



PHD THESIS

---

# Adaptive Influence Maximization: Bounding Adaptivity Gaps and Beyond

---

PHD PROGRAM IN COMPUTER SCIENCE: XXXIV CYCLE

*Author:*

Debashmita PODDAR  
debashmita.poddar@gssi.it

*Supervisors:*

Prof. Gianlorenzo D'ANGELO  
Dr. Cosimo VINCI  
gianlorenzo.dangelo@gssi.it  
cosimo.vinci@unisalento.it

March 2023

**GSSI Gran Sasso Science Institute**  
Viale Francesco Crispi, 7 - 67100 L'Aquila - Italy



# Abstract

The *influence maximization (IM)* problem is a discrete optimization problem [Kempe et al. [1]] of selecting the  $k$  most influential nodes in a network, and has ample applications ranging from viral marketing to outbreak detection. The IM problem is as follows: given a budget  $k$  and a graph  $G$ , our goal is to find a seed set  $S$  consisting of  $k$  nodes of  $G$  which maximizes the expected number of nodes influenced by  $S$ , according to some model of influence diffusion. The classic IM problem, which is also referred to as the *non-adaptive IM*, selects the seed set  $S$  before starting the diffusion process. Therefore, the focus is on the efficiency of the solution.

A few years later, Golovin and Krause [2] introduced the *adaptive IM (AIM)* to target the efficacy of the problem. In the adaptive setting, selection of the  $k$  seed nodes occurs one at a time, such that the  $j$ -th seed node is selected after observing the actual influence from the previously selected  $j - 1$  seeds. There are two main feedback models that have been studied: *full-adoption*, where the entire diffusion carried out by a seed node in the previous step is observed, and *myopic*, where only the neighbours of the previously selected seed nodes are considered as a feedback.

While adaptive policies are strictly stronger than non-adaptive ones, the latter are much easier to design and implement. Golovin and Krause [2] proposed a measure called the *adaptivity gap (AG)*, which is the ratio between the adaptive and non-adaptive optima. If the AG is small, we say that the non-adaptive policy provides a good approximation of the adaptive optimum.

In this thesis, we study AG with both feedback models. Under the full adoption feedback, we investigate the upper bound of AG over several graph classes, from in-arborescences to general graphs. We show the first sub-linear upper bound on AG that holds for any graph. Next, we have the experimental part, where we take different network models, such as random, scale-free and real world networks, and draw several comparisons between the adaptive greedy and the non-adaptive greedy algorithms under various network settings and parameters.

Eventually, we move on to the myopic feedback model, which lacks the property of submodularity. To recover from the drawbacks of being non-submodular, an artificial diffusion process is taken into account. We introduce a new technique to bound the AG without using the commonly used approaches, such as multi-linear extensions or random walks. We also study the adaptive greedy algorithm under the new setting and compare it to optimal adaptive policy. Our approach is of independent interest and can be used for further studies related to different adaptive optimization problems.



# *Acknowledgements*

I thank my supervisors for being a positive influence in both academia and my life in particular. Prof. Gianlorenzo D'Angelo, for being a patient and kind person. Gianlorenzo's pursuit of high standards in research have deeply shaped and impressed me over the years. He has invariably given me feedback on my work in the most benevolent way as possible. Dr. Cosimo Vinci as a forbearing supervisor, for consistently teaching and answering my questions without a speck of annoyance on his face. His unlimited enthusiasm in every aspect has motivated me to stick to this PhD during the bleaker times. Gianlorenzo and Cosimo always checked on my overall health before talking about work, and I am grateful for that.

My parents, for always letting me follow my dreams and being the pillars of strength. *Baba*, you really made me a strong and independent woman to deal with the world through your unusual ways. *Ma*, thank you for educating me to look beyond the appearance and have empathy.

My grandparents, who were amazing, and loved me till their last breath. I give them most of the credits for making me a good human being. There were times when I had to question my moral and their teachings reminded me of who I am. I miss them both, even though I don't talk about it often.

Pritom and Saili for being my only strand of family in Europe. I have spent most of the Christmas holidays in the course of my PhD with them, eating the delectable food Saili made for us. Thank you, brother, for always keeping me entertained, during the stressful periods.

Théo, for truly being my best friend and always believing in me with complete honesty. You have been an incredibly supportive and dedicated *ami* during the last seven and a half years I have known you. *Merci!*

A handful of people at the GSSI whom I call my adulting friends. These people have taken up the baton to teach me a lot of valuable life lessons. My heartfelt thanks to the entire GSSI community and the CS department for doing what they do the best.

Last but not the least, *grazie mille* to the *bella città* of L'Aquila, and the nice people I have encountered during these four years. Waking up to the beautiful view of the mountains definitely did good to my soul and helped me endure the shortcomings. Also, there is no better place than L'Aquila to learn about the Italian bureaucracy. This has been my second challenge, other than research, and there remains to be a lot of open problems which has been verified by many in the same boat as me.



# Contents

<b>Abstract</b>	<b>i</b>
<b>Acknowledgements</b>	<b>ii</b>
<b>List of Figures</b>	<b>vi</b>
<b>List of Tables</b>	<b>vii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 How connected are real world networks? . . . . .	1
1.2 Application of social networks: Viral Marketing . . . . .	2
1.3 From viral marketing towards maximizing influence in networks . . . . .	3
1.4 Necessity to bound the Adaptivity Gap . . . . .	5
1.5 Contributions . . . . .	6
Full-adoption feedback. . . . .	6
Myopic feedback. . . . .	7
1.6 Organization of the Thesis . . . . .	8
1.7 Table of Notations . . . . .	10
<b>2 Influence Maximization</b>	<b>12</b>
2.1 Main Definitions . . . . .	12
2.2 Diffusion Models . . . . .	13
2.2.1 Independent Cascade Model . . . . .	14
2.2.2 Linear Threshold (LT) Model . . . . .	15
2.2.3 Triggering Model . . . . .	16
2.3 Non-Adaptive Influence Maximization . . . . .	17
Monte-Carlo simulation-based approach. . . . .	18
Heuristic-based approach. . . . .	19
2.4 Adaptive Influence Maximization . . . . .	20
2.4.1 Adaptive Maximization . . . . .	20
2.4.2 Adaptive Influence Maximization . . . . .	21
2.4.3 Feedback Models of AIM . . . . .	21
2.4.3.1 Full-Adoption Feedback Model . . . . .	21
2.4.3.2 Myopic Feedback Model . . . . .	23
2.4.3.3 Partial and General Feedback Model . . . . .	23
2.4.4 Multi-Round Influence Maximization . . . . .	24
2.4.5 Other variations of AIM . . . . .	25

---

Non-monotone models. . . . .	25
DR-submodularity. . . . .	26
2.5 Stochastic Multi-Value Probing (SMP) . . . . .	26
<b>3 Better bounds on the Adaptivity Gap of Influence Maximization under Full-Adoption feedback</b>	<b>29</b>
3.1 Preliminaries . . . . .	32
3.2 Adaptivity Gap for In-arborescences . . . . .	34
3.3 Adaptivity Gap for General Influence Graphs . . . . .	39
3.4 Adaptivity Gap for Other Influence Graphs . . . . .	42
3.4.1 $\alpha$ -bounded-degree graphs . . . . .	42
3.4.2 $(\beta, \gamma)$ -bounded-activation graphs . . . . .	47
3.5 Experiments . . . . .	51
3.5.1 Testbed . . . . .	51
Erdős-Rényi. . . . .	52
Watts-Strogatz. . . . .	52
Barabási-Albert. . . . .	52
Facebook network. . . . .	53
Greedy Adaptivity Gap. . . . .	53
3.5.2 Independent Cascade . . . . .	53
Parameters. . . . .	53
Results. . . . .	53
3.5.3 Linear Threshold . . . . .	55
Parameters. . . . .	55
Results. . . . .	56
3.6 Summary . . . . .	57
<b>4 Improved Approximation Factor for Adaptive Influence Maximization via Simple greedy Strategies</b>	<b>58</b>
4.1 Stochastic Submodular Maximization . . . . .	62
4.2 Adaptive Influence Maximization under the Myopic Feedback Model: Preliminaries . . . . .	66
4.3 Efficiency of Non-Adaptive Greedy algorithm . . . . .	68
4.3.1 Adaptive Submodularity in the 2-Level Diffusion Model . . . . .	69
4.3.2 From the Ordinary to the 2-level Diffusion Model . . . . .	71
4.3.3 From the Adaptive to the Randomized Non-adaptive Policy . . . . .	72
4.3.4 Proof of Theorem 4.5 . . . . .	74
4.4 The Efficiency of the Adaptive Greedy Algorithm . . . . .	74
4.4.1 Proof of Theorem 4.13 . . . . .	78
4.5 Summary . . . . .	79
<b>5 Conclusion and Open Problems</b>	<b>81</b>
5.1 Applications of the Thesis . . . . .	83
In-arborescences. . . . .	83
Adaptive Stochastic Submodular Maximization. . . . .	83
5.2 Discussions and Open problems . . . . .	83
5.2.1 Full-Adoption Feedback . . . . .	84
5.2.2 Myopic Feedback . . . . .	85



**Bibliography**

**92**

# List of Figures

1.1	Connection between optimal and greedy policies. . . . .	5
2.1	Diffusion process under the independent cascade at several time steps. . . . .	14
2.2	Diffusion process under the linear threshold at several time steps. . . . .	14
2.3	A graph $G$ and two of its possible realisations. . . . .	15
3.1	In-arborescence . . . . .	34
3.2	Results under the Independent Cascade model with $p = [0.1, 1.0]$ . . . . .	54
3.3	Results under the Independent Cascade model with $p = [0.1, 0.2]$ . . . . .	55
3.4	Results under the Linear Threshold model with $p = [0.1, 1.0]$ . . . . .	57

# List of Tables

1.1	Table of Notations . . . . .	10
3.1	Adaptive ratios with $n = 64000$ and $k = 10$ under the IC model for generated graphs. . . . .	54
3.2	Adaptive ratios for the Facebook network with $k = 50$ . . . . .	54
3.3	Greedy adaptivity gap for the WS network under IC and LT with $k = 10$ . . . . .	56
5.1	Current bounds on AG and GAG for different models and feedback . . . . .	82



# Chapter 1

## Introduction

*“To develop a complete mind: Study the science of art; Study the art of science. Learn how to see. Realize that everything connects to everything else.”*

— Leonardo Da Vinci

### 1.1 How connected are real world networks?

Stanley Milgram in his classic paper, *The Small-World Problem* [3], put forth the notion of transitive dependency among people in a social network. He states that when a person, named Bob, is connected by a series of links to another person, called Alice, there are, on average, five intermediate people between Bob and Alice. Brown and Reingen [4] studied real networks and found the importance of stronger and weaker links associated with recommendation systems. Stronger links are seen as a measure of reliability for a certain recommendation, whereas weaker links are associated with the traversal of information from one cluster to another playing an important role in the networking ecosystem.

Social networking sites have seen a significant boom in the past two decades. These sites play a crucial role in spreading new ideas or innovations through its fundamental network structure consisting of individuals and their relationship with other individuals. Major networking sites with over a hundred million active users include *Facebook*, *Twitter*, *Instagram*, etc. In other words, *social networks* are a collection of *entities* which form *relationships* with other entities participating in the same network.

Due to the increase of information propagation in the networking sites, networks have become increasingly complex. Researchers have been trying to decode the properties and patterns exhibited by such massive networks. *Social Network Analysis (SNA)* has

emerged as the study of social networks using *collaboration graphs*, in which the entities are modelled by the nodes of a graph and the relationship between the entities are modelled by a set of edges. Formally, a *Social Graph*  $G = (V, E)$  is a directed graph (sometimes undirected), where  $V$  is the set of nodes representing the entities of the network, and  $E$  is the set of edges representing the relationship existing between those entities. SNA is the field of study at the intersection between *Graph Theory*, *Networking* and *Sociology*.

## 1.2 Application of social networks: Viral Marketing

*Marketing* is the promotion of a product by a company to its customer base. It takes into account the *word-of-mouth* effect, i.e., a customer who has used the products, can recommend it to her friends. *Viral Marketing (VM)* supports the same theory, and uses social networks as a platform for recommendations. The method follows the principle of contiguous viruses: if the flu infects one person, any person who comes in contact with the infected person also catches the flu with high probability.

In social networks, the company distributes some free or discounted products to specific individuals who are capable of marketing it. These individuals may recommend it to their neighbours after using it, who in turn might recommend it to their neighbours, and so on, following an iterative diffusion process. The direct analogy to this is *Influencer Marketing*. Influencers are those particular individuals in the network, who have a significant number of followers in a specific niche, and can affect the decisions of the people following her. Companies who want to launch new products in the market generally offer their products to the influencers, particularly to those who are closer to the specific customer base for selling the considered products.

Krackhardt et al. [5] states that a marketer can leverage her sales profit by knowing the topology of the underlying social network. They also introduced the concept of distributing free products to the influencers. Domingos and Richardson [6, 7] introduce some models of viral marketing, and raise the following algorithmic question: *in a vast connected network, if a seller markets a product, how should they choose a set of influential buyers (nodes) who can reach out to (influence) as many nodes as possible to buy that product?* Indeed, the number of such buyers are directly proportional to the seller's profit. Kempe et al. [1, 8, 9], in their seminal works, presented VM as a discrete optimization problem, and called it the *influence maximization (IM)* problem, discussed in details in the upcoming sections and chapters.

### 1.3 From viral marketing towards maximizing influence in networks

The IM problem as described by Kempe et al. [1] is an one-shot task: given a static social network  $G$ , a diffusion model with given parameters, select the seed set  $S$ , and with only one propagation pass through the entire network,  $S$  activates  $k$  nodes, with the expectation of maximizing the number of active nodes in the network.

The utility function,  $f$ , which needs to be maximized is shown to be submodular ( $f$  satisfies the “diminishing returns” property) and monotone (number of active nodes either increase or remain the same). Since the problem is NP-hard by nature (shown by reduction through *Set Cover Problem*), approximation algorithms needs to be used. From the results of Nemhauser et al. [10, 11], Kempe et al. [1] showed that the results obtained from a simple greedy algorithm will reach at least  $(1 - 1/e)$  of the optimal solution.

#### Independent Cascade (IC) Model

Goldenberg et al. [12], Goldberg and Liu [13] used the concept of *Cellular Automata* as a framework to model the diffusion of information in the network. The *Independent Cascade(IC)* model, named by Kempe et al. [1] models the dynamics of the viral marketing scenario and draws inspiration from the field of interacting particle systems. Initially, the diffusion model starts with a set of active nodes  $S$  such that  $|S| \leq k$ . Each edge  $e = (u, v) \in E$ , has an influence probability  $p_{uv}$  attached to it, so that node  $u$  will activate node  $v$  with a probability  $p_{uv}$ , independently from the other nodes.

