with high probability, some may not be covered at all). Such observations,

[1] https://www.pewresearch.org/journalism/2018/09/10/news-use-across-social-media-platforms-2018/

[2] https://www.buzzfeednews.com/article/craigsilverman/viral-fake-election-news-outperformed-real-news-on-facebook

have motivated researchers to take *fairness* issues with respect to information spread into account. More precisely, the social network may be composed of individuals or communities (based on sensitive attributes such as race, age, and gender) and the goal is to provide similar information access to all of them. General works have then shifted focus away from maximizing the spread of information, towards assuring that each of the communities gets its fair share of information (or coverage). For instance, when spreading information about a health awareness program or when advertising a job on social media, the information should be propagated among different communities in a way that sensitive attributes of individuals in the communities have no effect on their access to the information. There is a wide variety of fairness notions [11] and we will study some of them in this thesis.

In summary, as social networks may have a big impact on our lives, and more and more people access information via social networks, it is essential to spread information in a fair way. In this thesis, we study different notions of (group) fairness. We define different optimization problems in this scope, prove several NP-hardness and hardness of approximation results, and design approximation algorithms as well as heuristics for proposed problems. Our methodology also includes thorough experimental evaluation of proposed algorithmic techniques both on randomly generated as well as on real world instances.

1.1 Our Contribution

The contributions of this thesis are summarized as follows.

- In Chapter 2, we provide some preliminaries related to influence maximization problem and describe fairness notions that we use in this thesis.
- In Chapter 3, we study the *maximin* criterion for group fairness and introduce 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 (an input parameter), 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 3.2, 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 3.3, 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. [71]) 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. [38] 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 σ_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 ε 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 [82] and 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 3.4, 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. [71], 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. [71] and Fish et al. [38]. 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.

The summery of our theoretical results can be found in Table 1.1 together with references to the respective statements in later sections.

Node-based	NP-hard [Theorem 3.7] NP-hard to $1 - 1/e + \varepsilon$-approximate [Theorem 3.11]	Price of fairness is unbounded [Lemma 3.4]	$1 - 1/e$-approximate minus the ε term [Theorem 3.16]
Set-based	NP-hard [Theorem 3.7]	Price of fairness is unbounded [Lemma 3.4]	$1 - 1/e$-approximate minus the ε term [Theorem 3.20]

Table 1.1: Summary of our complexity results in Chapter 3. The number α can be any factor in $(0, 1]$ and ε in $(0, 1)$.

- In Chapter 4, we adopt a different and more strict view on fairness, that is, we consider fairness as a requirement that has to be ensured by the algorithm rather than a notion to be maximized. In terms of the optimization problems at hand, this results in fairness being taken into account via constraints instead of in the objective function, the obvious advantage being that the resulting fairness violation is strictly bounded. More precisely, we develop optimization problems that aim to maximize the overall spread (or spread within a target set) while satisfying the fairness constraints (using *equalized odds*, *demographic parity*, and *predictive parity* notion). While such strict fairness notions may easily result in infeasibility, we show how to bypass this problem by studying also *ex-ante fairness* rather than just ex-post fairness.

It is clear that such a strict approach to fairness as adopted here may lead to a big loss in efficiency, i.e., in overall spread and possibly also in time complexity of respective algorithms. One of our contributions, is to rigorously analyze these two kinds of loss. In Section 4.2, we provide the hardness of approximation of the proposed optimization problems. We in fact prove that it is NP-hard to approximate the problems to within any bounded factor, even if the fairness constraints are violated (multiplicatively and additively). In Section 4.3, we study two optimization problems (under demographic parity) that permit randomized strategies in seed selection process. We prove that the *price of fairness* may be unbounded in this context. We then proceed by studying the complexity of the proposed probabilistic problems. This includes both proving NP-hardness and hardness of approximation results, see Subsection 4.3.3, and developing an approximation algorithm for one of the problems, see Section 4.4. Our study here explicitly includes bi-criteria approximation, that is, we relax the fairness constraints or allow them to be violated within a limited amount (multiplicatively or additively). This permits us to propose algorithms that entitle the user to choose the tolerated amount of fairness violation freely instead of observing the fairness violation upon seeing the output of the algorithm. We proceed by developing efficient heuristics for the other problem and conclude with a detailed experimental study on the performance of the developed algorithms both in terms of efficiency and fairness in Section 4.5. For our experiments, we use random, synthetic, and real world data sets. Our experimental study shows that although our theoretical results are mainly pessimistic, our algorithms achieve a trade-off between fairness and overall coverage and in some cases even achieve similar coverage as state-of-the-art influence maximization algorithms while guaranteeing fairness on top.

Table 1.2 contains a summery of our theoretical results.

- In Chapter 5, we first study an optimization problem with the goal of modifying the network structure by adding at most b non-edges to the network in such a way that the minimum community coverage is maximized when information is spread using a purely efficiency oriented seeding strategy, i.e., a seed set S

$\mathrm{IM}^{\mathrm{eo}}$	NP-hard to (α, β)-approximate [Theorem 4.4] NP-hard to $(\alpha, \varepsilon)^{+}$-approximate [Theorem 4.5]	—	—
$\mathrm{IM}^{\mathrm{pp}}$	NP-hard to (α, β)-approximate [Theorem 4.6] NP-hard to $(\alpha, \varepsilon)^{+}$-approximate [Theorem 4.7]	—	—
$\mathrm{IM}^{\mathrm{dp}}$	NP-hard to (α, β)-approximate [Theorem 4.4] NP-hard to $(\alpha, \varepsilon)^{+}$-approximate [Theorem 4.5]	Price of fairness is unbounded [Lemma 4.10]	—
$\mathrm{iIM}^{\mathrm{dp}}$	NP-hard to approximate better than $1 - 1/e$ [Theorem 4.12]	Price of fairness is unbounded [Lemma 4.10]	$(1 - 1/e, 1 - 1/e)$-approximate [Theorem 4.13]
$\mathrm{pIM}^{\mathrm{dp}}$	NP-hard [Theorem 4.11]	Price of fairness is unbounded [Lemma 4.10]	Heuristics

Table 1.2: Summary of our complexity results in Chapter 4. The numbers α and β can be any factor in $(0, 1]$ and ε in $[0, 1)$.

of size k that maximizes the spread after adding links to the network. We call this the $\mathrm{FIM}_{\mathrm{AL}}$ problem – fair influence maximization by adding links. We study the complexity of solving $\mathrm{FIM}_{\mathrm{AL}}$ in Section 5.1 and provide plenty of evidence that solving $\mathrm{FIM}_{\mathrm{AL}}$ is challenging, both exactly and approximately. Maybe most importantly, we show that it is unlikely to be able to find an α-approximation to the optimal solution, for any $\alpha \in (0, 1]$, even when having access to an oracle that solves an NP-complete problem. We furthermore show that $\mathrm{FIM}_{\mathrm{AL}}$ remains NP-hard for constant b or k (in the latter case even to be approximated).

We thus turn to study a second problem (Section 5.2) that is possibly practically better motivated in the first place – the $\mathrm{FIM}^{\mathrm{g}}_{\mathrm{AL}}$ problem: Here instead of assuming that the efficiency oriented entity uses maximizing sets to spread information, we assume it to employ the greedy algorithm. This is a quite realistic assumption as the problem of finding a maximizing set is NP-hard, while the greedy algorithm can be used in order to obtain a $1 - 1/e - \varepsilon$-approximation for any $\varepsilon \in (0, 1)$ with high probability (w.h.p.) in $\mathrm{poly}(n, \varepsilon^{-1})$ time, i.e, polynomial time in $n = |V|$

and ε^{-1}. Even more, this approximation guarantee is essentially optimal [46]. Multiple implementations of the greedy algorithm for IM exist (e.g., [67, 68]) and they have been shown to be extremely efficient in practice. We observe that, in contrast to $\mathrm{FIM}_{\mathrm{AL}}$, the $\mathrm{FIM}^{\mathrm{g}}_{\mathrm{AL}}$ problem is polynomial time solvable when b (size of non-edges) is a constant – exactly in the (unrealistic) case of deterministic instances and up to an arbitrarily small additive error in the probabilistic case. While this highlights the difference between the two problems, the proposed algorithm is essentially a brute-force algorithm and is thus not promising in practice. We complement the finding of this algorithm for the special case of constant k (size of seed sets) with a lower bound showing that it is NP-hard to provide any approximation algorithm. We then propose a set of algorithms for $\mathrm{FIM}^{\mathrm{g}}_{\mathrm{AL}}$ and evaluate them against each other in a first experiment in Section 5.3. We then take the best performing algorithm for $\mathrm{FIM}^{\mathrm{g}}_{\mathrm{AL}}$ and, in a second experiment, compare the resulting fairness (i.e., fairness achieved by the greedy algorithm after adding the proposed non-edges to the graph) with competitor algorithms that choose seeds as to optimize fairness. We observe that already after adding very few edges to graphs with thousands of nodes, the fairness achieved by our algorithm outperforms the fairness achieved by the fairness-tailored algorithms. Maybe surprisingly, this even holds for algorithms that optimize ex-ante fairness.

We summarize our theoretical results for $\mathrm{FIM}_{\mathrm{AL}}$ and $\mathrm{FIM}^{\mathrm{g}}_{\mathrm{AL}}$ in Table 1.3.

	general	constant b	constant k
$\mathrm{FIM}_{\mathrm{AL}}$	Σ_2^p-hard [Theorem 5.4] Σ_2^p-hard to α-approximate [Theorem 5.5]	NP-hard [Theorem 5.7]	NP-hard to α-approximate [Theorem 5.6]
$\mathrm{FIM}^{\mathrm{g}}_{\mathrm{AL}}$	NP-hard to α-approximate [Corollary 5.9]	- Polynomial time (deterministic case) [Observation 5.8] - ε-approximate (probabilistic case) [Lemma 5.11]	NP-hard to α-approximate [Corollary 5.9]

Table 1.3: Summary of our complexity results in Chapter 5. The number α can be any factor in $(0, 1]$ and $\varepsilon \in (0, 1)$.

- In Chapter 6, we conclude and present several directions as future work.

1.2 A Survey of Related Works

In this section, we survey the previous works that have considered fairness issues in the context of influence maximization. We also review the works on the problem of recommending links in a social network.

1.2.1 Fairness Notions in Influence Maximization

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. [38], 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. [71], 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, including maximin, 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 our work in Chapter 3. Rahmattalabi et al. [62] further extend the group fairness approach of Tsang et al. [71] 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. [65] study how improving the diversity of nodes in the seed set can influence efficiency and fairness of the information diffusion process.

They consider a notion that is essentially equivalent to demographic parity. In a rather specific setting, where the network is generated using a biased preferential attachment model yielding two unequal communities, the authors experimentally show that, under certain conditions, degree-based seeding strategies that take into account the diversity of nodes in the seed set are more efficient and equitable. Ali et al. [4] 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. The authors consider fairness notions that are similar to demographic parity, but instead of maintaining the fairness constraints, they pass the group coverages through some monotone concave function and include it in the objective. Farnadi et al. [36] 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 (MILPs). Their framework includes also "equity" which coincides with demographic parity. The authors 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. As their approach requires solving a MILP, it is however unlikely to be applicable to large real world instances. In fact, they restrict their experimental study to the relatively small synthetic networks from the work of Wilder et al. [76]. Gershtein et al. [41] introduce multi-objective influence maximization problem that aims at maximizing the influence of each group in the network. The authors propose two algorithms by splitting the budget (i.e., seed set size) between the groups to get the desired influence and linear program of Maximum Coverage problem. Anwar et al. [5] investigate that how existence of structural and influence diffusion homophily can affect the influence among different groups on homophilic networks consisting of two groups majority and minority. The authors defined the concept of balance that preserve majority vs. the minority group and proposed an objective function that maximizes the total influence and balance of the reached nodes. They show that when the objective function is monotone and submodular, then the problem can be approximated to within a constant factor. Wang et al. [75] study the problem of information access equality in order to reach each group at similar rate. In their setting, networks consist of two specific groups and are generated with different properties. The authors experimentally measure the efficiency and equality of receiving information between groups under different diffusion models. Khajehnejad et al. [47] study fair influence maximization 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. In both the works of Lu et al. [51] and Yu et al. [83], the authors investigated a two stage setting in which the host first finds a set of seeds under a cardinality constraint (the sum of all budgets) which approximately maximizes influence. Then the authors considered the task of splitting this set to allocate the seeds to the different agents in a fair manner by either minimizing the maximum amplification factor [51] or maximizing the ratio of the minimum amplification factor over the maximum one. These two works differ by the diffusion model they used. Indeed, while the first paper uses a variant of the linear threshold model, the second one uses a variant of the independent cascade model.

In contrast to the previous work however, we define optimization problems in such a way that permit randomized strategies in the seed selection process rather than only deterministic ones. In one problem, we consider randomized strategy that pick nodes as seeds independently. In contrast, in the other problem, we allow any probabilistic strategies that choose seed sets of expected size k, i.e., not restricting to independent distributions. Introducing randomization allows us to circumvent the inapproximability result of the original deterministic maximin problem studied by Fish et al. [38], and propose approximation algorithms that achieve a constant multiplicative factor of $1 - 1/e$, see Chapter 3. To the best of our knowledge, this is the first work that uses randomization in the context of influence maximization. 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 [54]. It can also be used to incentivize participation [9] or to workaround impossibility results [20]. Lastly and closer to our work, using randomization is frequently used to obtain fairer solutions [8, 16, 45]. 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 [53]. 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* [57]. Despite being an old research topic, the study of randomized solutions is still a hot topic where many open problems remain to be solved [7, 21].

1.2.2 Fairness through Recommending Links

There are several works in which the authors add links to the network taking into account social influence and diffusion process, however they do so with a different objective.

Castiglioni et al. [23] and Corò et al. [32] study the problem of adding edges to a graph in order to maximize the influence from a given seed set in different models of diffusion. Castiglioni et al. [23] prove that, for the IC model, it is NP-hard to approximate the problem to within any constant factor. Corò et al. [32] considered the LT model and proposed a constant approximation algorithm. The experimental study show that adding edges to the network can increase the influence of a given seed set. D'Angelo et al. [34] study the problem of adding a set of edges *incident to a given seed set* with the same aim. In a setting, where the cost of adding each edge is 1, the authors showed that it is NP-hard to approximate the problem to within a factor better than $1 - 1/(2e)$, and they proposed an algorithm with an approximation factor of $1 - 1/e$ for the IC model. They extended the results to the general case where the cost of adding each edge is in $[0, 1]$. Wu et al. [77] consider also different intervention actions than just adding edges, e.g., increasing the weights of edges. The authors show that, for the IC model, the problem of maximizing spread under these interventions is NP-hard and the objective function is neither submodular nor supermodular. Khalil et al. [48] study both the edge addition and deletion problems in order to maximize/minimize influence in the linear threshold model. They showed that the objective functions of both problems are supermodular and therefore there are algorithms for the problems with provable approximation guarantees. Garimella et al. [40] address the problem of recommending a set of edges to minimize the controversy score of the graph. In their setting, the graph is partitioned into two disjoint sets and the controversy score is defined as the difference of the probability that a random walk starting from one partition will end in the same partition and the probability that the random walk will end in the different partition. The authors proposed an algorithm without providing any approximation guarantee. Moreover, they do not consider any diffusion process in the network. Tong et al. [69] transform the edge addition/deletion problem to the problem of maximizing/minimizing the eigenvalue of the adjacency matrix and experimentally show that their proposed method increases the dissemination of information. Chaoji et al. [24] study the content maximization problem by adding at most k edges per node. Under the Restricted Maximum Probability Path model, the authors show that the objective function is submodular and the problem can be approximated to within a constant factor. Yu et al. [84] propose a link recommendation method using the algebraic connectivity of the network to maximize the spread of contents and success rate of recommended links. The authors experimentally show that their method improves the spread of content in the network. D'Angelo et al. [33] address the problem of selecting a set of seed nodes and adding a set of edges incident to these seed nodes, without exceeding a given budget, to maximize the expected number of reached nodes. The authors consider two cases where the cost of adding each edge is at least a given constant and any value in $[0, 1]$, and all the seed nodes have the same cost. For both cases, they propose algorithms with constant approximation guarantees. Ma et al. [52] study the problem of individual influence maximization to maximize the influence of a target node by adding k edges to this node. They consider LT model and find a set of edges incident to the target node that minimizes the influence overlap between the target node and other nodes.

The following two works are the most closely related to our work in Chapter 5. Swift et al. [66] introduce a problem to suggest a set of edges that contains at most k edges incident to each node to maximize the expected number of reached nodes while satisfying a fairness constraint (reaching each group in the network with the same probability, i.e., achieving demographic parity). The authors consider the Restricted Maximum Probability Path model and assume that they are given a set of candidate edges and a set of seed nodes for propagating information. They show that the problem is NP-hard and even is NP-hard to approximate to within any bounded factor, unless P = NP. Then, by violating the fairness constraints, they propose an LP-based algorithm with a factor of $1 - 1/e$ on the total spread and $2e/(1 - 1/e)$ on fairness. Our setting in Chapter 5 is different from the problem in [66] in terms of objective function, fairness notion, and the diffusion model. Moreover, the set of seeds in their problem is fixed, known and independent of the added edges. Our aim is to achieve fairness automatically, when an external agent selects an efficient seed set that may explicitly depend on the added edges. The authors in [66] add more edges to the network, in fact k edges per node. While our budget b that shows the number of added edges should be asymptotically smaller than n ($b \leq 50$ in our experiments).

Bashardoust et al. [12] study the maximin criterion by adding edges to the network under IC model with a transaction probability $\alpha \in [0, 1]$. In their setting, each node can be the source of distinct and equally-important information and the goal is to add at most k edges to the network to maximize the minimum probability that a node receives information. After defining several notions, the authors propose heuristics without providing any approximation guarantee and experimentally show that adding edges to the network can increase the minimum probability that nodes receive information. The authors study only individual fairness and their work lacks theoretical results in terms of computational complexity and approximation algorithms.

The main difference between our work and most of the previous works in link recommendation that take into account social influence is that they do not consider any fairness criteria in the diffusion process. Moreover, most work that consider fairness, e.g., [4, 38, 71], assume that the entity that is spreading the information, i.e., the agent choosing seed set S, has an inherent interest in spreading the information fairly, otherwise why would they want to use the developed fair algorithms? This assumption may however be flawed in reality – the spreading entity may be, and probably mostly are, purely efficiency-oriented and not particularly interested in choosing fair seeding strategies.

Chapter 2

Background

In this chapter, we describe *influence maximization*, the most widely studied models for propagating information, and some notions and basic results regarding the influence maximization problem. We also describe the fairness notions that we will use in this thesis.

2.1 Influence Maximization

We consider the classical influence maximization setting where we are given a directed arc-weighted graph $G = (V, E, w)$ with V being the set of n nodes, E the set of arcs, and $w : V \times V \to [0, 1]$ 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 [46]. Each node in G is either *active* (influenced or reached) or *inactive*. Whenever a node becomes active, it stays active throughout the diffusion process. 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.

2.1.1 Information Diffusion Models

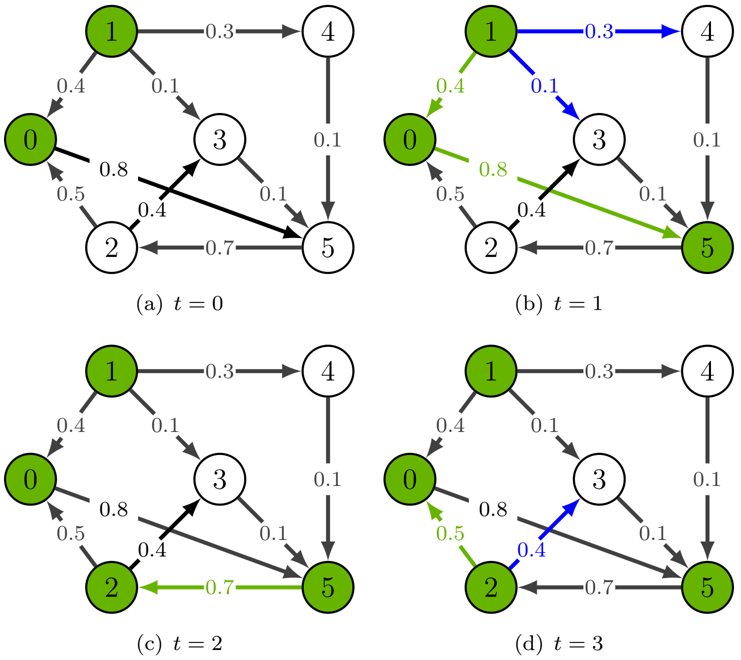
A lot of diffusion models have been designed to model information propagation in a social network. Next, we describe two of the most widely applied models, namely the Independent Cascade and the Linear Threshold models. Furthermore, we describe the Triggering Model that is a generalization of both the IC and LT models.

2.1.1.1 Independent Cascade Model

In the Independent Cascade (IC) model, the values $w_e \in [0,1]$ for $e \in E$ 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., $v \in S_{t+1}$. Once a node becomes active, it remains active for every succeeding time step. The process continues until no new nodes can be activated. Note that the process terminates after at most $|V|$ time steps. Figure 2.1 shows an example of a diffusion process under the IC model.

2.1.1.2 Linear Threshold Model

In the Linear Threshold (LT) model, the values $w_e \in [0,1]$ for $e \in E$ are such that, for each node v, it holds that $\sum_{(u,v)\in E} w_{uv} \leq 1$. 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 $\sum_{(u,v)\in E, u\in S_t} w_{uv} \geq \theta_v$ becomes active, i.e., $v \in S_{t+1}$, where the thresholds θ_v are chosen independently and uniformly at random from the interval $[0,1]$ for all nodes $v \in V$. The process continues until there is no new activated nodes. Figure 2.2 shows an example of a diffusion process under the LT model.



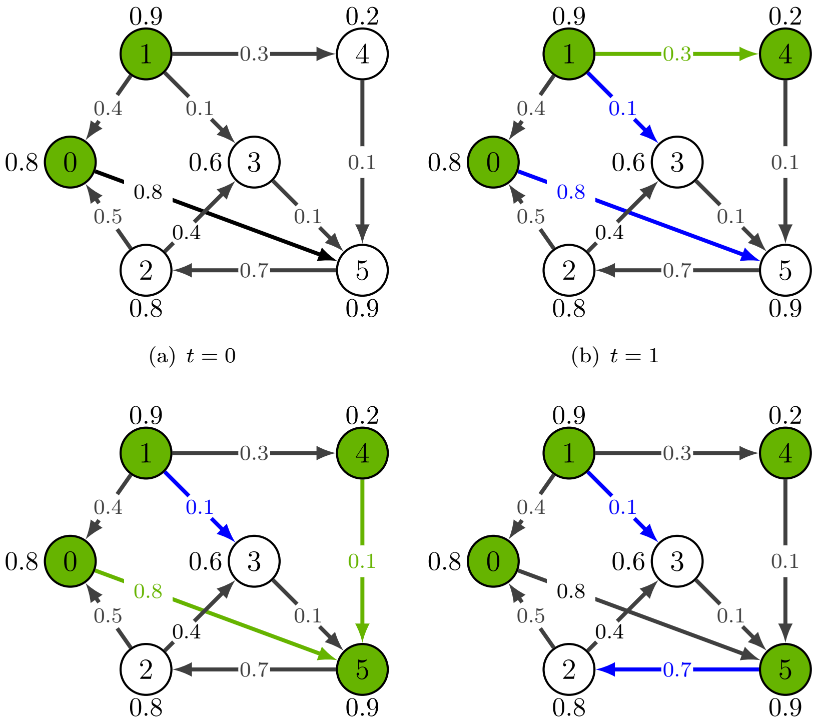
(a) $t = 0$ (b) $t = 1$

(c) $t = 2$ (d) $t = 3$

Figure 2.1: An example of the diffusion process of the IC model. Green nodes denote active nodes.

2.1.1.3 Triggering Model

Both the IC and LT models can be generalized to what is known as the *Triggering Model*, see [46, 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 $G_L = (V, E_L)$ denote the sub-graph of G where $E_L = \{(u, v) | v \in V, u \in T_v\}$. The graph G_L is frequently referred to as *live-edge* graph and the edges E_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 $1 - \sum_{u \in N_v} w_{uv}$.



(a) $t = 0$

(b) $t = 1$

(c) $t = 2$

(d) $t = 3$

Figure 2.2: An example of the diffusion process of the LT model. Green nodes denote active nodes.

2.1.2 Monotone and Submodular Set Functions

There are several definitions for submodular functions in the literature, but Nemhauser et al. [58] proved that all of these definitions are equivalent. They also introduced some properties of submodular set functions that used by Kempe et al. [46] to present an approximation algorithm for IM problem.

Definition 2.1. (Submodularity) A set function $f : 2^V \to \mathbb{R}$ is *submodular* if for any two sets $S \subseteq T \subseteq V$ and any element $x \in V \setminus T$, the marginal gain from adding the element x to S is at least as high as the marginal gain from adding the same element to the superset T. Formally

$$f(S \cup \{x\}) - f(S) \geq f(T \cup \{x\}) - f(T).$$

Definition 2.2. (Monotonicity) A function $f : 2^V \to \mathbb{R}$ is *monotone* if for any subsets $S \subseteq T \subseteq V$ it holds that $f(S) \leq f(T)$.

The following theorem states that for any monotone and submodular set function f, there is a greedy algorithm that finds a set of elements of limited size that maximizes the function f and provides a constant approximation factor.

Theorem 2.3 ([31, 58]). *For a monotone and submodular set function $f(\cdot)$, the greedy algorithm that in each of k iterations selects the element with the largest marginal increase in $f(\cdot)$ produces a set S^g of size k such that $f(S^g) \geq (1 - 1/e)\max_{|S|=k} f(S)$, where e is the base of the natural logarithm.*

2.1.3 Influence Maximization Problem

We now ready to formally define influence maximization problem.

Definition 2.4 (Influence Maximization). Given a graph $G = (V, E, w)$, a diffusion model, and a budget k, the objective is to find a seed set S with $|S| \leq k$ such that $\sigma(S)$ is maximized.

Kempe et al. [46] showed that the influence function $\sigma(\cdot)$ for the both IC and LT models is monotone and submodular. Thus using the greedy algorithm in Algorithm 1, we get an approximation algorithm with a factor of $1 - 1/e$. It is not feasible to evaluate the influence function $\sigma(\cdot)$ in polynomial time. It has been proven that it is #P-hard to compute $\sigma(\cdot)$ precisely both for the LT model [28] and the IC model [74]. However, using the Chernoff–Hoeffding bounds, the function can be approximated by sampling a sufficiently large number of live-edge graphs.

Algorithm 1 Greedy Hill Climbing

```
Input: Graph $G = (V, E, w)$ and a budget $k$
Output: Set $S \subseteq V$ with $|S| \leq k$
$S \leftarrow \emptyset$
while $|S| < k$ do
    $v \leftarrow \arg\max_{u \in V \setminus S}\{\sigma(S \cup u) - \sigma(S)\}$
    $S \leftarrow S \cup \{v\}$
end while
return $S$
```

Proposition 2.5 (Proposition 4.1 in [46]). *Let ε and δ be two small and real numbers. For a given seed set S, if we sample at least $\Omega(n^2/\varepsilon^2 \ln(1/\delta))$ live-edge graphs, with probability at least $1-\delta$, the average number of activated nodes over the live-edge graphs is a $(1 \pm \varepsilon)$-approximation to $\sigma(S)$.*

We now state the main theorem that shows the optimal solution for influence maximization can be approximated to within a constant factor.

Theorem 2.6 (Theorem 1.1 in [46])**.** *For the IC and LT models, there is a polynomial-time greedy algorithm that approximates the maximum influence to within a factor of* $1 - 1/e - \varepsilon$*, with probability at least* $1 - \delta$*, where e is the base of the natural logarithm and* ε *is any positive real number.*

2.2 Fairness Notions

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 [17]. The network consists of two communities or groups (set of nodes), 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. [68], in order to compute a seed set of size k with large expected coverage in the graph. As we can see in Figure 2.3 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. In the right plot in Figure 2.3, 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.

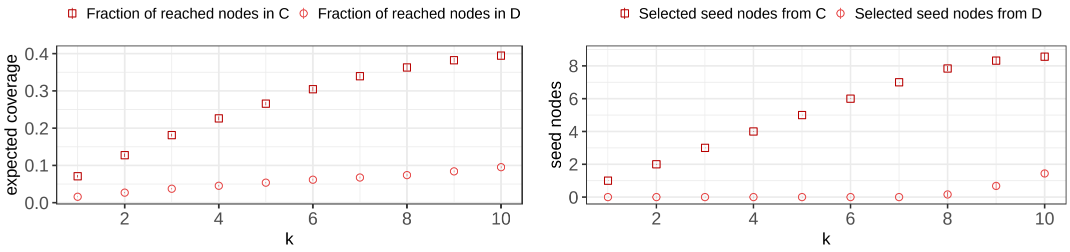


Figure 2.3: Results for the core-periphery model with a core of 50 nodes and a periphery of 150 nodes. The budget k is increasing from 1 to 10.

We proceed by reviewing the definition of fairness notions that we use in this thesis. There is not one specific definition of fairness. An intuitive criterion is the *maximin* criterion which requires that the utility of the worst-off group should be maximized. In the context of influence maximization, 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, where the objective is to maximize the minimum probability that nodes receive the information, has been considered by Fish et al. [38], who showed that the problem is hard to approximate to any constant approximation factor, unless P $=$ NP. The second problem, where the objective is to maximize the minimum probability that communities are reached, has been considered by Tsang et al. [71]. 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. To better understand this notion, consider the example of hiring with two groups male and female. The first problem aims to maximize the minimum probability that an applicant is hired (independent of the group membership). However, the second problem aims to maximize the minimum fraction of hired applicants per group. Another fairness notion that is used in the literature, for example in the machine learning community, see, e.g., Definition 1 in Chapter 2 in the book by Barocas et al. [11], is *demographic parity* (also referred to as independence) that falls into the category of group fairness and is actually defined as *equality* in probability of being selected conditioned on group membership. This notion is also considered in the context of influence maximization [4, 36, 65, 66]. In the hiring scenario, demographic parity requires that the fraction of hired applicants should be equal among the two groups, see Figure 2.2.