With the above parameters as input to the system, the diffusion process unravels in several time steps. At time step  $t$ , if node  $u$  is active, it is given a single opportunity to activate its neighbour  $v$  with probability  $p_{uv}$ . If node  $v$  becomes active at time step  $t$ , it will further attempt to activate its neighbours at time step  $t + 1$  according to the influence probability. The process iteratively continues and comes to a halt when there are no more activations. Since the diffusion process spreads in a cascading manner, and each activation is independent from the other, the model is named as the independent cascade model.

**Linear Threshold (LT).** Differently from the IC model, where each activation is independent from each other, the LT model follows a more complex contagion process. Each node  $v$  is associated with a threshold value  $\theta_v$ , and to activate  $v$ , the probability associated with the active edges from  $v$ 's neighbours need to cross the threshold  $\theta_v$ . This

model is not in the scope of the thesis except for the experimental part in Chapter 3. However, in Chapter 2, Section 2.2.2 we will discuss the technical part of the LT model.

## Adaptive Influence Maximization

In the *Adaptive Influence Maximization (AIM)* [14], we consider the adaptive version of the IC model. An adaptive policy  $\pi$  is a function associating the *network state* with a new node that should be added to the seed set  $S$ , where the network state is what a policy observes at each time. The function  $f$  that we want to maximize in expectation is the number of nodes that are reached by the spread of information.

The AIM problem wants to find a policy with a seed-set maximizing the expected value of  $f$  (according to the probability distribution of the realizations), where the cardinality of the seed set must be at most  $k$ . The seed nodes are selected in multiple time steps, based on the observations or feedback obtained from the previous steps. The quality of seed nodes selected under the adaptive setting is better and also the influence generated is further improved because of the feedback provided by the network.

## Feedback models

There are two commonly used feedback models: *full adoption* and *myopic*. A full adoption feedback model records the entire spread generated by a seed node. The adaptive policy  $\pi$  observes this feedback and then decides on the selection of the next seed node. As mentioned before, the underlying diffusion model considered in this thesis is the independent cascade model, which is adaptive submodular and monotone for the full adoption feedback [2]. Golovin and Krause [2] exploited the desired adaptive submodularity property of the objective function  $f$ , and applied the results from Nemhauser et al. [10, 11] to show that the solution provided by an adaptive greedy policy is at least  $(1 - 1/e)$  of the optimal adaptive policy.

The myopic feedback model observes the current state of a seed node's immediate neighbours only. This model is shown to be adaptive non-submodular by nature in [14]. In order to evaluate the performance of this model, we need to recover the adaptive submodularity via an artificial diffusion model, which gives a seed node multiple chances to activate its neighbours [15, 16]. The feedback models will be discussed later at length in Section 2.4.3.



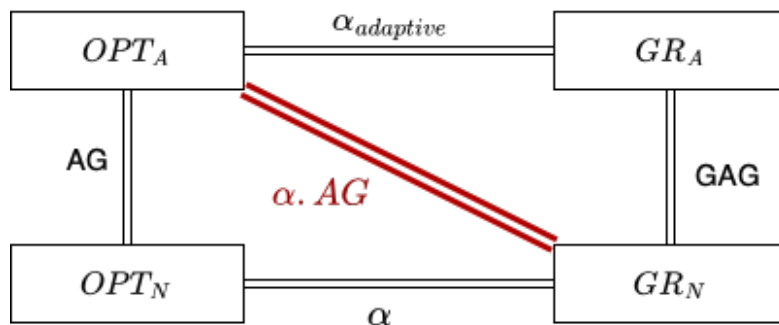


FIGURE 1.1: Connection between optimal and greedy policies.

### Adaptivity Gap

Another important term coined by Golovin and Krause [2] is the *Adaptivity Gap (AG)*. The AG quantifies how well does an optimal non-adaptive policy, that is a policy in which all the edges should be selected at beginning, without observing the realization, perform when compared to an optimal adaptive policy. To measure the efficiency of a non-adaptive greedy algorithm, we can multiply the adaptivity gap with the approximation factor to compare the performance ratio of the algorithm with the optimal adaptive policy.

## 1.4 Necessity to bound the Adaptivity Gap

Due to the feedback received after every round of seed selection, the efficiency in terms of time reduces for an adaptive policy. However, a non-adaptive policy is a straightforward simple process, and the overhead in time is negligible compared to the adaptive policies. Also, a non-adaptive algorithm is much simpler to design since the seed selection takes place in the beginning of the diffusion process, and we do not keep track of the nodes that are activated at each time step.

The adaptivity gap gives us the ratio to check how efficient the seed set selected by the optimal non-adaptive policy is compared to the adaptive optimal over different graph classes. A tighter bound is preferred, but the main challenge is to find an upper bound that is closer to 1, which will indicate that there exists an optimal non-adaptive policy that is as efficient as the optimal adaptive policy for that graph class. The thesis tries to find and then improve on these upper bounds on the different graph classes using newer techniques and by studying the ones that are used in the past.

In Section 3.5 we will encounter the efficiency of the adaptive greedy algorithm based on the memory consumption. With a faster best estimate, the non-adaptive greedy policy

tends to consume a huge part of the memory as the size of the graph grows, hence asking for better memory management. A non-adaptive greedy algorithm is generally preferred over an adaptive one due to the above stated reasons, and hence a measure is needed to calculate the efficiency of such algorithms. Another measure called the *greedy adaptivity gap (GAG)* [17], which measures the generated spread between the greedy adaptive algorithm and the greedy non-adaptive algorithm, helps us to navigate through the performance efficiency of a non-adaptive greedy algorithm over a adaptive greedy one. Figure 1.1 depicts the ratio between the optimals and the greedy algorithms (refer to table 1.7 for the notations used in the figure). The diagonal red line is the most important factor to bound, since it directly estimates by how much an optimal adaptive strategy outperforms a non-adaptive greedy algorithm, and helps us to design better algorithms by having an optimal measure in mind.

## 1.5 Contributions

1. Gianlorenzo D’Angelo, Debashmita Poddar, and Cosimo Vinci “Better bounds on the adaptivity gap of influence maximization under full-adoption feedback”, *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI, 2021*.

The extended version of the above paper is accepted at the *Artificial Intelligence Journal (AIJ)* [18].

**Full-adoption feedback.** Chapter 3 is mainly related to the results obtained in the aforementioned paper (that is, [19]), in which we conduct an extensive study on the full adaption feedback model under IC. First, we tackle the in-arborescences, which were already studied by [15] in the adaptive setting. Chen and Peng [15] derived an upper bound on adaptivity gaps using Poisson process and multi-linear extensions. We bypass those techniques and give a simpler and quicker solution to connect the non-adaptive and adaptive optimals. Eliminating the techniques used by [15] and building the proof on the marginals related to the hybrid non-adaptive policy reduces the upper bound on adaptivity gap for the in-arborescences from 3.16 to 2.31 (Section 3.2).

We further exploit our technique, to arrive at the first ever sub-linear upper bound on general graphs. To the best of our knowledge, this is the first and only bound known for general graphs with this setting. We show that under the Independent Cascade model with a full adoption feedback, the adaptivity gap of general graphs is  $\lceil n^{1/3} \rceil$ , where  $n$  is the number of nodes in the graph (Section 3.3).

Next, we derive results on more specific graphs such as the  $\alpha$ -bounded graphs that includes several undirected graph topologies like a star, cycle, clique (with more than 3 nodes), etc. The  $\alpha$ -bounded graphs are undirected graphs where for nodes more than degree 2, their degree summation is at most  $\alpha$ . We show that the adaptivity gap for such bounded graphs is  $\sqrt{\alpha} + O(1)$  and for 0-bounded graphs we get 3.16 as the upper bound (Section 3.4.1).

We also address a very interesting class of graphs called the  $(\beta, \gamma)$ -bounded-activation graphs which denotes a cluster of graphs where the nodes that have at most  $\gamma$  neighbours have  $\beta$  influence in expectation. For  $k \geq 2$ , the adaptivity gap for such influence graphs is  $\sqrt{\beta} + \frac{1}{1-\gamma}$  (Section 3.4.2).

The chapter is concluded by a series of experiments (Section 3.5) on a bunch of well known networks (generated), that simulate the network (Erdős-Rényi, Watts-Strogatz and Barabási-Albert) and a real world Facebook network. This part includes new results that have not yet been published, and shows how a state-of-the-art non-adaptive greedy algorithm ( $TIM^+$ ) performs when compared to its adaptive counterpart, which has the same underlying code as the non-adaptive greedy. We observe that the greedy adaptivity gap, which is the ratio between the adaptive greedy spread and the non-adaptive greedy, is closer to one for almost all the different network settings. This directly implies that the greedy non-adaptive algorithm ( $TIM^+$ ) used here, has an efficiency rate close to that of a greedy adaptive algorithm.

2. *Gianlorenzo D'Angelo, Debashmita Poddar, and Cosimo Vinci "Improved approximation factor for adaptive influence maximization via simple greedy strategies", 48th International Colloquium on Automata, Languages, and Programming, ICALP, 2021.*

**Myopic feedback.** Chapter 4 represents the aforementioned work (that is, [16]) which improves on [17]. The main difficulty faced with the myopic model is the lack of adaptive submodularity. This drawback makes the model non-trivial to analyze and hence we resort to an artificial model called the 2-level diffusion model. We use a similar concept introduced by [17], where the seed node gets multiple chances to activate its neighbours. However, instead of using random walks on optimal decision trees and multilinear extension like in [17], we directly analyze the non-adaptive greedy algorithm and relate it to an optimal adaptive solution which in turn improves the upper bound on the adaptivity gap from 4 to 3.164.

This flexibility of not passing through the adaptivity gap, gives us a more direct and improved approximation factor for both the non-adaptive greedy algorithm

and the adaptive greedy. For achieving the approximation ratio for non-adaptive greedy algorithm we resort to a randomized non-adaptive policy to relate the expected spread of the non-adaptive greedy with the adaptive policy and arrive at 0.316, which is an improvement from 0.158 (Section 4.3).

The approximation factor for the adaptive greedy algorithm is found in a similar fashion by introducing the strong 2-level diffusion model and a strong 2-level hybrid policy. The difference with the above mentioned policy lies in the fact that the strong hybrid policy is conditioned by the partial realisations, which is not the case in the simple hybrid adaptive policy. The approximation factor is bounded by 0.393 for the adaptive greedy (Section 4.4).

## 1.6 Organization of the Thesis

This thesis approaches on bounding the adaptivity gap using different techniques, and elucidates on the feedback models with different graph classes. The underlying diffusion model considered here is the independent cascade model and a part of the experiment is conducted on linear threshold.

- *Chapter 1* gives a general overview of the contents of this thesis and gives a high-level introduction on the topics that will be tackled here. A table of notations is provided to ease the access of all the symbols that will be used throughout the thesis.
- *Chapter 2* carries out an in-depth analysis of the IM problem with the different diffusion models. The chapter continues to provide further details on the AIM problem, an exhaustive analysis of the feedback models and their adaptivity gaps. Furthermore, the chapter navigates through the related state-of-the-art research that has been conducted in this field.
- *Chapter 3* studies the full-adoption feedback under the IC model with different graph classes (in-arborescences, general graphs,  $\alpha$ -bounded-degree graphs,  $(\beta, \gamma)$ -bounded-activation graphs). The chapter contains new techniques and the corresponding results on the upper bound of the AG related to the above mentioned graphs. The last part of the chapter deals with experiments which is important to uncover the efficiency of the greedy algorithms with different networks and parameters, giving us a further insight of the complexity to bound both the adaptivity gap and the greedy adaptivity gap.

- 
- *Chapter 4* contains the results related to the infamous adaptive non-submodular myopic feedback model with IC and explores a new way of reducing the upper bound on AG. The chapter also explores the greedy algorithms to achieve better approximations on the adaptive greedy and the non-adaptive greedy algorithms.
  - *Chapter 5* concludes the thesis by summarizing the main results obtained during the PhD. There is a dedicated section related to some application of the graph classes and feedback models studied in the thesis to motivate further research on this topic. Lastly, we discuss the open problems that can be tackled in the future by referring to the works conducted in this PhD thesis.

## 1.7 Table of Notations

TABLE 1.1: Table of Notations

Symbol	Meaning
$[k]_h$	$\{h, h + 1, \dots, k\}; h \leq k$
$G(V, E)$	A graph with $V$ as the set of vertices and $E$ as the set of edges.
$(u, v)$	An edge between nodes $u$ and $v$ : if directed, $u$ has an outgoing edge to $v$ .
$p(uv)$	Activation probability associated to the edge $(u, v) \in E$
$n$	Total number of nodes in the graph ; $ V  := n$
$m$	Total number of edges in the graph
$k$	Total number of selected seed nodes
$i$ or $j$	Generic nodes
$S$	Set of selected seed nodes; $S \subseteq V,  S  \leq k$
$A_t$	Activated nodes at step $t$ .
$L(V, L(E))$	A random graph $L$ made from $G$ and called a live edge graph; $L(E) \subseteq E$
$R(S, L)$	Set of nodes activated by $S$ in $L$ .
$\sigma(S)$	Expected influence spread of $S$ ; $\mathbb{E}_L[ R(S, L) ]$
$\phi_L$	Realization associated with $L$ , $\phi_L : V \rightarrow 2^V$
$\psi(v)$	Partial realization, set of nodes activated by $v$ ; $\psi : V \rightarrow 2^S$
$dom(\psi)$	Domain of $\psi$
$R(\psi)$	Set of nodes activated by $dom(\psi)$
$f(\psi)$	Number of nodes activated by $dom(\psi)$ ; $f(\psi) =  R(\psi) $
$\psi'$	Sub-realization of $\psi$ if $dom(\psi') \subseteq dom(\psi)$
$\pi$	Adaptive policy
$\pi(\psi)$	Seed node returned by $\pi$ after taking $\psi$ as an input
$\sigma(\pi)$	Expected influence spread of $\pi$
$\pi^*$	Optimal adaptive policy
$OPT_N(G, k)$	Optimal solution of the non-adaptive problem
$OPT_A(G, k)$	Optimal solution of the adaptive problem
$AG(G, k)$	Adaptivity Gap
$T(V, F)$	A rooted tree
$\Delta(i \psi)$	Expected marginal gain of $i$ when it is added to $\psi$
$\rho$	Random variable; $\mathbb{P}[\rho = i] = x_i/k$
$\partial(R(\psi))$	Set of boundary nodes of $R(\psi)$ ; Lemma 3.3 of Chapter 3.
$deg_v(G)$	Degree of node $v$ in $G$

Symbol	Meaning
$\mathcal{T}$	Rooted binary decision tree
$d$	Depth of the cascade in the partial feedback model
$\tau$	Threshold value related to the RIS algorithm
$\mathcal{R}$	Set of reverse reachable (RR) sets in RIS
$r$	Number of rounds
$\alpha$	Approximation factor of a greedy algorithm (not the $\alpha$ in Chapter 3)
$X$	Random variable
$N(u)$	Set of out-neighbors of node $u$ ; Section 3.4.2 Chapter 3
$\beta$	$\beta > 0$ is an integer; Section 3.4.2 Chapter 3
$\hat{S}$	Set of nodes such that $ \hat{S}  \leq \beta$ ; Section 3.4.2 Chapter 3
$\gamma$	$\gamma \in [0, 1)$ such that $\gamma \geq \sum_{v \in N(u)} p_{u,v}$ ; Section 3.4.2 Chapter 3
$b$	Number of nodes in each batch
$G_u$	Residual graph
$Rand_t$	Randomized non-adaptive policy
$\xi$	Partial state; Chapter 4
$\hat{\mathbf{L}}$	A live edge graph distributed as $\mathbf{L}$ , but independent from it
$\mathbf{L}^2(S)$	2-level-live edge graph with a seed set $S$
$\sigma_{\mathbf{L}, \hat{\mathbf{L}}}^2(S)$	The influence spread of $S$ in $\mathbf{L}^2(S)$
$\sigma^2(S)$	Expected influence spread of $S$ under the 2-level diffusion model
$Hyb_t$	Hybrid adaptive policy; Chapter 4
$\hat{\Phi}_\pi$	Random realization returned by $\pi$
$\Delta_S^2(v)$	Expected increment of $v$ , when it has two chances to activate its neighbor
$GR_N(G, t)$	Expected influence of $\sigma(S_t)$ , first $t$ nodes are selected by a greedy algorithm
$GR_A(G, t)$	Expected spread of an adaptive greedy policy after selecting the first $t$ seeds

## Chapter 2

# Influence Maximization

In this chapter, first we give an exhaustive definition of the influence maximization problem under the different diffusion models used for information propagation. Next, we present the different algorithmic approaches used for the non-adaptive greedy algorithm. Section 2.4 gives a brief overview of the adaptive IM which is the main scope of the thesis. We introduce the adaptivity gaps and the different feedback models associated with the adaptive problem. A dedicated related works section for the AIM with the feedback models is presented. We also explore the work done on the unconventional non-monotone and DR-submodular models w.r.t adaptive IM. Finally, we discuss the commonly used techniques related closely to our work, and on which the thesis improves upon in section 2.5.