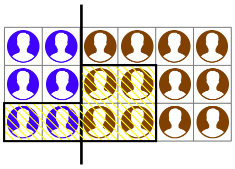


Figure 2.4: The fraction of hired applicants in each group is 1/3. The vertical line separates the groups and the dashed area shows the hired applicants.

Another fairness notions that we use in this thesis are *equalized odds* (also called separation), see, e.g., Definition 2 in Chapter 2 in the book by Barocas et al. [11] and *predictive parity* (referred to as outcome test) [29, 73]. Equalized odds requires that conditioned on target value (e.g., applicants with a good and bad resume), the probability of being selected should be the same among all communities (groups). In the example of hiring, let T and $\bar{T}$ be the sets of applicants with good and bad resume, respectively. Equalized odds implies that the probability of an applicant with an actual good resume to be correctly hired and the probability of an applicant with an actual bad resume to be incorrectly hired should both be the same for male and female applicants, see Figure 2.2.

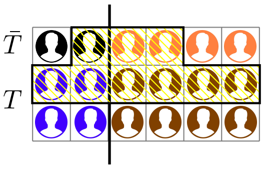


Figure 2.5: The fraction of hired applicants with good and bad resume out of all applicants with good and bad resume in each group is 1/2 and 1/2, respectively. The vertical line separates the groups and the dashed area shows the hired applicants.

Predictive parity requires that the fraction of selected targeted users (e.g., applicants who are hired for a job and have good resume) out of all selected users should be the same for each community. In the example of hiring, predictive parity implies that for both male and female applicants, the probability of an applicant that is hired to actually have a good resume should be the same, see Figure 2.2.

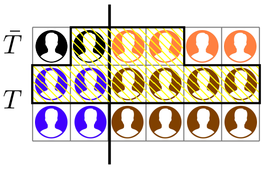


Figure 2.6: The fraction of hired applicants with good resume out of all hired applicants in each group is 2/3. The vertical line separates the groups and the dashed area shows the hired applicants.

2.3 Further Notation

For a seed set $S \subseteq V$, we define $\sigma_v(S) := \Pr_{\mathcal{L}}[v \in \rho_{\mathcal{L}}(S)]$ as the probability that node $v \in V$ is reached from seed nodes S. Note that the expected spread is the sum over all these probabilities, i.e., $\sigma(S) = \mathbb{E}_{\mathcal{L}}[|\rho_{\mathcal{L}}(S)|] = \sum_{v \in V} \Pr_{\mathcal{L}}[v \in \rho_{\mathcal{L}}(S)] = \sum_{v \in V} \sigma_v(S)$. We extend this notation in a natural way, that is, for a set (or group) of nodes $C \subseteq V$, we denote by $\sigma_C(S) = \frac{1}{|C|} \cdot \sum_{v \in C} \sigma_v(S)$ the average probability of nodes being reached in C or equivalently this is the expected group coverage of C, i.e., the expected fraction of nodes from C that are reached.

We use $\mathbb{1}$ for the all-ones vector (of suitable dimension) and $\mathbb{1}_i$ for the i'th unit vector. Furthermore, with a slight abuse of notation, we use $\mathbb{1}_P$ to be the indicator function that equals 1 if P holds and 0 otherwise. We also say that an event holds with high probability (w.h.p.), if it holds with probability at least $1 - n^{-\alpha}$ for a constant α that can be made arbitrarily large.

Approximation Algorithms. For $N \in \mathbb{N}$, we use $[N]$ to denote the integers from 1 to N. We will consider maximization problems of the form $\max\{F(x) : x \in R \text{ and } \exists \gamma : A_i(x) = \gamma \text{ for all } i \in [m]\}$, where R is a feasibility region, the functions $A_i : R \to \mathbb{R}_{\geq 0}$, for $i \in [m]$, define a set of (additional) constraints that can possibly be violated or hold only approximately, and $F : R \to \mathbb{R}_{\geq 0}$ is an objective function. We consider approximation algorithms (possibly) with constraint violation. Let $\alpha, \beta \in (0, 1]$ be real values. Then, we say that $x \in R$ is *β-feasible* if $A_i(x) \geq \beta \cdot A_j(x)$ for all pairs of $i, j \in [m]$. We say that $x \in R$ is an *(α, β)-approximation* if x is β-feasible and $F(x) \geq \alpha \cdot \mathrm{opt}$, where opt is the optimum value. We call an algorithm a *(α, β)-approximation algorithm*, if it is a polynomial-time algorithm whose output solutions are (α, β)-approximations. Similarly, for a given $\varepsilon \in [0, 1)$, we say that $x \in R$ is *ε^+-feasible* if $\left|A_i(x) - A_j(x)\right| \leq \varepsilon$ for all $i, j \in [m]$ and $i \neq j$. For $\alpha \in (0, 1]$ and $\varepsilon \in [0, 1)$, an *$(\alpha, \varepsilon)^+$-approximation* algorithm produces an ε^+-feasible $x \in R$ such that $\sigma(x) \geqslant \alpha\, \mathrm{opt}$.

Chapter 3

Maximin Fairness through Randomization

In this chapter, we study the problem of determining key seed nodes for influence maximization in social networks in an efficient and fair manner. Similar to previous works like Fish et al. [38] and Tsang et al. [71], we study the *maximin* criterion for (group) fairness. We extend these works by studying the impact of randomization on fairness.

3.1 Problem Definition

We start by introducing the problem that has been investigated by Fish et al. [38] and Tsang et al. [71].

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. [38] and Tsang et al. [71], 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. [38]. 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. [38] considered the special case where the diffusion model is the Independent Cascade model and in which all arcs have the same probability of diffusion α. 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.

3.1.1 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 3.1 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.

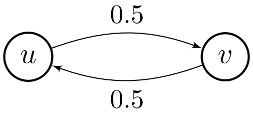


Figure 3.1: Simple instance showing that randomization allows to increase fairness in influence maximization.

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., $\mathcal{P} := \{p \in [0,1]^{2^V} : \mathbb{1}^T p = 1, \sum_{S\subseteq V} p_S|S| \leq k\}$ and let $S \sim p$ denote the random process of sampling S according to the distribution p. We now consider the optimization problem

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

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 [53].

We note that in the conference paper [14] we studied the set-based probabilistic maximin 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 the conference paper [14]. 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 Subsection 3.1.2 in 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 $\mathcal{X} := \{x \in [0,1]^n : \mathbb{1}^T x \leq k\}$ 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 $f(x) := \mathbb{E}_{S \sim x}[f(S)]$.

3.1.2 Relationship between Problems

We first observe that, for $x \in \mathcal{X}$, the vector p^x defined as $p^x_S := \prod_{i \in S} x_i \prod_{j \in V \setminus S}(1 - x_j)$, 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 3.1. *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 3.2. *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 $y^p_i := \text{Pr}_{S \sim p}[i \in S] = \sum_{S \subseteq V : i \in S} p_S$. The correlation gap [2, 80] of $f : 2^V \to \mathbb{R}_{\geq 0}$ 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. [2, Corollary 1.2] or Yan [80, 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

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

$$= \frac{e}{e-1} \cdot \min_{C \in \mathcal{C}} \mathbb{E}_{S \sim y^p}[\sigma_C(S)] \leq \frac{e}{e-1} \cdot \mathrm{opt}_{\mathcal{X}}(G, \mathcal{C}, k),$$

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 $y^p \in \mathcal{X}$. □

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

Lemma 3.3. *There exists a graph G, community structure $\mathcal{C}$, and integer k, such that* $\mathrm{opt}_{\mathcal{P}}(G, \mathcal{C}, k) \geq \frac{5}{4} \cdot \mathrm{opt}_{\mathcal{X}}(G, \mathcal{C}, k)$ *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 3.1 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 $1/2 + 1/2 \cdot 2/3 = 5/6$. □

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.

3.2 Price of Fairness and Hardness

3.2.1 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

$$F_{\mathcal{X}}(G, \mathcal{C}, k) = \arg\max_{x \in \mathcal{X}} \min_{C \in \mathcal{C}} \mathbb{E}_{S \sim x}[\sigma_C(S)] \quad \text{and} \quad F_{\mathcal{P}}(G, \mathcal{C}, k) = \arg\max_{p \in \mathcal{P}} \min_{C \in \mathcal{C}} \mathbb{E}_{S \sim p}[\sigma_C(S)],$$

respectively. Then, the respective prices of fairness $\mathrm{PoF}_{\mathcal{X}}(G,\mathcal{C},k)$ and $\mathrm{PoF}_{\mathcal{P}}(G,\mathcal{C},k)$ incurred by restricting to strategies in $F_{\mathcal{X}}(G,\mathcal{C},k)$ and $F_{\mathcal{P}}(G,\mathcal{C},k)$ are given by

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

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

Lemma 3.4. *For any even $n > 0$, there is a graph G with n nodes and a community structure $\mathcal{C}$ such that $\mathrm{PoF}_{\mathcal{X}}(G,\mathcal{C},1) = \mathrm{PoF}_{\mathcal{P}}(G,\mathcal{C},1) = (n+2)/4$, 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 and $k = 1$. 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 $F_{\mathcal{X}}(G,\mathcal{C},1) = \{\rho\cdot\mathbb{1}_{I\cup\{w\}}\}$ 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 $\mathrm{opt}_{\mathcal{X}}(G,\mathcal{C},1) = \mathrm{opt}_{\mathcal{P}}(G,\mathcal{C},1) = \rho$. 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, $\mathrm{PoF}_{\mathcal{X}}(G,\mathcal{C},1) = \mathrm{PoF}_{\mathcal{P}}(G,\mathcal{C},1) = (\frac{n}{2}+1)/2 = (n+2)/4$. □

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

Lemma 3.5. *For any graph G, community structure $\mathcal{C}$ and number k, it holds that $\mathrm{PoF}_{\mathcal{X}}(G,\mathcal{C},k) \leq n/k$ and $\mathrm{PoF}_{\mathcal{P}}(G,\mathcal{C},k) \leq n/k$.*

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., $\mathbb{1}^T x = k$ 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. □

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 $\mathrm{PoF}_{\mathcal{X}}$ and $\mathrm{PoF}_{\mathcal{P}}$ 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

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

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 $\overline{\mathrm{PoF}}_{\mathcal{X}}(G,\mathcal{C},k) \geq \mathrm{PoF}_{\mathcal{X}}(G,\mathcal{C},k)$ and $\overline{\mathrm{PoF}}_{\mathcal{P}}(G,\mathcal{C},k) \geq \mathrm{PoF}_{\mathcal{P}}(G,\mathcal{C},k)$. Moreover, we note that Lemma 3.4 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 3.5 does not transfer to the pessimistic notion. In fact, we obtain only the following weaker lemma.

Lemma 3.6. *For any graph G, community structure $\mathcal{C}$ and number k, it holds that $\overline{\mathrm{PoF}}_{\mathcal{X}}(G,\mathcal{C},k) \leq n$ and $\overline{\mathrm{PoF}}_{\mathcal{P}}(G,\mathcal{C},k) \leq n$.*

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., $\mathbb{1}^T x \geq 1$. 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. □

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 $\overline{\mathrm{PoF}}_{\mathcal{X}}(G,\mathcal{C},2) \geq n$ and $\overline{\mathrm{PoF}}_{\mathcal{P}}(G,\mathcal{C},2) \geq n$. 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.

3.2.2 Hardness

Fish et al. [38] show that the standard maximin problem as introduced in Section 3.1 is NP-hard and even inapproximable. In this subsection, we provide hardness results for our set-based and node-based probabilistic maximin problems that we introduced in Subsection 3.1.1. 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 3.7. *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 $T = \{v_{i_1}, \ldots, v_{i_k}\}$ 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 3.2. 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 $I_e = \{u_{e^1}, \ldots, u_{e^\lambda}\}$ 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 α 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 α. 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 size k in G if

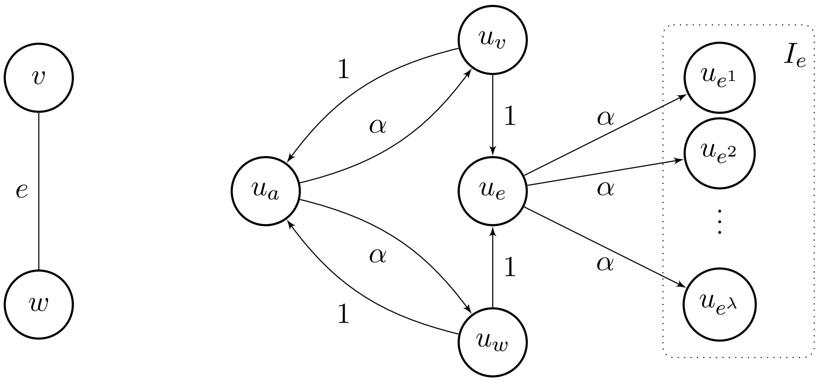


Figure 3.2: Illustration of the reduction: Scheme in $\overline{G}$ that is obtained for one edge $e = \{v, w\}$ in G. Note that there is just one node u_a in $\overline{G}$, while there is a node u_e and a set of nodes I_e for each edge $e \in E$.

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 3.8. *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 $\min_{u \in U} \mathbb{E}_{S \sim x}[\sigma_u(S)] \geq \alpha$. □

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

Observation 3.9.

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 $p' := p - p_S \mathbb{1}_S + p_S \mathbb{1}_{S'}$ (resp. $x' := x - x_u \mathbb{1}_u + x_u \mathbb{1}_{u'}$) 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 $\sigma_{u_{e^i}}(S) \leq \alpha(1 - (1 - \alpha)^2)$. □

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

Lemma 3.10. *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 3.9, 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 λ. 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

$$\begin{aligned} \sigma_{u_{e^1}}(p^*) &= \Pr_{S\sim p^*}[u_{e^1} \in S] + \mathbb{E}_{S\sim p^*}[\sigma_{u_{e^1}}(S) \mid u_{e^1} \notin S] \cdot \Pr_{S\sim p^*}[u_{e^1} \notin S] \\ &\leq \frac{k}{\lambda} + \mathbb{E}_{S\sim p^*}[\sigma_{u_{e^1}}(S) \mid u_{e^1} \notin S]. \end{aligned} \tag{3.1}$$

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 3.9, we get

$$\mathbb{E}_{S\sim p^*}[\sigma_{u_{e^1}}(S) \mid u_{e^i} \notin S] \leq \alpha(1 - (1-\alpha)^2) \cdot \rho_e + \alpha \cdot (1 - \rho_e) = \alpha - \rho_e \cdot \alpha(1-\alpha)^2. \tag{3.2}$$

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

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

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 (3.2) and this again into (3.1) 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. □

The above two lemmata directly prove Theorem 3.7.

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

Theorem 3.11. *For any $\varepsilon > 0$, the node-based probabilistic maximin problem, cannot be approximated to within a factor $1 - 1/e + \varepsilon$, unless $\mathrm{P} = \mathrm{NP}$.*

Proof. The proof is by reduction from the Max-Coverage problem. An instance of Max-Coverage is given by an integer κ 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 κ 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 $\mathrm{P} = \mathrm{NP}$ [37, 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 ℓ 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

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

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

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

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. □

3.3 Approximation Algorithms

In this section, we show that there are algorithms with constant approximation factors 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 ε for any $\varepsilon > 0$.

3.3.1 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 cannot be multiplicatively approximated 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 additive ε-approximations to $\text{opt}_{\mathcal{P}}(G, \mathcal{C}, k)$ and $\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k)$, respectively.

Formally, for $T \in \mathbb{Z}_{\geq 0}$, we let $L_1, \ldots, L_T$ 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 3.12. *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 $|\tilde{\sigma}_v(S) - \sigma_v(S)| \leqslant \varepsilon$ 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 [56]. 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, $\Pr[|\tilde{\sigma}_v(S) - \sigma_v(S)| \geqslant \varepsilon] \leq 2e^{-2T\varepsilon^2} \leq \delta \cdot 2^{-n}/n$ 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 $|\tilde{\sigma}_v(S) - \sigma_v(S)| \leqslant \varepsilon$ for all $v \in V$ and for all $S \in 2^V$. □

We now observe that the absolute ε-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 ε-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 ε-approximations. Hence we get the following lemma.

Lemma 3.13. *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 $\tilde{\text{opt}}_{\mathcal{P}}(G, \mathcal{C}, k) = \max_{p \in \mathcal{P}} \min_{C \in \mathcal{C}} \tilde{\sigma}_C(p)$ and $\tilde{\text{opt}}_{\mathcal{X}}(G, \mathcal{C}, k) = \max_{x \in \mathcal{X}} \min_{C \in \mathcal{C}} \tilde{\sigma}_C(x)$ and $p \in \mathcal{P}$ and $x \in \mathcal{X}$ be solutions such that $\min_{C \in \mathcal{C}} \tilde{\sigma}_C(p) \geq \alpha \cdot \tilde{\text{opt}}_{\mathcal{P}}(G, \mathcal{C}, k) - \beta$ and $\min_{C \in \mathcal{C}} \tilde{\sigma}_C(x) \geq \alpha \cdot \tilde{\text{opt}}_{\mathcal{X}}(G, \mathcal{C}, k) - \beta$, respectively. Then p and x are solutions such that $\min_{C \in \mathcal{C}} \sigma_C(p) \geq \alpha \cdot \text{opt}_{\mathcal{P}}(G, \mathcal{C}, k) - (\beta + \varepsilon)$ and $\min_{C \in \mathcal{C}} \sigma_C(x) \geq \alpha \cdot \text{opt}_{\mathcal{X}}(G, \mathcal{C}, k) - (\beta + \varepsilon)$ 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 3.12 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

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

The proof for the node-based problem is completely analogous. □

3.3.2 Probabilistically Choosing Nodes

We start with the node-based problem. It entails to solve the optimization problem $\mathrm{opt}_{\mathcal{X}}(G, \mathcal{C}, k) := \max_{x \in \mathcal{X}} \min_{C \in \mathcal{C}} \sigma_C(x)$, where $\sigma_C(x) = \mathbb{E}_{S \sim x}[\sigma_C(S)]$ for $C \in \mathcal{C}$ and $x \in \mathcal{X} := \{x \in [0,1]^n : \mathbb{1}^T x \leq k\}$. 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 3.13 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. [25] in combination with a binary search on a threshold can be used in order to get a $1-1/e$-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 lemma.

Lemma 3.14. *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 $\{i \in V : v \in \rho_L(i)\} =: \{1, \ldots, r\} = R$, i.e., R is the set of nodes that can reach node v in L. Using the geometric-arithmetic mean inequality, we get

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

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

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

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 $\prod_{i=1}^{r-1}(1-x_i) \leq 1$. □

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 3.15. *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 $1-1/e$-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 Lemma 3.14 yield the result. □

Together with Lemma 3.13 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 3.16. *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 $\min_{C\in\mathcal{C}} \sigma_C(x) \geq (1 - \frac{1}{e}) \operatorname{opt}_{\mathcal{X}}(G, \mathcal{C}, k) - \varepsilon$.*

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:

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

By combining Lemma 3.15 and Lemma 3.13, we get that, with probability at least $1 - \delta$, the optimal solution to the above LP is a multiplicative $1 - 1/e$-approximation plus the additive $-\varepsilon$ term to $\operatorname{opt}_{\mathcal{X}}(G, \mathcal{C}, k)$. □

3.3.3 Probabilistically Choosing Sets

Recall that the set-based probabilistic maximin problem is defind as $\operatorname{opt}_{\mathcal{P}}(G, \mathcal{C}, k) := \max_{p\in\mathcal{P}} \min_{C\in\mathcal{C}} \sigma_C(p)$, where $\sigma_C(p) = \mathbb{E}_{S\sim p}[\sigma_C(S)]$ for $C \in \mathcal{C}$ and $p \in \mathcal{P} := \{p \in [0, 1]^{2^V} : \mathbb{1}^T p = 1, \sum_{S\subseteq V} p_S|S| \leq k\}$. In the light of Lemma 3.13, we focus on finding approximate solutions to $\max_{p\in\mathcal{P}} \min_{C\in\mathcal{C}} \tilde{\sigma}_C(p)$.

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. However, the problem is NP-hard. 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. 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 subsection, 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 [82] 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 [82] 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 $f : P \to \mathbb{R}^\mu$ be a μ-dimensional linear function over P. Assume that $0 \leq f_j(x) \leq \omega$ for all $j \in [\mu]$ and $x \in P$, where ω 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 3.17 (Young [82]). *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 an α-approximation for $\max_{p \in \mathcal{P}} \min_{C \in \mathcal{C}} \tilde{\sigma}_C(p)$, if we provide an oracle with multiplicative approximation α. Hence, let us take a closer look at the requirements of Theorem 3.17 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

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

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

$$\mathrm{opt}_{\mathcal{O}}(G, \mathcal{C}, k, \omega) := \max_{p \in \mathcal{P}} \tilde{\sigma}^\omega(p). \tag{3.3}$$

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

Lemma 3.18. *It holds that* $\mathrm{opt}_{\mathcal{O}}(G, \mathcal{C}, k, \omega) = \mathrm{opt}_{\mathcal{Q}}(G, \mathcal{C}, k, \omega)$ *with*

$$\mathrm{opt}_{\mathcal{Q}}(G, \mathcal{C}, k, \omega) := \max_{q \in \mathcal{Q}} \sum_{i=1}^{n} q_i \tilde{\sigma}^{\omega}(S_i^*), \tag{3.4}$$

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

Proof. Let q^* be an optimal solution to (3.4). Define $p \in \mathcal{P}$ as $p_S = q_i^*$ if $S = S_i^*$ and 0 otherwise. Then, p is a feasible solution to (3.3) and thus $\mathrm{opt}_{\mathcal{O}}(G, \mathcal{C}, k, \omega) \ge \mathrm{opt}_{\mathcal{Q}}(G, \mathcal{C}, k, \omega)$. It remains to prove that $\mathrm{opt}_{\mathcal{O}}(G, \mathcal{C}, k, \omega) \le \mathrm{opt}_{\mathcal{Q}}(G, \mathcal{C}, k, \omega)$. For this sake let $p^* \in \mathcal{P}$ be an optimal solution to (3.3). Assume $p_S^* > 0$ for some set S of cardinality i, but $S \neq S_i^*$. Consider the solution $p' := p - p_S \mathbb{1}_S + p_S \mathbb{1}_{S_i^*}$. Clearly, $\tilde{\sigma}^{\omega}(p) \le \tilde{\sigma}^{\omega}(p')$ 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 (3.3) 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 (3.4) and thus $\mathrm{opt}_{\mathcal{O}}(G, \mathcal{C}, k, \omega) \le \mathrm{opt}_{\mathcal{Q}}(G, \mathcal{C}, k, \omega)$. □

In other words, among the vectors that attain the optimum $\mathrm{opt}_{\mathcal{O}}(G, \mathcal{C}, k, \omega)$, 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 $1 - 1/e$-approximations to $\max\{\tilde{\sigma}^{\omega}(S) : S \subseteq V, |S| \le i\}$. 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

$$\mathrm{opt}_{\mathcal{Q}}^{\mathrm{gr}}(G, \mathcal{C}, k, \omega) := \max_{q \in \mathcal{Q}} \sum_{i=1}^{n} q_i \tilde{\sigma}^{\omega}(S_i) \tag{3.5}$$

that is identical to (3.4) up to the replacement of S_i^* by S_i. We obtain the following lemma.

Lemma 3.19. *The vector $\mathbb{1}_k$ is an optimal solution to the problem in* (3.5)*. Consequently, $\tilde{\sigma}^\omega(S_k) = \mathrm{opt}^{\mathrm{gr}}_{\mathcal{Q}}(G, \mathcal{C}, k, \omega) \geq (1 - 1/e) \cdot \mathrm{opt}_{\mathcal{O}}(G, \mathcal{C}, k, \omega)$.*

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 $\mathcal{Q} := \{q \in [0,1]^n : \mathbb{1}^T q = 1, \sum_{i=1}^n i \cdot q_i \leq k\}$ 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, $\Delta_1 \geq \Delta_2 \geq \ldots \geq \Delta_n$ and thus the optimum is $\sum_{i=1}^k \Delta_i$. This implies that

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

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

Theorem 3.20. *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. $\min_{C \in \mathcal{C}} \sigma_C(p) \geq (1 - \frac{1}{e}) \mathrm{opt}_{\mathcal{P}}(G, \mathcal{C}, k) - \varepsilon$. 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 ω 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 $1 - 1/e$-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 3.17 with $\eta = \varepsilon/2$ thus implies that Young's algorithm returns a solution $p \in \mathcal{P}$ with $\min_{C \in \mathcal{C}} \tilde{\sigma}_C(p) \geq \alpha \cdot (1 - \frac{\varepsilon}{2}) \max_{p \in \mathcal{P}} \min_{C \in \mathcal{C}} \tilde{\sigma}_C(p) \geq \alpha \cdot \max_{p \in \mathcal{P}} \min_{C \in \mathcal{C}} \tilde{\sigma}_C(p) - \frac{\varepsilon}{2}$ 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 3.13 with $\varepsilon/2$ leads that we get a multiplicative $1 - 1/e$-approximation minus an additive ε term to $\mathrm{opt}_{\mathcal{P}}(G, \mathcal{C}, k)$ in polynomial time with probability at least $1 - \delta$. $\square$

We now state the multiplicative-weight routine for the set-based problem, see Algorithm 2. We assume a routine Greedy$(\tilde{\sigma}^\omega(\cdot), k)$ that performs k iterations of the greedy hill climbing algorithm on $\tilde{\sigma}^{\omega}(\cdot)$ and returns a $1 - 1/e$-approximation to the oracle problem.

Algorithm 2 **MultiplicativeWeight**$(G, \mathcal{C}, k, \varepsilon, \delta, \eta)$

```
Sample $T \geq 4\varepsilon^{-2} \cdot [n + \log n + \log \delta^{-1}]$ live-edge graphs
$p \leftarrow 0$, $i \leftarrow 1$, $\Pi \leftarrow -\infty$, $\Delta \leftarrow \infty$, $z_C \leftarrow 1$ and $s_C \leftarrow 0$ for $C \in \mathcal{C}$
while $\Pi \geq (1 - \eta) \cdot \Delta$ do
  $\omega_v \leftarrow \sum_{C \in \mathcal{C}: v \in C} \frac{z_C}{|C|}$ for all $v \in V$
  $S \leftarrow \textsc{Greedy}(\tilde{\sigma}^{\omega}(\cdot), k)$
  $p \leftarrow p + \mathbb{1}_S$
  $s_C \leftarrow \frac{i-1}{i} \cdot s_C + \frac{1}{i} \cdot \tilde{\sigma}_C(S)$ and $z_C \leftarrow z_C \cdot (1 - \eta \cdot \tilde{\sigma}_C(S))$, for all $C \in \mathcal{C}$
  $\Pi \leftarrow \min_C \{s_C\}$ and $\Delta \leftarrow \min \left\{\Delta, \frac{\tilde{\sigma}^{\omega}(S)}{\mathbb{1}^T z}\right\}$
  $i \leftarrow i + 1$
end while
return $p/i$
```

3.4 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. The routine for the set-based problem is the one described in Subsection 3.3.3. We refer to this algorithm as **set_based** in what follows. For the node-based problem, an implementation of the LP-based algorithm from Subsection 3.3.2 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 3.15 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. [38] as well as the more involved method due to Tsang et al. [71]. We continue by describing the experimental setting in detail.

3.4.1 Experimental Setting

Competitors. The methods of Fish et al. [38] 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 also consider this method in a case that there is a set of community in the network. We refer to this algorithm as **grdy_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. 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**.

The algorithm of Tsang et al. [71] 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. [25] and Udwani [72], 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 **moso**.

We also compare our algorithm to the standard greedy algorithm for influence maximization. We use the slightly more involved and very efficient TIM implementation [68]. While 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 **grdy_im** 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 **uniform**.

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 [44] 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 η parameter of the multiplicative weight routine (see Theorem 3.17) to be 0.1.

We use the IC model as diffusion model in all our experiments. For the methods due to Fish et al. [38] and Tsang et al. [71] (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 ε-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. [71], 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.

Networks. We use the following networks: (1) Random instances generated according to the Barabási-Albert model [3] 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. [71]. (4) Real world instances from the SNAP database [50] and a paper by Guimerà et al. [42], 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 3.1. 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.

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 3.1: Properties of random, synthetic and real world networks

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 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.

3.4.2 Results

Barabási-Albert Graphs. For the Barabási-Albert graphs, we explore singleton communities, the BFS community structure with $m = k$,[1] and random imbalanced community structures of sizes $4n/10, 3n/10, 2n/10, n/10$. The results are reported in Figure 3.3. In the left plots in Figure 3.3, 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 **grdy_im** outperforms other methods for all values of n. We note however that the advantage in efficiency of **grdy_im** 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.