### 2.1 Main Definitions

We study the following dynamics consisting in multiple time steps. Given a graph  $G = (V, E)$  as an input instance, we have a set  $S \subseteq V$  of *active nodes*<sup>1</sup>, called the *seed set*, which is responsible for the activation of nodes at the first time step,  $t$ . Furthermore, we have an underlying *diffusion model*  $M$ , determining how the set of active nodes, at each time step, influences some inactive nodes, that become active at the next time step,  $t + 1$ . The *influence function*  $\sigma_{G,M} : 2^V \rightarrow \mathbb{R}_{\geq 0}$  associates to each possible initial seed set  $S$ , the expected value  $\sigma_{G,M}(S)$  of nodes which are active at the end of the process.

The influence maximization problem can be formally defined as follows: given a budget of  $k$  users, a graph  $G$ , and a diffusion model  $M$ , the goal is to select a seed set  $S^*$  of  $k$

---

<sup>1</sup>In this thesis, we will use the terms active and influenced nodes interchangeably.



users from  $V$  in order to maximize the influence function, i.e., a set  $S^*$  such that

$$\sigma_{G,M}(S^*) = \arg \max_{S \subseteq V: |S| \leq k} \sigma_{G,M}(S). \quad (2.1)$$

Initially, all the nodes present in the seed set  $S$  are active. As the nodes in the graph starts getting influenced, an inactive node  $v$  will be activated by its active neighbours eventually, according to the considered diffusion model  $M$ . A node  $u \in V$  is said to be the *neighbour* of  $v$  if there is an edge  $(u, v) \in E$ . The diffusion process comes to a halt when there are no more activations. One classic example of influence function is the expected number of influenced nodes.

The main objective of the IM problem is to find a seed set of at most  $k$  nodes that maximizes the value of  $\sigma_{G,M}$  at the end of the diffusion process. However, maximizing  $\sigma_{G,M}$  is often a computationally hard problem. To overcome this issue, one can resort to an  $\alpha$ -*approximation algorithm*, that computes in polynomial time an approximate optimal seed set  $S$  with an *approximation factor* of  $\alpha \leq 1$ , i.e., a seed set  $S$  such that  $\sigma_{G,M}(S) \geq \alpha \cdot \sigma_{G,M}(S^*)$ . Most of the approximation algorithms considered for the IM problem require that function  $\sigma_{G,M}$  verify two desirable properties: *monotone* and *submodular*. A monotone function ensures that adding nodes to the seed set will not reduce the value of  $\sigma_{G,M}$ , whereas submodular means that the marginal gain of adding a node to the seed set does not increase as the size of the seed set increases. In particular,  $\sigma_{G,M}$  is *monotone* if, for any  $S \subseteq V$  and for any  $x \in V \setminus S$ , we get  $\sigma_{G,M}(S) \leq \sigma_{G,M}(S \cup \{x\})$ . Furthermore,  $\sigma_{G,M}$  is *submodular* if, for any  $S, P \subseteq V$  such that  $S \subseteq P$  and for any  $x \in V \setminus P$ , we get  $\sigma_{G,M}(P \cup \{x\}) - \sigma_{G,M}(P) \leq \sigma_{G,M}(S \cup \{x\}) - \sigma_{G,M}(S)$ . In subsection 2.3, we will formally describe a greedy approximation algorithm for the non-adaptive IM that exploits these properties to guarantee a good approximation ratio.

## 2.2 Diffusion Models

The design of diffusion models is an integral aspect of the IM problem. There are two kinds of diffusion processes: *progressive* and *non-progressive*. In the progressive model, once a node is activated, deactivation is impossible. In contrast, the non-progressive model allows an activated node to be deactivated later in the diffusion process. Even-Dar et al. [20] studied the IM problem by exploiting the *Voter Model* [21], which is an example of a non-progressive case. In this thesis, we restrict our study only to progressive models.

Models can also be *time-unbounded* and *time-bounded*. In time-unbounded models, termination occurs when the node activation comes to a halt. Whereas, the time-bounded

models have a separate time parameter for the completion of an ongoing diffusion process. Most of the previous work and the works in this thesis are mainly focus on time-unbounded models. Some examples of time-unbounded models are the independent cascade, linear threshold and *triggering*. For AIM and this thesis, only the IC model is considered, however, there is a section dedicated to the experiments on LT. The following subsections explain the IC, LT and triggering model in details.

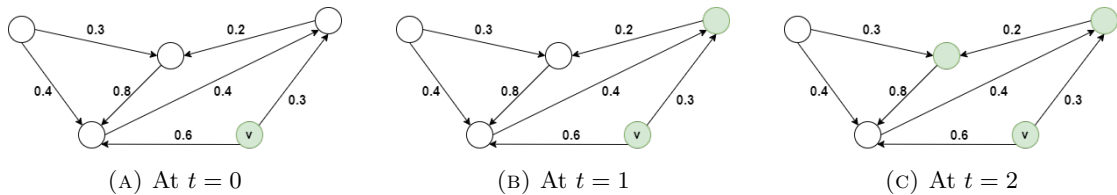


FIGURE 2.1: Diffusion process under the independent cascade at several time steps.

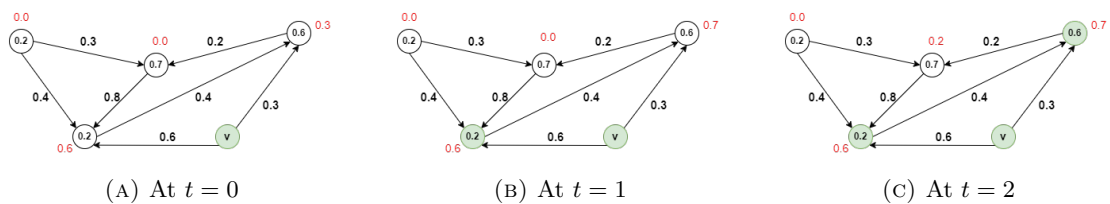


FIGURE 2.2: Diffusion process under the linear threshold at several time steps.

## 2.2.1 Independent Cascade Model

In the IC model, we have an *influence graph*  $G = (V = [n], E, (p_{uv})_{(u,v) \in E})$ , where edges are directed and  $p_{uv} \in [0, 1]$  is an *activation probability* associated to each edge  $(u, v) \in E$ . Given a set of *seeds*  $S \subseteq V$  which are initially *active*, the diffusion process in the IC model is defined in  $t \geq 0$  discrete steps as follows: (i) let  $A_t$  be the set of active nodes which are activated at each step  $t \geq 0$ ; (ii)  $A_0 := S$ ; (iii) given a step  $t \geq 0$ , for any edge  $(u, v)$  such that  $u \in A_t$ , node  $u$  can activate node  $v$  with probability  $p_{uv}$  independently from any other node, and in case of success,  $v$  is included in  $A_{t+1}$ ; (iv) the diffusion process ends at a step  $r \geq 0$  such that  $A_r = \emptyset$ , i.e., no more nodes in the graph can be activated. The size of  $\bigcup_{t \leq r} A_t$ , i.e. the number of nodes activated/reached by the diffusion process, is the *influence spread*. Figure 2.1 shows the influence unravelling at different time steps for the IC model.

We point out that the activation of an edge  $(u, v)$  is a random event determined by the outcome of a coin toss that gives **head** with probability  $p_{uv}$ . Before the start of the diffusion process, the biased coin is flipped for every edge in the graph  $G$ . By looking at the influence probabilities; one can conclude whether a node  $v$  will be active after the termination of the cascade process or not.

The above diffusion process can be equivalently defined as follows. The *live-edge graph*  $\mathbf{L} = (V, \mathbf{L}(E))$  is a random graph of  $G$ , such that each edge  $(u, v) \in E$  is included in  $\mathbf{L}(E)$  with probability  $p_{uv}$ , independently from the other edges, i.e.,  $\mathbb{P}[\mathbf{L} = L] = \prod_{(u,v) \in L} p_{uv} \prod_{(u,v) \in E \setminus L} (1 - p_{uv})$ .

With a little abuse of notation, we may denote  $\mathbf{L}(E)$  with  $L$ . Given  $L \subseteq E$ , let  $R_L(S) := \{v \in V : \text{there exists a path from } u \text{ to } v \text{ in } L \text{ for some } u \in S\}$ , i.e., the set of nodes reached by  $S$  in the graph  $L$ . Informally, if  $S$  is the set of seeds, and  $L$  is a realisation of the live-edge graph,  $R_L(S)$  equivalently denotes the set of nodes which are reached/activated by the above diffusion process. Let  $\sigma_L(S) := |R_L(S)|$  denote the *influence spread* generated by the set of seeds  $S$  if the realised live-edge graph is  $L$ , and let  $\sigma(S) := \mathbb{E}_L[\sigma_L(S)]$  be the *expected influence spread* generated by  $S$ . For a clear visualization of the realisations of a live-edge graph, refer to figure 2.3.

Kempe et al. [1] shows that finding the set of influential nodes maximizing the influence function in the IC model is NP-hard, and they use a reduction to the *Maximum Coverage* problem. A particular model of IC is that of *Weighted Cascade*. In such a model, each influence probability is defined as  $p_{uv} = 1/d_{in}(v)$ , where  $d_{in}(v)$  is the in-degree of  $v$ . In this process, a node is equally influenced by each of its neighbours.

Other models assume that the influence probabilities are not known. Saito et al. [22] predict the influence probabilities by using the *Expectation Maximization* algorithm to compute the diffusion probabilities of all the edges in  $G$ , repeatedly, to maximize the objective function. Goyal et al. [23] developed a learning algorithm which takes as input a social graph and the *Action Log*, that reports the actions performed by all the nodes.

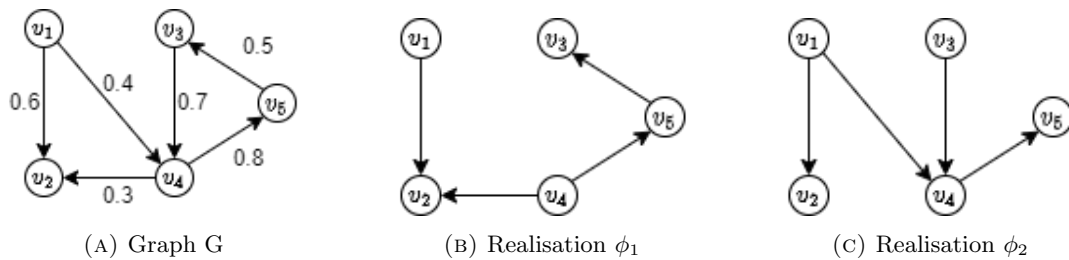


FIGURE 2.3: A graph  $G$  and two of its possible realisations.

### 2.2.2 Linear Threshold (LT) Model

Granovetter [24] introduced the *Linear Threshold model*, where he proposed a diffusion model in which each node  $v$  becomes active if the weighted sum of its active neighbours reaches a certain threshold. More formally, the model takes as input a graph  $G$ , the seed set  $S$ , and an edge weight  $W_{u,v}$  for any edge  $(u, v) \in E$ . Before the diffusion starts,

a *threshold value*  $\theta_v$  is picked uniformly at random in the interval  $[0, 1]$  by each node  $v \in V$ , independently from the other nodes.

At each time step  $t$  a node  $v$  will get activated if the summation of edge weights of its active neighbours reach the threshold value  $\theta_v$ , i.e.,

$$\sum_{(u,v) \in E: u \text{ is active}} W_{u,v} \geq \theta_v \quad (2.2)$$

The process proceeds in several time steps and terminates when no more activations are possible. Figure 2.2, shows the steps of information diffusion in this model. The number inside the node indicates its threshold. A red counter on the side indicates the current active edge weight of a node. If the number in the red counter equals or exceeds the threshold of a specific node, then the node gets activated. Kempe et al. [9] showed that finding the set  $S$  maximizing the influence function in the LT model is also NP-hard, and they use a reduction to the *Vertex Cover* problem.

A *Generalized Threshold Model* can be designed by removing the linearity from LT. The model follows the same structure as LT. However,  $v$  now relies upon an arbitrary *monotone threshold function*  $f_v$  which maps the subsets of  $v$ 's neighbours into the interval  $[0, 1]$ , with  $f_v(\emptyset) = 0$ . Let  $N$  represent the set of neighbours of  $v$  that are active at time step  $t - 1$ . Node  $v$  becomes active at time step  $t$  if and only if  $f_v(N) \geq \theta_v$ . The LT model is a special case of the generalized threshold model. The threshold function of the LT model takes the following form:

$$f_v(N) = \min \left( 1, \sum_{u \in N} W_{u,v} \right). \quad (2.3)$$

The General Threshold model is *submodular* if each function  $f_v$  is submodular. If this model is not submodular, its analysis may be intractable.

### 2.2.3 Triggering Model

The IC and the LT are specialized cases of the *triggering* model also introduced by Kempe et al. [9]. In this model, instead of having a threshold value, a node  $v$  has a quiescent subset  $T_v$ , called the *triggering set*, which affects the state of  $v$ . Initially, the process starts with a seed set  $S$ , and each node  $v$  chooses a triggering set  $T_v$  independently from each other, according to some distribution over the subsets of its neighbours. An inactive node  $v$  can be activated at time step  $t$ , if it has an active neighbour  $u$  in  $T_v$  at time step  $t - 1$ .

Another way to think of this process is by considering the *live-edge* technique. Edges of a graph  $G = (V, E)$  can be either live or blocked. If a node  $u$  is present in the triggering set  $T_v$  of  $v$ , the edge  $(u, v) \in E$  will be declared as live and claimed to be blocked otherwise. So,  $v$  is reachable from the seed set  $S$  by traversing a path made entirely out of live edges. The influence function  $\sigma$  is shown to be submodular in the triggering model.

## 2.3 Non-Adaptive Influence Maximization

The *non-adaptive influence maximization* is a computational problem, where given an influence graph  $G$  and an integer  $k \geq 1$ , we are asked to find a set of seeds  $S \subseteq V$  with  $|S| \leq k$  such that  $\sigma(S)$  is maximized. Without loss of generality, we assume that  $k \in [n]$ , and since the objective function is monotone,  $|S| = k$  for any solution of  $S$ .

Kempe et al. [1, 9] showed that function  $\sigma$  is monotone and submodular, therefore the following greedy algorithm achieves a  $1 - \frac{1}{e}$  approximation factor. Algorithm 1 is as follows: (i) start with an empty set of seeds  $S := \emptyset$ ; (ii) at each iteration  $t \in [k]$ , add to  $S$  the node  $v$  that maximizes the expected influence spread  $\sigma(S \cup \{v\})$ . Note that the greedy algorithm requires at each iteration to compute the value of function  $\sigma$ , for some set of seeds.

---

**Algorithm 1:** Non-Adaptive algorithm

---

**Require:** an influence graph  $G$ ;

- 1:  $S \leftarrow \emptyset$ ;
  - 2: **while**  $|S| < k$  **do**
  - 3:    $v^* \leftarrow \arg \max_{v \in V \setminus S} (\sigma(S \cup \{v\}) - \sigma(S))$ ;
  - 4:    $S \leftarrow S \cup \{v^*\}$ ;
  - 5: **end while**
  - 6: **return**  $S$ ;
- 

By using the fact that  $\sigma$  is a monotone submodular function with  $\sigma(\emptyset) = 0$ , the results of Nemhauser and Wolsey [10] on submodular function approximation imply that algorithm 1 will always produce a solution guaranteeing at least  $1 - [(k-1)/k]^k \geq 1 - 1/e$  times the optimal value, where  $e \simeq 2.71$  is the *Nepero number*.

At each step of algorithm 1, we are required to compute the value  $\sigma(S \cup \{v\})$  for any node  $v \in V \setminus \{v\}$  to find the local optimal influence function, but this has been shown to be computationally intractable as it is  $\#P$ -hard [25]. However, standard Chernoff bounds allow us estimate the value of  $\sigma$  through a polynomial number of Monte-Carlo simulations by introducing an arbitrarily small additive error  $\epsilon > 0$ , which depends on

the number of simulations [9]. In the remainder of the thesis, we will omit the additional term  $\epsilon$  to avoid unnecessary complicated formulas (except for the experimental part, where we set  $\epsilon = 0.05$ ). We will refer to this algorithm as the *non-adaptive greedy algorithm* in this thesis.

The two most widely studied non-adaptive IM algorithms are classified under the following categories: i) Monte-Carlo simulations-based approach [1] and, ii) Heuristic-based approach [26].

**Monte-Carlo simulation-based approach.** Kempe et al.’s non-adaptive greedy algorithm is an effective and simple algorithm that produces near optimal solution with a theoretical guarantee due to submodularity. They use Monte Carlo simulations to overcome the drawback of  $\#P$ -hard complexity. A Monte Carlo simulation is used to predict a model, by assigning a set of generated values based on a probability distribution to the uncertain variables. The simulations are computed independently on all the randomly generated values to obtain the results, which are then averaged to arrive at an estimate. Multiple instances of the graph  $G$  are created and the number of reachable nodes  $R(S)$  are computed for each of the generated instances. The value of  $R(S)$  is averaged and the expected number of reachable nodes by  $S$  is  $\sigma(S)$ .

Nevertheless, the overhead cost that is incurred with a simple greedy algorithm is huge, because for each candidate seed node, the greedy algorithm simulates the influence cascade for estimating the influence increment. The number of Monte Carlo simulations required by the greedy algorithm is  $O(mR)$ ,  $O(m)$  is the time required for each simulation and  $R$  is the number of simulations required ( $R \approx 10000$  [1] for a good estimate). The resulting time complexity for the greedy algorithm is  $O(kmnR)$ , where  $k$  is the total number of iterations, and  $O(n)$  is the number of simulations for each iteration.

Leskovec et al. [27] introduced the CELF algorithm which uses “lazy-forward” optimization to improve on the time complexity for over 700 times by avoiding unnecessary simulations. They studied the structural property of social networks to find that real networks are mostly power-law graphs, which means majority of the nodes have a small influence. The algorithm maintains a priority queue to store the marginal influence of a node  $v$  and the iteration number where it was computed. A node with a small influence in its previous iterations will not have a bigger marginal gain due to submodularity. Hence the simulation omit these nodes from its calculation without affecting the effectiveness of the algorithm.

The CELF algorithm was further improved by Goyal et al. [28] with CELF++ which ran 35% – 55% faster than CELF. Other simulation based approaches include that of

Cheng et al. [29] to reduce the number of computations of CELF. Another simulation based paper was by Ohsaka et al. [30], where they prune the number of BFSs required for reachability tests by reducing the number of vertices visited during each BFS. However, Monte Carlo simulations are computationally expensive and has scalability issues when the graph starts to grow.

**Heuristic-based approach.** In order to tackle with the expensive computations and large-scale networks, Chen et al. [31] used the degree discount technique which reduces the running time but has a much lower efficiency compared to the Monte Carlo approach. Borgs et al. [26] introduced the *reverse sampling technique (RIS)* and utilizes the concept of *reverse reachable (RR) sets* and *Random RR sets*. An instance  $g$  of graph  $G$  is generated by using the value of  $p_{(uv)}$  to add edges in  $g$  between nodes  $u$  and  $v$  of  $g$ . The RR set for a node  $u$  in  $g$  is the set of nodes that have a directed path to  $u$  in  $g$ . Due to the randomness of generating  $g$  using the probability on edges, we have a distribution  $\mathcal{G}$  of  $g$ . To generate a random RR set on  $g$ , the instance  $g$  is randomly sampled from  $\mathcal{G}$ .

The idea behind Borgs et al.'s algorithm RIS, is to generate a large set of random RR sets, denoted as  $\mathcal{R}$ . Using the greedy maximum coverage algorithm from [32] the goal is to find a seed set  $S^*$  of size  $k$  which covers the maximum number of RR-sets, this implies that  $S^*$  will have the maximum expected spread over all the size  $k$  seed sets of  $G$ . Borgs et al.'s technique reduces the estimation of a large number of node sets hence reducing the overhead time complexity induced by simulation based approaches. To calculate the number of RR sets that needs to be generated, RIS proposed a predefined threshold value  $\tau$ . Defining a proper  $\tau$  value results in a seed set  $S$  being generated at almost linear time complexity with an approximation ratio of  $(1 - \frac{1}{e} - \epsilon)$ . The RIS still has some overhead cost due to the dependency of  $\tau$  on the number of nodes and edges examined.

Tang et al. [33] provided an improved version of the RIS algorithm called  $TIM^+$ . They mainly cut down the threshold cost by replacing it with a predefined constant  $\theta$ . TIM removes the dependency from the edges, and uses Chernoff bounds to estimate the value for  $\theta$ . The estimated time complexity of the  $TIM^+$  algorithm is  $O((k+l)(m+n)\log n/\epsilon^2)$ , where  $l$  is a constant and is set to 1. For the experimental part of the thesis (section 3.5), we use  $TIM^+$  as the base non-adaptive greedy algorithm. Tang et al. [34] further improved on the efficiency of  $TIM^+$  with the IMM algorithm by bounding the size of  $\mathcal{R}$  using a martingale approach.

More improvements on the IMM algorithm was brought by Nguyen et al. [35], Huang et al. [36], Tang et al. [37] where they curb the excessive generation of RR sets when the value of  $k$  is large. They work with the *Stop-and-Stare* algorithm [35], where they

set a stopping time limit  $T$ . The sampling process is stopped when  $T$  is reached and the algorithm checks if the seed set  $S$  fulfils the  $(1 - 1/e - \epsilon)$ -approximation. Once the current  $S$  reaches the approximation, the algorithm terminates.

The heuristic-based approach has better scalability and time complexity, but lacks in the department of performance guarantees that the simulation-based approach can provide.

## 2.4 Adaptive Influence Maximization

### 2.4.1 Adaptive Maximization

Golovin and Krause [14] studied the *Adaptive Maximization (AM)* setting, and later applied it to the context of Influence Maximization by introducing its adaptive version. The general problem of AM is defined as follows: consider a graph  $G = (V, E)$  and a set of states  $O = \{alive, dead\}$ . Each edge  $e \in E$  is in one of the possible states, so that the state of the edges can be represented using a function  $\phi : E \mapsto O$ , called the *realisation*. Hence, we can state that  $\phi(e)$  is the state of  $e$  under realisation  $\phi$ . A random variable  $\Phi$  taking values from all the possible realisations, according to a given probability distribution, is called a *random realisation*. For a given realisation  $\phi$ , let  $p(\phi)$  denote the probability that  $\Phi = \phi$ .

We probe each  $e \in E$  sequentially and observe its state  $\Phi(e)$ . The observations made after each pick can be represented as a *partial realisation* function  $\psi$ , over some subset of  $E$ . The domain of  $\psi$ , i.e., the set of edges observed in  $\psi$  is represented by  $dom(\psi)$ . A partial realisation function is said to be consistent with a realisation  $\phi$  if they are equal in  $dom(\psi)$  and is written as  $\phi \sim \psi$ . When  $\psi$  and  $\psi'$  are both consistent with some  $\phi$ , and  $dom(\psi) \subseteq dom(\psi')$ ,  $\psi$  is called the *subrealisation* of  $\psi'$ .

The node selection for the next round is strategized by an *adaptive policy*  $\pi$ , which is a function from a set of partial realisations to  $E$ . When  $\psi$  is not present in the domain of  $\pi$ , the termination of the policy occurs upon observations of  $\psi$ . The domain of  $\pi$ , denoted by  $dom(\pi)$ , should be closed under subrealisations. The function we want to maximize is in the form of  $f : 2^E \times O^E \rightarrow \mathbb{R}_{\geq 0}$ , and takes as input the realisation  $\phi$ , and the set of edges  $E(\pi, \phi)$  selected by the considered adaptive policy  $\pi$  under realisation  $\phi$ . Unfortunately, we do not know what is the final realisation  $\phi$ . The AIM overcomes this problem, and its goal is to find a policy  $\pi^*$  that maximizes the expected value  $\mathbb{E}_{\Phi}(f(E(\pi, \Phi), \Phi))$  subject to  $E(\pi, \Phi) \leq k$ , where  $k$  is the maximum number of edges that a policy can select.



## 2.4.2 Adaptive Influence Maximization

Now, we adapt the previous framework of adaptive maximization to the IM problem and get the AIM Problem. Differently from the non-adaptive setting, in which all the seeds are selected at the beginning, an adaptive policy activates the seeds sequentially in  $k$  steps, one seed at each step, and the decision on the next seed to select is based on the feedback resulting from the observed spread of previously selected nodes. The feedback models considered in this work are the full-adoption feedback: where the adaptive policy observes the entire spread of the selected seed node; and myopic: when a node is selected, the adaptive policy observes just the state of its neighbours. Chapters 3 and 4 provide the technical definitions related to the adaptive influence maximization under specific feedback models.

**Adaptivity Gap.** The adaptivity gap on an input instance is defined as the ratio between the optimal adaptive policy and the optimal non-adaptive policy. Since an adaptive policy is strictly better than a non-adaptive one, if the value of AG is closer to 1, the non-adaptive policy is considered to be as efficient as the adaptive policy, and we can use the non-adaptive policy as a good approximation to the adaptive one, due to its simplicity. Chapter 3 Section 3.1 provides a more formal definition of the adaptivity gap.

## 2.4.3 Feedback Models of AIM

Depending on the type of network state that a policy can observe at each time, we have two general models of AIM: *edge-feedback model*, in which the network state is a set of active edges, and *node-feedback model*, in which the network state is a set of active nodes.

In this section, we discuss some particular models of AIM, that can be either node-feedback, or edge-feedback. In particular, here we discuss the full-adoption feedback, myopic feedback, and the *general feedback* model under the edge feedback.

### 2.4.3.1 Full-Adoption Feedback Model

The full-adoption feedback selects a node and returns the entire cascade as feedback for the next round. Each node is selected one by one, and the selection needs to be done at the beginning of each round. Once the budget  $k$  is reached, the process terminates and returns the seed set. To solve the AIM problem in the full-adoption feedback

model, Golovin and Krause [14] designed an adaptive version of the non-adaptive greedy algorithm. Golovin and Krause [14] showed that the full adoption feedback can guarantee a good approximation ratio by applying the concepts of monotone and submodular functions to the adaptive framework.

The algorithm greedily selects nodes one by one from the graph  $G$  until the budget  $k$  has been reached. For each node  $u \in V$ , the conditional expected marginal gain is determined. The node with the largest estimate for the gain is added to  $S$ , and the random realisation of the selected node  $u$  is observed. The set of partial realisations is updated by adding the edges that has been activated by  $u$ . Golovin and Krause [14] proved that for the adaptive IM problem, the greedy policy  $\pi^G$  determined by a greedy adaptive algorithm achieves at least  $(1 - \frac{1}{e})$  of the optimal policy, when  $f$  is adaptive monotone and submodular.

Nevertheless, the greedy adaptive algorithm can not reach the above approximation guarantee in polynomial time, because to select a seed node adaptively in each round, we need to identify the node  $u$  with the highest expected spread on the residual graph  $G_u$ , which is obtained by removing all the activated nodes from the previous rounds via the feedback. However, computing the exact spread is  $\#P$ -hard [25]. Vaswani and Lakshmanan [38] overcame this drawback by allowing errors on the expected spread estimation. By introducing a parameter  $\eta$  to bound the error and using the greedy approach of Dar and Shapira [20], an approximation guarantee of  $1 - 1/e^\eta$  is reached. Vaswani and Lakshmanan also extended their approach to batch mode, where instead of selecting one seed node and observing the influence, we can select a batch of seed nodes, and observe its spread to do the node selection for the next batch.