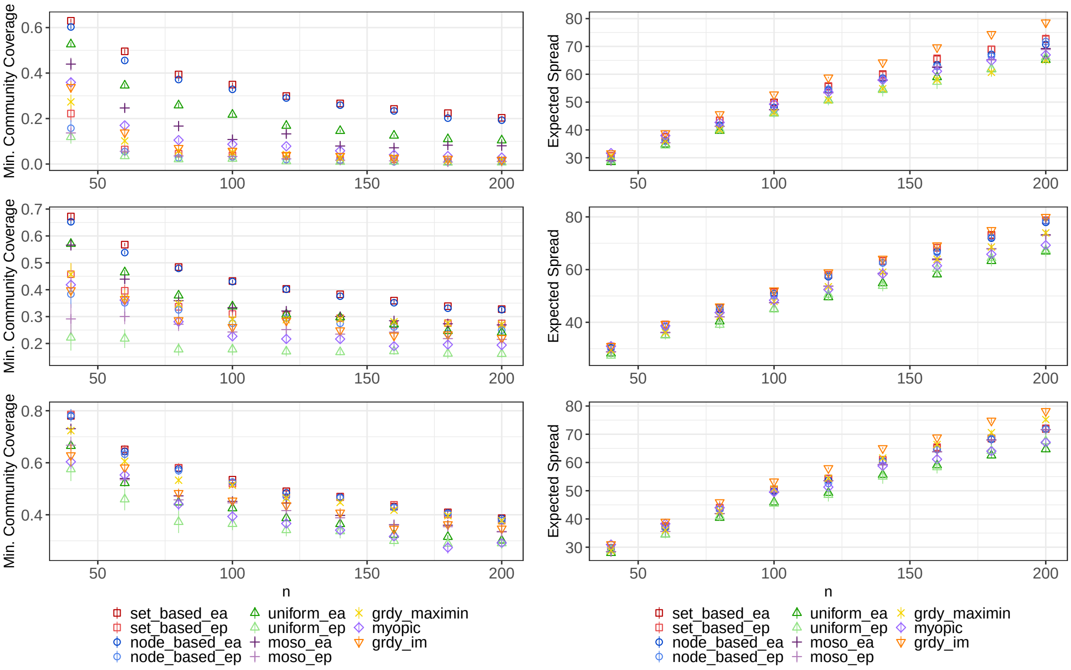


Figure 3.3: Results for Barabási-Albert instances: $k = 20$, n increasing from 40 to 200 in steps of 20. The minimum community probability is shown on the left, while expected spread is shown on the right. From top to bottom, we see (1) singleton community structure, (2) BFS community structure, and (3) random imbalanced community structure.

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

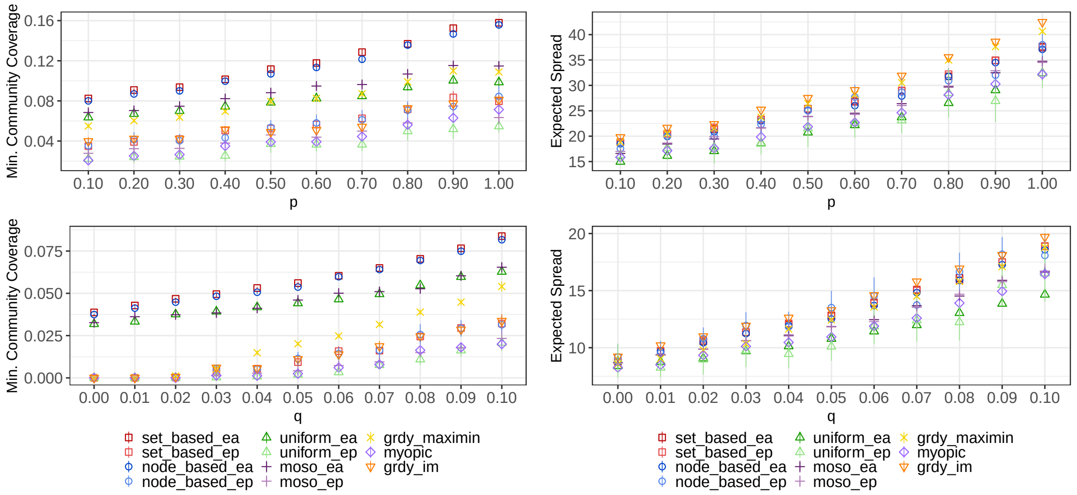


Figure 3.4: Results for block stochastic graphs: edge weights constant 0.05, $k = 8$, $n = 200$. The minimum community probability is shown on the left, while expected spread is shown on the right. From top to bottom, we see (1) $q = 0.1$ and p increasing from q to 1 in steps of 0.1, (2) $p = 0.1$ and q increasing from 0 to p in steps of 0.01.

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 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 3.4. 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 **grdy_im** 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.

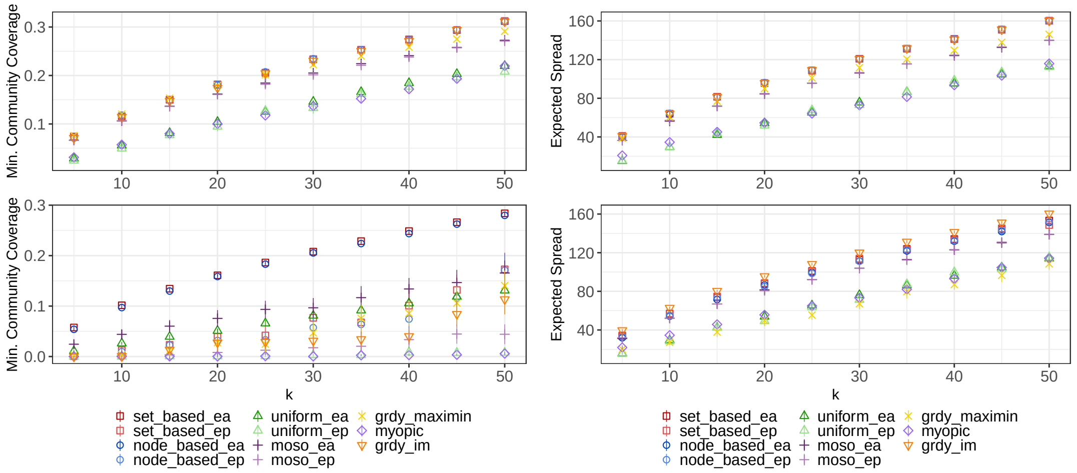


Figure 3.5: Results for the instances of Tsang et al. [71]: k increasing from 5 to 50 in steps of 5. The minimum community probability is shown on the left, while expected spread is shown on the right. From top to bottom, we see (1) community structure induced by attribute gender, (2) community structure induced by attributes region, gender and ethnicity.

Instances of Tsang et al. Next we evaluate the algorithms on the instances used by Tsang et al. [71]. These are synthetic networks introduced by Wilder et al. [76] 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. [71], all algorithms w.r.t. group fairness of the communities induced by some of those attributes. The results are reported in Figure 3.5. 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**, **moso**, and **uniform** are all almost identical to 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 **grdy_im** and our methods are very close to each other.

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

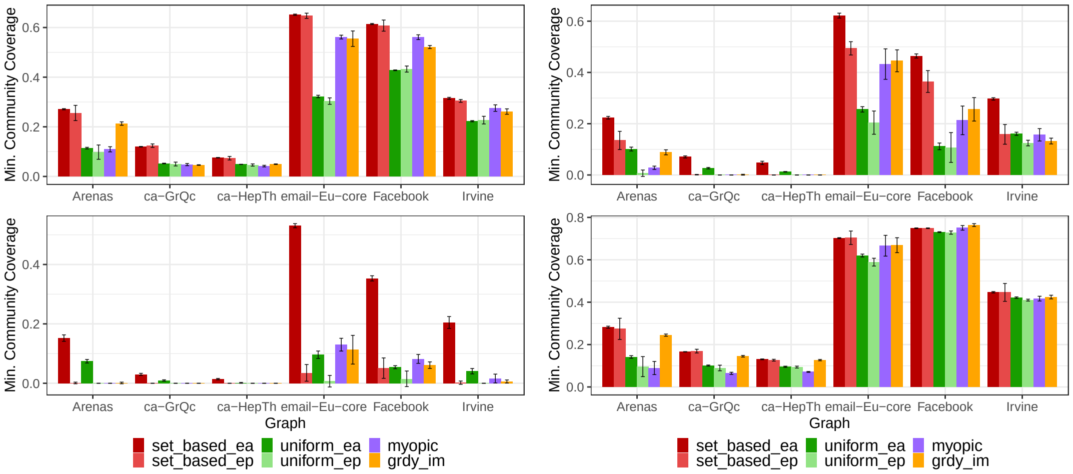


Figure 3.6: Results for the instances used by Fish et al. for $k = 100$. BFS community structure with (top left) 10 communities, (top right) $n/10$ communities, (bottom left) $n/2$ communities. (bottom right) Random imbalanced community structure with 16 communities.

Arenas [42] 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 [49] 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 [49] 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 [55] This dataset represents a part of the Facebook network, where nodes are users and edges indicate friendships.

Irvine [59] 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 [81] 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 **grdy_maximin** 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.

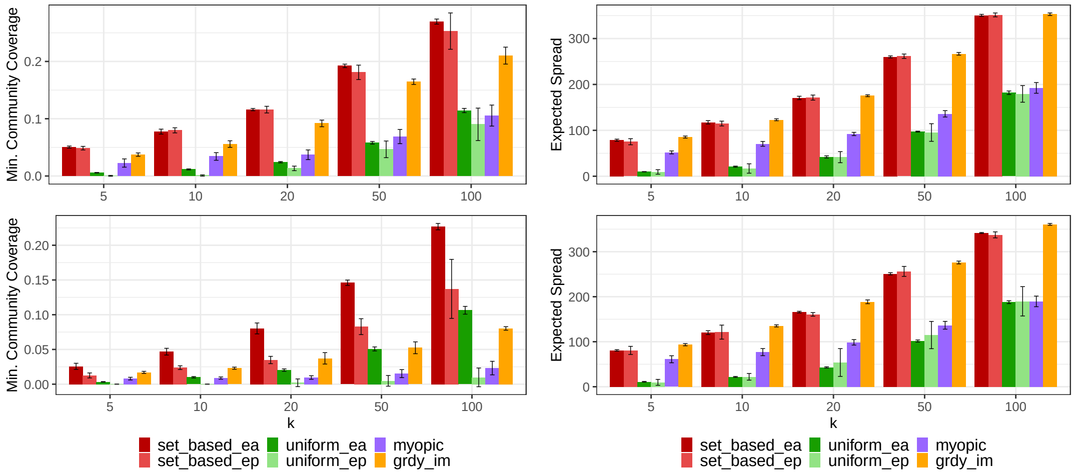


Figure 3.7: Results for the Arenas network for increasing $k = 5, 10, 20, 50, 100$. The minimum community probability is shown on the left, while expected spread is shown on the right. BFS community structure with (1) 10 communities, (2) $n/10$ communities.

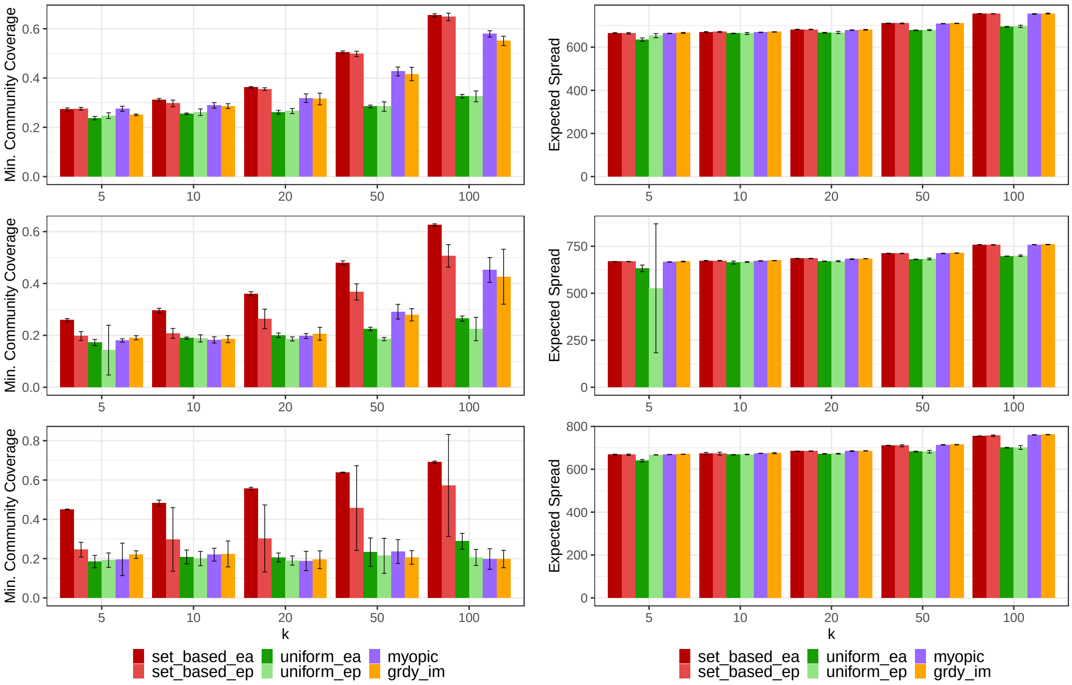


Figure 3.8: Results for email-Eu-core network for increasing $k = 5, 10, 20, 50, 100$. The minimum community probability is shown on the left, while expected spread is shown on the right. (1) BFS community structure with 10 communities, (2) BFS community structure with $n/10$ communities, (3) community structure induced by departments.

Evaluation on Networks used by Fish et al. We start with the networks that were also used in the study of Fish et al. [38]. We experiment with different community structures, both the BFS community structures with different community sizes and the random imbalanced community structure. The results can be found in Figure 3.6. 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 **grdy_im** 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 3.7 and 3.8. 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, **grdy_im** 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 **uniform** for small values of k) perform almost identical.

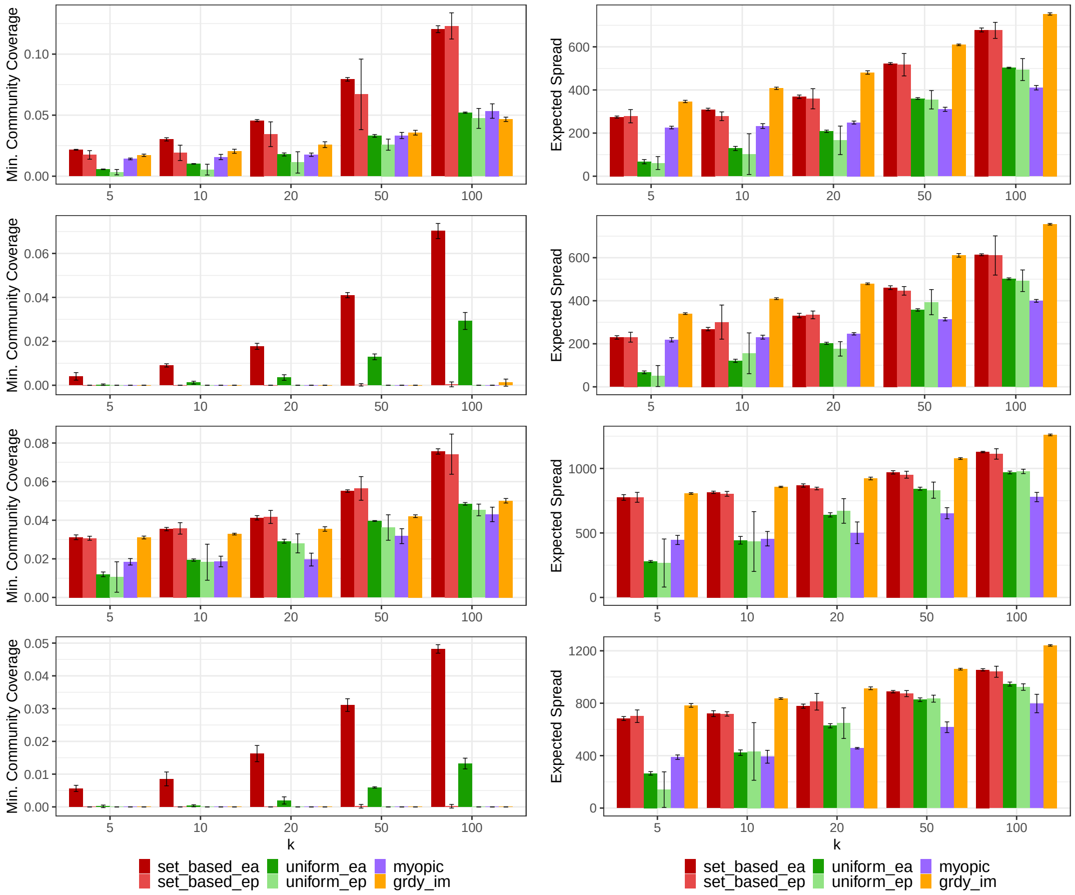


Figure 3.9: Results for co-authorship networks for increasing $k = 5, 10, 20, 50, 100$. The minimum community probability is shown on the left, while expected spread is shown on the right. BFS community structure (1) ca-GrQc with 10 communities, (2) ca-GrQc with $n/10$ communities, (3) ca-HepTh with 10 communities (4) ca-HepTh with $n/10$ communities.

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 3.9. 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 **uniform** do achieve a significantly non-zero ex-ante value, the results of **set_based** being more than twice as high compared to the values of **uniform**. 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 **grdy_im** and **set_based** perform the best, while there is a bigger advantage for **grdy_im** in this case than with most other instances tested. Note however that **set_based** does achieve significantly better fairness values as compared to **grdy_im** in all settings where it falls behind **grdy_im** 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 3.10. 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 algorithm are very similar. Note that **set_based** performs almost the same as **grdy_im** in the expected spread while being significantly better than **grdy_im** in terms of fairness.

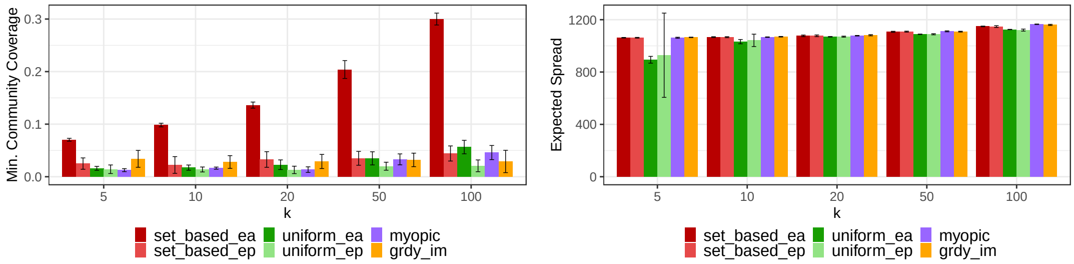


Figure 3.10: Results for the com-Youtube network for increasing $k = 5, 10, 20, 50, 100$. The minimum community probability is shown on the left, while expected spread is shown on the right.

Chapter 4

Demographic Parity through Randomization

In this chapter, we investigate optimization problems using various notions of (group) fairness that aim at maximizing the total spread or spread over specific set of users while satisfying fairness constraints.

As access to information via social networks may have a big impact on our life, see, e.g. [10], researchers have taken also *fairness* issues with respect to information spread into account, see related works in Section 1.2 of Chapter 1. Here, an essential question arises, namely: What do we mean by fair? There is a large variety of fairness notions [11] and in fact different notions have been investigated also in this scope, with the most common one being the maximin criterion [15, 38, 71]. What all three previously mentioned works, have in common however is that they consider fairness as a measure to be optimized, namely via maximizing the minimum coverage.

This raises, however, a conceptual question. When maximizing the minimum coverage, we may still end up in a situation where the values of two groups differ a lot. More precisely, consider an example with two groups, say C and D. All the three mentioned approaches would prefer an outcome where C gets a coverage of 0.5 while D gets a coverage of 1 over an outcome where both receive a coverage of 0.499. Now, while fairness is a debatable concept, the second outcome may be considered more fair by many. In fact, if we take a closer look at demographic parity, e.g., [11, Definition 2 in Chapter 2], we observe that, demographic parity is actually defined as equality in probability of being selected conditioned on group membership. In the above example, this is satisfied in the second outcome, but far from being satisfied in the first. More fundamentally, the following question arises. In all of these works fairness is considered as a notion to be optimized. But is this the right way of considering fairness? Is fairness not instead something that we want algorithms to *guarantee*, i.e., do not we want to restrict algorithms to satisfy certain levels of fairness independent of their objective?

4.1 Problem Definition

To address fairness in influence maximization, here we focus on defining optimization problems using various notions of group fairness. In the classical influence maximization problem, given a graph G and an integer k, the objective is to find a set of k seeds that maximizes the expected spread, i.e., $\max_{S\in\mathcal{S}}\{\sigma(S)\}$, where $\mathcal{S} := \{S \subseteq V : |S| \leq k\}$ is the set of subsets of nodes of size at most k. We refer to the optimal value of this optimization problem as $\mathrm{opt}(G, k)$. Motivated by real world applications such as marketing, let $T \subseteq V$ and $\bar{T} = V \setminus T$ be the sets of targeted (e.g., a set of users who *likes* the product that a company is promoting) and non-targeted nodes (e.g., a set of users who *hates* the product that a company is promoting) in G, respectively. In addition to G and k, we are given a community structure $\mathcal{C}$. The communities $\mathcal{C} = \{C_1, \ldots, C_m\}$ is a partition if and only if $C_i \cap C_j = \emptyset$ for all $i \neq j$ and $\bigcup_{i\in[m]} C_i = V$. We also use $\nu_A(S) = \mathbb{E}_{\mathcal{L}}[|\rho_{\mathcal{L}}(S) \cap A|]$ to denote the expected number of nodes reached in a set $A \subseteq V$ from seed set S.

Requiring Equalized Odds. We are now ready to formally define our first optimization problem, we refer to it as $\mathrm{IM}^{\mathrm{eo}}$, standing for influence maximization under equalized odds:

$$\max_{S\in\mathcal{S}}\{\nu_T(S) : \exists t, \bar{t} \text{ s.t. } \sigma_{C\cap T}(S) = t \text{ and } \sigma_{C\cap\bar{T}}(S) = \bar{t} \text{ for all } C \in \mathcal{C}\}. \tag{$\mathrm{IM}^{\mathrm{eo}}$}$$

The goal is to find a set S of size at most k that maximizes the expected number of reached targeted nodes while the fraction of reached targeted and non-targeted nodes in each community is the same among all communities. For an instance, consisting of a graph G, communities $\mathcal{C}$, and an integer k, we call $\mathrm{opt}^{\mathrm{eo}}_{\mathcal{S}}(G, \mathcal{C}, k)$ the optimum of $\mathrm{IM}^{\mathrm{eo}}$.

Observation 4.1. Assume that $\mathcal{C}$ is a partition of the node set V. Let S be a set of at most k seed nodes that satisfies $\sigma_{C\cap T}(S) = t$ and $\sigma_{C\cap \bar{T}}(S) = \bar{t}$ for all $C \in \mathcal{C}$, then $\nu_T(S) = t \cdot |T|$.

Proof. Using that $\mathcal{C}$ is a partition, we get

$$\begin{aligned}\nu_T(S) &= \mathbb{E}_{\mathcal{L}}[|\rho_{\mathcal{L}}(S) \cap T|] = \sum_{v\in V} \mathbb{E}_{\mathcal{L}}[\mathbb{1}_{v\in\rho_{\mathcal{L}}(S)\wedge v\in T}] = \sum_{v\in T} \mathbb{E}_{\mathcal{L}}[\mathbb{1}_{v\in\rho_{\mathcal{L}}(S)}] \\ &= \sum_{C\in\mathcal{C}} \sum_{v\in C\cap T} \mathbb{E}_{\mathcal{L}}[\mathbb{1}_{v\in\rho_{\mathcal{L}}(S)}] = \sum_{C\in\mathcal{C}} \sum_{v\in C} \mathbb{E}_{\mathcal{L}}[\mathbb{1}_{v\in\rho_{\mathcal{L}}(S)\wedge v\in T}] = \sum_{C\in\mathcal{C}} \mathbb{E}_{\mathcal{L}}[|\rho_{\mathcal{L}}(S) \cap (C\cap T)|] \\ &= \sum_{C\in\mathcal{C}} \nu_{C\cap T}(S) = \sum_{C\in\mathcal{C}} t\cdot |C\cap T| = t\cdot|T|,\end{aligned}$$

where in the second to last step we used that S is the set that satisfies $\sigma_{C\cap T}(S) = \frac{\nu_{C\cap T}(S)}{|C\cap T|} = t$ for every $C \in \mathcal{C}$. □

Requiring Demographic Parity. This notion is the special case of the equalized odds notion where $T = V$. Under the demographic parity, our goal is to find a set S of size at most k that maximizes the total spread while the fraction of reached nodes in each community is equal among all communities. We define the following optimization problem, we refer to it as $\mathrm{IM}^{\mathrm{dp}}$, standing for influence maximization under demographic parity:

$$\max_{S\in\mathcal{S}} \big\{\sigma(S) : \exists\gamma \text{ s.t. } \sigma_C(S) = \gamma \text{ for all } C \in \mathcal{C}\big\}. \tag{$\mathrm{IM}^{\mathrm{dp}}$}$$

For an instance G, $\mathcal{C}$, and k, we call $\mathrm{opt}^{\mathrm{dp}}_{\mathcal{S}}(G, \mathcal{C}, k)$ the optimum of $\mathrm{IM}^{\mathrm{dp}}$.

Observation 4.2. Assume that $\mathcal{C}$ is a partition of the node set V. Let S be a seed set of size at most k with $\sigma_C(S) = \gamma$ (for some γ) for all $C \in \mathcal{C}$, then $\sigma(S) = \gamma \cdot |V|$.

Proof. Using that $\mathcal{C}$ is a partition and S is the set that satisfies $\sigma_C(S) = \frac{\nu_C(S)}{|C|} = \gamma$ for every $C \in \mathcal{C}$, we get

$$\begin{aligned}\sigma(S) &= \mathbb{E}_{\mathcal{L}}[|\rho_{\mathcal{L}}(S)|] = \sum_{v\in V} \mathbb{E}_{\mathcal{L}}[\mathbb{1}_{v\in\rho_{\mathcal{L}}(S)}] = \sum_{C\in\mathcal{C}} \sum_{v\in C} \mathbb{E}_{\mathcal{L}}[\mathbb{1}_{v\in\rho_{\mathcal{L}}(S)}] = \sum_{C\in\mathcal{C}} \mathbb{E}_{\mathcal{L}}[|\rho_{\mathcal{L}}(S) \cap C|] \\ &= \sum_{C\in\mathcal{C}} \nu_C(S) = \sum_{C\in\mathcal{C}} \gamma \cdot |C| = \gamma\cdot|V|.\end{aligned}$$

□

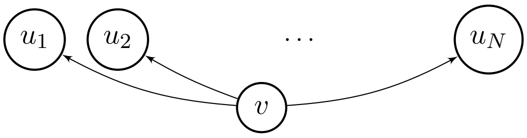


Figure 4.1: Instance used in the proof of Lemma 4.3 illustrating the contrast between the maximin criterion and demographic parity. The edge probabilities are set to $(1+\varepsilon)/N$ and the IC model is used as diffusion model.

Requiring Predictive Parity. We consider the following optimization problem under predictive parity, we denote it by $\mathrm{IM}^{\mathrm{pp}}$, standing for influence maximization under predictive parity:

$$\max_{S\in\mathcal{S}}\{\nu_T(S) : \exists s \text{ s.t. } \frac{\nu_{C\cap T}(S)}{\nu_C(S)} = s \text{ for all } \ C \in \mathcal{C}\}. \tag{$\mathrm{IM}^{\mathrm{pp}}$}$$

Here the objective is to find a seed set S of size at most k that maximizes the coverage of targeted nodes while the ratio of the expected number of reached targeted nodes to the expected number of reached nodes in each community is the same among all communities. For an instance G, $\mathcal{C}$, and k, we denote the optimum of $\mathrm{IM}^{\mathrm{pp}}$ with $\mathrm{opt}^{\mathrm{pp}}_{\mathcal{S}}(G,\mathcal{C},k)$.

4.1.1 Demographic Parity vs. Maximin

We proceed by giving an example that illustrates that considering the maximin criterion as done in Chapter 3 and demographic parity in our strict sense can lead to drastically different outcomes. More precisely, we construct an instance where the optimal maximin solution suffers linear multiplicative violation in demographic parity, while achieving an expected coverage that is only around twice as good as a solution that achieves perfect demographic parity. This is formalized below.

Lemma 4.3. *Let $\varepsilon > 0$. There is an instance $G, \mathcal{C}, k$ with n nodes, in which the optimal maximin strategy achieves an overall expected coverage of $2+\varepsilon$, but suffers a violation in demographic parity of $(n-1)/(1+\varepsilon) = \Theta(n)$. On the other hand, the expected total coverage achieved by optimal demographic parity strategy is* $\mathrm{opt}(G,\mathcal{C},k) = (n+1)/(n-\varepsilon) = 1 + \Theta(1/n)$.

Proof. Consider the graph G in Figure 4.1 consisting of $n = N+1$ nodes $v, u_1, \ldots, u_N$. Let $\mathcal{C}$ be the community structure consisting of all singleton communities, i.e. $\mathcal{C} =$

$\{\{v\}, \{u_1\}, \ldots, \{u_N\}\}$. There is an edge (v, u_i), for each $i \in [N]$ with probability $(1+\varepsilon)/N$. Furthermore, we assume that the IC model is used and set $k = 1$. Note that by the choice of the edge probability, the optimal maximin strategy q will assign probability 1 to the set $\{v\}$. This results in $\sigma(q) = 2+\varepsilon$ and $\sigma_{u_i}(q) = (1+\varepsilon)/N$ for each $i \in [N]$. As $\sigma_v(q) = 1$, this leads to a multiplicative violation in demographic parity of $N/(1+\varepsilon) = \Theta(n)$. On the other hand, consider the probabilistic strategy p that assigns $1/(N-\varepsilon)$ to the set $\{v\}$ and $(1-1/(N-\varepsilon))/N$ to each set $\{u_i\}$, for $i \in [N]$. It is clear that $\sigma_v(p) = 1/(N-\varepsilon)$ and furthermore $\sigma_{u_i}(p) = (1-1/(N-\varepsilon))/N + (1+\varepsilon)/(N(N-\varepsilon))$ for $i \in [N]$, which equals $1/(N-\varepsilon)$. Hence, the expected group coverage is identical for all groups. Furthermore, the overall spread is $(N+1)/(N-\varepsilon) = 1+\Theta(1/n)$, which is a lower bound on $\mathrm{opt}(G, \mathcal{C}, k)$. □

4.2 Hardness Results

In this section, we give several hardness of approximation results for $\mathrm{IM}^{\mathrm{eo}}$, $\mathrm{IM}^{\mathrm{dp}}$, and $\mathrm{IM}^{\mathrm{pp}}$. We show that it is NP-hard to approximate $\mathrm{IM}^{\mathrm{eo}}$, $\mathrm{IM}^{\mathrm{dp}}$, and $\mathrm{IM}^{\mathrm{pp}}$ to within any bounded factor. Indeed, we prove two stronger and more general statements: One cannot find in polynomial time a solution that approximates the optimums of $\mathrm{IM}^{\mathrm{eo}}$, $\mathrm{IM}^{\mathrm{dp}}$, and $\mathrm{IM}^{\mathrm{pp}}$, even if we allow the fairness constraints to be violated by a multiplicative or an additive term, unless P = NP.