Together with the problem of finding an optimal adaptive policy, determining the adaptivity gap for the full adoption feedback is also a challenge. Chen and Peng [15] derive the upper and lower bounds on the AG when the graphs are *in-arborescences* and *out-arborescences*. An arborescence is a directed rooted tree. An in-arborescence (resp. out-arborescence) is an arborescence when the edges are directed from the leaves to the root (resp. from the root to the leaves).

Since the information propagates from the leaves to the root in an in-arborescence, the boundary of active nodes shrinks. The leaves are the set of discovered nodes, and the boundary forms the subset of the set of nodes reachable from the leaves. Using these properties, the AG has been shown to be between  $\frac{e}{(e-1)}$  and  $\frac{2e}{(e-1)}$ . In the case of an out-arborescence graph, the main observation has been that the predecessors of each node in the graph forms a directed line. In order to activate a certain node  $u$ , all its predecessors must be activated one at a time. For out-arborescences the AG has been shown to be between  $\frac{e}{(e-1)}$  and 2.

### 2.4.3.2 Myopic Feedback Model

The main drawback of the full adoption feedback model is the delay in the process: select nodes one by one, observe its complete diffusion, return it as a feedback for the selection of the next seed node, and continue this over  $k$  steps. To overcome this potential set back, we could select a seed node  $u$  and check which of its neighbours are getting influenced by it, and return the set of neighbours influenced by  $u$  as a feedback for the next round. Since only the active neighbours of the selected seeds are returned as feedback, the model is called myopic. The myopic model has been mentioned by Golovin and Krause [14], and in a revised arXiv version of the same paper, the authors claimed that the objective function  $f$  in the myopic feedback model is not adaptive submodular. The adaptive submodularity condition is violated since the seed node selection process is not made after a diffusion terminates. Instead, the diffusion of a certain round ceases after the current node activates its neighbours.

Peng and Chen [39] confirmed the conjecture of Golovin and Krause [14], which states that the adaptive greedy algorithm with myopic feedback is a constant approximation of the adaptive optimal solution. They obtain an upper and lower bound for the AG under the IC model with myopic feedback,  $[\frac{e}{e-1}, 4]$ . They also proved that the adaptive and the non-adaptive greedy algorithms have an approximation ratio of  $\frac{1}{4}(1 - \frac{1}{e})$  under the myopic model. The authors conclude by stating that the myopic feedback is not beneficial since the approximation ratios for both the adaptive and non-adaptive greedy algorithms are the same. Also, the solution is  $\frac{e^2+1}{(e+1)^2} \approx 60\%$  of the optimal value, which is less than the results obtained by Kempe et al. [9]. In simple words, the performance is not commendable since the feedback received is not enough to determine the selection of the best possible node in the next round. However, for both practical and theoretical reasons, studying the myopic model is useful.

### 2.4.3.3 Partial and General Feedback Model

Yuan and Tang [40] have expressed the trade-off between performance and delay in the full adoption and the myopic feedback model. They developed the *partial feedback* model, which however is not adaptive submodular, due to the same reasons as the myopic model. The partial feedback model can capture a more realistic scenario of the marketing world. The model considers a general  $\alpha$ -greedy policy, depending on a *control parameter*  $\alpha \in [0, 1]$  given as input. At each round  $r$ , a policy selects a new seed node  $v$ , and the cascade proceeds via *time slots* in which the depth of the cascade increases by one; we can equivalently denote each time slot with the depth  $d$  of the cascade generated by  $v$ . At each round  $r$  and time slot  $d$ , the policy observes the state of all the edges

involved in the cascade, and the round ends when the following condition is verified:

$$\alpha \leq \frac{\sum_{v \in V} p_v(S_r, \psi_{r,d})}{|V \setminus O_{r,d}|}, \quad (2.4)$$

where  $\psi_{r,d}$  represent the observations,  $O_{r,d}$  represents the set of nodes whose activation probabilities are zero at round  $r$  and at time slot  $d \geq 0$ , and  $\sum_{v \in V} p_v(S_0, \psi_{r,d})$  is the expected number of activated nodes, under the activation probability of  $p_v(S_r, \psi_r)$ , with  $S_r$  being the seed set. After this, a new round starts, and the previous process continues iteratively until the seed set contains  $k$  nodes. Informally, each round terminates when the expected probability that a node becomes active is higher than  $\alpha$ . Observe that, the partial model becomes a full adoption feedback when  $\alpha = 1$ , and non-adaptive IM when  $\alpha = 0$ .

Tong and Wang [41] introduced the *general feedback* model. Such a model is defined as the partial model, except for the fact that each round terminates after exactly  $d$  time slots, where  $d \geq 0$  is an integer given as input. Observe that for  $d = 1$  we get the myopic feedback, and for  $d = \infty$  we get the full adoption feedback model. Since the model is not adaptive submodular, one cannot immediately guarantee that the greedy strategy is a good approximation algorithm. To overcome this problem, the authors introduced a new metric called the *regret ratio*. The regret ratio  $R$  of a diffusion process is the maximum ratio between (i) the best marginal profit obtained for a given seed set and a given round, when a new seed node is added to  $S$  at the beginning of round  $r$ , and (ii) the best marginal profit achieved when the new seed node is added to  $S$  at the end of round  $r$ , i.e., after  $d$  time slots. The regret ratio has been used to show that the greedy algorithm guarantees an approximation factor of  $1 - e^{-1/R}$ .

#### 2.4.4 Multi-Round Influence Maximization

Sun et al. [42] works on a marketing scenario, where the advertiser considers multiple rounds to market a product. In particular, instead of considering one round for the diffusion process, the advertiser conducts the diffusion process over  $T$  rounds. The advertiser has a budget  $k$  for each round, i.e., a seed set  $S_t$  of at most  $k$  nodes can be selected at each round. An independent diffusion process is carried out in each round  $t$ , where a subset of nodes are activated, starting from the initial seed set  $S_t$ . The total influence spread over the  $T$  rounds is calculated as the union of the sets of nodes activated over all the  $T$  rounds. The goal is to find the seed sets that maximize the expected total influence spread. The authors work on both the adaptive and non-adaptive version and design three greedy algorithms.

In the non-adaptive version, the advertiser needs to find all the  $T$  seed sets before the start of the diffusion process. The authors consider two different algorithms: *cross-round* and *within-round*. The within-round has an approximation ratio of  $1 - e^{-(1-\frac{1}{e})} \approx 0.46$ . However, the cross-round has a better approximation factor of  $1/2 - \varepsilon$ , but has a higher running time.

For the adaptive version, in each round, the advertiser needs to select the seed set for that round and observe its influence spread. Once the budget  $k$  is reached for a specific round  $r$ , the next round  $r + 1$  receives the influence spread of the previous  $r$  rounds as feedback. In the next round, based on the observations of the propagation made by the previous rounds, the seed set for that round is selected. The authors designed a greedy approximation algorithm called *AdaGreedy*, and show that the algorithm guarantees a  $1 - e^{-(1-\frac{1}{e})} - \varepsilon$  approximation ratio.

### 2.4.5 Other variations of AIM

**Non-monotone models.** Gotovos et al. [43] proposed the *adaptive random greedy policy* which generalizes the random greedy algorithm by Buchbinder et al. [44]. The policy can be used for both monotone and non-monotone functions, however for non-monotone functions the policy achieves a  $1/e$  approximation ratio. The algorithm introduces dummy nodes to the ground set which ensures that the algorithm never picks a node with a negative marginal gain. Instead of selecting the node with the highest marginal gain, the algorithm randomly selects a node from a set, which contains  $k$ -nodes with the highest marginal gain.

The time required for the algorithm is majorly dependent on the number of times the objective function is evaluated and it is  $O(nk)$  for [43] which is their main drawback. Tang [45] proposes a linear-time policy for the same problem, where the time is not dependent on the value of  $k$  and requires  $O(\frac{n}{\varepsilon^2} \frac{1}{\log \varepsilon})$  evaluations achieving the same approximation factor of  $1/e$ . Their algorithm also speeds up on the monotone variant of the problem. The idea behind their policy is to sample a random set of size  $\min[qn, n]$ , where  $q$  is a parameter dependent on the value of  $k$  and used to bound the time. Out of that random set, the seed with the  $i$ -th largest marginal gain w.r.t the current partial realisation is selected, where  $i$ 's value is sampled at random from  $(0, s]$ , and  $s$  is another parameter relying on  $n$  and  $k$ . If the selected node's marginal gain is non-negative, it is added to the seed set.

A recent work of Amanatidis et al. [46] on non-monotone adaptive submodular function subject to knapsack constraints, which are more stringent by nature, achieves a constant factor approximation and runs  $O(n \log(n))$  evaluations.

**DR-submodularity.** Guo and Wu [47] studied AIM when the model is adaptive DR-submodular or *diminishing return submodular*. DR-submodularity is a generalization of the submodularity property and are generally defined over integer lattices (seeding vectors) instead of seed sets. Inspired by the works of Chen et al. [48] on influence maximization over lattices, Guo and Wu extended their work to the adaptive setting. They consider a new scenario where after activating a seed node, we wait to see if the selected node is willing to be the seed node. If the selected node refuses to be the seed, we either raise the cost of the selected seed to be the seed, or we select a new node to become the seed. Only when a node accepts to be the seed, we move on to observing its spread. They named this the *Adaptive-IMMA* problem which stands for AIM with multiple activations. They show that the AIM problem with dr-submodularity returns an approximation ratio of  $(1 - 1/e)$ .

## 2.5 Stochastic Multi-Value Probing (SMP)

Previous works on finding the adaptivity gap under independent cascade includes the usage of the stochastic multi-value probing model which was introduced in the adaptive setting by Bradac et al. [49]. The SMP problem is a generalized version of the stochastic probing model, whose adaptive version was studied by Gupta et al. [50].

The stochastic probing model is defined as follows, we are given an universal set  $V = [n]$  and set of edges  $E$  with all the combination of possible edges. Each of these edges  $e \in E$  has a probability  $p_e$  associated with it and denotes the activation probability of  $e$ . We have a set  $A \subseteq V$  chosen at random from a family of subsets of  $V$ ,  $\mathcal{S}$ . We probe each of the elements,  $v \in A$  to reveal whether  $v$  is active in  $A$  or not due to the probability on the edges. However, we are only allowed to probe certain subsets  $S \in \mathcal{S}$ , which fulfills the budget requirements (constraints). The goal here is to find a probing strategy that maximizes the expected number of active nodes in the set  $A$ , which is  $\mathbb{E}_A[f(A \cup S)]$ .

The utility function is monotone submodular and the probing strategy can be transformed to an adaptive setting, where after probing each element  $v \in A$ , we observe its outcome, and then continue with the next probe. The adaptive probing strategy is designed using a binary rooted decision tree,  $\mathcal{T}$ . The nodes in the tree represents the elements in the ground set  $V$ , and each of these nodes (except the leaves) have two outgoing edges, *yes*, when the edge  $e$  is active with probability  $p_e$ , and *no*, when the edge  $e$  is inactive with probability  $1 - p_e$ . Note that when the edge is active, it automatically activates the inactive node, its direct child. The probing process terminates, when we

reach the leaf nodes, where no more activations are possible. The stem of a tree represents the path of unsuccessful probes. Whenever a stem is observed, it is stitched back to the root which causes a loss and the final adaptivity gap becomes 3 [50].

The above mentioned problem can be generalised by using more than the Bernoulli random variables, and allowing multiple values on the edges. Bradac et al. [49] considered adding multiple values to the edges of the rooted decision tree  $\mathcal{T}$ , which no longer restricts itself to binary values. Due to the loss caused by sticking back of the stem to perform the induction over its subtrees, Bradac et al. [49] observes at each step a single node, instead of an entire stem and thus retrieving the loss and upper bounding the AG to 2.

The idea is the following, we have a non-adaptive strategy that randomly chooses a path in the decision tree with the same probability as an optimal adaptive strategy would have. This randomised non-adaptive strategy can be described as follows: starting from the root, one of its children is probed randomly with the same probability that the optimal adaptive policy would have probed this child. The result is stored under a partial realisation  $T_{v_1}$ , and the iteration continues till a leaf is reached. The non-adaptive policy selects this path from the root to the leaf.

By combining the non-adaptive policy with optimal adaptive policy, Bradac et al. [49] defines the super adaptive strategy, which gives a node two chances of probing. For any  $i \geq 1$ , when the non-adaptive policy decides to include the  $i$ -th node in the path towards the leaf node, we notice that the expected increment in the utility of the non-adaptive policy will be at least half of the expected increment achieved by the super adaptive strategy. Asadpour and Nazerzadeh [51] showed that the adaptivity gap of SMP under cardinality constraints is upper bounded by 1.58. Where as, Bradac et al. [49] proved that the upper bound of adaptivity gaps under the downward-closed families of set constraints (cardinality, knapsack and matroid), which means if a set can be probed, even their subsets can be probed, are 2 and tight for both the theorems [49, 51].

Using the above techniques to find the adaptivity gap for SMP problems, Chen and Peng [15] studied different graph classes with AIM under full adoption and IC. They use multilinear extension and Poisson process from [51] to model the AIM problem as a SMP problem. Since the results of [51] are based on cardinality constraints, modelling it for the IM problem becomes easier. For myopic feedback under IC, Peng and Chen [39] relies on the decision tree technique from [49] to build the artificial diffusion model, which gives each seed multiple chances to influence its neighbour and bound the AG for the myopic model.

The techniques involved in this thesis are inspired from the literature above, however, we resort to a simpler and direct analysis to get better bounds on the adaptivity gap under IC with both the feedback models.



## Chapter 3

# Better bounds on the Adaptivity Gap of Influence Maximization under Full-Adoption feedback

In this chapter, we consider the independent cascade model with full-adoption feedback, and show the first sub-linear upper bound on the adaptivity gap, (ratio between the optimal adaptive policy and the optimal non-adaptive policy [3.1]) that holds for general graphs. In detail, we show that the adaptivity gap is at most  $\lceil n^{1/3} \rceil$  (Theorem 3.5), where  $n$  is the number of nodes in the graph. Moreover, we tighten the upper bound on the adaptivity gap for in-arborescences by showing that it is at most  $\frac{2e^2}{e^2-1} < \frac{2e}{e-1}$  (Theorem 3.1).

Using similar techniques, we study the adaptivity gap of two classes of influence graphs:  $\alpha$ -bounded-degree graphs, which is the class of influence graphs where the sum of node degrees higher than two is at most  $\alpha$ , and  $(\beta, \gamma)$ -bounded-activation graphs, where the nodes are partitioned into influential nodes, which are at most  $\beta$  nodes, and non-influential, where each of these nodes can infect at most  $\gamma$  neighbours in expectation, where the value of  $\gamma$  is less than 1.  $\alpha$ -bounded-degree graphs can be encountered in several graph topologies (see, for instance, Example 3.1), and  $(\beta, \gamma)$ -bounded-activation graphs are well-motivated by social networks in which most nodes have a limited power of influence. We show that the adaptivity gap of  $\alpha$ -bounded-degree and  $(\beta, \gamma)$ -bounded-activation graphs is upper-bounded by  $\sqrt{\alpha} + O(1)$  and  $\sqrt{\beta} + \frac{1}{1-\gamma}$  (Theorems 3.7 and 3.11) respectively, and these values are smaller than that of  $O(n^{1/3})$  for several influence graph classes. Furthermore, in 0-bounded-degree graphs, i.e. undirected graphs in which each connected component is a path or a cycle, the adaptivity gap is at most  $\frac{3e^3}{e^3-1}$  (Theorem 3.9).

To prove our bounds, we introduce new techniques to connect adaptive policies with non-adaptive ones that might be of their own interest (further details are given in the paragraph “General outline of the proof technique” in Section 3.2). In particular, we resort to a simple randomized *hybrid non-adaptive policy*, that differs from the main approaches previously used in adaptive influence maximization and other adaptive optimization problems: (i) the *Poisson process* [52] combined with the *multi-linear extension of submodular set-functions* [53], which represent the main probabilistic technique adopted by Asadpour and Nazerzadeh [51] and Chen and Peng [15], and (ii) the *random walk on the decision tree* [54, 55], that is a tool applied by Bradac et al. [49], Gupta et al. [50] and Peng and Chen [39].

In the experimental section 3.5, we take into account a cutting age non-adaptive greedy algorithm, called  $TIM^+$ , and design its adaptive version to compare their performance ratio, called the greedy adaptivity gap. To compute the efficiency of the seed set selected by  $TIM^+$ , we design another adaptive algorithm, which take as input the predetermined seed set selected by  $TIM^+$ , and running the adaptive algorithm on top of it. We observe that the ratio is closer to one, thus implying the efficiency of  $TIM^+$  when compared to its adaptive counterpart.

## Related Works.

Adaptivity gaps for the problem of maximizing stochastic monotone submodular functions have been studied by Asadpour and Nazerzadeh [51]. There exists a series of work engaged with adaptivity gaps for a two-step adaptive influence maximization problem [56–59]. Gupta et al. [50, 55] and Bradac et al. [49] worked on the adaptivity gaps for stochastic probing.

A recent line of studies [15, 17, 39] focuses on finding the adaptivity gap on different graph classes using the classical feedback models. Peng and Chen [39] confirmed a conjecture of Golovin and Krause [2], which states that the adaptivity gap of the independent cascade model with myopic feedback is constant. In particular, they showed that the adaptivity gap belongs to  $[\frac{\epsilon}{\epsilon-1}, 4]$ . Chen et al. [17] introduced the greedy adaptivity gap, which compares the performance of the adaptive and the non-adaptive greedy algorithms. They show that the infimum of the greedy adaptivity gap is  $1 - \frac{1}{\epsilon}$  for every combination of diffusion and feedback models.

The most relevant related work to our results in this chapter is that of Chen and Peng [15], as they derive upper and lower bounds on the adaptivity gap under the independent cascade model with full-adoption feedback, when the considered graphs are in-arborescences, out-arborescences, or one-directional bipartite graphs. In particular,

they show that the adaptivity gaps of in-arborescences and out-arborescences are in the intervals  $\left[\frac{e}{e-1}, \frac{2e}{e-1}\right]$  and  $\left[\frac{e}{e-1}, 2\right]$  respectively, and they provide a tight bound of  $\frac{e}{e-1}$  on the adaptivity gap of one-directional bipartite graphs. Under the more general triggering model, a constant upper bound of 2 on the adaptivity gap of one-directional bipartite graphs was provided by Fujii and Sakaue [60].

### Contributions.

Our main notable result in this chapter is the first sub-linear upper bound that holds for any graph. Specifically, we show that the adaptivity gap is upper-bounded by  $\lceil n^{1/3} \rceil$ , where  $n$  is the number of nodes in the graph. Moreover, we improve over the known upper bound for in-arborescences from  $2e/(e-1) \approx 3.16$  to  $2e^2/(e^2-1) \approx 2.31$ .

Then, we study  $\alpha$ -bounded-degree graphs, that is the class of undirected graphs in which the sum of node degrees higher than two is at most  $\alpha$ , and show that the adaptivity gap is upper-bounded by  $\sqrt{\alpha} + O(1)$ ; we also show that in 0-bounded-degree graphs, i.e. undirected graphs in which each connected component is a path or a cycle, the adaptivity gap is at most  $3e^3/(e^3-1) \approx 3.16$ .

We also consider  $(\beta, \gamma)$ -bounded-activation graphs [18], where all nodes but  $\beta$  influence in expectation has at most  $\gamma \in [0, 1)$  neighbors each; for this class of influence graphs we show that the adaptivity gap is at most  $\sqrt{\beta} + \frac{1}{1-\gamma}$ . To prove our bounds, we introduce new techniques to relate adaptive policies with non-adaptive ones that might be of their own interest.

Finally, we have some experiments to determine the performance efficiency of a specific widely used non-adaptive greedy algorithm,  $TIM^+$ . We build a greedy adaptive equivalent of  $TIM^+$  and discuss about the quality of seed set generated by both the versions of  $TIM^+$ , the memory, and time overhead. We conclude that  $TIM^+$  as a greedy non-adaptive algorithm performs quite well when compared to its adaptive counterpart and the main focus should be on the requirements of the computational costs to run these algorithms.

### Organization of the Chapter.

The preliminaries for this chapter is defined in Section 3.1. Sections 3.2–3.4 are devoted to the main technical contribution of the chapter, i.e., upper bounds on the adaptivity gap of in-arborescences, general graphs and other influence graphs ( $\alpha$ -bounded-degree and  $(\beta, \gamma)$ -bounded-activation graphs). Section 3.5 has some experiments comparing the results of  $TIM^+$ , adaptive greedy under  $TIM^+$ , and adaptive greedy with the seeds

generated by  $TIM^+$  for different network types and parameters under the IC and LT models.

### 3.1 Preliminaries

For two integers  $h$  and  $k$ ,  $h \leq k$ , let  $[k]_h := \{h, h + 1, \dots, k\}$  and  $[k] := [k]_1$ .

**Non-adaptive Influence Maximization** The *non-adaptive influence maximization problem under the IC model* is a computational problem that is defined as follows: given an influence graph  $G$  and an integer  $k \geq 1$ , we are asked to find a set of seed nodes  $S \subseteq V$  with  $|S| = k$  such that  $\sigma(S)$  is maximized. Without loss of generality, we assume that  $k \in [n]$  and, since the objective function is monotone,  $|S| = k$  for any solution  $S$ . A detailed overview is presented in Sections 2.1–2.3 of Chapter 2.

**Adaptive Influence Maximization** Differently from the non-adaptive setting, in which all the seed nodes are activated at the beginning and then the influence spread is observed, an *adaptive policy* activates the seeds sequentially in  $k$  steps, one seed node at each step, and the decision on the next seed node to select is based on the feedback resulting from the observed spread of the previously selected nodes. The feedback model considered in this work is *full-adoption*: when a node is selected, the adaptive policy observes its entire influence spread.

An adaptive policy under the full-adoption feedback model is formally defined as follows. Given a live-edge graph  $L$ , the *realization*  $\phi_L : V \rightarrow 2^V$  associated to  $L$  assigns to each node  $v \in V$  the value  $\phi_L(v) := R(\{v\}, L)$ , i.e., the set of nodes activated by  $v$  under a live-edge graph  $L$ . Given a set  $S \subseteq V$ , a *partial realization*  $\psi : S \rightarrow 2^V$  is the restriction to  $S$  for some realization, i.e., there exists a live-edge graph  $L$  such that  $\psi(v) = \phi_L(v)$  for any  $v \in S$ . Given a partial realization  $\psi : S \rightarrow 2^V$ , let  $dom(\psi) := S$ , i.e.,  $dom(\psi)$  is the domain of partial realization  $\psi$ , let  $R(\psi) := \bigcup_{v \in S} \psi(v)$ , i.e.,  $R(\psi)$  is the set of nodes reached/activated by the diffusion process when the set of seed nodes is  $S$ , and let  $f(\psi) := |R(\psi)|$ . A partial realization  $\psi'$  is a *sub-realization* of  $\psi$  (or, equivalently,  $\psi' \subseteq \psi$ ), if  $dom(\psi') \subseteq dom(\psi)$  and  $\psi'(v) = \psi(v)$  for any  $v \in dom(\psi')$ . We observe that a partial realization  $\psi$  can be equivalently represented as  $\{(v, R(\{v\}, L)) : v \in dom(\psi)\}$  for some live-edge graph  $L$ .

An adaptive policy  $\pi$  takes as input a partial realization  $\psi$  and, either returns a node  $\pi(\psi) \in V$  and activates it as seed, or interrupts the activation of new seed nodes, e.g., by returning a string  $\pi(\psi) := STOP$ ; furthermore, we assume (w.l.o.g.) that an

adaptive policy  $\pi$  chooses each seed based only on the observation of  $R(\psi)$  (i.e., the set of nodes activated by the previous seeds) and  $|\text{dom}(\psi)|$  (i.e., the number of seeds previously selected), that is,  $\pi(\psi) = \pi(R(\psi), |\text{dom}(\psi)|)$ . An adaptive policy  $\pi$  can be run as in Algorithm 2. The *expected influence spread* of an adaptive policy  $\pi$  is defined as  $\sigma(\pi) := \mathbb{E}_L[f(\psi_{\pi,L})]$ , i.e., it is the expected value (taken on all the possible live-edge graphs) of the number of nodes reached by the diffusion process at the end of Algorithm 2. We say that  $|\pi| = k$  if policy  $\pi$  always return a partial realization  $\psi_{\pi,L}$  with  $|\text{dom}(\psi_{\pi,L})| = k$ . The *adaptive influence maximization problem under the IC model* is the computational problem that, given an influence graph  $G$  and an integer  $k \geq 1$ , asks to find an adaptive policy  $\pi$  that maximizes the expected influence spread  $\sigma(\pi)$  subject to a constraint  $|\pi| = k$ .

---

**Algorithm 2:** Adaptive algorithm
 

---

**Require:** an influence graph  $G$  and an adaptive policy  $\pi$ ;

**Ensure:** a partial realization;

- 1: let  $L$  be the live-edge graph;
  - 2: let  $\psi := \emptyset$  (i.e.,  $\psi$  is the empty partial realization);
  - 3: **while**  $\pi(\psi) \neq \text{STOP}$  **do**
  - 4:  $v := \pi(\psi)$ ;
  - 5:  $\psi := \psi \cup \{(v, R(\{v\}, L))\}$ ;
  - 6: **end while**
  - 7: **return**  $\psi_{\pi,L} := \psi$ ;
- 

### Adaptivity Gap.

Given an influence graph  $G$  and an integer  $k \geq 1$ , let  $\text{OPT}_N(G, k)$  (resp.  $\text{OPT}_A(G, k)$ ) denote the optimal value of the non-adaptive (resp. adaptive) influence maximization problem with input  $G$  and  $k$ . Given a class of influence graphs  $\mathcal{G}$  and an integer  $k \geq 1$ , the *k-adaptivity gap* of  $\mathcal{G}$  is defined as

$$AG(\mathcal{G}, k) := \sup_{G \in \mathcal{G}} \frac{\text{OPT}_A(G, k)}{\text{OPT}_N(G, k)},$$

and measures by how much does an adaptive policy outperform a non-adaptive solution for the influence maximization problem applied to influence graphs in  $\mathcal{G}$ , when the maximum number of seed nodes is  $k$ . The *adaptivity gap* of  $\mathcal{G}$  is defined as  $AG(\mathcal{G}) := \sup_{k \geq 1} AG(\mathcal{G}, k)$ . We observe that for  $k = 1$  or  $n \leq k$  the *k-adaptivity gap* is trivially equal to 1, thus we omit such cases in the following.

### 3.2 Adaptivity Gap for In-arborescences

An *in-arborescence* is a graph  $G = (V, E)$  that can be constructed from a rooted tree  $T = (V, F)$ , by adding an edge  $(v, u)$  in  $E$ , if  $u$  is a father of  $v$  in tree  $T$ . By exploiting the shrinking boundary property, refer to figure 3.1b and lemma 3.3, an upper bound of  $\frac{2e}{e-1} \approx 3.16$  on the adaptivity gap of in-arborescences has been provided in [15]. In this section we provide an improved upper bound for such graphs.

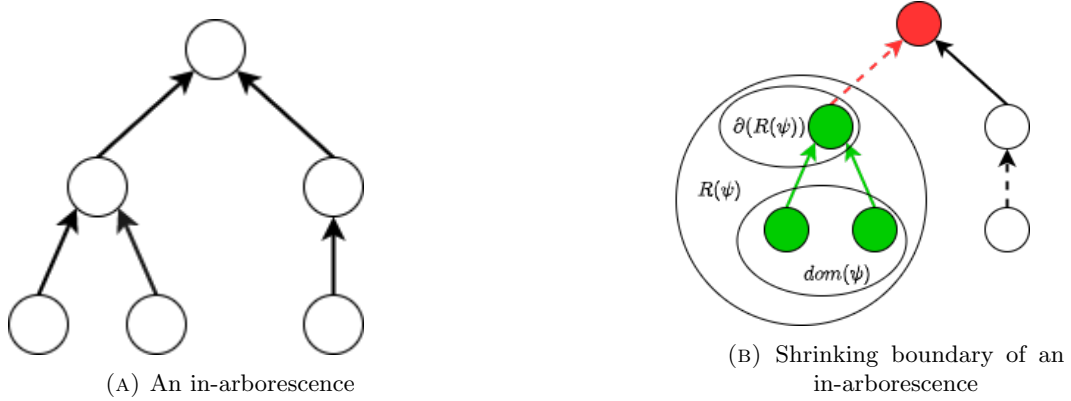


FIGURE 3.1: In-arborescence

**Theorem 3.1.** *If  $\mathcal{G}$  is the class of all the in-arborescences, then*

$$AG(\mathcal{G}, k) \leq \frac{2}{1 - (1 - 2/k)^k} \leq \frac{2e^2}{e^2 - 1} \approx 2.31, \quad \forall k \geq 2.$$

Let  $G = (V = [n], E, (p_{uv})_{(u,v) \in E})$  be an in-arborescence, where  $n > k$  is the number of nodes. To show the claim of Theorem 3.1, we need some preliminary notations and lemmas. Given a partial realization  $\psi$ , and a node  $v \in V$ , let

$$\Delta(v|\psi) := \mathbb{E}_L[f(\psi \cup \{(v, R(\{v\}, L))\})] - f(\psi)|\psi \subseteq \phi_L],$$

i.e.,  $\Delta(v|\psi)$  is the expected increment of the influence spread due to node  $v$  when the observed partial realization is  $\psi$ . We have the following claim (from [2]), holding even for general graphs, whose proof is trivial.