4.2.1 Hardness of $\mathrm{IM}^{\mathrm{eo}}$ and $\mathrm{IM}^{\mathrm{dp}}$

Note that $\mathrm{IM}^{\mathrm{dp}}$ is the special case of $\mathrm{IM}^{\mathrm{eo}}$, thus all the hardness results for $\mathrm{IM}^{\mathrm{dp}}$ also hold for $\mathrm{IM}^{\mathrm{eo}}$. We start with the multiplicative case.

Theorem 4.4. *For any $\alpha \in (0, 1]$, $\beta \in (0, 1]$, there is no (α, β)-approximation algorithm for $\mathrm{IM}^{\mathrm{dp}}$, unless* P = NP.

Proof. Let β' be the largest $\beta' \leq \beta$ such that $1/\sqrt{\beta'}$ is integer. We show the stronger statement for β' instead of β. We reduce from Set Cover, where we are given a ground set $U = \{U_1, \ldots, U_\nu\}$, a collection of subsets $D = \{D_1, \ldots, D_\mu\}$ over U, and an integer κ, and we aim to determine whether there exists a subset $D' \subseteq D$ of size κ whose union is U. Given an instance of Set Cover, we define an instance of $\mathrm{IM}^{\mathrm{dp}}$. W.l.o.g. we can

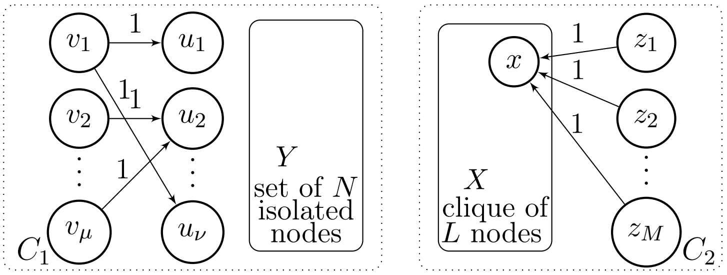


Figure 4.2: Construction of G from an instance of Set Cover.

assume the instance to be large enough, that is $\mu > 1/\sqrt{\beta'}$. Furthermore, we assume that information spread follows the IC model. The graph $G = (V, E, w)$ in the IM$^{\text{dp}}$ instance is constructed as illustrated in Figure 4.2. We define an integer $\lambda := 1/\sqrt{\beta'}$ that depends on β' and influences the size of G. The node set V consists of two disjoint and disconnected communities $\mathcal{C} = \{C_1, C_2\}$. The first community C_1 consists of (1) one node v_j for each $D_j \in D$, (2) one node u_i for each $U_i \in U$, and (3) a set of $N = (\mu \cdot \lambda - 1) \cdot (\mu + \nu)$ (isolated) nodes Y. The only edges in C_1 are those defined by the Set Cover instance, i.e., there is an edge from v_j to u_i, whenever $U_i \in D_j$. The second community C_2 consists of (1) a bidirected clique X of $L = \lambda \cdot (\kappa + \nu)$ nodes, and (2) a set Z of $M = \mu \cdot (\mu + \nu) - \lambda \cdot (\kappa + \nu)$ nodes. Besides, the edges in X, there is one edge from each node $z \in Z$ to one specific node $x \in X$. The edge probabilities of all edges are 1. We set $k := \kappa + 1$ and note that $M = \mu \cdot (\mu + \nu) - 1/\sqrt{\beta'} \cdot (\kappa + \nu) > 0$ by the definition of λ and the assumptions that $\mu > 1/\sqrt{\beta'}$.

We now show that there exists a set cover D' of size κ if and only if there is a β'-feasible solution of size k with strictly positive spread. For brevity, let us denote $B := (\kappa + \nu)/(\mu(\mu + \nu))$. (i) First assume that there is a set cover D' of size κ. Setting S to be the set of nodes corresponding to D' plus the node x achieves a spread of $\sigma(S) = (\kappa + \nu)(\lambda + 1) > 0$. To verify that S is β'-feasible we observe that $\sigma_{C_1}(S) = B/\lambda$ and $\sigma_{C_2}(S) = \lambda B$ and thus $\sigma_{C_2}(S) \geq \sigma_{C_1}(S) \geq \beta' \sigma_{C_2}(S)$. (ii) We now show the opposite direction: If there is a β'-feasible seed set S that has positive spread, it has to hold that $|S| \geq 1$. Then, by the fairness constraints and the fact that the communities are disconnected, the set S has to contain at least one node from C_2. This implies that $\sigma_{C_2}(S) \geq \lambda T$. By the β'-feasibility, we have that $\sigma_{C_1}(S) \geq \beta' \sigma_{C_2}(S) \geq \beta' \lambda B = B/\lambda$. This implies that there is a set of size at most $k - 1 = \kappa$ that covers at least $\kappa + \nu$ nodes in community C_1 and thus there is a set cover of size at most κ.

Now, assume that there exists a polynomial-time (α, β')-approximation algorithm A

for $\mathrm{IM}^{\mathrm{dp}}$. Then, if there exists a set cover of size κ, A will output a solution S such that $\sigma(S) \geq \alpha \cdot \mathrm{opt}^{\mathrm{dp}}_{\mathcal{S}}(G, \mathcal{C}, k) > 0$. Otherwise, A must return the only β'-feasible seed set $S = \emptyset$ with $\sigma(S) = 0$. Therefore, by using A we can decide in polynomial time whether or not there exists a set cover of size κ, and so no such algorithm can exist unless P = NP. □

We now consider the case where the fairness constraints are violated by an additive term. Using a similar reduction we show the following theorem.

Theorem 4.5. *For $\alpha \in (0, 1]$, $\varepsilon \in [0, 1)$, there is no $(\alpha, \varepsilon)^+$-approximation algorithm for* $\mathrm{IM}^{\mathrm{dp}}$*, unless* P = NP.

Proof. The proof is based on a reduction from the Set Cover problem similar to the one used in Theorem 4.4. Let ε' be the smallest value such that $\varepsilon < \varepsilon' < 1$ and $\varepsilon' \cdot (\mu^2 + \nu)$ is integer. We prove the stronger statement for ε' instead of ε. W.l.o.g. we assume that $\mu > 4/(1 - \varepsilon')$ and that $\nu \geq \kappa$. Moreover, we assume that $\mu \geq \nu/2$ since Set Cover remains NP-hard in this case (see, e.g., [39, Theorem 3.3]). We assume the IC model as underlying diffusion model. Consider the graph $G = (V, E, w)$ in Figure 4.2, where $N = \mu \cdot (\mu - 1)$, $L = |X| = \varepsilon' \cdot (\mu^2 + \nu) + \nu + k$, and $M = (1 - \varepsilon') \cdot (\mu^2 + \nu) - \nu - k$. In addition, for every node $v \in C_1$, there is an edge from v to the specific node x in X with probability one. We also set $k = \kappa$. Note that $M = (1 - \varepsilon') \cdot (\mu^2 + \nu) - \nu - k > 0$ by the assumptions that $\mu > 4/(1 - \varepsilon')$ and $\mu \geq \nu/2$. In fact, $(1 - \varepsilon') \cdot (\mu^2 + \nu) \geq (1 - \varepsilon') \cdot \mu^2 > (1 - \varepsilon') \cdot \frac{4\mu}{1 - \varepsilon'} > 2\nu > \nu + k$

We show that there exists a set cover D' of size κ if and only if there exists an $(\varepsilon')^+$-feasible solution S such that $\sigma(S) > 0$. For brevity, let $B := (k + \nu)/(\mu^2 + \nu)$. (i) If there exists a set cover D' of size $\kappa = k$. Then, we can construct an $(\varepsilon')^+$-feasible seed set S of size k by selecting the nodes corresponding to the subsets in D' and obtain $\sigma(S) = \varepsilon' \cdot (\mu^2 + \nu) + 2 \cdot (\nu + k) > 0$. The set S is $(\varepsilon')^+$-feasible since $\sigma_{C_1}(S) = B$ and $\sigma_{C_2}(S) = (\varepsilon' \cdot (\mu^2 + \nu) + \nu + k)/(\mu^2 + \nu) = \sigma_{C_1}(S) + \varepsilon'$. (ii) If there exists an $(\varepsilon')^+$-feasible seed set S such that $\sigma(S) > 0$, then we must have that $|S| > 1$. Since all the nodes in G reach the node $x \in X$ with probability 1 and from node x all nodes in X are reached with probability 1, we have that $\sigma_{C_2}(S) \geq B + \varepsilon'$. By the $(\varepsilon')^+$-feasibility of S, this bound on $\sigma_{C_2}(S)$ implies that $\sigma_{C_1}(S) \geq B$. Hence, there exists a set of seed nodes of size at most $k = \kappa$ in community C_1 that reaches at least $k + \nu$ nodes, thus there is a set cover of size at most κ.

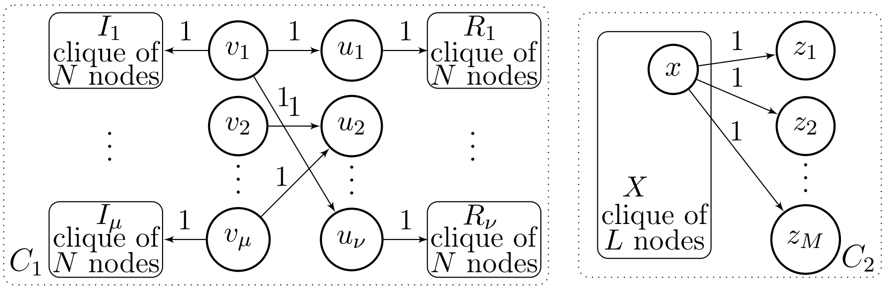


Figure 4.3: Construction of graph G in the instance of IM^{pp} from an instance of Set Cover.

Let us assume that there exists polynomial-time $(\alpha, \varepsilon')^+$-approximation algorithm A for IM^{dp}. If there exists a set cover of size κ, then A outputs an $(\varepsilon')^+$-feasible set S such that $\sigma(S) \geq \alpha \cdot \text{opt}_{\mathcal{S}}^{\text{dp}}(G, \mathcal{C}, k) > 0$. Otherwise, A outputs $S = \emptyset$ with $\sigma(S) = 0$. Hence, A can be used to solve the set cover problem in polynomial time, a contradiction to $\text{P} \neq \text{NP}$. □

4.2.2 Hardness of IM^{pp}

We now show that it is NP-hard to approximate IM^{pp} to within any bounded factor, even if the fairness constraint are violated by a multiplicative or an additive term. We first show the result for the multiplicative case.

Theorem 4.6. *For any $\alpha \in (0, 1]$, $\beta \in (0, 1]$, there exists no (α, β)-approximation algorithm for IM^{pp}, unless $\text{P} = \text{NP}$.*

Proof. We use Set Cover problem, where we are given a collection of subsets $D = \{D_1, \ldots, D_\mu\}$ over a ground set $U = \{U_1, \ldots, U_\nu\}$ and an integer κ, and we are asked whether there exists a collection D' of κ subsets that covers U.

Let β' be the largest $\beta' \leq \beta$ such that $1/\sqrt{\beta'}$ is integer. We show the stronger statement for β' instead of β. Given an instance of Set Cover, we define an instance of problem IM^{pp}. The graph $G = (V, E, w)$ in the instance of problem IM^{pp} is shown in Figure 4.3. W.l.o.g. we can assume that $\mu > 1/\sqrt{\beta'}$, and assume that we are using the IC model. Let $\lambda := 1/\sqrt{\beta'}$ be an integer that influences the size of G. Let $\mathcal{C}$ be the community structure consisting of two disjoint and disconnected communities C_1 and C_2. The first community C_1 consists of (1) one node v_j for each $D_j \in D$, (2) one node u_i for each $U_i \in U$, and (3) a set F of $\mu+\nu$ cliques of size $N = \lambda \cdot \mu - 1$, i.e., $F = \{I_1, \ldots, I_\mu, R_1, \ldots, R_\nu\}$.

There is an edge from v_j to u_i, whenever $U_i \in D_j$. For each v_j and u_i, there is an edge from v_j and u_i to one of the nodes in the cliques I_j and R_i, respectively. The second community C_2 consists of (1) a clique X of $L = \lambda \cdot \nu + \kappa$ nodes, and (2) a set of nodes Z of size $M = \mu \cdot (\kappa + \nu) - (\lambda \cdot \nu + \kappa)$. There is an edge from one specific node $x \in X$ to each node $z \in Z$. The edge probabilities of all edges are 1. Note that the set of nodes $u_1, \ldots, u_\nu$ in C_1 and $\lambda \cdot \nu$ nodes in X are targeted nodes and other nodes are non-targeted ones. We set $k := \kappa + 1$ and note that $M = \mu \cdot (\kappa + \nu) - (1/\sqrt{\beta'} \cdot \nu + \kappa) > 0$ by the definition of λ and the assumption that $\mu > 1/\sqrt{\beta'}$.

We now show that there exists a set cover D' of size κ if and only if there is a β'-feasible set S such that $\nu_T(S) > 0$. For brevity, let $B := \nu/(\mu(\kappa + \nu))$. (i) Assume that there is a set cover D' of size κ. Then, by selecting the nodes corresponding to the subsets in D' plus the node x we can construct a seed set S of size $k = \kappa + 1$ that achieves a spread of $\nu_T(S) = (\lambda + 1)\nu > 0$. To verify that S is β'-feasible we observe that $\sigma^T_{C_1}(S) = \frac{\nu_{C_1 \cap T}(S)}{\nu_{C_1}(S)} = B/\lambda$ and $\sigma^T_{C_2}(S) = \frac{\nu_{C_2 \cap T}(S)}{\nu_{C_2}(S)} = \lambda B$ and thus $\sigma^T_{C_2}(S) \geq \sigma^T_{C_1}(S) \geq \beta' \sigma^T_{C_2}(S)$. (ii) Now assume that if there is a β'-feasible seed set S that satisfies $\nu_T(S) > 0$, then we should have that $|S| \geq 1$. Since the communities are disjoint and disconnected, in order to satisfy the fairness constraints the set S has to contain at least one node from X. This implies that $\sigma^T_{C_2}(S) \geq \lambda B$. By the β'-feasibility, we have that $\sigma^T_{C_1}(S) \geq \beta' \sigma^T_{C_2}(S) \geq \beta' \lambda B = B/\lambda$. This implies that there is a set of size at most $k - 1 = \kappa$ that covers all targeted nodes $u_1, \ldots, u_\nu$ in community C_1 and thus there is a set cover of size at most κ.

Now assume that there is a polynomial-time (α, β')-approximation algorithm A for our problem. If there is a set cover of size κ, A will output a solution S such that $\nu_T(S) \geq \alpha \cdot \mathrm{opt}^{\mathrm{pp}}_{\mathcal{S}}(G, \mathcal{C}, k) > 0$. Otherwise, A returns the only β'-feasible seed set $S = \emptyset$ with $\nu_T(S) = 0$. Hence, A can be used to decide in polynomial time whether or not there exists a set cover of size κ, and so no such algorithm can exist unless P = NP. □

We proceed by proving the result for the additive case.

Theorem 4.7. *For any $\alpha \in (0, 1]$, $\varepsilon \in [0, 1)$, there exists no $(\alpha, \varepsilon)^+$-approximation algorithm for* $\mathrm{IM}^{\mathrm{PP}}$*, unless* P = NP*.*

Proof. The proof is based on a reduction from the Set Cover problem, where we are given a ground set $U = \{U_1, \ldots, U_\nu\}$, a collection of subsets $D = \{D_1, \ldots, D_\mu\}$ over U, and an integer κ, and we aim to determine whether there exists a subset $D' \subseteq D$ of size

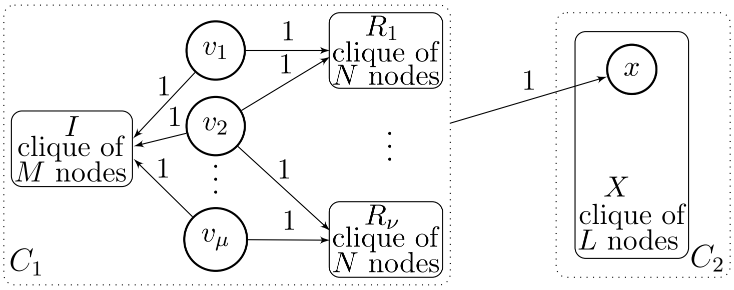


Figure 4.4: Construction of G from a Set Cover instance.

κ whose union is U. Let ε' be the smallest value such that $\varepsilon < \varepsilon' < 1$ and $\varepsilon' \cdot (\kappa + \mu)$ is integer. We prove the stronger statement for ε' instead of ε. Given an instance of Set Cover problem, we create an IM$^{\text{pp}}$ instance $G, \mathcal{C}, k$ as follows. W.l.o.g. we can assume that $\nu \geq 1/\varepsilon'$. Consider the graph $G = (V, E, w)$ in Figure 4.4, where there are two disjoint communities $\mathcal{C} = \{C_1, C_2\}$. Community C_1 consists of (1) one node v_j for each $D_j \in D$, (2) a clique R_i of $N = (1 - \varepsilon') \cdot (\kappa + \mu)$ nodes for each $U_i \in U$, and (3) a clique I of $M = \varepsilon' \cdot \nu(\kappa + \mu) - \kappa$ nodes. For each $U_i \in D_j$, there is an edge from v_j to the specific node r_i in R_i. There is also an edge from each node v_j to the specific node $w \in I$. Community C_2 consists of a clique X of $L = \nu(\kappa + \mu)$ nodes. For each node $v \in C_1$, there is an edge from v to the specific node $x \in X$. The edge probabilities of all edges are 1. We assume the IC model be the underlying diffusion model and set $k = \kappa$. Note that all nodes in $R_1, \ldots, R_\nu$ and all nodes in C_2 are targeted nodes, and other nodes are non-targeted ones. Also note that $M = \varepsilon' \cdot \nu(\kappa + \mu) - \kappa \geq \mu > 0$, by the choice of $\nu \geq 1/\varepsilon'$.

We show that there exists a set cover D' of size κ if and only if there exists an $(\varepsilon')^+$-feasible solution S such that $\nu_T(S) > 0$. (i) Assume that there exists a set cover D' of size $\kappa = k$. Then, by selecting the nodes corresponding to the subsets in D' we can construct a seed set S of size k and obtain $\nu_T(S) = \nu(2 - \varepsilon')(\kappa + \mu) > 0$. To show that the set S is $(\varepsilon')^+$-feasible, we observe that $\sigma^T_{C_1}(S) = \frac{\nu_{C_1 \cap T}(S)}{\nu_{C_1}(S)} = 1 - \varepsilon'$ and $\sigma^T_{C_2}(S) = \frac{\nu_{C_2 \cap T}(S)}{\nu_{C_2}(S)} = 1 = \sigma^T_{C_1}(S) + \varepsilon'$. (ii) Now assume that there exists an $(\varepsilon')^+$-feasible seed set S that satisfies $\nu_T(S) > 0$, then we must have that $|S| > 1$. Note that all the nodes in G can reach the node $x \in X$ and node x can reach all nodes in X. Since the probability on all edges is 1, then by selecting any node in G as a seed all nodes in X will be reached with probability 1. Thus, we have that $\sigma^T_{C_2}(S) = 1$. By the $(\varepsilon')^+$-feasibility of S, this bound on $\sigma^T_{C_2}(S)$ implies that $\sigma^T_{C_1}(S) \geq \sigma^T_{C_2}(S) - \varepsilon' = 1 - \varepsilon'$.

Hence, there exists a set of seed nodes of size at most $k = \kappa$ in community C_1 that reaches all targeted nodes in $R_1, \dots, R_\nu$, thus there is a set cover of size at most κ.

Let us assume that there exists a polynomial-time $(\alpha, \varepsilon')^+$-approximation algorithm A for our problem. If there exists a set cover of size κ, then A outputs a set S such that $\nu_T(S) \geq \alpha \cdot \mathrm{opt}^{\mathrm{pp}}_{\mathcal{S}}(G, \mathcal{C}, k) > 0$. Otherwise, A outputs $S = \emptyset$ with $\nu_T(S) = 0$. Hence, A would be able to decide in polynomial time, whether there exists a set cover of size κ, a contradiction to P $\neq$ NP. □

4.3 Fairness via Randomization

The above results imply that it is NP-hard to approximate $\mathrm{IM}^{\mathrm{eo}}$, $\mathrm{IM}^{\mathrm{dp}}$, and $\mathrm{IM}^{\mathrm{pp}}$ to within any bounded factor. In this section, we consider demographic parity notion and in addition to $\mathrm{IM}^{\mathrm{dp}}$, we define optimization problems that permit randomized strategies in the seed selection process rather than only deterministic ones, in an analogous way to what we did in Chapter 3 for the maximin criterion. We introduce two different probabilistic settings, a general one and one that chooses seed nodes independently.

In the first problem, $\mathrm{pIM}^{\mathrm{dp}}$, standing for probabilistic influence maximization under demographic parity, feasible solutions are distributions over node sets. Formally, we let $\mathcal{P} := \{p \in [0,1]^{2^V} : \mathbb{1}^T p = 1, \sum_{S \subseteq V} p_S |S| \leq k\}$ be the set of distributions over node sets of expected size at most k and denote by $S \sim p$ the random process of sampling S according to $p \in \mathcal{P}$. Now, the goal in $\mathrm{pIM}^{\mathrm{dp}}$ is to find the distribution $p \in \mathcal{P}$ that maximizes the expected number of reached nodes, while ensuring that perfect demographic parity is satisfied in expectation, i.e., that the expected probability to be reached is the same among all communities. Formally, $\mathrm{pIM}^{\mathrm{dp}}$ is defined as

$$\max_{p \in \mathcal{P}} \{\sigma(p) : \exists \gamma \text{ s.t. } \sigma_C(p) = \gamma \text{ for all } C \in \mathcal{C}\}, \tag{$\mathrm{pIM}^{\mathrm{dp}}$}$$

where we extend set functions to vectors in a straightforward way, i.e., for a set function f, we let $f(p) := \mathbb{E}_{S \sim p}[f(S)]$. For an instance $G, \mathcal{C}, k$, we use $\mathrm{opt}_{\mathcal{P}}(G, \mathcal{C}, k)$ to show the optimum of $\mathrm{pIM}^{\mathrm{dp}}$.

In the second probabilistic variant of $\mathrm{IM}^{\mathrm{dp}}$, we restrict to independent probability distributions, that is, in a feasible solution each node is selected as a seed independently with some probability in such a way that the expected size of the seed set is at most

k. Formally, we let $\mathcal{X} := \{x \in [0,1]^n : \mathbb{1}^T x \leq k\}$ and, for $x \in \mathcal{X}$, we denote with $S \sim x$ the process of randomly generating a set S from x, where each i is included in S independently with probability x_i. We then obtain independent probabilistic influence maximization under demographic parity problem iIM^{dp} as:

$$\max_{x \in \mathcal{X}} \{\sigma(x) : \exists \gamma \text{ s.t. } \sigma_C(x) = \gamma \text{ for all } C \in \mathcal{C}\}, \tag{iIM$^{\text{dp}}$}$$

where again for a set function f and a vector $x \in \mathcal{X}$, we let $f(x) := \mathbb{E}_{S \sim x}[f(S)]$. Again, for an instance $G, \mathcal{C}, k$, we denote with $\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k)$ the optimum of iIM^{dp}.

4.3.1 Relationship between IM^{dp}, pIM^{dp}, and iIM^{dp}

We first observe that clearly every feasible solution of IM^{dp} corresponds to a feasible solution of iIM^{dp} and pIM^{dp}, respectively. Furthermore, every feasible solution of iIM^{dp} directly corresponds to a feasible solution of pIM^{dp} via the following transformation: For $x \in \mathcal{X}$ define the vector p^x as $p^x_S := \prod_{i \in S} x_i \prod_{j \in V \setminus S} (1 - x_j)$, for $S \subseteq V$. Then, observe that $\sigma(x) = \sigma(p^x)$, $p^x \in \mathcal{P}$, and $\sigma_C(x) = \sigma_C(p^x)$, for any $C \in \mathcal{C}$. Hence, we obtain the following lemma. In the rest of this chapter, we use $\text{opt}_{\mathcal{S}}(G, \mathcal{C}, k)$ (other than $\text{opt}^{\text{dp}}_{\mathcal{S}}(G, \mathcal{C}, k)$) to show the optimum of IM^{dp}.

Lemma 4.8. *For every instance $G, \mathcal{C}, k$, it holds that*

$$\text{opt}_{\mathcal{S}}(G, \mathcal{C}, k) \leq \text{opt}_{\mathcal{X}}(G, \mathcal{C}, k) \leq \text{opt}_{\mathcal{P}}(G, \mathcal{C}, k).$$

A natural question is then whether a similar relation holds also in the other direction. We observe that this is not the case, $\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k)$ cannot be upper bounded in terms of $\text{opt}_{\mathcal{S}}(G, \mathcal{C}, k)$ multiplicatively and $\text{opt}_{\mathcal{P}}(G, \mathcal{C}, k)$ not in terms of $\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k)$. Formally:

Lemma 4.9. *Assume information spread to follow the IC model. There exist instances $G, \mathcal{C}, k$ s.t.*

$$(i)\ \frac{\text{opt}_{\mathcal{S}}(G, \mathcal{C}, k)}{\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k)} = 0, \text{ and } (ii)\ \frac{\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k)}{\text{opt}_{\mathcal{P}}(G, \mathcal{C}, k)} = 0$$

as well as (iii) $\text{opt}_{\mathcal{P}}(G, \mathcal{C}, k) - \text{opt}_{\mathcal{X}}(G, \mathcal{C}, k) = \Omega(n)$.

Proof. In order to prove *(i)*, consider the graph on the left in Figure 4.5 consisting of two nodes a and b that are connected by an edge with probability 3/4. Let $\mathcal{C}$ be the singleton community structure and $k = 1$. It is clear that a deterministic solution that

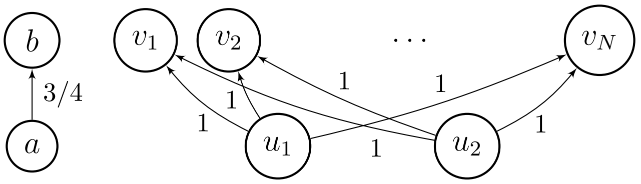


Figure 4.5: Instance showing that the optimum of pIM$^{\text{dp}}$ cannot be upper bounded in terms of iIM$^{\text{dp}}$.

chooses any seed cannot achieve demographic parity and thus $\text{opt}_{\mathcal{S}}(G, \mathcal{C}, k) = 0$. On the other hand, consider the solution $x \in \mathcal{X}$ for iIM$^{\text{dp}}$ defined by $x_a = 2/3$ and $x_b = 1/3$. It satisfies the demographic parity constraints, since $\sigma_a(x) = \sigma_b(x) = 2/3$, and achieves an overall expected coverage $\sigma(x)$ of $2 \cdot 2/3 = 4/3 > 0$ and thus $\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k) > 0$.

For *(ii)* consider the graph G in Figure 4.5 on the right consisting of two nodes $\{u_1, u_2\}$ and a set of N nodes $I = \{v_1, \ldots, v_N\}$. For each node u_i, there is an edge to all nodes in I with edge probability 1. Let $\mathcal{C}$ be the singleton community structure and $k = 1$. We first observe that a feasible solution for pIM$^{\text{dp}}$ is obtained by a distribution p that selects the set $\{u_1, u_2\}$ and the empty set both with probability 1/2, this solution p achieves an expected spread $\sigma(p)$ of $N/2+1$, thus $\text{opt}_{\mathcal{P}}(G, \mathcal{C}, k) \geq N/2+1 > 0$. Instead, we show that the only feasible solution $x \in \mathcal{X}$ for iIM$^{\text{dp}}$ is the zero solution, i.e., the solution x^0 with $x^0_v = 0$ for all $v \in \{u_1, u_2, v_1, \ldots, v_n\}$ and thus $\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k) = 0$. In order to show this, we first observe that u_1 and u_2 have no incoming edges and thus $\sigma_{u_j}(x) = x_{u_j}$ for any $x \in \mathcal{X}$ and $j \in [2]$. Moreover, due to the demographic parity constraints, we must have $x_{u_1} = x_{u_2}$. Let us call this value ρ and observe that $\rho \leq 1/2$ as $k = 1$. Now assume for the purpose of contradiction that $\rho > 0$. Then, for any $v \in I$, $\sigma_v(x) = x_v + (1 - x_v)(1 - (1 - \rho)^2)$ which is at least $1 - (1 - \rho)^2 = \rho(2 - \rho) > \rho$. As $\rho = \sigma_{u_1}(x)$, this contradicts the demographic parity constraints and thus $\rho = 0$. As a consequence $x_{v_i} = 0$ for all $i \in [N]$ due to the demographic parity constraints and thus $\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k) = 0$. This shows the first statement. Finally, $\text{opt}_{\mathcal{P}}(G, \mathcal{C}, k) - \text{opt}_{\mathcal{X}}(G, \mathcal{C}, k) \geq N/2 + 1 = \Omega(n)$. □

4.3.2 Price of Fairness

The price of (group) fairness is a measure of loss in efficiency due to fairness. More precisely, for $X \in \{\mathcal{S}, \mathcal{X}, \mathcal{P}\}$, we define $\text{PoF}_X(G, \mathcal{C}, k)$ as the ratio of the maximum coverage in the absence of fairness constraints, i.e., $\text{opt}(G, k)$ to the optima of the corresponding problem involving demographic parity fairness constraints, in other words,

$\mathrm{PoF}_X(G, \mathcal{C}, k) := \mathrm{opt}(G, k)/\,\mathrm{opt}_X(G, \mathcal{C}, k)$. Due to Lemma 4.8, we have the following relation $\mathrm{PoF}_{\mathcal{S}}(G, \mathcal{C}, k) \geq \mathrm{PoF}_{\mathcal{X}}(G, \mathcal{C}, k) \geq \mathrm{PoF}_{\mathcal{P}}(G, \mathcal{C}, k)$. We proceed by showing that the PoF can be unbounded for $\mathrm{pIM}^{\mathrm{dp}}$ and thus in all three cases.

Lemma 4.10. *Assume that information spread follows the IC model. For any even $n > 0$, there is an instance $G, \mathcal{C}, k$ s.t. $\mathrm{PoF}_X(G, \mathcal{C}, k) = \Omega(n)$ for $X \in \{\mathcal{S}, \mathcal{X}, \mathcal{P}\}$.*

Proof. In the light of the comment above it suffices to show the claim for $\mathrm{PoF}_{\mathcal{P}}$. Consider the graph G consisting of two disjoint sets I and J, each of size $n/2$. For one specific node $w \in J$, there is an edge from w to each node in I with probability 1. Let $\mathcal{C}$ be the singleton community structure and $k = 1$. Let us call p an optimal solution p for $\mathrm{pIM}^{\mathrm{dp}}$. Since nodes in J have no incoming edges, it holds that $\sigma_v(p) = \Pr_{S\sim p}[v \in S]$ for all $v \in J$. Let us call this value ρ. By the fairness constraints, it must also hold that $\sigma_v(p) = \rho$ for the nodes $v \in I$. As a result $\sigma(p) = n\rho$. Furthermore,

$$\frac{n}{2} \cdot \rho = \sum_{v\in J} \Pr_{S\sim p}[v \in S] = \sum_{v\in J} \sum_{S:v\in S} p_S = \sum_{S\subseteq V} p_S|S| \leq 1,$$

where the inequality holds because $p \in \mathcal{P}$. Hence, $\rho \leq 2/n$ and $\mathrm{opt}_{\mathcal{P}}(G, \mathcal{C}, k) = n\rho \leq 2$. On the other hand, $\mathrm{opt}(G, k) \geq \sigma(\{w\}) = n/2 + 1$ and thus $\mathrm{PoF}_{\mathcal{P}}(G, \mathcal{C}, k) \geq (n/2 + 1)/2 = \Omega(n)$. □

4.3.3 Hardness of $\mathrm{pIM}^{\mathrm{dp}}$ and $\mathrm{iIM}^{\mathrm{dp}}$

In the following, we show that the $\mathrm{pIM}^{\mathrm{dp}}$ problem is NP-hard and for the $\mathrm{iIM}^{\mathrm{dp}}$ problem we prove that it cannot be approximated to within a factor better than $1 - 1/e$.

Theorem 4.11. *The* $\mathrm{pIM}^{\mathrm{dp}}$ *problem is* NP*-hard.*

Proof. We reduce from the Set Cover problem, where we are given a collection of subsets $D = \{D_1, \ldots, D_\mu\}$ over a ground set $U = \{U_1, \ldots, U_\nu\}$ and an integer κ, and we are asked whether there exists a collection of κ subsets covering U. We can assume w.l.o.g. that every element from U appears in at least one subset from D as otherwise the instance is trivially false.

Given a Set Cover instance, we create a $\mathrm{pIM}^{\mathrm{dp}}$ instance $G, \mathcal{C}, k$ as follows. The graph $G = (V, E)$ has a node set $V = A \cup B$, where $A = \{v_1, \ldots, v_\mu\}$, $B = \{u_1, \ldots, u_\nu\}$ and there is a directed edge from v_j to u_i whenever $U_i \in D_j$ with probability 1. For an illustration see the construction of the bipartite graph on the left in Figure 4.2. The community structure $\mathcal{C}$ consists of only one community $C = V$, we set $k = \kappa$, and use the IC model. We proceed by showing that there exists a set cover of size κ if and only if there exists a fair solution $p \in \mathcal{P}$ with $\sigma(p) = k + \nu$. We note that the demographic parity fairness constraint is always fulfilled as there is a single community. (i) First, assume that there exists a set cover D' of size κ. Then we can construct a probability distribution $p \in \mathcal{P}$ by setting $p_S = 1$ for $S = \{v_i : D_i \in D'\}$ and 0 elsewhere. Clearly, $\sigma(p) = k + \nu$. (ii) Now assume that there is $p \in \mathcal{P}$ with $\sigma(p) = k + \nu$. Note that the expected spread restricted to A is no more than k as nodes in A have no incoming edges, formally $\sum_{v \in A} \sigma_v(p) = \sum_{v \in A} \sum_{S \subseteq V : v \in S} p_S = \sum_{S \subseteq V} |S \cap A| p_S \leq \sum_{S \subseteq V} |S| p_S \leq k$. Hence, from $\sigma(p) = k + \nu$, we conclude that $\sigma_u(p) = 1$ for all $u \in B$. Note however that $\sigma_u(p) = \sum_{S \subseteq V : R_u \cap S \neq \emptyset} p_S$, where $R_u = \{u\} \cup N_u$. As $\sum_{S \subseteq V} p_S = 1$, we conclude that $p_S = 0$ for all sets $S \subseteq V$ whenever $R_u \cap S = \emptyset$ for some $u \in B$. The contrapositive of the latter statement is that $p_S > 0$ implies $R_u \cap S \neq \emptyset$ for all $u \in B$. Since $\sum_{S \subseteq V} p_S \cdot |S| \leq k$, there is at least one set $S \subseteq V$, such that $|S| \leq k$ and $p_S > 0$. Hence, there is $S \subseteq V$ such that $|S| \leq k$ such that $R_u \cap S \neq \emptyset$ for all $u \in B$. If S contains a node from B, we can replace it with an arbitrary in-neighbor from A that has to exist by our assumption on the Set Cover instance. We obtain a set $S' \subseteq A$ of size at most k that reaches all nodes in B and the set $D' := \{D_i \in D : v_i \in S'\}$ is thus a set cover of size at most κ. □

For iIM$^{\text{dp}}$ we show an ever stronger result via a reduction from Max-Coverage: It cannot be approximated better than within $1 - 1/e$, unless P = NP.

Theorem 4.12. *There is no $(\alpha, 0)$-approximation algorithm for* iIM$^{\text{dp}}$ *for a constant $\alpha > 1 - 1/e$, unless* P = NP.

Proof. We reduce from the Max-Coverage problem, where given a collection of subsets $D = \{D_1, \ldots, D_\mu\}$ over a ground set $U = \{U_1, \ldots, U_\nu\}$ and an integer κ, the goal is to find a subset $D' \subseteq D$ of size at most κ that maximizes $|\bigcup_{S \in D'} S|$, the number of covered elements in U. We can assume w.l.o.g. that every element from U appears in at least one subset from D as otherwise also the optimum solution cannot cover it.

Given a Max-Coverage instance, we define an iIM$^{\text{dp}}$ instance $G, \mathcal{C}, k$ as follows. The directed graph $G = (V, E)$ consists of a node set $A = \{v_1, \ldots, v_\mu\}$ and a node set

$B = \cup_{i\in[\nu]} B_i$, where $B_i = \{u_i^1, \dots, u_i^{k\nu}\}$. There is an edge from v_j to u_i^ℓ, for all $\ell \in [\kappa\nu]$, whenever $U_i \in D_j$. The construction is similar to the one in Theorem 4.11 with the difference that every node in the set B is copied $\kappa\nu$ times. We adopt the IC model and set the probabilities of all edges to 1. The community structure $\mathcal{C}$ consists of only one community $C = V$ and we set $k = \kappa$. We proceed by showing the following claim: If there is a fair solution $x \in \mathcal{X}$, we can in polynomial time construct a set $S \subseteq A$ of size at most k with $\sigma(S) \geq \sigma(x)$ and furthermore $\sigma(S) = k + z \cdot k\nu$ for some $z \in \{0, \dots, \nu\}$. We note that we can write $\sigma(x) = \sum_{v\in V} \sigma_v(x) = \sum_{v\in V}(1 - \prod_{w\in R_v}(1 - x_w))$, where $R_v = \{v\} \cup N_v$. We now note that, for any $\varepsilon > 0$, the function σ satisfies the ε-convexity condition from Ageev and Sviridenko [1] and thus Pipage rounding can be used in order to, in polynomial time, construct a set $S \subseteq V$ of size at most k such that $\sigma(S) \geq \sigma(x)$. If S contains a node from B, we can replace it with an in-neighbor from A only increasing the overall coverage of S. Hence we get a set $S \subseteq A$ of size at most k with $\sigma(S) \geq \sigma(x)$ and clearly S reaches itself plus some $z \cdot k\nu$ nodes from B, thus $\sigma(S) = k + z \cdot k\nu$.

Now assume that we have an α-approximation algorithm for iIM$^{\text{dp}}$ with some $\alpha > 1 - 1/e$. For the given Max-Coverage instance, we then solve the constructed iIM$^{\text{dp}}$ instance, obtaining a fair solution $x \in \mathcal{X}$ such that $\sigma(x) \geq \alpha\cdot\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k)$. We can now, using the above claim, in polynomial time, construct a set $S \subseteq A$ with $\sigma(S) = k+z\cdot k\nu \geq \sigma(x) \geq \alpha \cdot \text{opt}_{\mathcal{X}}(G, \mathcal{C}, k)$ with some $z \in \{0, \dots, \nu\}$. Let now D^* be an optimal solution of size at most $\kappa = k$ of the Max-Coverage instance and let $S^* = \{v_i \in A : D_i \in D^*\}$ be the corresponding node set in A. Then, $\text{opt}_{\mathcal{X}}(G, \mathcal{C}, k) \geq \sigma(S^*) = k + \text{opt}\cdot k\nu$, where opt is the coverage of D^*. It follows that $z \geq \alpha \cdot \text{opt} - (1-\alpha)/\nu \geq (\alpha - 1/\nu)\cdot\text{opt}$, since $\text{opt} \geq 1 \geq 1-\alpha$. Recalling that $\alpha > 1-1/e$, for large enough ν also $\alpha-1/\nu > 1-1/e$ and thus we obtain an approximation algorithm for Max-Coverage with approximation factor bigger than $1 - 1/e$, which is impossible unless P = NP [37]. □

4.4 Algorithms for pIM$^{\text{dp}}$ and iIM$^{\text{dp}}$

We proceed with algorithms for pIM$^{\text{dp}}$ and iIM$^{\text{dp}}$. First note that it is not feasible to evaluate the functions σ and σ_C involved in the optimization problems exactly [74]. It is however well understood that the functions can be approximated by the functions $\tilde{\sigma}_C$ and $\tilde{\sigma}$ that are obtained by sampling a polynomial number of live-edge graphs, see, e.g., Lemma 3.12 in Chapter 3 and Proposition 2.5 in Chapter 2.

4.4.1 Approximation Algorithm for $\mathrm{iIM}^{\mathrm{dp}}$

We start by giving an approximation algorithm for $\mathrm{iIM}^{\mathrm{dp}}$. Given the above discussion, we consider $\tilde{\sigma}$ and $\tilde{\sigma}_C$ instead of σ and σ_C:

$$\max_{x\in\mathcal{X}}\{\tilde{\sigma}(x) : \exists\gamma \text{ s.t. } \tilde{\sigma}_C(x) = \gamma \text{ for all } C \in \mathcal{C}\}. \tag{apx-ipIM$^{\mathrm{dp}}$}$$

As discussed above, an (α, β)-approximation x for an instance $(G, \mathcal{C}, k)$ of $\text{apx-ipIM}^{\mathrm{dp}}$ approximates $\mathrm{iIM}^{\mathrm{dp}}$ by adding a multiplicative error in the objective and an additive error in the fairness violation, that is it satisfies $\sigma(x) \geq (\alpha - \varepsilon)\,\mathrm{opt}_{\mathcal{X}}(G, \mathcal{C}, k)$ and $\sigma_C(x) \geq \beta\sigma_{C'}(x) - \varepsilon$, for any arbitrary small $\varepsilon > 0$. We can thus focus on giving an approximation algorithm for $\text{apx-ipIM}^{\mathrm{dp}}$. Formally, we prove the following theorem.

Theorem 4.13. *There exists a* $(1-1/e, 1-1/e)$*-approximation algorithm for* $\text{apx-ipIM}^{\mathrm{dp}}$.

We first note that the objective function of $\text{apx-ipIM}^{\mathrm{dp}}$ is not linear, since the probability of sampling a seed set S from a distribution $x \in \mathcal{X}$ is $\prod_{i\in S} x_i \prod_{i\notin S}(1 - x_i)$. Our approach here is to approximate $\text{apx-ipIM}^{\mathrm{dp}}$ by a linear program (LP) of polynomial size. For a live-edge graph L and a node $v \in V$, we let $q_v(L, x)$ be the probability of sampling a set S that can reach v in live-edge graph L, that is $q_v(L, x) = \Pr_{S\sim x}[v \in \rho_L(S)]$. We can write $q_v(L, x) = 1 - \prod_{i\in V : v\in\rho_L(i)}(1 - x_i)$. It is easy to observe, see, e.g., Lemma 3.14 in Chapter 3, that $q_v(L, x)$ can be approximated within a constant factor by a function $p_v(L, x) := \min\{1, \sum_{i\in V : v\in\rho_L(i)} x_i\}$. More precisely,

$$q_v(L, x) \in [(1 - 1/e) \cdot p_v(L, x), p_v(L, x)]. \tag{4.1}$$

By defining $\lambda_v(x) := \frac{1}{T}\sum_{t=1}^{T} p_v(L_t, x)$, $\lambda(x) := \sum_{v\in V} \lambda_v(x)$, as well as $\lambda_C(x) := \frac{1}{|C|}\sum_{v\in C} \lambda_v(x)$ we obtain linear functions. Recalling that $\tilde{\sigma}_v(x) = \frac{1}{T}\sum_{t=1}^{T} q_v(L_t, x)$ together with the relation between p_v and q_v directly implies that λ_v, λ, and λ_C approximate σ_v, σ, and σ_C for all nodes $v \in V$ and communities $C \in \mathcal{C}$, respectively. Thus, we consider the following optimization problem

$$\max_{x\in\mathcal{X}}\{\lambda(x) : \exists\gamma \text{ s.t. } \lambda_C(x) = \gamma \text{ for all } C \in \mathcal{C}\}. \tag{pp$_\lambda$}$$

We then get the following lemma.

Lemma 4.14. *Let* $x \in \mathcal{X}$ *be an optimal solution to* pp_λ*, then* x *is a* $(1 - 1/e, 1 - 1/e)$*-approximation to* $\text{apx-ipIM}^{\mathrm{dp}}$.

Proof. Let x^* be optimal for apx-ipIM$^{\mathrm{dp}}$. Then

$$\begin{aligned}\tilde{\sigma}_v(x) &= \frac{1}{T}\sum_{t=1}^{T} q_v(L_t, x) \geq \big(1-\frac{1}{e}\big)\cdot\frac{1}{T}\sum_{t=1}^{T} p_v(L_t, x)\\ &\geq \big(1-\frac{1}{e}\big)\cdot\frac{1}{T}\sum_{t=1}^{T} p_v(L_t, x^*) \geq \big(1-\frac{1}{e}\big)\cdot\frac{1}{T}\sum_{t=1}^{T} q_v(L_t, x^*) = (1-\frac{1}{e})\cdot\tilde{\sigma}_v(x^*),\end{aligned}$$

where we used two times observation 4.1, and the optimality of x. We recall that $\tilde{\sigma}(x) = \sum_{v\in V}\tilde{\sigma}_v(x)$ for any x, thus, this shows the approximation on the objective function. For $C, C' \in \mathcal{C}$, we have

$$\tilde{\sigma}_C(x) \geq \big(1-\frac{1}{e}\big)\cdot\lambda_C(x) = \big(1-\frac{1}{e}\big)\cdot\lambda_{C'}(x) \geq \big(1-\frac{1}{e}\big)\cdot\tilde{\sigma}_{C'}(x)$$

again, using observation (4.1) twice as well as the feasibility of x. Similarly,

$$\tilde{\sigma}_C(x) \leq \lambda_C(x) = \lambda_{C'}(x) \leq \frac{e}{e-1}\cdot\tilde{\sigma}_{C'}(x),$$

and thus x is $(1-1/e)$-feasible for apx-ipIM$^{\mathrm{dp}}$. □

Note that the optimization problem $\max_{x\in\mathcal{X}}\{\lambda(x) : \exists\gamma \text{ s.t. } \lambda_C(x) = \gamma \text{ for all } C \in \mathcal{C}\}$ can be modeled as a linear program of polynomial size. The idea is to model the minimum in the definition of $p_v(L, x)$ by a variable y_{v,L_t}, for every $t \in [T]$.

Lemma 4.15. *The problem* pp_λ *can be solved in polynomial time using linear programming.*

Proof. The problem can be formulated as the following polynomial size linear program

$$\begin{aligned} \max \ & \sum_{v\in V}\sum_{t\in[T]} y_{v,L_t}\\ \text{s.t. } & \frac{1}{|C|}\sum_{v\in C}\frac{1}{T}\sum_{t\in[T]} y_{v,L_t} = \gamma \quad \forall C\in\mathcal{C}\\ & \sum_{i: v\in\rho_{L_t}(i)} x_i = y_{v,L_t} \quad \forall v\in V, t\in[T] \qquad (4.2)\\ & x\in\mathcal{X}, \gamma\in[0,1], y_{v,L_t}\in[0,1] \quad \forall v\in V, t\in[T].\end{aligned}$$

□

Lemmata 4.15 and 4.14 directly imply that there is a polynomial time $(1-1/e, 1-1/e)$-approximation for apx-ipIM$^{\mathrm{dp}}$ and thus also establishes Theorem 4.13. In our experimental study we refer to the described algorithm as **ind_lp**.

4.4.2 Algorithms for pIM$^{\mathrm{dp}}$

In this subsection, we present algorithms for pIM$^{\mathrm{dp}}$ that are based on greedy strategies and solving a (comparatively) small linear program. We again focus on algorithms for the problem with the approximate functions $\tilde{\sigma}$ and $\tilde{\sigma}_C$, formally

$$\max_{p \in \mathcal{P}} \{\tilde{\sigma}(p) : \exists \gamma \text{ s.t. } \tilde{\sigma}_C(p) = \gamma \text{ for all } C \in \mathcal{C}\}. \qquad (\text{apx-pIM}^{\mathrm{dp}})$$

Differently from the node-based problem, the objective function of apx-pIM$^{\mathrm{dp}}$ is linear and hence it can be formulated as a linear program by introducing a variable for each seed set $S \subseteq 2^V$. However, the size of such a linear program would be $\Theta(2^n)$, the dimension of $\mathcal{P}$. Our approach here is to restrict to a subset $\mathcal{Q} \subseteq \mathcal{P}$ in such a way that the linear program at hand becomes more tractable. More precisely, the two heuristics that we propose are based on solving the following linear program for two different choices of $\mathcal{Q}$

$$\max_{p \in \mathcal{Q}} \{\tilde{\sigma}(p) : \exists \gamma \text{ s.t. } \tilde{\sigma}_C(p) = \gamma \text{ for all } C \in \mathcal{C}\}. \qquad (\text{pp}_{\mathcal{Q}})$$

In the first heuristic, that we call **grdy_grp+lp**, we choose $\mathcal{Q}$ by restricting the set of non-zero variables to sets that either (1) have a large coverage with respect to a certain community, or (2) have a large overall coverage. Formally, we let $\mathcal{Q} := \{p \in \mathcal{P} : p_S = 0 \text{ for all } S \notin \mathcal{S}_1 \cup \mathcal{S}_2\}$, where $\mathcal{S}_1 = \{S_i : i \in [m]\}$ with $S_i = \arg\max_{S \in V}\{\tilde{\sigma}_{C_i}(S) : |S| \leq k\}$, $\mathcal{S}_2 := \{T_i : i \in \{0\} \cup [2k]\}$ with $T_i := \arg\max_{S \in \mathcal{S}}\{\tilde{\sigma}(S) : |S| \leq i\}$. Here the choice of $2k$ in the definition of $\mathcal{S}_2$ is more or less arbitrary, the rationale being that due to submodularity of σ it is unlikely that choosing a set of size twice the allowed expectation leads to a profitable gain in overall spread. Clearly, the idea behind this choice of $\mathcal{Q}$ is to provide the LP with sufficiently many degrees of freedom to both achieve a high overall coverage and a good coverage for each community.

In the second heuristic, that we refer to as **maximin+lp**, we define $\mathcal{Q} := \{p \in \mathcal{P} : p = \lambda_0 \cdot \mathbb{1}_\emptyset + \sum_{i \in [m]} \lambda_i \mathbb{1}_{S_i} + \lambda_{m+1} q\}$, where $\mathbb{1}_S$ is the 2^n-dimensional vector that is 1 at position $S \subseteq V$ and zero elsewhere, and $q \in \mathcal{P}$ is the distribution computed by the set-based algorithm in Chapter 3 for the maximin criterion. In other words, we restrict to probability distributions in $\mathcal{P}$ that are linear combinations of (1) a distribution computed for the maximin criterion and (2) the degenerate distributions of the empty set and the sets maximizing the respective community coverage. The rationale of this choice of $\mathcal{Q}$ is to profit from the efficiency of the maximin solution but enabling the LP solver to improve the incurred violation in demographic parity by putting additional probability on the deterministic distributions corresponding to under-represented communities.

4.5 Experiments

In this section, we report on a detailed experimental study. We evaluate a diverse set of algorithms for influence maximization in terms of their efficiency (both overall coverage and run-time) and demographic parity fairness. In our evaluation, we use random, synthetic, and real data sets.

Algorithms. In addition to **ind_lp**, **grdy_grp+lp**, and **maximin+lp**, our study includes the following competitors: **grdy_im** the greedy algorithm for IM, **grdy_maximin** the algorithm that greedily maximizes the minimum community coverage, **grdy_prop** a simple heuristic that greedily maximizes σ_{C_i} for $i \in [m]$ using $k|C_i|/n$ seeds, **milp** the MILP of Farnadi et al. [36], **moso** an algorithm based on multi-objective submodular optimization due to Tsang et al. [71], **set_based** the multiplicative weights routine for the set-based problem proposed in Chapter 3, **myopic** a simple heuristic by Fish et al. [38], and **uniform** the uniform solution to iIM^{dp}.

We proceed with a note on the **milp** algorithm by Farnadi et al. [36] that we use under their equity fairness notion (equivalent to demographic parity) relaxed by an additive 0.1 as they propose, we would like to remark the following. The mixed-integer linear program (MILP) that the authors solve is very similar to the LP that we propose in the proof of Lemma 4.15 with the main differences that the authors restrict the x-variables to be binary and require the constraint in (4.2) to hold with $\geq$ rather than equality. We stress that the y-variables in their MILP (called α in their paper) are not decision variables that indicate whether a node is covered anymore. More precisely, as a consequence of the fairness constraints, these variables may take any value between 0 and 1. As a result the seed set computed by **milp** may not satisfy the relaxed fairness constraints at all. We note that **grdy_maximin**, **set_based**, **moso**, and **myopic** were designed for the maximin criterion. We emphasize that **set_based**, **ind_lp**, **grdy_grp+lp**, **maximin+lp**, and **uniform** compute distributions and are thus designed for achieving ex-ante guarantees, while the other algorithms compute deterministic seed sets. For our algorithms from the previous section we relax the strict demographic parity constraints for some parameter $\eta \in [0, 1)$ as follows.

For **grdy_grp+lp** and **maximin+lp**, we simply replace γ in the demographic parity constraints in pp_Q by $\gamma \pm \eta$. We then choose $\eta \in \{0, x/16, x/8, x/4\}$, where x is the violation in demographic parity that the output of **grdy_im** suffers for **grdy_grp+lp** and **maximin+lp**. For the algorithm **ind_lp**, we substitute the constraints in (4.2) with $y_{v,L_t} \in [\sum_{i:v\in\rho_L(i)} x_i - \eta, \sum_{i:v\in\rho_L(i)} x_i]$ for each $v, t \in [T]$ and choose $\eta \in \{0, 1/4, 1/3, 1/2\}$.

Instances. We use random, synthetic and real world graphs. We refer to Table 3.1 in Chapter 3 for further details on the instances. We use the IC model with uniformly random weights in $[0, 0.4]$ for the random and synthetic networks and $[0, 0.2]$ for real world instances.

We consider the following different community structures. (1) Singleton communities. (2) Random communities: each node is assigned uniformly at random to a community. (3) BFS communities. (4) Random-overlap communities: for a given m, a node is, each with probability $1/(m + 2)$, (i) in community C_i for $i \in [m]$, (ii) in no community, or (iii) in all m communities. (5) Leidenalg communities: communities detected by a common algorithm for community detection [70]. (6) Given communities for the synthetic networks and for some of the real world instances.

Experimental Setting. As all evaluated algorithms are randomized, we repeat each run 10 times per graph, for random and synthetic graphs, we in addition average over 5 graphs, thus resulting in 50 runs per algorithm. In all our 2-dimensional plots, we also show averages of the projections onto each dimension together with 95% confidence intervals. For algorithms that output distributions rather than sets, i.e., giving ex-ante guarantees, we evaluate both their overall coverage and their demographic parity violation *in expectation.* We ran experiments with a large variety of parameter settings and, here we only report on a subset of the experiments performed. In our plots the overall (expected) coverage (as ratio of overall nodes) is on the vertical axis while the violation in demographic parity is on the horizontal axis. We see the averages and confidence intervals for overall coverage and violation in demographic parity as projected onto the right and top of the plot, respectively. We note that a perfect algorithm would achieve maximum overall coverage, while suffering zero violation in demographic parity, thus ending up in the top left of the plots.

For **grdy_im**, we use the TIM implementation by Tang et al. [68]. We implement **ind_lp**, **grdy_grp+lp**, and **maximin+lp** in C++, use the TIM implementation in order to compute the sets $\mathcal{S}_1$ and $\mathcal{S}_2$ and gurobi 9.5.0 [43] for solving the LPs. For **moso** we also choose gurobi as solver. For **grdy_prop**, if the resulting seed set is of size less than k (because of overlaps or due to rounding) the seed set is extended with nodes that maximize the total spread.

The tested algorithms are implemented in two different programming languages: **ind_lp**, **grdy_grp+lp**, **maximin+lp**, **grdy_im**, **grdy_prop**, **set_based** are implemented in C++ (compiled with g++ 7.5.0), while the algorithms **grdy_maximin**, **milp**, **moso**, **myopic**, **uniform** are implemented in Python (version 3.7.6). For consistency, the final evaluation of the computed solutions of all algorithms is still done in the same language (Python). For this final evaluation, we use a constant number of 100 live-edge graphs for simulating the diffusion process. We note that using a constant number of live-edge graphs is a frequent choice [36, 38], still, our algorithm's output is actually based on a larger number of live-edge graphs, 1000 in the case of **ind_lp**, and an even larger number for **grdy_grp+lp** and **maximin+lp**, namely as many as generated by the TIM implementation when computing $\mathcal{S}_1$ and $\mathcal{S}_2$. For the final evaluation of $\sigma_v(x)$ for $x \in \mathcal{X}$, we generate a number of sets $S \sim x$ sufficient to get an additive ε-approximation with probability at least $1 - \delta$, we use $\delta = \varepsilon = 0.1$.

4.5.1 Results

Running Times. We measure the running times of all algorithms on the random instances for increasing values of $n = 50, 100, 200$, see Table 4.1. We exclude **uniform** as it takes constant time and **milp** for $n > 50$ as it does not terminate in less than 30 mins. We observe that **grdy_im**, **ind_lp**, and **myopic** are fastest. As we will see, unfortunately, the fairness achieved by **grdy_im** and **myopic** is very poor. From the competitor algorithms, **grdy_maximin**, **milp** and **moso** perform the worst in terms of running times and as their fairness values are not too good either, we exclude them from

Algorithm	$n = 50$	$n = 100$	$n = 200$
grdy_grp+lp	3.20 ± 0.13	16.57 ± 0.61	116.64 ± 5.72
maximin+lp	6.54 ± 0.35	39.78 ± 1.02	232.90 ± 10.43
ind_lp	0.62 ± 0.04	1.05 ± 0.06	1.90 ± 0.07
grdy_im	0.07 ± 0.01	0.21 ± 0.01	0.74 ± 0.04
grdy_maximin	9.33 ± 0.26	54.80 ± 1.97	150.30 ± 9.03
grdy_prop	2.44 ± 0.09	16.08 ± 0.59	115.15 ± 5.45
milp	70.32 ± 4.61	–	–
moso	87.25 ± 3.81	138.77 ± 7.30	194.16 ± 12.59
set_based	3.35 ± 0.26	20.15 ± 0.48	97.23 ± 3.63
myopic	2.04 ± 0.13	4.12 ± 0.47	4.77 ± 0.74

Table 4.1: Running times on random instances ($k = 25$, singleton community structure) with 95% confidence intervals.

further experiments. We also exclude **ind_lp** as it is not performing too well in terms of fairness and coverage in comparison to our other two algorithms **grdy_grp+lp** and **maximin+lp**.

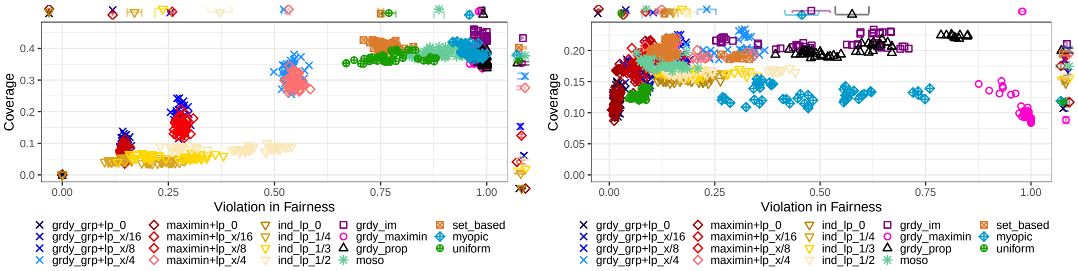


Figure 4.6: (left) Random instances ($k = 25$, $n = 200$, singleton communities), (right) synthetic instances ($k = 25$, $n = 500$, communities induced by gender and region).