*Claim 1* (Adaptive Submodularity [2]). Let  $G$  be an arbitrary influence graph. For any partial realizations  $\psi, \psi'$  of  $G$  such that  $\psi \subseteq \psi'$ , and any node  $v \notin R(\psi')$ , we have that  $\Delta(v|\psi') \leq \Delta(v|\psi)$ .

An adaptive policy  $\pi$  is called *randomized* if, for any partial realization  $\psi$ , node  $\pi(\psi)$  is not selected deterministically (in general), but randomly (according to a probability distribution  $p_\psi$  depending on  $\psi$ ). Given a vector  $\mathbf{y} = (y_1, \dots, y_n)$  such that  $y_v \in [0, 1]$  for any  $v \in V$ , we say that  $\mathbb{P}(\pi) = \mathbf{y}$  if the probability that each node  $v$  belongs to

$\text{dom}(\psi_{\pi,L})$  is  $y_v$ , where  $\psi_{\pi,L}$  is the partial realisation with policy  $\pi$ . Let  $\text{OPT}_A(G, \mathbf{y})$  be the optimal expected influence spread  $\sigma(\pi)$  over all the randomized adaptive policies  $\pi$  such that  $\mathbb{P}(\pi) = \mathbf{y}$ .<sup>1</sup>

Let  $\pi^*$  be an optimal adaptive policy for the adaptive influence maximization problem (with  $|\pi^*| = k$ ), and let  $\mathbf{x} = (x_1, \dots, x_n)$  be the vector such that  $\mathbb{P}(\pi^*) = \mathbf{x}$ . As  $|\pi^*| = k$ , we have that  $\sum_{v \in V} x_v = k$ .

For any  $t \in [k]_0$ , let  $S_t$  be the optimal set of  $t$  seed nodes in the non-adaptive influence maximization problem, i.e., such that  $\text{OPT}_N(G, t) = \mathbb{E}_L(|R(S_t, L)|)$ . Let  $\psi_{t,L}$  be the random variable denoting the sub-realisation of  $\phi_L$  such that  $\text{dom}(\psi_{t,L}) = S_t$ . Let  $\rho$  be the random variable equal to node  $v \in V$  with probability  $x_v/k$ . Observe that the above random variable is well-defined, as  $\sum_{v \in V} (x_v/k) = k/k = 1$ . For any  $t \in [k]$ , let  $\psi_{\rho,t,L}$  be the random variable denoting the sub-realisation of  $\phi_L$  such that  $\text{dom}(\psi_{\rho,t,L}) = S_{t-1} \cup \{\rho\}$ .

**General outline of the proof technique** We observe that  $\psi_{\rho,t,L}$  is the partial realization coming from the following *hybrid non-adaptive policy*: initially, we activate the first  $t - 1$  seed nodes as in the optimal non-adaptive solution guaranteeing an expected influence spread of  $\text{OPT}_N(G, t - 1)$ ; then, we randomly choose a node  $v$  according to random variable  $\rho$  and we select  $v$  as  $t$ -th seed node (if not already selected as seed). We use this hybrid non-adaptive policy as a main tool to obtain an improved upper bound on the adaptivity gap for in-arborescences. In Lemma 3.2, holding even for general graphs, we relate the hybrid non-adaptive policy and the optimal non-adaptive solution, with the optimal adaptive policy. Lemma 3.2, together with Lemma 3.3 (that is similar to Lemma 3.8 in [15]), is used in the main proof of the theorem to relate  $\text{OPT}_N(G, t)$  with  $\text{OPT}_A(G, k)$  for any  $t \in [k]$ , and this leads to our upper bound.

The proof structure of Lemma 3.2 exhibits some similarities with Lemma 6 of [51] and Lemma 3.3 of [15], but in their approach, they relate non-adaptive policies based on the Poisson process and multi-linear extensions, with the optimal adaptive policy. One disadvantage of the Poisson process adopted in [15] is that the number  $X$  of seed nodes selected by the corresponding non-adaptive policy is equal to  $k$  under expectation (i.e.,  $\mathbb{E}(X) = k$ ), and determining the expected influence spread w.r.t. random variable  $X$  has implied a further loss in the final upper bound (see Lemma 3.9 and inequality (21) of Theorem 3.1 in [15]). Instead, by using the hybrid-non-adaptive policy, we guarantee that the number of selected seed nodes at each step  $t \in [k]$  is exactly equal to  $t$ , independently from the considered random execution. This property allow us to avoid

<sup>1</sup>We observe that, if  $\mathbf{y}$  is arbitrary, a deterministic policy  $\pi$  verifying  $\mathbb{P}(\pi) = \mathbf{y}$  might not exists, and the introduction of randomization solves this issue.

the expectations w.r.t. the number of selected seed nodes, and this leads to a further improvement of the resulting upper bound on the adaptivity gap.

**Lemma 3.2.** *Let  $G$  be an arbitrary influence graph. For any  $t \in [k]$ , and any fixed partial realization  $\psi$  of  $G$  such that  $\mathbb{P}[\psi_{t-1,L} = \psi] > 0$ , we have*

$$OPT_A(G, k) \leq \sigma(R(\psi)) + k \cdot \mathbb{E}_{L,\rho} [f(\psi_{\rho,t,L}) - f(\psi_{t-1,L}) | \psi_{t-1,L} = \psi].$$

*Proof.* We have

$$\begin{aligned} & k \cdot \mathbb{E}_{L,\rho} [f(\psi_{\rho,t,L}) - f(\psi_{t-1,L}) | \psi_{t-1,L} = \psi] \\ &= k \cdot \sum_{v \in V} \mathbb{P}[\rho = v] \cdot \Delta(v | \psi) \\ &= k \cdot \sum_{v \in V \setminus R(\psi)} \frac{x_v}{k} \cdot \Delta(v | \psi) \end{aligned} \tag{3.1}$$

$$= \sum_{v \in V \setminus R(\psi)} x_v \cdot \Delta(v | \psi), \tag{3.2}$$

where (3.1) holds since  $\Delta(v | \psi) = 0$  for any  $v \in R(\psi)$ .

Let  $\mathbf{x}' = (x'_1, \dots, x'_n)$  be the vector such that  $x'_v = 1$  if  $v \in R(\psi)$ , and  $x'_v = x_v$  otherwise. As  $x'_v \geq x_v$  for any  $v \in V$  we have

$$OPT_A(G, k) \leq OPT_A(G, \mathbf{x}) \leq OPT_A(G, \mathbf{x}'). \tag{3.3}$$

Let  $\pi'$  be the optimal randomized adaptive policy such that  $\mathbb{P}(\pi') = \mathbf{x}'$ . Policy  $\pi'$  selects each node in  $R(\psi)$  with probability 1, thus we can assume that such seed nodes are selected at the beginning and that the adaptive policy starts by observing the resulting partial realization. Furthermore, we can assume that, for any partial realization  $\psi'$ ,  $\pi'$  does not select any node  $v \in R(\psi')$ , otherwise there is no increase of the influence spread. Given  $j \in [n]$ , let  $\Delta'(j)$  denote the expected increment of the influence spread when  $\pi'$  selects the  $j$ -th seed node (in order of selection, and without considering in the count the initial seeds of  $R(\psi)$ ); analogously, let  $\Delta'(j|v)$  denote the expected increment of the influence spread when  $\pi'$  selects the  $j$ -th seed node, conditioned by the fact that



the  $j$ -th seed is node  $v$ .<sup>2</sup> We get

$$\begin{aligned}
& OPT_A(G, \mathbf{x}') \\
&= \sigma(R(\psi)) + \sum_j \Delta'(j) \\
&= \sigma(R(\psi)) + \sum_j \sum_{v \in V \setminus R(\psi)} \mathbb{P}[\text{the } j\text{-th seed node is } v] \cdot \Delta'(j|v) \\
&= \sigma(R(\psi)) + \sum_{v \in V \setminus R(\psi)} \sum_j \mathbb{P}[\text{the } j\text{-th seed node is } v] \cdot \Delta'(j|v) \\
&= \sigma(R(\psi)) + \sum_{v \in V \setminus R(\psi)} \sum_j \mathbb{P}[\text{the } j\text{-th seed node is } v] \cdot \\
&\quad \cdot \mathbb{E}_{\pi'}[\Delta(v|\psi')|v = \pi'(\psi') \text{ for some } \psi' \supseteq \psi \text{ observed at step } j] \\
&\leq \sigma(R(\psi)) + \sum_{v \in V \setminus R(\psi)} \sum_j \mathbb{P}[\text{the } j\text{-th seed node is } v] \cdot \Delta(v|\psi) \tag{3.4} \\
&= \sigma(R(\psi)) + \sum_{v \in V \setminus R(\psi)} \mathbb{P}[v \text{ is selected as seed}] \cdot \Delta(v|\psi) \\
&= \sigma(R(\psi)) + \sum_{v \in V \setminus R(\psi)} x'_v \cdot \Delta(v|\psi) \\
&= \sigma(R(\psi)) + \sum_{v \in V \setminus R(\psi)} x_v \cdot \Delta(v|\psi), \tag{3.5}
\end{aligned}$$

where (3.4) holds since  $\Delta(v|\psi') \leq \Delta(v|\psi)$  for any partial realization  $\psi' \supseteq \psi$  by adaptive submodularity (Claim 1). By putting together (3.2), (3.3), and (3.5), we get

$$\begin{aligned}
& \sigma(R(\psi)) + k \cdot \mathbb{E}_{L, \rho} [f(\psi_{\rho, t, L}) - f(\psi_{t-1, L}) | \psi_{t-1, L} = \psi] \\
&= \sigma(R(\psi)) + \sum_{v \in V \setminus R(\psi)} x_v \cdot \Delta(v|\psi) \\
&\geq OPT_A(G, \mathbf{x}') \\
&\geq OPT_A(G, k),
\end{aligned}$$

thus showing the claim. □

**Lemma 3.3.** *If the input influence graph  $G$  is an in-arborescence, then*

$$\sigma(R(\psi_{t-1, L})) \leq f(\psi_{t-1, L}) + OPT_N(G, t-1)$$

for any live-edge graph  $L$  and  $t \in [k]$ .

*Proof.* Given a subset  $U \subseteq V$ , let  $\partial U := \{u \in U : \exists (u, v) \in E, v \notin U\}$ . We have that  $\sigma(R(\psi)) \leq |R(\psi)| + \sigma(\partial R(\psi)) = f(\psi) + \sigma(\partial R(\psi))$  for any partial realization  $\psi$ . Thus,

<sup>2</sup>If an execution of  $\pi'$  requires less than  $j$  steps, we assume that the increase of the influence spread at step  $j$  (that contributes to the expected values  $\Delta'(j)$  and  $\Delta'(j|v)$ ) is null.

to show the claim, it suffices to show that  $\sigma(\partial R(\psi_{t-1,L})) \leq OPT_N(G, t-1)$ . For in-arborescences, we have that  $|\partial R(\psi_{t-1,L})| \leq |dom(\psi_{t-1,L})| = t-1$ , thus  $\sigma(\partial R(\psi_{t-1,L})) \leq OPT_N(G, t-1)$ .  $\square$

Armed with the above lemmas, we can now prove Theorem 3.1.

*Proof of Theorem 3.1.* For any  $t \in [k]$ , we have

$$\begin{aligned} & k \cdot (OPT_N(G, t) - OPT_N(G, t-1)) \\ &= k \cdot (\sigma(S_t) - \sigma(S_{t-1})) \\ &= k \cdot (\mathbb{E}_L[f(\psi_{t,L})] - \mathbb{E}_L[f(\psi_{t-1,L})]) \\ &\geq k \cdot (\mathbb{E}_{L,\rho}[f(\psi_{\rho,t,L})] - \mathbb{E}_L[f(\psi_{t-1,L})]) \end{aligned} \quad (3.6)$$

$$\begin{aligned} &= k \cdot (\mathbb{E}_{L,\rho}[f(\psi_{\rho,t,L})] - \mathbb{E}_{L,\rho}[f(\psi_{t-1,L})]) \\ &= k \cdot \mathbb{E}_{L,\rho}[f(\psi_{\rho,t,L}) - f(\psi_{t-1,L})] \\ &= \mathbb{E}_{\psi_{t-1,L}} [k \cdot \mathbb{E}_{L,\rho}[f(\psi_{\rho,t,L}) - f(\psi_{t-1,L}) | \psi_{t-1,L}]] \\ &\geq \mathbb{E}_{\psi_{t-1,L}} [OPT_A(G, k) - \sigma(R(\psi_{t-1,L}))] \end{aligned} \quad (3.7)$$

$$\begin{aligned} &\geq \mathbb{E}_{\psi_{t-1,L}} [OPT_A(G, k) - f(\psi_{t-1,L}) - OPT_N(G, t-1)] \\ &= \mathbb{E}_{\psi_{t-1,L}} [OPT_A(G, k)] - \mathbb{E}_{\psi_{t-1,L}} [f(\psi_{t-1,L})] - \mathbb{E}_{\psi_{t-1,L}} [OPT_N(G, t-1)] \\ &= OPT_A(G, k) - \sigma(S_{t-1}) - OPT_N(G, t-1) \end{aligned} \quad (3.8)$$

$$= OPT_A(G, k) - 2 \cdot OPT_N(G, t-1), \quad (3.9)$$

where (3.6) holds since  $dom(\psi_{t,L})$  is the optimal set of  $t$  seed nodes for the non-adaptive influence maximization problem, (3.7) comes from Lemma 3.2, and (3.8) comes from Lemma 3.3. Thus, by (3.9), we get  $k \cdot (OPT_N(G, t) - OPT_N(G, t-1)) \geq OPT_A(G, k) - 2 \cdot OPT_N(G, t-1)$ , that after some manipulations leads to the following recursive relation:

$$OPT_N(G, t) \geq \frac{1}{k} \cdot OPT_A(G, k) + \left(1 - \frac{2}{k}\right) \cdot OPT_N(G, t-1), \quad \forall t \in [k]. \quad (3.10)$$

By applying (3.10) iteratively, we get

$$OPT_N(G, k) \geq \frac{1}{k} \cdot \sum_{t=0}^{k-1} \left(1 - \frac{2}{k}\right)^t \cdot OPT_A(G, k) = \frac{1 - (1 - 2/k)^k}{2} \cdot OPT_A(G, k),$$

that leads to

$$\frac{OPT_A(G, k)}{OPT_N(G, k)} \leq \frac{2}{1 - (1 - 2/k)^k} \leq \frac{2}{1 - e^{-2}} = \frac{2e^2}{e^2 - 1},$$

and this shows the claim.  $\square$

### 3.3 Adaptivity Gap for General Influence Graphs

In this section, we exhibit upper bounds on the  $k$ -adaptivity gap of general graphs. In the following theorem, we first give an upper bound that is linear in the number of seed nodes.