Random and Synthetic Networks. We start with the random networks, see the left plot of Figure 4.6. We exclude **milp** from this and all further experiment as it does not solve a single instance in less than 30 mins. All competitor algorithms suffer a fairness violation of more than 0.75 and achieve a coverage between 0.35 and 0.45. In the case of **grdy_im**, there is a fairness violation of almost 1. Next, note that our algorithms that are restricted to find perfectly fair solutions, i.e., **grdy_grp+lp_0**, **maximin+lp_0**, and **ind_lp_0** obtain zero overall coverage. As we are in the setting of singleton communities, perfect demographic parity is a very strong requirement. Instead, if we use **grdy_grp+lp_x/4** (**maximin+lp_x/4**), where x is the violation of

grdy_im (here ≈ 1), we still achieve 75% (67%) of **grdy_im**'s coverage while suffering a fairness violation of only 0.5. More generally, **grdy_grp+lp** and **maximin+lp** allow for a trade-off between coverage and fairness. If the user is for example willing to tolerate only a fairness violation of around 0.25, he can use **grdy_grp+lp_x/8** (or **maximin+lp_x/8**) and would still achieve 41% (or 35%) of **grdy_im**'s coverage. Note that the algorithm **ind_lp** performs worse than **grdy_grp+lp** and **maximin+lp** in terms of coverage with similar fairness values.

For the synthetic data sets of Wilder et al. [76], see the right plot in Figure 4.6, we show results for the community structure induced by the attributes gender and region consisting of 15 communities of largely varying sizes. The best competitor algorithm in terms of fairness violation is **uniform** with a fairness violation of around 0.07, on the other hand it achieves a coverage of only around 0.13. The **moso** algorithm of Tsang et al. [71] achieves a fairness violation of around 0.13 while achieving a coverage of around 0.18. The **grdy_im** algorithm achieves the biggest coverage of around 0.21, but suffers a huge fairness violation of around 0.5. Here, our algorithms **grdy_grp+lp** and **maximin+lp** even achieve a decent overall coverage of 55% and 60% of **grdy_im**'s (comparable to, e.g., **moso**) when we restrict to no fairness violation at all (note that there is still a tiny violation in fairness as the final evaluation is done with an independent sample of live-edge graphs). Furthermore, when we allow a fairness violation of $x/16$, where x is the violation of **grdy_im**, our algorithms **grdy_grp+lp_x/16** and **maximin+lp_x/16** achieve a fairness violation of 0.08 and 0.07 with an overall coverage of 81% and 85% of **grdy_im**'s, respectively – thus strictly dominating over **grdy_maximin**, **moso** and **myopic**, while beating competitors in terms of fairness. We exclude **ind_lp** as it is not performing too well in terms of fairness and coverage in comparison to **grdy_grp+lp** and **maximin+lp** for further experiments.

Real World Instances. We turn to the real world instances.

Email Networks. We start with Arenas and email-Eu-core networks. We report on the results for Arenas network with different community structures: random, BFS, random-overlap, and leidenalg community structure. For the email-Eu-core network, we consider the community structure induced by the departments. See Figures 4.7 and 4.8 for some results on these networks. Our algorithms **grdy_grp+lp** and **maximin+lp** achieve the best demographic parity values by far. On the Arenas network, for example the plot in the bottom left of Figure 4.7, we achieve a violation in demographic parity of only 0.008, while getting more than 88% of **grdy_im**'s coverage that in turn suffers an around 5 times higher fairness violation. On the email-Eu-core network, our algorithm **maximin+lp_x/8** achieves a fairness violation around 0.2 (a quarter of **grdy_im**), while still achieving essentially the same coverage. We also note that all algorithms but **grdy_grp+lp**, **maximin+lp**, and **set_based** perform comparable to **uniform** in terms of both coverage and fairness on email-Eu-core.

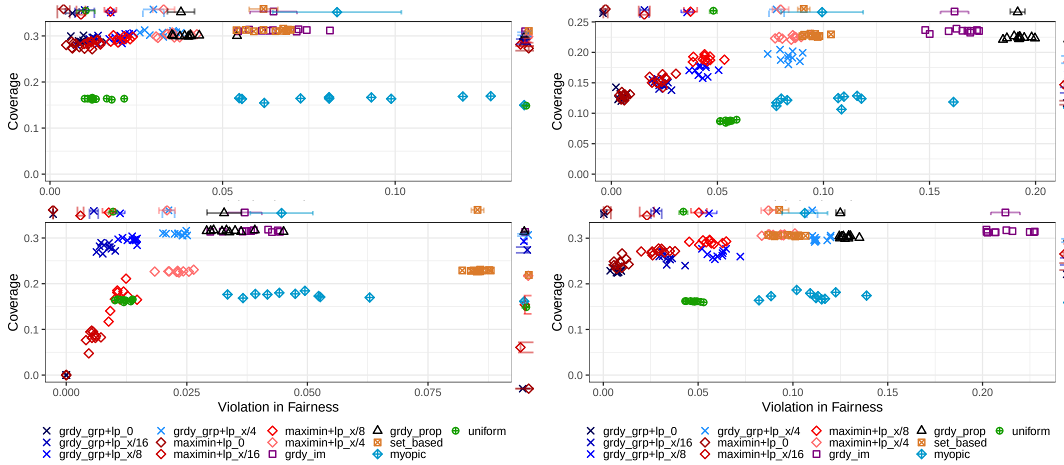


Figure 4.7: Arenas: (top left) Random communities, $m = 10$, $k = 100$. (top right) BFS communities, $m = 10$, $k = 50$. (bottom left) Random-overlap communities, $m = 10$, $k = 100$. (bottom right) Leidenalg communities, $k = 100$.

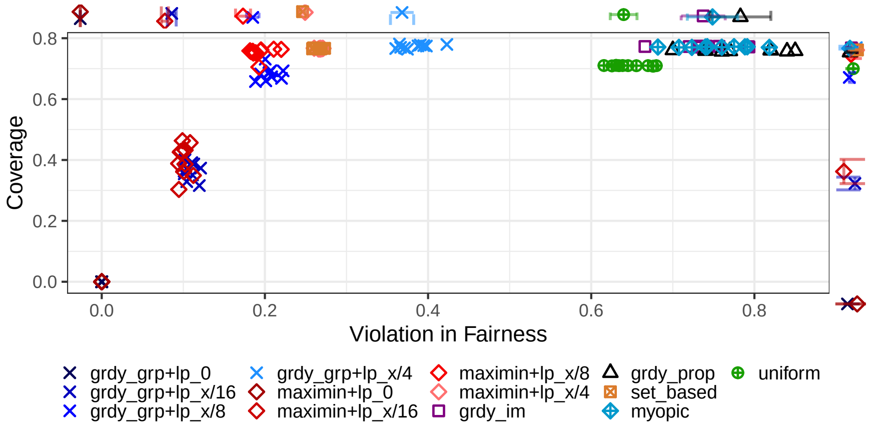


Figure 4.8: email-Eu-core with real communities, $k = 100$.

Irvine Network. For the results on Irvine network, see Figure 4.9. Again the demographic parity values achieved by **grdy_grp+lp** and **maximin+lp** are the best among all algorithms. Our **grdy_grp+lp_x/4** and **maximin+lp_x/4** reach almost the same coverage as **grdy_im**'s coverage on three of the plots (top left, bottom left, and bottom right plot). We note that all algorithms but **grdy_grp+lp**, **maximin+lp** perform comparable to **grdy_im** in terms of coverage. We also note that the simple heuristic **grdy_prop** and **set_based** perform even worse in terms of fairness than **grdy_im** in the plot on the top right and the plots on the top left and bottom left, respectively.

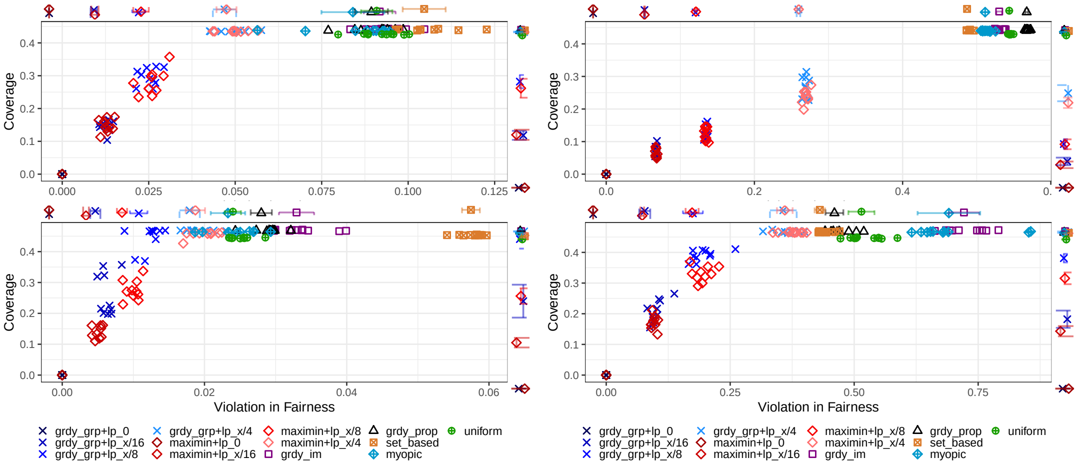


Figure 4.9: Irvine: (top left) Random communities, $m = 10$, $k = 50$. (top right) BFS communities, $m = 10$, $k = 50$. (bottom left) Random-overlap communities, $m = 10$, $k = 100$. (bottom right) Leidenalg communities, $k = 100$.

Co-Authorship Networks. For the co-authorship networks ca-GrQc and ca-HepTh, we report the results with different community structures in Figures 4.10 and 4.11. Due to running times we further restrict the evaluated algorithms by excluding also **maximin+lp** and **set_based**. Again **grdy_grp+lp** achieves the best fairness values by far. We again see a trade-off between fairness violation and overall coverage, i.e., in some cases no algorithm achieves low fairness violation while maintaining high coverage. Still in some other cases our algorithms achieve exactly that. On the ca-GrQc network with BFS community structure (the plot on the top right of Figure 4.10), for example, **grdy_grp+lp_0** has a fairness violation very close to 0, while getting more than 80% of **grdy_im**'s coverage.

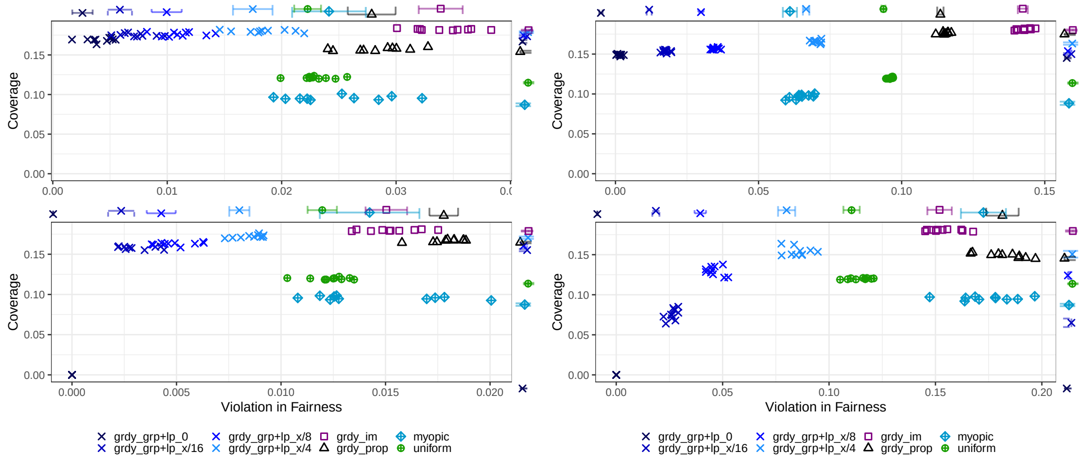


Figure 4.10: ca-GrQc: (top left) Random communities, $m = 10$, $k = 100$. (top right) BFS communities, $m = 2$, $k = 100$. (bottom left) Random-overlap communities, $m = 10$, $k = 100$. (bottom right) Leidenalg communities, $k = 100$.

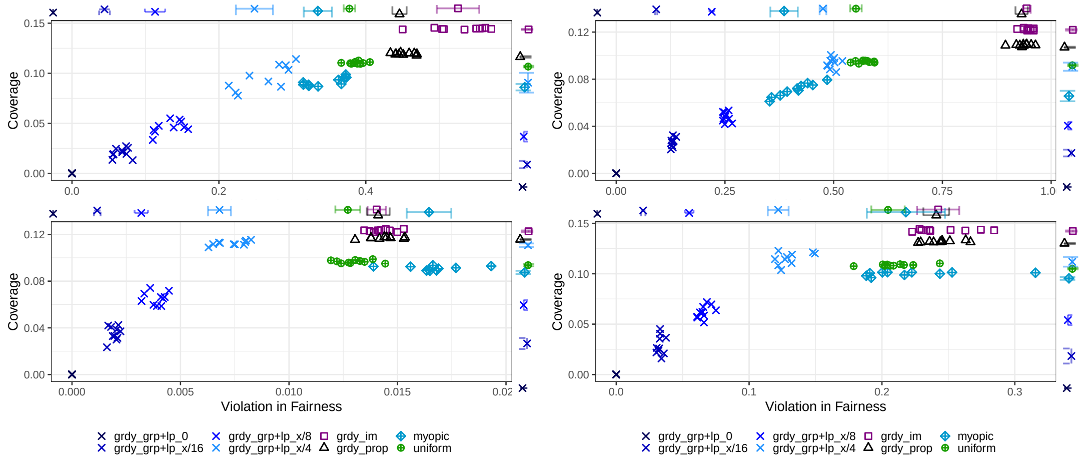


Figure 4.11: ca-HepTh: (top left) Random communities, $m = n/10$, $k = 100$. (top right) BFS communities, $m = n/10$, $k = 50$. (bottom left) Random-overlap communities, $m = 20$, $k = 50$. (bottom right) Leidenalg communities, $k = 100$.

Facebook Network. Lastly, we report on the results for the Facebook network. Again we observe that our **grdy_grp+lp** algorithm achieve the best fairness values. In the plot on the top right, for example, **grdy_grp+lp_x/16** obtains 55% of **grdy_im**'s coverage with only 7% of its fairness violation. Maybe even better, **grdy_grp+lp_x/8** obtains 99% of **grdy_im**'s coverage with only 23% of its fairness violation. We note that the fairness values and coverage achieved by algorithms other than **grdy_grp+lp** are comparable to each other.

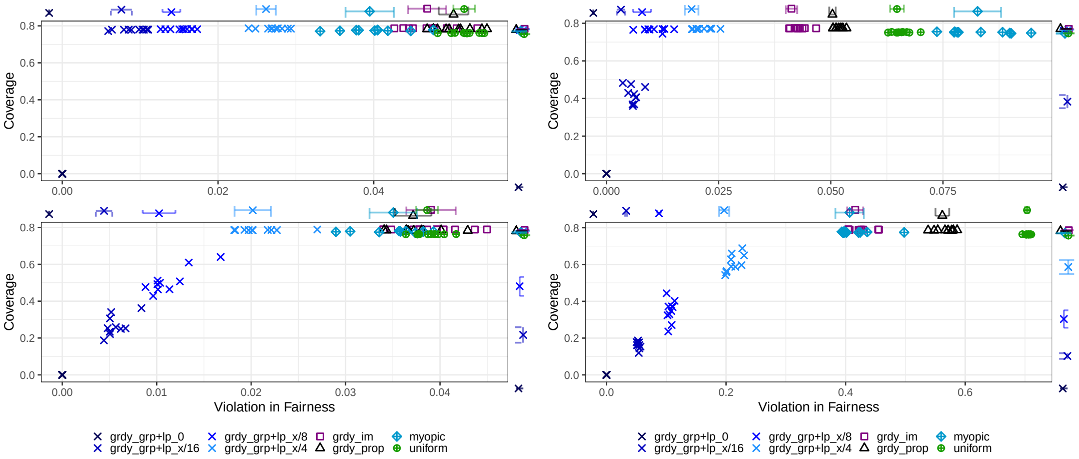


Figure 4.12: Facebook: (top left) Random communities, $m = 10$, $k = 100$. (top right) BFS communities, $m = 2$, $k = 50$. (bottom left) Random-overlap communities, $m = 20$, $k = 100$. (bottom right) Leidenalg communities, $k = 100$.

Chapter 5

Maximin Fairness by Adding Links

In this chapter, we investigate optimization problems with the goal of adding links to a social network to maximize the minimum community coverage when information is spread using a purely efficiency oriented seeding strategy.

Most works in the context of fairness in influence maximization study the question of how to find seeding strategies (deterministic or probabilistic) such that nodes or communities in the network get their fair share of coverage. In this chapter we take a different approach to fairness. We do not rely on the good will of the information spreading entity, but instead modify the underlying social network in such a way as to make efficiency-oriented information spreading automatically fair. The modification of the network may be done by the network owner or any other entity interested in guaranteeing fairness. While different ways of modifying the network are perceivable, we choose the possibly most natural one – we improve the network's connectivity by adding links. Here, we take the rather realistic approach to assume the information spreading entity to be indifferent rather than adversarial towards fairness.

5.1 The $\mathrm{FIM}_{\mathrm{AL}}$ Problem: Making Spread Maximizers Fair

5.1.1 Problem Definition

Consider a directed weighted graph $G = (V, E, w)$, two integers k and b, and a community structure $\mathcal{C}$. We use the *Independent Cascade model* for describing the random process of information diffusion. We let $\bar{E} := (V \times V) \setminus E$ denote the set of *non-edges* in G. For a set of non-edges $F \subseteq \bar{E}$ and a set of seed nodes $S \subseteq V$, we define $\sigma(S, F)$ as the expected number of nodes reached from S in the graph $G' = (V, E \cup F)$ that results from adding F to G. Similarly, for a node $v \in V$, $\sigma_v(S, F)$ is the probability that v is reached from S in G' and, for a community $C \subseteq V$, we define $\sigma_C(S, F)$ to be the average probability of nodes in C being reached from S in G'. Moreover, we define $\mathcal{M}(F, k) := \arg\max_{S \subseteq V}\{\sigma(S, F) : |S| \leq k\}$ to be the set of size k maximizers to $\sigma(\cdot, F)$. We are now ready to formally define the FIM_{AL} problem:

$$\max_{F \subseteq \bar{E}:|F| \leq b} \big\{\tau : \min_{C \in \mathcal{C}} \sigma_C(S, F) \geq \tau,\ \forall S \in \mathcal{M}(F, k)\big\}.$$

We denote with $\text{opt}_{\text{AL}}(G, \mathcal{C}, b, k)$ the optimum of FIM_{AL}. Clearly, our goal in FIM_{AL} is to find a set of at most b non-edges $F \subseteq \bar{E}$, that, when added to G, maximizes the minimum community coverage when information is spread in a purely "efficiency-oriented" way, i.e., from a set of at most k seed nodes that is chosen such that the set function $\sigma(\cdot, F)$ is maximized. The motivation behind studying FIM_{AL} is to, e.g., as the network owner, change the structure of a social network in such a way that an efficiency-oriented entity that wants to spread information in G automatically spreads information in a more fair way.

In what follows, we give several NP-hardness and hardness of approximation results for FIM_{AL}. We start by showing that the decision version of the general FIM_{AL} problem is Σ_2^p-hard. We even show that it is unlikely that FIM_{AL} can be approximated to within any factor. We then turn to special cases of FIM_{AL} where either $b = 1$ or $k = 1$ and show that the problem remains NP-hard also in these special cases – for $k = 1$ even hard to approximate to within any factor.

For better comprehensibility, we first note that in the the decision version of FIM_{AL}, in addition to the graph $G = (V, E)$, the communities $\mathcal{C}$, and the integers b, k, we are given a threshold t and the task is to decide if there exists $F \subseteq \bar{E}$ with $|F| \leq b$ such that for all $S \in \mathcal{M}(F, k)$: $\min_{C \in \mathcal{C}} \sigma_C(S, F) \geq t$.

Note that when the edge probabilities belong to $\{0, 1\}$, we refer to the instance as the *deterministic case*, in this case $\sigma(S)$ is the (deterministic) number of nodes reachable from seeds S in G.

5.1.2 Σ_2^p-Hardness

We start by recalling the definition of the complexity class Σ_2^p.

Definition 5.1 (Definition 5.1 in [6])**.** The class Σ_2^p is defined to be the set of all languages L for which there exists a polynomial-time Turing machine M and a polynomial q such that $x \in L$ if and only if $\exists u \in \{0,1\}^{q(|x|)} : \forall v \in \{0,1\}^{q(|x|)} : M(x,u,v) = 1$.[1]

We next introduce the $\Sigma_2\,$SAT problem which is Σ_2^p-complete, see, e.g., Exercise 1 in Chapter 5 of the book by Arora and Barak [6].

Definition 5.2 (Example 5.6 in [6])**.** Given a boolean expression $\phi(X,Y)$ in 3-CNF with variables $X = (x_1,\ldots,x_\nu)$ and $Y = (y_{\nu+1},\ldots,y_\mu)$, the $\Sigma_2\,$SAT problem entails to decide if $\exists x \forall y : \phi(x,y) = \top$, where $x : X \to \{0,1\}$ and $y : Y \to \{0,1\}$ are assignments to the variables X and Y, respectively.

For ease of presentation, we assume the indices of Y to start at $\nu+1$, such that indices of X and Y are disjoint. Our goal now is to show that the decision version of $\mathrm{FIM}_{\mathrm{AL}}$ is Σ_2^p-hard. We will describe a reduction from $\Sigma_2\,$SAT to the decision version of $\mathrm{FIM}_{\mathrm{AL}}$. We assume that $\phi(X,Y)$ contains m clauses $\phi_1,\ldots,\phi_m$ and for a clause ϕ_r we call $r(s)$, $s \in [3]$, the indices of the three variables corresponding to ϕ_r's three literals (in arbitrary fixed order).

Given an instance of $\Sigma_2\,$SAT, we create an instance $(G,\mathcal{C},b,k,t)$ of the decision version of $\mathrm{FIM}_{\mathrm{AL}}$ as follows, see Figure 5.1 for an illustration. Fix a constant $M := \mu + \nu + 6m + 1$. The node set V of G consists of

- $U = \{q, P\}$, where $P = p_1,\ldots,p_{M-1}$,
- $V^\exists = \{v_i, \bar{v}_i : i \in [\nu]\}$ and $V^\forall = \{v_j, \bar{v}_j, L_j : j \in [\mu] \setminus [\nu]\}$, where $L_j = l_{j,1},\ldots,l_{j,M-2}$, and
- $W = \{w_1^r, w_{\bar{1}}^r, w_2^r, w_{\bar{2}}^r, w_3^r, w_{\bar{3}}^r : r \in [m]\}$.

The edge set E consists of

[1] Equivalently, see, e.g., Theorem 5.12 and Remark 5.16 in the same book, Σ_2^p can be defined as the set of all languages that can be decided by a non-deterministic Turing machine with access to an oracle that solves some NP-complete problem.

- $E^{\text{var}} := \{(v_{r(s)}, w_s^r), (\bar{v}_{r(s)}, w_{\bar{s}}^r) : s \in [3], r \in [m]\}$,
- E^L that consists of all edges from the nodes v_j, $\bar{v}_j$ to all nodes $v \in L_j$, for $j \in [\mu] \setminus [\nu]$,
- E^P that consists of edges from q to all nodes in P, and
- $Z := V^2 \setminus (E^{\text{var}} \cup E^L \cup E^P \cup E(q, V^{\exists}))$, where $E(q, V^{\exists}) := \{(q, v) : v \in V^{\exists}\}$.

We note that as a result $\bar{E} = E(q, V^{\exists})$. The edge weight function is defined as $w_e = 0$ for all edges $e \in Z$ and $w_e = 1$ otherwise. The community structure $\mathcal{C}$ consists of: (1) communities $C_1, \ldots, C_m$, where each C_r is of cardinality 3 and for $s \in [3]$, $w_s^r \in C_r$ if $x_{r(s)} \in \phi_r$ (or $y_{r(s)} \in \phi_r$) and $w_{\bar{s}}^r \in C_r$ if $\bar{x}_{r(s)} \in \phi_r$ (or $\bar{y}_{r(s)} \in \phi_r$); and (2) communities $C_{m+1}, \ldots, C_{m+\nu}$, with $C_{m+i} = \{v_i, \bar{v}_i\}$ for each $i \in [\nu]$. We set $k = \mu + 1$, $b = \nu$ and $t = 1/3$.

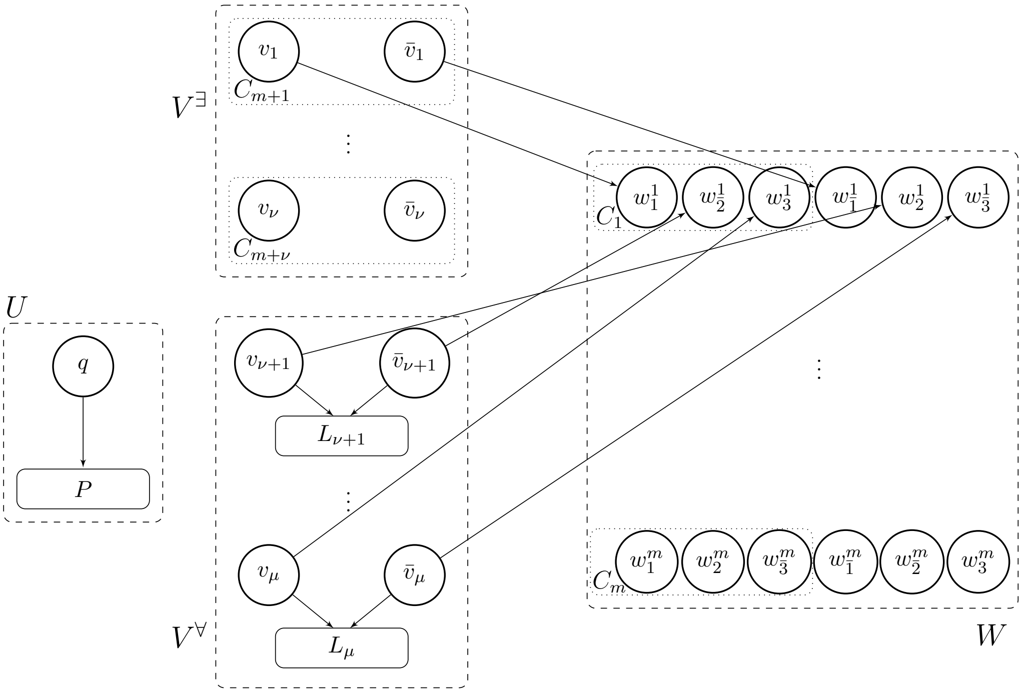


Figure 5.1: Construction of G from a Σ_2 SAT instance. Only the edges to the nodes corresponding to the first clause ϕ_1 are drawn. All drawn edges have weight 1. The only edges that are not in G are the ones from q to $V^{\exists}$.

Our goal is now to show that the Σ_2 SAT instance is a yes-instance if and only if the constructed FIM_{AL} instance is. We first need the following lemma.

Lemma 5.3. *Let $F \subseteq \bar{E} = E(q, V^{\exists})$ with $|F| \leq \nu$. It holds that $S \in \mathcal{M}(F, \mu + 1)$ if and only if $q \in S$ and $S \cap \{v_j, \bar{v}_j\} \neq \emptyset$ for all $j \in [\mu] \setminus [\nu]$.*

Proof. Fix a set F as in the statement of the lemma and let us call $P(S)$ for the property that $q \in S$ and $S \cap \{v_j, \bar{v}_j\} \neq \emptyset$ for all $j \in [\mu] \setminus [\nu]$. ($\Rightarrow$) First note that any set S that satisfies $P(S)$, achieves $\sigma(S, F) \geq M + (M-1)(\mu - \nu)$ and that a set T that does not satisfy $P(T)$ achieves $\sigma(T, F) \leq n - M$. Now, notice that $n = M + 2\nu + (\mu - \nu)M + 6m$ and thus $\sigma(T, F) \leq 2\nu + (\mu - \nu)M + 6m$. Using that $M = \mu + \nu + 6m + 1$, shows that $\sigma(S, F) > \sigma(T, F)$. This shows that T cannot be in $\mathcal{M}(F, k)$ and thus this completes the proof of this direction. ($\Leftarrow$) It is enough to show that all sets that satisfy property $P(S)$ achieve the same value $\sigma(S, F)$. From the construction of E^{var} it follows that the set W can be partitioned into $W^{\forall}$ and $W^{\exists}$ in a way that the nodes in $W^{\forall}$ have an in-edge from a node in $V^{\forall}$, while the nodes in $W^{\exists}$ have an in-edge from $V^{\exists}$. Now, let S be an arbitrary set satisfying property $P(S)$. It then follows that

$$\sigma(S, F) = \sigma(\{q\}, F) + \frac{|W^{\forall}|}{2} + (M-1)(\mu - \nu).$$

As the latter does not depend on S the proof is complete. □

We are now ready to prove the theorem.

Theorem 5.4. *The decision version of* FIM$_{\text{AL}}$ *is* Σ_2^p*-hard even in the deterministic case.*

Proof. We show that the Σ_2 SAT instance is a yes-instance if and only if the constructed FIM$_{\text{AL}}$ instance is.

($\Rightarrow$) Assume that the Σ_2 SAT instance is a yes-instance, i.e, there exists an assignments x to the variables X such that for all assignment y to the variables Y, it holds that $\phi(x, y) = \top$. We will now show that there exists $F \subseteq \bar{E}$ with $|F| \leq \nu$ such that for all $S \in \mathcal{M}(F, \mu + 1)$, it holds that $\min_{C \in \mathcal{C}} \sigma_C(S, F) \geq 1/3$. Let $F \subseteq \bar{E} = E(q, V^{\exists})$ be equal to the set of edges from q to $V^{\exists}$ that correspond to the assignment x. Now, let $S \in \mathcal{M}(F, \mu + 1)$ be arbitrary. It then follows using Lemma 5.3 that $S = \{q\} \cup \dot{S}$, where $\dot{S}$ corresponds to an assignment y of Y. As $\phi(x, y) = \top$ it follows that, for every clause ϕ_r at least one literal is true, thus for every community C_r with $r \in [m]$, at least one node $w \in C_r$ is reached and hence $\sigma_{C_r}(S, F) \geq 1/3$. For communities C_i with $i \in [m+1, m+\nu]$, we obtain that $\sigma_{C_i}(S, F) \geq 1/2$, as F corresponds to an assignment and S contains q according to Lemma 5.3.

($\Leftarrow$) Now, assume that the FIM_{AL} instance admits a solution $F \subseteq \bar{E}$ with $|F| \leq \nu$ such that for all $S \in \mathcal{M}(F, \mu+1)$, it holds that $\min_{C \in \mathcal{C}} \sigma_C(S, F) > 0$. Notice that $\sigma_C(S, F) > 0$ for every $S \in \mathcal{M}(F, \mu+1)$ together with Lemma 5.3 implies that F consists of a set of edges to $V^{\exists}$ that corresponds to an assignment. Let now y be an arbitrary assignment to Y and let S be the set containing q and all nodes from $V^{\forall}$ that correspond to the assignment y. Again using Lemma 5.3 it follows that $S \in \mathcal{M}(F, \mu+1)$ and thus $\sigma_{C_r}(S, F) > 0$ for all $r \in [m]$. This means that at least one node in every community C_i is reached or equivalently at least one literal in every clause ϕ_r is true in the assignments x and y. It follows that $\phi(x, y) = \top$. □

From the same reduction, we can even conclude that it is unlikely to find an arbitrary approximation to FIM_{AL} as shown in the next theorem. The class Δ_2^p is the class of all languages decided by polynomial-time Turing Machines that have access to an oracle for some NP-complete problem. It is widely believed that Σ_2^p and Δ_2^p are distinct (see Section 17.2 in [60]).

Theorem 5.5. *Let $\alpha \in (0, 1]$. If computing an α-approximation to* FIM_{AL} *is in Δ_2^p, then $\Sigma_2^p = \Delta_2^p$.*

Proof. Note that we have shown above that the Σ_2 SAT instance is a yes-instance if and only if the constructed FIM_{AL} instance admits a solution $F \subseteq \bar{E}$ with $|F| \leq \nu$ such that for all $S \in \mathcal{M}(F, \mu+1)$, it holds that $\min_{C \in \mathcal{C}} \sigma_C(S, F) > 0$. Note also that the FIM_{AL} instance there is deterministic.

Now, let $\alpha \in (0, 1]$ and assume that we have an algorithm computing an α-approximation to FIM_{AL} that runs in polynomial time when given access to an oracle for some NP-complete problem, i.e., computing an α-approximate solution to FIM_{AL} is in $\Delta_2^p = \text{P}^{\text{NP}}$. Given a Σ_2 SAT instance, we can then build the FIM_{AL} instance as described and compute an α-approximation to it. We then get a set $F \subseteq \bar{E}$ with $|F| \leq \nu$ such that for all $S \in \mathcal{M}(F, \mu+1)$, it holds that $\min_{C \in \mathcal{C}} \sigma_C(S, F) \geq \alpha \cdot \text{opt}_{\text{AL}}(G, \mathcal{C}, b, k)$. Therefore, the original Σ_2 SAT instance is a yes-instance if and only if $\min_{C \in \mathcal{C}} \sigma_C(S, F) > 0$, for all $S \in \mathcal{M}(F, \mu+1)$, and, if we can check this last condition, then we can decide whether the Σ_2 SAT instance is a yes-instance. We now show how to check this condition by using a polynomial number of calls to an oracle for some NP-complete problem.

We equivalently show how to check whether there exists a solution $S \in \mathcal{M}(F, \mu+1)$ such that $\min_{C \in \mathcal{C}} \sigma_C(S, F) = 0$. In deterministic instances, it is NP-complete to check whether there exists a seed set S such that $\sigma(S, F) \geq \tau$, for some parameter τ. We can then, using a polynomial number of calls to the oracle, find an $S \in \mathcal{M}(F, \mu + 1)$. In fact, since the instance is deterministic, it is enough to guess all $\tau \in [|V|]$. Let now $\tau^* = \sigma(S, F)$. Then we again use an oracle to solve the NP-complete problem of checking whether there exists a seed set S such that $\sigma(S, F) = \tau^*$ and $\min_{C \in \mathcal{C}} \sigma_C(S, F) = 0$. As the above algorithm overall requires a polynomial number of calls to the oracle, the proof is complete. □

5.1.3 Still Hard Special Cases

While we have shown above that the general problem is Σ_2^p-hard, we will now show that not even in the apparently simple case where $k = 1$, we can hope to find any approximation, unless P = NP.

Theorem 5.6. *For any, $\alpha \in (0, 1]$, it is* NP*-hard to approximate* FIM$_{\text{AL}}$ *to within a factor of α, even in the deterministic case and if $k = 1$.*

Proof. We reduce from Set Cover, where we are given a collection of sets $\mathcal{D} = \{D_1, \dots, D_\mu\}$ over a ground set $\mathcal{U} = \{U_1, \dots, U_\nu\}$ and an integer κ, and the task is to decide whether there exists a set cover of size at most κ, i.e., a collection $\mathcal{S} \subseteq \mathcal{D}$ with $|\mathcal{S}| \leq \kappa$ such that $\bigcup_{D \in \mathcal{S}} D = \mathcal{U}$.

Given a Set Cover instance, we create an instance $(G, \mathcal{C}, b, 1)$ of FIM$_{\text{AL}}$ as follows. The graph $G = (V, E, w)$ has node set $V := A \cup B \cup \{q\}$, where $A := \{v_1, \dots, v_\mu\}$ and $B := \{u_1, \dots, u_\nu\}$ and edge set $E := E^{sc} \cup Z$, where $E^{sc} := \{(v_j, u_i) : U_i \in D_j\}$ and $Z := V^2 \setminus (E^{sc} \cup E_{q,A})$, where $E_{q,A} := \{q\} \times A$. The edge-weight function w is defined as $w_e = 1$ for $e \in E^{sc} \cup E_{q,A}$ and $w_e = 0$ otherwise, i.e., for $e \in Z$. The communities $\mathcal{C}$ consist of $\nu + 1$ singletons $C_q = \{q\}$ and $C_i = \{u_i\}$ for $i \in [\nu]$. We set $b = \kappa$.

We now show that there exists a set cover $\mathcal{S}$ of size at most κ if and only if there exists a set of non-edges $F \subseteq \bar{E}$ with $|F| \leq b$, such that $\min_{C \in \mathcal{C}} \sigma_C(S, F) \geq 1$ for all $S \in \mathcal{M}(F, k)$: ("$\Rightarrow$") Assume that there exists a set cover $\mathcal{S}$ of size at most κ. Consider the set $F = \{(q, v_j) : D_j \in \mathcal{S}\}$ that is of cardinality at most $b = \kappa$. We now observe that $\mathcal{M}(F, k) = \{\{q\}\}$ and thus $\min_{C \in \mathcal{C}} \sigma_C(S, F) \geq 1$ for all $S \in \mathcal{M}(F, k)$ by the choice of F. ("$\Leftarrow$") Now assume that there exists a set $F \subseteq \bar{E}$ with $|F| \leq b = \kappa$ such that $\min_{C \in \mathcal{C}} \sigma_C(S, F) \geq 1$ for all $S \in \mathcal{M}(F, k)$. Note that $F \subseteq \bar{E} = E_{q,A}$ and thus again

$\mathcal{M}(F,k) = \{\{q\}\}$ and $\sigma(S,F) = \nu + \kappa + 1$ for all $S \in \mathcal{M}(F,k)$. Hence, it follows that $\{D_j : (q,v_j) \in F\}$ is a set cover of size at most κ.

Now, let $\alpha \in (0,1]$ and assume that there exists a polynomial time α-approximation algorithm $\mathcal{A}$ for FIM_{AL}. If there is a set cover of size κ, then $\text{opt}_{\text{AL}}(G,\mathcal{C},b,k) = 1$ and $\mathcal{A}$ outputs a set F such that $\min_{C\in\mathcal{C}} \sigma_C(S,F) \geq \alpha \cdot \text{opt}_{\text{AL}}(G,\mathcal{C},b,k) > 0$ for all sets $S \in \mathcal{M}(F,k)$. If however there is no set cover of size κ, then $\text{opt}_{\text{AL}}(G,\mathcal{C},b,k) < 1$ and as the instance is deterministic this means that $\text{opt}_{\text{AL}}(G,\mathcal{C},b,k) = 0$. Thus $\mathcal{A}$ must return a solution F such that $\sigma_C(S,F) = 0$ for some community $C \in \mathcal{C}$ and some set $S \in \mathcal{M}(F,k)$. Therefore, by using $\mathcal{A}$ we can decide in polynomial time whether or not there exists a set cover of size κ by running $\mathcal{A}$ and then checking if there exists a community $C \in \mathcal{C}$ and a set $S \in \mathcal{M}(F,k)$ such that $\sigma_C(S,F) = 0$. Note that we can compute $\mathcal{M}(F,k)$ in polynomial time by evaluation of all different n choices – recall that $k = 1$. It follows that it is NP-hard to approximate FIM_{AL} to within a factor of α. □

A natural next question is whether the problem remains hard also if $b = 1$. We show that this is the case:

Theorem 5.7. *The decision version of* FIM_{AL} *is* NP-*hard even in the deterministic case and if* $b = 1$.

Proof. We reduce from Set Cover, where we are given a collection of sets $\mathcal{D} = \{D_1,\ldots,D_\mu\}$ over a ground set $\mathcal{U} = \{U_1,\ldots,U_\nu\}$ and an integer κ, and the task is to decide whether there exists a set cover of size at most κ, i.e., a collection $\mathcal{S} \subseteq \mathcal{D}$ with $|\mathcal{S}| \leq \kappa$ such that $\bigcup_{D\in\mathcal{S}} D = \mathcal{U}$. W.l.o.g., we can assume that every U_i appears in at least one set D_j as otherwise the instance is trivially a no-instance.

Given a Set Cover instance, we create an instance $(G,\mathcal{C},1,k,t)$ of the decision version of FIM_{AL} as follows (here t denotes the threshold to be reached). The graph $G = (V,E,w)$ has node set $V := A \cup B \cup \{q\}$, where $A := \{v_1,\ldots,v_\mu\}$ and $B := \{u_1,\ldots,u_\nu\}$ and edge set $E := E^{sc} \cup Z$, where $E^{sc} := \{(v_j,u_i) : U_i \in D_j\}$ and $Z = V^2 \setminus (E^{sc} \cup E_{B,q})$ with $E_{B,q} := B \times \{q\}$. The edge-weight function w is defined as $w_e = 1$ for $e \in E^{sc} \cup E_{B,q}$ and $w_e = 0$ otherwise, i.e., for $e \in Z$. The community structure $\mathcal{C}$ consists of $\nu + 1$ singleton communities $C_q = \{q\}$ and $C_i = \{u_i\}$ for every $i \in [\nu]$. We set $k = \kappa$ and $t = 1$.

We now show that the set cover instance is a yes-instance if and only if the FIM_{AL} instance is, i.e., if there exists a set of non-edges $F \subseteq \bar{E}$ with $|F| \leq b$, such that $\min_{C \in \mathcal{C}} \sigma_C(S, F) \geq 1$ for all $S \in \mathcal{M}(F, k)$: ("$\Rightarrow$") Assume that there is a set cover $\mathcal{S}$ of size at most κ. Let $F = \{(u, q)\}$ for some arbitrary node $u \in B$. Then $S = \{v_j : D_j \in \mathcal{S}\}$ achieves $\sigma(S, F) = \nu + \kappa + 1$. Note that nodes in A have no ingoing edges with positive probability and thus no set that is not a subset of A can achieve a higher coverage than S thus $\mathcal{M}(F, k) = \{S\}$. As a consequence $\min_{C \in \mathcal{C}} \sigma_C(S_{F,k}, F) \geq 1$ for all $S \in \mathcal{M}(F, k)$. ("$\Leftarrow$") Now assume that there exists a set $F \subseteq \bar{E}$ with $|F| \leq b = 1$, such that $\min_{C \in \mathcal{C}} \sigma_C(S, F) \geq 1$ for all $S \in \mathcal{M}(F, k)$. Note that $F \subseteq \bar{E} = E_{B,q}$ and thus from $\min_{C \in \mathcal{C}} \sigma_C(S, F) \geq 1$, it follows that $\sigma_C(S, \emptyset) \geq 1$ for every $C = \{u_i\}$ and $S \in \mathcal{M}(F, k)$. By the assumption on the Set Cover instance, the set S can be transformed into a subset S' of A such that still $\sigma_C(S', \emptyset) \geq 1$ for every $C = \{u_i\}$. We can thus conclude that $\{D_i : v_i \in S'\}$ is a set cover of size at most κ. □

5.2 The $\text{FIM}^{\text{g}}_{\text{AL}}$ Problem: Towards Fairness in Practice

5.2.1 Problem Definition

We have seen a lot of evidence above that FIM_{AL} is intractable. We thus continue by proposing an alternative problem that not only turns out to be more computationally tractable, but also is possibly practically better motivated in the first place in the following sense: The problem of finding a set of at most k nodes that maximizes $\sigma(\cdot, F)$ is however an NP-hard optimization problem and thus it is unrealistic to assume the entity to spread information using a maximizing set. Instead what is frequently used in practice for the computation of an efficient seed set is the greedy algorithm. In fact, the choice of the greedy algorithm is also well-founded in theory, as, for a fixed set of non-edges F, the set function $\sigma(\cdot, F)$ is monotone and submodular and thus one is guaranteed to achieve an essentially optimal approximation factor of $1 - 1/e - \varepsilon$ for any $\varepsilon > 0$, see Theorem 2.6 in Chapter 2. Hence, an optimization problem that is practically better motivated than FIM_{AL}, assumes that the efficiency-oriented entity, in order to spread information, uses the greedy algorithm for computing the seed set. The greedy algorithm for $\sigma(\cdot, F)$ is however a randomized algorithm, as it relies on simulating information spread using a polynomial number of live-edge graphs (or reverse reachable (RR) sets, depending on the implementation). It becomes thus necessary that we consider the output of the algorithm to be a distribution over seed sets of size k, rather than just a single set. For a set of non-edges $F \subseteq \bar{E}$ and an integer k, let us denote this distribution with $p(F, k)$. We then define the $\mathrm{FIM}_{\mathrm{AL}}^{\mathrm{g}}$ problem as:

$$\max_{F \subseteq \bar{E}:|F| \leq b} \big\{\tau : \mathbb{E}_{S \sim p(F,k)}[\sigma_C(S, F)] \geq \tau \; \forall C \in \mathcal{C}\big\}.$$

Intuitively, our goal in the optimization problem $\mathrm{FIM}_{\mathrm{AL}}^{\mathrm{g}}$ is to find a set of at most b non-edges $F \subseteq \bar{E}$, that, when added to G, maximizes the minimum community coverage (in expectation) when information is spread using the greedy algorithm – a quite realistic assumption.

Here, we do not assume to have access to $p(F, k)$, not even for one set F, as it would generally require exponential space to be encoded. Instead, we assume to have access to the greedy algorithm in an oracle fashion, i.e., for a given set F, we can call the greedy algorithm on $\sigma(\cdot, F)$ with budget k and get a set S. One can then show using an easy Hoeffding bound argument, see below, that $\mathbb{E}_{S \sim p(F,k)}[\sigma_C(S, F)]$ can be approximated arbitrarily well w.h.p. for every F.

It is also worth mentioning that our approach can be extended to a setting where we want to be fair w.r.t. multiple implementations of the greedy algorithm or even more generally to multiple implementations of multiple algorithms (different from the greedy algorithm). This can be achieved as follows. Assume that $(p_i)_{i \in [N]}$ are a priori-likelihoods of using one of N different algorithms and assume $p^i(F, k)$ to reflect the probability distribution of seed sets corresponding to algorithm i. Then the distribution with $p_S(F, k) := \sum_i p_i \cdot p_S^i(F, k)$ for $S \subseteq V$ reflects the distribution over seed sets resulting from using all N algorithms. The only condition here, for our algorithmic results below to keep working, is that the algorithms are polynomial time.

5.2.2 Polynomiality of Deterministic Case with Constant b

We now first observe that in the deterministic case with constant b, it is simple to solve the problem exactly in polynomial time, simply by going through all at most $\binom{n^2-m}{b} \leq n^{2b}$ possible sets of non-edges F, computing the deterministic set S_F that the greedy algorithm outputs for maximizing $\sigma(\cdot, F)$, and checking what is the value $\tau_F = \min_{C \in \mathcal{C}} \sigma_C(S_F, F)$. Then return the set F that achieves the maximum τ_F. Although this seems trivial, we notice that such an approach cannot work for $\mathrm{FIM}_{\mathrm{AL}}$, for which we showed that the problem remains NP-hard in the deterministic case even if $b = 1$, see Theorem 5.7.

Observation 5.8. There is a polynomial time algorithm to compute an optimal solution to $\text{FIM}^{\text{g}}_{\text{AL}}$ in the deterministic case when b is constant.

5.2.3 Hardness

In the language of parameterized complexity, Observation 5.8 shows that the deterministic $\text{FIM}^{\text{g}}_{\text{AL}}$ problem belongs to the class XP when parameterized by b. A natural question is therefore whether there exists an FPT algorithm that solves or approximates $\text{FIM}^{\text{g}}_{\text{AL}}$ in deterministic instances. In fact, already Theorem 5.6 answers negatively to this question as the proof shows a polynomial-time reduction from the Set Cover problem to the deterministic case of $\text{FIM}^{\text{g}}_{\text{AL}}$ in which b is equal to the size of a set cover κ. As Set Cover is W[2]-hard w.r.t. κ, $\text{FIM}^{\text{g}}_{\text{AL}}$ does not admit an FPT algorithm w.r.t. b, even in the deterministic case, unless W[2] = FPT. Moreover, under the same condition, no parameterized α-approximation algorithm exists since the optimum of a $\text{FIM}^{\text{g}}_{\text{AL}}$ instance is strictly positive if and only if there exists a set cover of size κ.

A natural next question is what happens for general b, but with $k = 1$. The problem remains hard in this case. Consider the instance constructed in the reduction in Theorem 5.6. As $k = 1$ and as the instance is deterministic, it is clear that the greedy algorithm, for any set $F \subseteq \bar{E}$ of non-edges, simply computes a maximizing set of cardinality 1. Hence the following statement can be shown in the same way as in the proof of Theorem 5.6: there exists a set cover $\mathcal{S}$ of size at most κ if and only if there exists a set of non-edges $F \subseteq \bar{E}$ with $|F| \leq b$, such that $\min_{C \in \mathcal{C}} \mathbb{E}_{S \sim p(F,k)} \sigma_C(S, F)] \geq 1$. This yields the following corollary to Theorem 5.6.

Corollary 5.9. *For any $\alpha \in (0, 1]$, it is* NP*-hard to approximate the* $\text{FIM}^{\text{g}}_{\text{AL}}$ *problem to within a factor of α, even in the deterministic case and if $k = 1$.*

As mentioned above, we will see below that $\text{FIM}^{\text{g}}_{\text{AL}}$ for general constant b turns out to be arbitrarily well approximable. To prove this, we first turn back to the question of approximating $\mathbb{E}_{S \sim p(F,k)}[\sigma_C(S, F)]$ for a fixed F.

5.2.4 Approximating $p(F,k)$

As mentioned above, we do not assume access to $p(F,k)$, instead we show that, using the greedy algorithm in an oracle fashion, we can approximate $\mathbb{E}_{S\sim p(F,k)}[\sigma_C(S,F)]$ arbitrarily well using a Hoeffding bound. We first recall that already σ_C cannot be evaluated exactly but has to be approximated using $\text{poly}(n,\varepsilon^{-1})$ many samples of live-edge graphs.

Lemma 5.10. *Given an instance $(G,\mathcal{C},b,k)$ of* $\text{FIM}^{\text{g}}_{\text{AL}}$ *with constant b, one can in* $\text{poly}(n,m,\varepsilon^{-1})$ *time, compute functions f_C such that, $|f_C(F)-\mathbb{E}_{S\sim p(F,k)}[\sigma_C(S,F)]| \leq \varepsilon$ for all $C \in \mathcal{C}$ and $F \subseteq \bar{E}$ with $|F| \leq b$ w.h.p. Here $m = |C|$.*

Proof. We assume to have access to approximations $\tilde{\sigma}_C$ of σ_C for all $C \in \mathcal{C}$ such that, w.h.p., $|\sigma_C(S,F) - \tilde{\sigma}_C(S,F)| \leq \varepsilon/2$ for all $C \in \mathcal{C}$, $S \subseteq V$, and $F \subseteq \bar{E}$ with $|F| \leq b$. Such approximations can, e.g., be computed as in Lemma 3.12 of Chapter 3. Concluding from the bound on T there, this can be done in $\text{poly}(n,m,\varepsilon^{-1})$ time. We can now, for every $F \subseteq \bar{E}$ with $|F| \leq b$, call the greedy algorithm $N = \Omega(\varepsilon^{-2}\log(nm))$ times and obtain sets $S_1,\dots,S_N$ of size k. For every $C \in \mathcal{C}$, define $f_C(F) := \frac{1}{N}\sum_{i=1}^N \tilde{\sigma}_C(S_i,F)$ and $\bar{f}_C(F) := \frac{1}{N}\sum_{i=1}^N \sigma_C(S_i,F)$. Then using a Hoeffding bound, see, e.g., Theorem 4.12 in the book by Mitzenmacher and Upfal [56], it holds that $\Pr[|\bar{f}_C(F) - \mathbb{E}_{S\sim p(F,k)}[\sigma_C(S,F)]| \geq \varepsilon/2] \leq (nm)^{-\Omega(1)}$. After applying a union bound, we obtain that w.h.p., we have $|\bar{f}_C(F) - \mathbb{E}_{S\sim p(F,k)}[\sigma_C(S,F)]| \leq \varepsilon/2$ for all $C \in \mathcal{C}$ and $F \subseteq \bar{E}$ with $|F| \leq b$. Hence, w.h.p.,

$$\begin{aligned}\left|f_C(F) - \mathbb{E}_{S\sim p(F,k)}[\sigma_C(S,F)]\right| &\leq \left|\bar{f}_C(F) - \mathbb{E}_{S\sim p(F,k)}[\sigma_C(S,F)]\right| \\ &\quad + \left|f_C(F) - \bar{f}_C(F)\right| \leq \varepsilon.\end{aligned}$$ □

5.2.5 General Approximation for Constant b

The above lemma enables us to provide a polynomial time algorithm for $\text{FIM}^{\text{g}}_{\text{AL}}$ when b is constant that finds a set $F \subseteq \bar{E}$ that is ε-close to optimal (in an additive sense) w.h.p. After proving the above lemma, the idea is simple and similar to the deterministic case: Again, go through all at most n^{2b} possible sets of non-edges F, compute $\varepsilon/2$-approximations $(f_C(F))_{C\in\mathcal{C}}$ as in Lemma 5.10, and return the set with maximum value $\tau_F = \min_{C\in\mathcal{C}} f_C(F)$. This set is an additive ε-approximation of the maximizing set F^* (using the approximation guarantee once for F and once for F^*).

Lemma 5.11. *Let $\varepsilon \in (0,1)$, there is a polynomial time algorithm to compute an additive ε-approximation to the optimal solution of* $\mathrm{FIM}^{\mathrm{g}}_{\mathrm{AL}}$ *when b is constant.*

5.2.6 Practical Algorithms

For the case with general budget b, recall that the problem is inapproximable, unless $\mathrm{P} = \mathrm{NP}$ according to Corolllary 5.9. We still propose several algorithms in this subsection that perform well in practice as we will show later on. All our algorithms are of a greedy flavour and based on restricting to the evaluation of increments of non-edges that seem promising to improve fairness. In the following, we describe the proposed methods.

grdy_al. The algorithm that, starting with $F = \emptyset$, in b iterations, chooses the non-edge e into F that maximizes the increment $\min_C \mathbb{E}_{S\sim p(F,k)}[\sigma(S, F \cup \{e\})] - \min_C \mathbb{E}_{S\sim p(F,k)}[\sigma(S, F)]$. For efficiency we restrict to evaluate only non-edges that are (1) incident to S_p, the union over all sets with positive support in $p(F,k)$, and (2) are inter-community edges. Note that at the beginning of each iteration, we recompute $p(F,k)$ as F changes.

to_minC_infl. The algorithm that, starting from the empty set $F = \emptyset$, adds the non-edge $e = (u,v) \in \bar{E} \setminus F$ to F that connects a node from S_p with a node that maximizes $f(e) := \Pr_{S\sim p(F,k)}[u \in S] \cdot w_e \cdot \mathbb{E}_{S\sim p(F,k)}[\sigma_{\bar{C}}(S \cup \{v\}, F)]$, where $\bar{C}$ is the community of minimum coverage. We refer the reader to the pseudo-code in Algorithm 4. The rationale being to choose the non-edge that connects a seed node with a node that has large influence in the community $\bar{C}$ taking into account both the probability that u is a seed and the edge weight w_e.

to_minC_min. The algorithm that, starting from the empty set, adds a non-edge to the node $\bar{v}$ with minimum probability of being reached in the community that currently suffers the smallest community coverage. Among all these non-edges we choose the non-edge $(u, \bar{v})$ that maximizes the product $\Pr_{S\sim q}[u \in S] \cdot w_{(u,\bar{v})}$. The pseudo-code is given in Algorithm 5.

We highlight two techniques that we use speed up our implementations: (1) a pruning technique for **grdy_al**: Let δ denote the best increment of an edge that we have seen so far. Before evaluating the exact increment of a non-edge $e = (u,v) \in A \setminus F$, we compute an upper bound on the increment achievable by e via evaluating the expected community coverages $\mathbb{E}_{S\sim p(F,k)}[\sigma_C(\{v\},F)]$ that would be achieved by choosing v as a seed. We refer the reader to the pseudo-code in Algorithm 3 for further details. (2) A way to update RR sets rather than recompute them from scratch after adding edges: In all our algorithms, we change the graph by adding edges to it. As a consequence the functions σ and σ_C need to be approximated based on different simulations or, here, based on different RR sets. We observe however that after adding one edge, say $e=(u,v)$ to the graph, we do not need to entirely resample the RR sets, but, instead, can update and reuse them as follows. For every RR set R that contains the node v, we update R by re-starting the RR set construction from u with probability $w_{(u,v)}$ and adding the resulting nodes to R.

Algorithm 3 grdy_al

```
Require: instance $\mathcal{I} = (G, \mathcal{C}, b, k)$
Ensure: set $F \subseteq \bar{E}$ with $|F| \leq b$
  $F \leftarrow \emptyset$
  while $|F| < b$ do
    $q \leftarrow p(F,k)$
    $\delta \leftarrow -\infty$
    $A \leftarrow \{(u,v) \in \bar{E} : u \in S$ for some $S : q_S > 0$ and $v \notin S$ for all $S : q_S > 0\}$
    for $(u,v) = e \in A \setminus F$ do
      $\tau_C(v) \leftarrow \mathbb{E}_{S\sim q}[\sigma_C(S,F)] + \mathbb{E}_{S\sim q}[\sigma_C(\{v\},F)]$, for all $C \in \mathcal{C}$
      if $\min_{C\in\mathcal{C}}\{\tau_C(v)\} > \delta$ then
        $\lambda \leftarrow \min_{C\in\mathcal{C}}\{\mathbb{E}_{S\sim q}[\sigma_C(S, F \cup \{e\})]\}$
        if $\lambda > \delta$ then
          $\delta \leftarrow \lambda$
          $\bar{e} \leftarrow e$
        end if
      end if
    end for
    $F \leftarrow F \cup \{\bar{e}\}$
  end while
  return $F$
```

5.3 Experiments

In this section, we report on two experiments involving the $\text{FIM}^{\text{g}}_{\text{AL}}$ problem. In the first experiment, we compare the algorithms presented above in terms of quality and running time. In a second experiment, we evaluate the best performing algorithm against other

Algorithm 4 **to_minC_infl**

```
Require: instance $\mathcal{I} = (G, \mathcal{C}, b, k)$
Ensure: set $F \subseteq \bar{E}$ with $|F| \leq b$
  $F \leftarrow \emptyset$
  while $|F| < b$ do
    $q \leftarrow p(F, k)$
    $\bar{C} \leftarrow \arg\min_{C \in \mathcal{C}} \{\mathbb{E}_{S \sim q}[\sigma_C(S, F)]\}$
    $\bar{e} \leftarrow \arg\max_{(u,v) \in \bar{E} \setminus F} \{\Pr_{S \sim q}[u \in S] \cdot w_{(u,v)} \cdot \mathbb{E}_{S \sim q}[\sigma_{\bar{C}}(S \cup \{v\}, F)]\}$
    $F \leftarrow F \cup \{\bar{e}\}$
  end while
  return $F$
```

Algorithm 5 **to_minC_min**

```
Require: instance $\mathcal{I} = (G, \mathcal{C}, b, k)$
Ensure: set $F \subseteq \bar{E}$ with $|F| \leq b$
  $F \leftarrow \emptyset$
  while $|F| < b$ do
    $q \leftarrow p(F, k)$
    $\bar{C} \leftarrow \arg\min_{C \in \mathcal{C}} \{\mathbb{E}_{S \sim q}[\sigma_C(S, F)]\}$
    $\bar{v} \leftarrow \arg\min_{v \in \bar{C}} \{\mathbb{E}_{S \sim q}[\sigma_v(S, F)]\}$
    $\bar{e} \leftarrow \arg\max_{(u,\bar{v}) \in \bar{E} \setminus F} \{\Pr_{S \sim q}[u \in S] \cdot w_{(u,\bar{v})}\}$
    $F \leftarrow F \cup \{\bar{e}\}$
  end while
  return $F$
```

fairness-tailored seeding algorithms. We show, for several settings, that already adding just a few edges can lead to a situation where purely efficiency-oriented information spreading becomes automatically fair. We proceed by describing the experimental setup.

Experimental Setting. In our experiments we use random, synthetic and real world instances. Properties of synthetic and real world instances are given in Table 3.1 in Chapter 3. We choose the non-edge weights uniformly at random from the interval $[0, 1]$. The algorithms **grdy_al**, **to_minC_infl**, and **to_minC_min** are implemented in C++ and were compiled with g++ 7.5.0.

Experiment 1. In addition to the three algorithms described in Section 5.2, we evaluate the following two base lines: **random**: the algorithm that chooses b non-edges uniformly at random, and **max_weight**: the algorithm that chooses the b non-edges of maximal weight. The results can be found in Figure 5.2 for the random and synthetic instances. We observe that, despite the pruning approach described above, **grdy_al**'s running time is the worst. Furthermore, the fairness that it achieves is worse than the one of **to_minC_infl**. We thus exclude **grdy_al** from further experiments. **random** and **max_weight** are fastest but the fairness achieved by them is very poor.

In Figure 5.3, we can see the results for the real world instances Arenas, ca-GrQc and email-Eu-core. We observe that the running times of both algorithms **to_minC_infl** and **to_minC_min** are comparable, while **to_minC_infl** achieves better values of fairness. We thus choose **to_minC_infl** as the best performing algorithm as a result of this experiment.

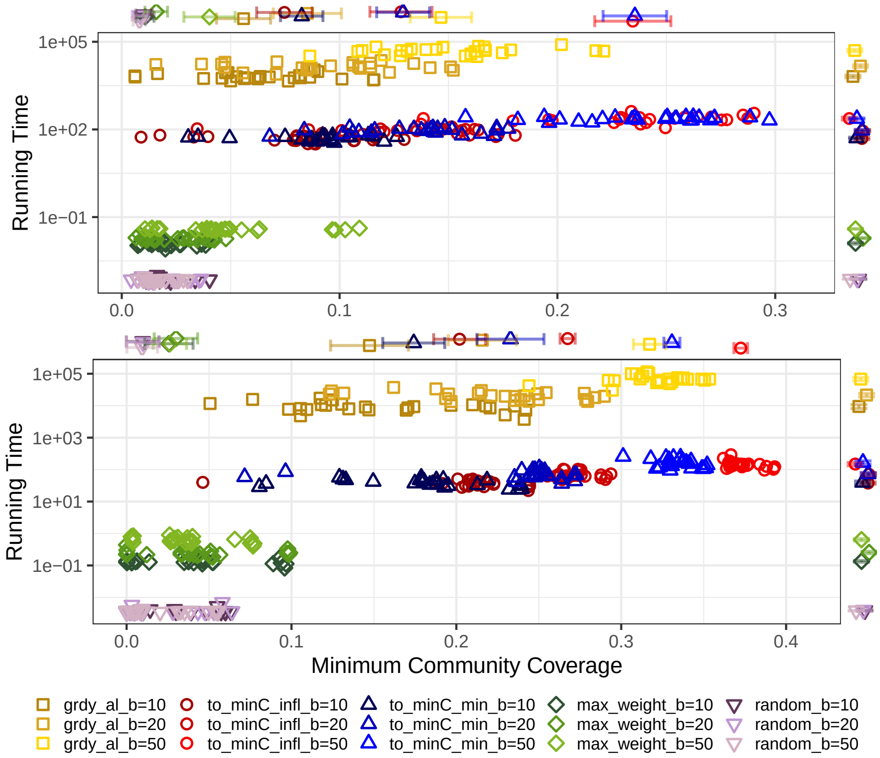


Figure 5.2: Results Experiment 1: (1) Random instances ($k = 25$, $n = 200$, singleton communities), (2) synthetic instances ($k = 25$, $n = 500$, communities induced by gender and region). The running time is on the logarithmic vertical axis, while the minimum community coverage is on the horizontal axis.

Experiment 2. The goal of the second experiment is to analyze how many links we need to add in order to make the standard greedy algorithm for IM satisfy similar or better fairness guarantees than fairness-tailored algorithms. To this end, we compare our method **to_minC_infl** with the following competitors: **grdy_im**, **grdy_maximin**,

myopic, **set_based**, and **moso**. We refer to Subsections 3.3.3 and 3.4.1 in Chapter 3 for further details on the methods.

We note that the algorithms **set_based** and **moso** are designed to compute distributions over seed sets and nodes, respectively, and thus they can be used to obtain both ex-ante and ex-post fairness guarantees. Hence, for these two algorithms we include both there ex-post and ex-ante values in our evaluations. It is worth pointing out that is much easier (especially in settings with many communities) to achieve good values ex-ante rather than ex-post.

We show the results for the random and synthetic instances in Figure 5.4. Already for small values of b, i.e., after adding just a few edges, our algorithm surpass all ex-post fairness values of the competitors. Even better and maybe surprisingly, our algorithm also achieves ex-post values higher than the ex-ante values of **set_based** and **moso**. We exclude the algorithms **grdy_maximin** and **moso** from experiments with the real world instance as they perform the worst in terms of running time. We turn to the real world instances, see Figure 5.5, on which we evaluate our algorithm for three fixed values of $b = 10, 20, 50$. We observe that by adding only 10 edges, the fairness values obtained by our algorithm dominate over the ex-post fairness values achieved by the competitors. We also observe that after adding only 50 edges, the fairness values of our method are larger than (or comparable to) the ex-ante fairness values achieved by **set_based**, on all instances.

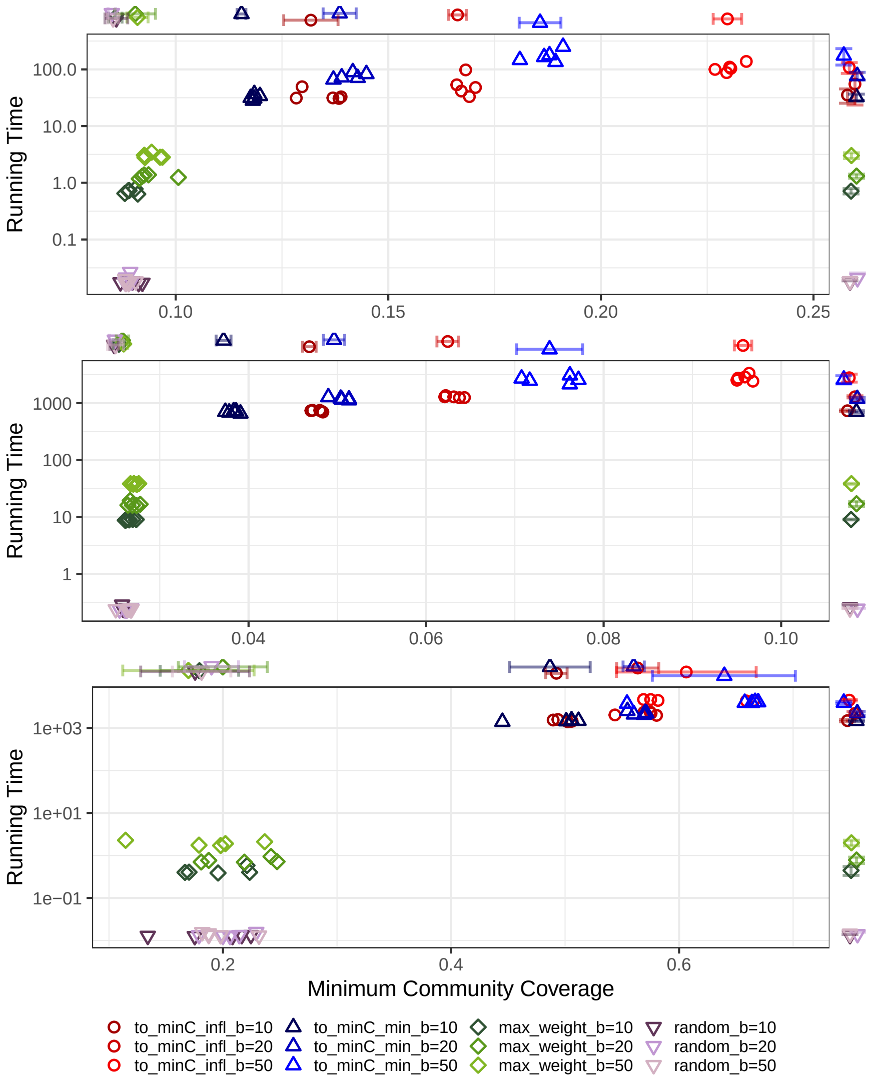


Figure 5.3: Results Experiment 1: (1) Arenas with BFS communities ($m = 10$), $k = 20$, (2) ca-GrQc with BFS communities ($m = 10$), $k = 20$, (3) email-Eu-core with real communities, $k = 20$. Again, the running time is on the logarithmic vertical axis, while the minimum community coverage is on the horizontal axis.

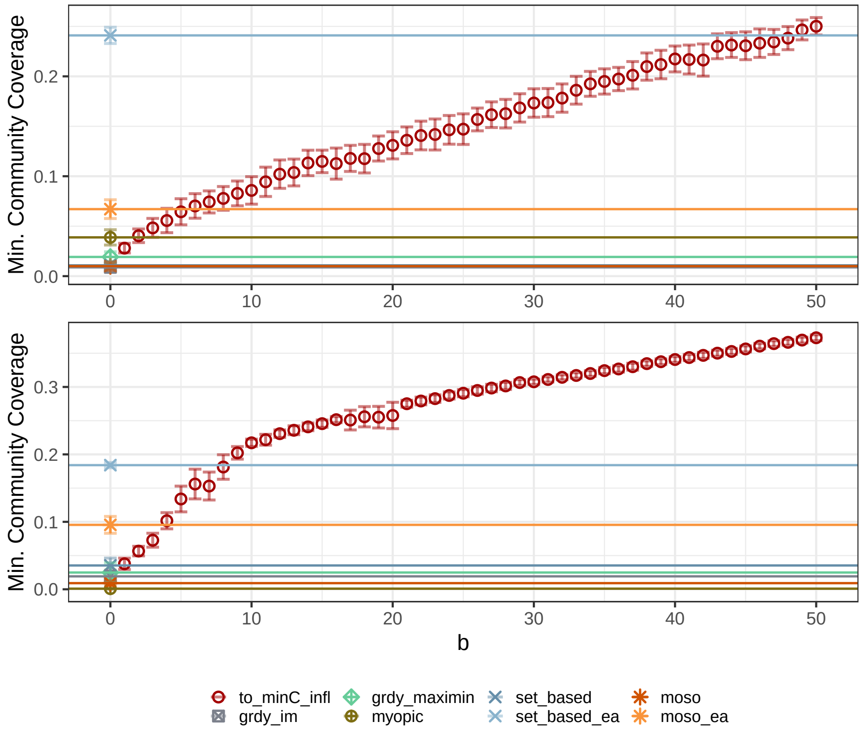


Figure 5.4: Results Experiment 2: (1) Random instances ($k = 25$, $n = 200$, singleton communities), (2) synthetic instances ($k = 25$, $n = 500$, communities induced by gender and region), minimum community coverage on the vertical, b on the horizontal axis.

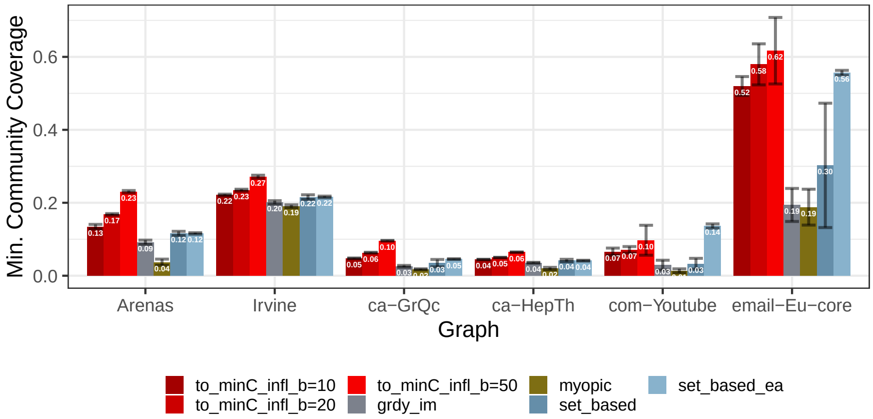


Figure 5.5: Results Experiment 2: Real world graphs with BFS communities ($m = 10$) for Arenas, Irvine, ca-GrQc, ca-HepTh, and real communities for email-Eu-core and com-Youtube, $k = 20$, minimum community coverage on the vertical axis, different instances on the horizontal.

Chapter 6

Conclusion and Future Work

In this thesis, we investigated different optimization problems under various notions of group fairness. Under the maximin criterion, we studied the problem of determining key seed nodes to maximize the minimum expected probability that communities are reached, and designed approximation algorithms achieving a constant multiplicative factor. Using other notions of fairness, e.g., equalized odds, demographic parity, and predictive parity notion, we proposed several optimization problems that aim at maximizing the overall spread or spread over some specific users while satisfying fairness via constraints (either ex-post or ex-ante). After studying the complexity of the proposed optimization problems, for one of the probabilistic problems we designed an algorithm with both constant approximation factor and fairness violation as well as heuristics for the other one. 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 confirms the increase in ex-ante fairness achieved over previous methods, indicating that randomness as source of fairness in influence maximization is very promising to be further explored. We also observed that our probabilistic algorithms give good results in terms of ex-post fairness values. We then studied optimization problems with the goal to modify the network structure by adding links in such a way that efficiency-oriented information spreading becomes automatically fair.

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. One possible way of showing the hardness of approximation for the set-based problem can be that, one needs to design a rounding technique (like Pipage rounding in [1]) in order to compute a deterministic solution S from a probabilistic strategy p. Tightening the result on the gap between the node-based and set-based problem and solving the set-based problem using different approaches, e.g., the row or column-generation approach, are another open questions.

A more interesting and challenging direction of work would be to design an approximation algorithm for the $\mathrm{pIM}^{\mathrm{dp}}$ problem. Also improving the fairness violation or achieving exact demographic parity for the $\mathrm{iIM}^{\mathrm{dp}}$ problem seems an interesting problem.

Moreover, studying the $\mathrm{FIM}_{\mathrm{AL}}$ and $\mathrm{FIM}^{\mathrm{g}}_{\mathrm{AL}}$ problems under different intervention actions, e.g., increasing the weights of edges, is another research direction. We believe that one can use similar approaches to what we did in Chapter 5 and give the NP-hardness and hardness of approximation results for the problems.

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. [71].

Lastly, studying the parameterized complexity of the proposed problems is actually an interesting direction of work.

Bibliography

[1] Alexander A. Ageev and Maxim Sviridenko. Pipage rounding: A new method of constructing algorithms with proven performance guarantee. *Journal of Combinatorial Optimization*, 8(3):307–328, 2004.

[2] Shipra Agrawal, Yichuan Ding, Amin Saberi, and Yinyu Ye. 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*, pages 1087–1096, 2010.

[3] Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74(1):47–97, January 2002.

[4] Junaid Ali, Mahmoudreza Babaei, Abhijnan Chakraborty, Baharan Mirzasoleiman, Krishna P. Gummadi, and Adish Singla. On the fairness of time-critical influence maximization in social networks (extended abstract). In *38th IEEE International Conference on Data Engineering, ICDE 2022, Kuala Lumpur, Malaysia, May 9-12, 2022*, pages 1541–1542. IEEE, 2022.

[5] Md Sanzeed Anwar, Martin Saveski, and Deb Roy. Balanced influence maximization in the presence of homophily. In *WSDM '21, The Fourteenth ACM International Conference on Web Search and Data Mining, Virtual Event, Israel, March 8-12, 2021*, pages 175–183. ACM, 2021.

[6] Sanjeev Arora and Boaz Barak. *Computational Complexity - A Modern Approach.* Cambridge University Press, 2009.

[7] Haris Aziz. A probabilistic approach to voting, allocation, matching, and coalition formation. In *The Future of Economic Design*, pages 45–50. Springer, 2019.

[8] Haris Aziz, Felix Brandt, and Paul Stursberg. On popular random assignments. In *International Symposium on Algorithmic Game Theory*, pages 183–194. Springer, 2013.

[9] Haris Aziz, Pang Luo, and Christine Rizkallah. Rank maximal equal contribution: A probabilistic social choice function. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 910–916. AAAI Press, 2018.

[10] Abhijit Banerjee, Arun G Chandrasekhar, Esther Duflo, and Matthew O Jackson. The diffusion of microfinance. *Science*, 341(6144):1236498, 2013.

[11] Solon Barocas, Moritz Hardt, and Arvind Narayanan. *Fairness and Machine Learning*. fairmlbook.org, 2019. http://www.fairmlbook.org.

[12] Ashkan Bashardoust, Sorelle A. Friedler, Carlos Eduardo Scheidegger, Blair D. Sullivan, and Suresh Venkatasubramanian. Reducing access disparities in networks using edge augmentation. *CoRR*, abs/2209.07616, 2022.

[13] Ruben Becker, Federico Corò, Gianlorenzo D'Angelo, and Hugo Gilbert. 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*, pages 3–10. AAAI Press, 2020.

[14] Ruben Becker, Gianlorenzo D'Angelo, Sajjad Ghobadi, and Hugo Gilbert. 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*, pages 14684–14692, 2021.

[15] Ruben Becker, Gianlorenzo D'Angelo, Sajjad Ghobadi, and Hugo Gilbert. Fairness in influence maximization through randomization. *Journal of Artificial Intelligence Research*, 73:1251–1283, 2022.

[16] Anna Bogomolnaia and Hervé Moulin. A new solution to the random assignment problem. *Journal of Economic theory*, 100(2):295–328, 2001.

[17] Stephen P. Borgatti and Martin G. Everett. Models of core/periphery structures. *Soc. Networks*, 21(4):375–395, 2000.

[18] Christian Borgs, Michael Brautbar, Jennifer T. Chayes, and Brendan Lucier. 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*, pages 946–957, 2014.

[19] A. Borodin, M. Braverman, B. Lucier, and J. Oren. Strategyproof mechanisms for competitive influence in networks. *Algorithmica*, 78(2):425–452, 2017.

[20] Florian Brandl, Felix Brandt, and Hans Georg Seedig. Consistent probabilistic social choice. *Econometrica*, 84(5):1839–1880, 2016.

[21] Felix Brandt. Collective choice lotteries. In *The Future of Economic Design*, pages 51–56. Springer, 2019.

[22] Ceren Budak, Divyakant Agrawal, and Amr El Abbadi. 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*, pages 665–674, 2011.

[23] Matteo Castiglioni, Diodato Ferraioli, and Nicola Gatti. Election control in social networks via edge addition or removal. 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*, pages 1878–1885. AAAI Press, 2020.

[24] Vineet Chaoji, Sayan Ranu, Rajeev Rastogi, and Rushi Bhatt. Recommendations to boost content spread in social networks. In *Proceedings of the 21st World Wide Web Conference 2012, WWW 2012, Lyon, France, April 16-20, 2012*, pages 529–538. ACM, 2012.

[25] Chandra Chekuri, Jan Vondrák, and Rico Zenklusen. 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, pages 575–584, 2010.

[26] Wei Chen and Shang-Hua Teng. 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)*, pages 967–976, 2017.

[27] Wei Chen, Yajun Wang, and Siyu Yang. Efficient influence maximization in social networks. In John F. Elder IV, Françoise Fogelman-Soulié, Peter A. Flach, and Mohammed Javeed Zaki, editors, *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, June 28 - July 1, 2009*, pages 199–208. ACM, 2009.

[28] Wei Chen, Yifei Yuan, and Li Zhang. Scalable influence maximization in social networks under the linear threshold model. In *ICDM 2010, The 10th IEEE International Conference on Data Mining, Sydney, Australia, 14-17 December 2010*, pages 88–97. IEEE Computer Society, 2010.

[29] Alexandra Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big Data*, 5(2):153–163, 2017.

[30] Edith Cohen, Daniel Delling, Thomas Pajor, and Renato F. Werneck. 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*, pages 629–638, 2014.

[31] Gerard Cornuejols, Marshall L Fisher, and George L Nemhauser. Exceptional paper—location of bank accounts to optimize float: An analytic study of exact and approximate algorithms. *Management science*, 23(8):789–810, 1977.

[32] Federico Coro, Gianlorenzo D'Angelo, and Yllka Velaj. Link recommendation for social influence maximization. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 15(6):94:1–94:23, 2021.

[33] Gianlorenzo D'Angelo, Lorenzo Severini, and Yllka Velaj. Selecting nodes and buying links to maximize the information diffusion in a network. In *42nd International Symposium on Mathematical Foundations of Computer Science, MFCS*

2017, August 21-25, 2017 - Aalborg, Denmark, volume 83 of *LIPIcs*, pages 75:1–75:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017.

[34] Gianlorenzo D'Angelo, Lorenzo Severini, and Yllka Velaj. Recommending links through influence maximization. *Theoretical Computer Science*, 764:30–41, 2019.

[35] Pedro M. Domingos and Matthew Richardson. Mining the network value of customers. In Doheon Lee, Mario Schkolnick, Foster J. Provost, and Ramakrishnan Srikant, editors, *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, San Francisco, CA, USA, August 26-29, 2001*, pages 57–66. ACM, 2001.

[36] Golnoosh Farnadi, Behrouz Babaki, and Michel Gendreau. A unifying framework for fairness-aware influence maximization. In *Companion Proceedings of the Web Conference 2020*, pages 714–722, 2020.

[37] Uriel Feige. A threshold of ln n for approximating set cover. *Journal of the ACM (JACM)*, 45(4):634–652, 1998.

[38] Benjamin Fish, Ashkan Bashardoust, Danah Boyd, Sorelle Friedler, Carlos Scheidegger, and Suresh Venkatasubramanian. Gaps in information access in social networks? In *The World Wide Web Conference*, pages 480–490. ACM, 2019.

[39] M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman, 1979.

[40] Kiran Garimella, Gianmarco De Francisci Morales, Aristides Gionis, and Michael Mathioudakis. Reducing controversy by connecting opposing views. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, WSDM 2017, Cambridge, United Kingdom, February 6-10, 2017*, pages 81–90. ACM, 2017.

[41] Shay Gershtein, Tova Milo, and Brit Youngmann. Multi-objective influence maximization. In *Proceedings of the 24th International Conference on Extending Database Technology, EDBT 2021, Nicosia, Cyprus, March 23 - 26, 2021*, pages 145–156, 2021.

[42] Roger Guimerà, Leon Danon, Albert Díaz-Guilera, Francesc Giralt, and Alex Arenas. Self-similar community structure in a network of human interactions. *Physical review E*, 68(6):065103, 2003.

[43] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2022.

[44] Aric A. Hagberg, Daniel A. Schult, and Pieter J. Swart. Exploring network structure, dynamics, and function using networkx. In Gaël Varoquaux, Travis Vaught, and Jarrod Millman, editors, *Proceedings of the 7th Python in Science Conference*, pages 11 – 15, Pasadena, CA USA, 2008.

[45] Akshay-Kumar Katta and Jay Sethuraman. A solution to the random assignment problem on the full preference domain. *Journal of Economic theory*, 131(1):231–250, 2006.

[46] David Kempe, Jon M. Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. *Theory of Computing*, 11:105–147, 2015.

[47] Moein Khajehnejad, Ahmad Asgharian Rezaei, Mahmoudreza Babaei, Jessica Hoffmann, Mahdi Jalili, and Adrian Weller. Adversarial graph embeddings for fair influence maximization over social networks. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, IJCAI'20, 2021.

[48] Elias Boutros Khalil, Bistra Dilkina, and Le Song. Scalable diffusion-aware optimization of network topology. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 1226–1235. ACM, 2014.

[49] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Graph evolution: Densification and shrinking diameters. *ACM transactions on Knowledge Discovery from Data (TKDD)*, 1(1):2–es, 2007.

[50] Jure Leskovec and Andrej Krevl. SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data, June 2014.

[51] Wei Lu, Francesco Bonchi, Amit Goyal, and Laks VS Lakshmanan. 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*, pages 928–936, 2013.

[52] Guowei Ma, Qi Liu, Enhong Chen, and Biao Xiang. Individual influence maximization via link recommendation. In *Web-Age Information Management - 16th*

International Conference, WAIM 2015, Qingdao, China, June 8-10, 2015. Proceedings, volume 9098 of *Lecture Notes in Computer Science*, pages 42–56. Springer, 2015.

[53] Mark J Machina. Dynamic consistency and non-expected utility models of choice under uncertainty. *Journal of Economic Literature*, 27(4):1622–1668, 1989.

[54] David Manlove. *Algorithmics of matching under preferences*, volume 2. World Scientific, 2013.

[55] Julian J. McAuley and Jure Leskovec. Learning to discover social circles in ego networks. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*, pages 548–556, 2012.

[56] Michael Mitzenmacher and Eli Upfal. *Probability and Computing: Randomization and Probabilistic Techniques in Algorithms and Data Analysis.* Cambridge University Press, USA, 2nd edition, 2017.

[57] Hervé Moulin. *Axioms of Cooperative Decision Making.* Econometric Society Monographs. Cambridge University Press, 1991.

[58] George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. An analysis of approximations for maximizing submodular set functions - I. *Mathematical programming*, 14(1):265–294, 1978.

[59] Tore Opsahl and Pietro Panzarasa. Clustering in weighted networks. *Social networks*, 31(2):155–163, 2009.

[60] Christos H. Papadimitriou. *Computational complexity.* Addison-Wesley, 1994.

[61] Hannah Jane Parkinson. Click and elect: how fake news helped donald trump win a real election. *The Guardian*, 14, 2016.

[62] Aida Rahmattalabi, Shahin Jabbari, Himabindu Lakkaraju, Phebe Vayanos, Max Izenberg, Ryan Brown, Eric Rice, and Milind Tambe. Fair influence maximization: a welfare optimization approach. 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, pages 11630–11638. AAAI Press, 2021.

[63] Max Read. Donald trump won because of facebook. *New York Magazine*, 9, 2016.

[64] Matthew Richardson and Pedro M. Domingos. Mining knowledge-sharing sites for viral marketing. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, July 23-26, 2002, Edmonton, Alberta, Canada*, pages 61–70. ACM, 2002.

[65] Ana-Andreea Stoica, Jessy Xinyi Han, and Augustin Chaintreau. Seeding network influence in biased networks and the benefits of diversity. In *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, pages 2089–2098, 2020.

[66] Ian P. Swift, Sana Ebrahimi, Azade Nova, and Abolfazl Asudeh. Maximizing fair content spread via edge suggestion in social networks. *Proc. VLDB Endow.*, 15(11):2692–2705, 2022.

[67] Youze Tang, Yanchen Shi, and Xiaokui Xiao. 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*, pages 1539–1554, 2015.

[68] Youze Tang, Xiaokui Xiao, and Yanchen Shi. 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*, pages 75–86, 2014.

[69] Hanghang Tong, B. Aditya Prakash, Tina Eliassi-Rad, Michalis Faloutsos, and Christos Faloutsos. Gelling, and melting, large graphs by edge manipulation. In *21st ACM International Conference on Information and Knowledge Management, CIKM'12, Maui, HI, USA, October 29 - November 02, 2012*, pages 245–254. ACM, 2012.

[70] V. A. Traag, L. Waltman, and N. J. van Eck. From louvain to leiden: guaranteeing well-connected communities. *Scientific Reports*, 9(1):5233, 2019.

[71] Alan Tsang, Bryan Wilder, Eric Rice, Milind Tambe, and Yair Zick. Group-fairness in influence maximization. In *Proceedings of the 28th International Joint*

Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019, pages 5997–6005, 2019.

[72] Rajan Udwani. 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*, pages 9513–9524, 2018.

[73] Sahil Verma and Julia Rubin. Fairness definitions explained. In Yuriy Brun, Brittany Johnson, and Alexandra Meliou, editors, *Proceedings of the International Workshop on Software Fairness, FairWare@ICSE 2018, Gothenburg, Sweden, May 29, 2018*, pages 1–7. ACM, 2018.

[74] Chi Wang, Wei Chen, and Yajun Wang. Scalable influence maximization for independent cascade model in large-scale social networks. *Data Mining and Knowledge Discovery*, 25(3):545–576, 2012.

[75] Xindi Wang, Onur Varol, and Tina Eliassi-Rad. Information access equality on network generative models. *CoRR*, abs/2107.02263, 2021. Available at SSRN.

[76] Bryan Wilder, Han-Ching Ou, Kayla de la Haye, and Milind Tambe. 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*, pages 841–849, 2018.

[77] Xiaojian Wu, Daniel Sheldon, and Shlomo Zilberstein. Efficient algorithms to optimize diffusion processes under the independent cascade model. *NIPS Work. on Networks in the Social and Information Sciences*, 1(1), 2015.

[78] Amulya Yadav, Hau Chan, Albert Xin Jiang, Haifeng Xu, Eric Rice, and Milind Tambe. Using social networks to aid homeless shelters: Dynamic influence maximization under uncertainty. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems, Singapore, May 9-13, 2016*, pages 740–748. ACM, 2016.

[79] Amulya Yadav, Bryan Wilder, Eric Rice, Robin Petering, Jaih Craddock, Amanda Yoshioka-Maxwell, Mary Hemler, Laura Onasch-Vera, Milind Tambe, and Darlene Woo. Bridging the gap between theory and practice in influence maximization:

Raising awareness about HIV among homeless youth. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 5399–5403, 2018.

[80] Qiqi Yan. 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*, pages 710–719, 2011.

[81] Jaewon Yang and Jure Leskovec. Defining and evaluating network communities based on ground-truth. *Knowledge and Information Systems*, 42(1):181–213, 2015.

[82] Neal E. Young. 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*, pages 170–178, 1995.

[83] Ying Yu, Jinglan Jia, Deying Li, and Yuqing Zhu. Fair multi-influence maximization in competitive social networks. In *International Conference on Wireless Algorithms, Systems, and Applications*, pages 253–265. Springer, 2017.

[84] Zhi Yu, Can Wang, Jiajun Bu, Xin Wang, Yue Wu, and Chun Chen. Friend recommendation with content spread enhancement in social networks. *Information Sciences*, 309:102–118, 2015.