**Theorem 3.4.** *Given an arbitrary class of influence graphs  $\mathcal{G}$  and  $k \geq 2$ , we get  $AG(\mathcal{G}, k) \leq k$ .*

*Proof.* Let  $G = (V = [n], E, (p_{uv})_{(u,v) \in E})$  be an arbitrary influence graph. Let  $\pi^*$  be an optimal adaptive policy subject to  $|\pi^*| = k$ , and let  $\psi_{t,\pi^*,L}$  be the partial realization observed when the  $t$ -th seed node has been selected by Algorithm 2 with policy  $\pi^*$ . Fix  $t \in [k]$ , a partial realization  $\psi$  such that  $\mathbb{P}[\psi_{t,\pi^*,L} = \psi] > 0$ , and let  $v = \pi^*(\psi)$  be the node selected by policy  $\pi^*$  under partial realization  $\psi$ . We have that

$$\begin{aligned} & \mathbb{E}_L[f(\psi_{t,\pi^*,L}) - f(\psi_{t-1,\pi^*,L}) | \psi_{t-1,\pi^*,L} = \psi] \\ &= \Delta(v | \psi) \\ &\leq \Delta(v | \emptyset) \end{aligned} \tag{3.11}$$

$$\begin{aligned} &= \sigma(\{v\}) \\ &\leq OPT_N(G, 1), \end{aligned} \tag{3.12}$$

where (3.11) holds by adaptive submodularity (Claim 1). Thus, we get

$$\begin{aligned} OPT_A(G, k) &= \mathbb{E}_L[f(\psi_{k,\pi^*,L})] \\ &= \sum_{t=1}^k \mathbb{E}_L[f(\psi_{t,\pi^*,L}) - f(\psi_{t-1,\pi^*,L})] \\ &= \sum_{t=1}^k \mathbb{E}_{\psi_{t-1,\pi^*,L}}[\mathbb{E}_L[f(\psi_{t,\pi^*,L}) - f(\psi_{t-1,\pi^*,L}) | \psi_{t-1,\pi^*,L}]] \\ &\leq k \cdot \mathbb{E}_{\psi_{t-1,\pi^*,L}}[OPT_N(G, 1)] \end{aligned} \tag{3.13}$$

$$\begin{aligned} &= k \cdot OPT_N(G, 1) \\ &\leq k \cdot OPT_N(G, k), \end{aligned} \tag{3.14}$$

where (3.13) comes from (3.12), and the claim follows.  $\square$

In the next theorem we give an upper bound on the adaptivity gap that is sublinear in the number of nodes of the considered graph.

When considering general graphs having a bounded number of nodes, we get an upper bound on the adaptivity gap that is sublinear in the maximum number of nodes of the considered graphs.

**Theorem 3.5.** *If  $\mathcal{G}$  is the class of influence graphs having at most  $n$  nodes, we get  $AG(\mathcal{G}) \leq \lceil n^{1/3} \rceil$ .*

Let  $G = (V, E, (p_{uv})_{(u,v) \in E})$  be the input influence graph. To show Theorem 3.5, we recall the preliminary notations considered for the proof of Theorem 3.1, and we give a further preliminary lemma.

**Lemma 3.6.** *Given a set  $U \subseteq V$  of cardinality  $h \geq k$ , we have  $\sigma(U) \leq \frac{h}{k} \cdot OPT_N(G, k)$ .*

*Proof.* For any  $t \in [h]_0$ , let  $U_t := \emptyset$  if  $t = 0$ , and  $U_t := U_{t-1} \cup \{v_t\}$ , where  $v_t \in \arg \max_{v \in U \setminus U_{t-1}} (\sigma(U_{t-1} \cup \{v\}) - \sigma(U_{t-1}))$ . We have that  $\Delta_t := \sigma(U_t) - \sigma(U_{t-1})$  is non-increasing in  $t \in [h]$ . Indeed, given  $t \in [k-1]$ , we have that

$$\begin{aligned} \Delta_{t+1} &= \sigma(U_{t+1}) - \sigma(U_t) \\ &= \sigma(U_t \cup \{v_{t+1}\}) - \sigma(U_t) \\ &\leq \sigma(U_{t-1} \cup \{v_{t+1}\}) - \sigma(U_{t-1}) \end{aligned} \tag{3.15}$$

$$\begin{aligned} &\leq \max_{v \in U \setminus U_{t-1}} (\sigma(U_{t-1} \cup \{v\}) - \sigma(U_{t-1})) \\ &= \sigma(U_{t-1} \cup \{v_t\}) - \sigma(U_{t-1}) \\ &= \Delta_t, \end{aligned} \tag{3.16}$$

where (3.15) holds since  $\sigma$  is a submodular set-function (see [9]). Thus, we necessarily have

$$\begin{aligned} \frac{\sigma(U)}{h} &= \frac{\sum_{t=1}^h \Delta_t}{h} \\ &\leq \frac{\sum_{t=1}^k \Delta_t (h/k)}{h} \end{aligned} \tag{3.17}$$

$$\begin{aligned} &= \frac{\sum_{t=1}^k \Delta_t}{k} \\ &= \frac{\sigma(U_k)}{k} \\ &\leq \frac{OPT_N(G, k)}{k}, \end{aligned} \tag{3.18}$$

where (3.17) comes from (3.16). By (3.18), the claim follows.  $\square$

We use Theorem 3.4 and Lemma 3.6 to show Theorem 3.5.

*Proof of Theorem 3.5.* We assume w.l.o.g. that  $k > \lceil n^{1/3} \rceil$  and that  $OPT_N(G, k) < (\lceil n^{1/3} \rceil)^2$ . Indeed, if  $k \leq \lceil n^{1/3} \rceil$ , by Theorem 3.4 the claim holds, and if  $OPT_N(G, k) \geq (\lceil n^{1/3} \rceil)^2$ , then  $\frac{OPT_A(G, k)}{OPT_N(G, k)} \leq \frac{|V|}{OPT_N(G, k)} \leq \frac{n}{(\lceil n^{1/3} \rceil)^2} \leq \lceil n^{1/3} \rceil$ , and the claim holds as well. For any  $t \in [k]$ , we have

$$\begin{aligned} & k \cdot (OPT_N(G, t) - OPT_N(G, t-1)) \\ & \geq k \cdot (\mathbb{E}_{L, \rho}[f(\psi_{\rho, t, L})] - \mathbb{E}_{L, \rho}[f(\psi_{t-1, L})]) \\ & = \mathbb{E}_{\psi_{t-1, L}} [k \cdot \mathbb{E}_{L, \rho}[f(\psi_{\rho, t, L}) - f(\psi_{t-1, L}) | \psi_{t-1, L}]] \\ & \geq \mathbb{E}_{\psi_{t-1, L}} [OPT_A(G, k) - \sigma(R(\psi_{t-1, L}))] \end{aligned} \quad (3.19)$$

$$\begin{aligned} & = \mathbb{E}_{\psi_{t-1, L}} [OPT_A(G, k)] - \mathbb{E}_{\psi_{t-1, L}} [\sigma(R(\psi_{t-1, L}))] \\ & \geq \mathbb{E}_{\psi_{t-1, L}} [OPT_A(G, k)] - \mathbb{E}_{\psi_{k, L}} [\sigma(R(\psi_{k, L}))] \\ & \geq \mathbb{E}_{\psi_{t-1, L}} [OPT_A(G, k)] - \mathbb{E}_{\psi_{k, L}} \left[ \frac{|R(\psi_{k, L})|}{k} \cdot OPT_N(G, k) \right] \end{aligned} \quad (3.20)$$

$$\begin{aligned} & = OPT_A(G, k) - \frac{\mathbb{E}_{\psi_{k, L}} [|R(\psi_{k, L})|]}{k} \cdot OPT_N(G, k) \\ & \geq OPT_A(G, k) - \frac{\mathbb{E}_{\psi_{k, L}} [|R(\psi_{k, L})|]}{\lceil n^{1/3} \rceil + 1} \cdot ((\lceil n^{1/3} \rceil)^2 - 1) \end{aligned} \quad (3.21)$$

$$\begin{aligned} & = OPT_A(G, k) - (\lceil n^{1/3} \rceil - 1) \cdot \mathbb{E}_{\psi_{k, L}} [|R(\psi_{k, L})|] \\ & = OPT_A(G, k) - (\lceil n^{1/3} \rceil - 1) \cdot OPT_N(G, k), \end{aligned} \quad (3.22)$$

where (3.19) comes from Lemma 3.2, (3.20) comes from Lemma 3.6, and (3.21) comes from the hypothesis  $k > \lceil n^{1/3} \rceil$  and  $OPT_N(G, k) < (\lceil n^{1/3} \rceil)^2$ . By (3.22), we get  $OPT_N(G, t) - OPT_N(G, t-1) \geq (OPT_A(G, k) - (\lceil n^{1/3} \rceil - 1) \cdot OPT_N(G, k))/k$  for any  $t \in [k]$ , and by summing such inequality over all  $t \in [k]$ , we get

$$\begin{aligned} & OPT_N(G, k) \\ & = \sum_{t=1}^k (OPT_N(G, t) - OPT_N(G, t-1)) \\ & \geq \sum_{t=1}^k \frac{OPT_A(G, k) - (\lceil n^{1/3} \rceil - 1) \cdot OPT_N(G, k)}{k} \\ & = OPT_A(G, k) - (\lceil n^{1/3} \rceil - 1) \cdot OPT_N(G, k). \end{aligned} \quad (3.23)$$

Finally, (3.23) implies that  $OPT_A(G, k) \leq \lceil n^{1/3} \rceil \cdot OPT_N(G, k)$ , and this shows the claim.  $\square$

### 3.4 Adaptivity Gap for Other Influence Graphs

In this section, we extend the results obtained in Theorem 3.1, and we get upper bounds on the adaptivity gap of other classes of influence graphs.

#### 3.4.1 $\alpha$ -bounded-degree graphs

We first consider the class of  $\alpha$ -bounded-degree graphs: a class of undirected influence graphs parametrized by an integer  $\alpha \geq 0$  that includes several known graph topologies. In the following, when we refer to undirected influence graphs, we assume that, for any undirected edge  $\{u, v\}$ , there are two directed edges  $(u, v)$  and  $(v, u)$  having respectively two (possibly) distinct probabilities  $p_{uv}$  and  $p_{vu}$ .

Given an undirected graph  $G = (V, E)$  and a node  $v \in V$ , let  $\deg_v(G)$  denote the degree of node  $v$  in graph  $G$ . Given an integer  $\alpha \geq 0$ , an influence graph  $G = (V, E, (p_{uv})_{(u,v) \in E})$  is an  $\alpha$ -bounded-degree graph if it is undirected and  $\sum_{v \in V: \deg_v(G) > 2} \deg_v(G) \leq \alpha$ , i.e., the sum all the node degrees higher than 2 is at most  $\alpha$ ; we observe that the definition of  $\alpha$ -bounded degree graphs does not depends on the influence probabilities, but on the graph topology only.

**Example 3.1.** *Given an undirected graph  $G$ , a simple subpath  $P$  (resp. cycle  $C$ ) of  $G$  is standard if all the nodes of  $P$  but the first and the last one (resp. all the nodes of  $C$ ) have degree 2. The standard contraction of  $G$  is the multigraph  $G'$  obtained by replacing each maximal simple subpath  $P = (v_1, \dots, v_t)$  of  $G$  such that  $\deg_{v_2}(G) = \deg_{v_3}(G) = \dots = \deg_{v_{t-1}}(G) = 2$ , with an edge and by deleting each simple cycle  $C = (v_1, v_2, \dots, v_{t-1}, v_t = v_1)$  such that  $\deg_{v_2}(G) = \deg_{v_3}(G) = \dots = \deg_{v_{t-1}}(G) = 2$ . There are several interesting classes of  $\alpha$ -bounded-degree graphs characterized by the topological structure of their standard contraction:*

- *The set of 0-bounded-degree graphs is made of all the graphs  $G$  such that each connected component of  $G$  is either an undirected path or an undirected cycle; equivalently, the set of 0-bounded-degree graphs is made of all the graphs  $G$  whose standard contraction is the (possibly empty) union of several disconnected edges.*
- *If the standard contraction of a graph  $G$  is homeomorphic to a star with  $h$  edges, then  $G$  is a  $h$ -bounded-degree graph.*
- *If the standard contraction of a graph  $G$  is homeomorphic to a parallel-link multigraph with  $h$  edges, then  $G$  is a  $2h$ -bounded-degree graph.*

- If the standard contraction of a graph  $G$  is homeomorphic to a cycle with  $h$  chords, then  $G$  is a  $6h$ -bounded-degree graph.
- If the standard contraction of a graph  $G$  is homeomorphic to a clique with  $h$  nodes, then  $G$  is a  $h(h-1)$ -bounded-degree graph.

In the following, we provide an upper bound on the adaptivity gap of  $\alpha$ -bounded-degree graphs for any  $\alpha \geq 0$ .

**Theorem 3.7.** *Given  $\alpha \geq 0$ , let  $\mathcal{G}$  be the class of  $\alpha$ -bounded-degree graphs. Then*

$$AG(\mathcal{G}, k) \leq \min \left\{ k, \frac{\alpha}{k} + 2 + \frac{1}{1 - (1 - 1/k)^k} \right\} \leq \frac{\sqrt{4(e-1)^2\alpha + (3e-2)^2} + 3e - 2}{2(e-1)}$$

for any  $k \geq 2$ , i.e.,  $AG(\mathcal{G}) \leq \sqrt{\alpha} + O(1)$ .

Let  $G = (V = [n], E, (p_{uv})_{(u,v) \in E})$  be an  $\alpha$ -bounded-degree graph, and we recall the preliminary notations from Theorem 3.1. The proof of Theorem 3.7 is a non-trivial generalization of Theorem 3.1. In particular, the proof resorts to Theorem 3.4 to get the upper bound of  $k$ , and, by following the approach of Theorem 3.1, the following technical lemma is used in place of Lemma 3.3 to get the final upper bound.

**Lemma 3.8.** *When the input influence graph  $G$  is an  $\alpha$ -bounded-degree graph with  $\alpha \geq 0$ , we have that  $\sigma(R(\psi_{t-1,L})) \leq f(\psi_{t-1,L}) + \left(\frac{\alpha}{k} + 2\right) \cdot OPT_N(G, k)$  for any  $t \in [k]$  and live-edge graph  $L$ .*

*Proof.* Given a subset  $U \subseteq V$ , let  $\partial U := \{u \in U : \exists (u, v) \in E, v \notin U\}$ . We have that  $\sigma(R(\psi)) \leq |R(\psi)| + \sigma(\partial R(\psi)) = f(\psi) + \sigma(\partial R(\psi))$  for any partial realization  $\psi$ . Thus, to show the claim, it suffices to show that

$$\sigma(\partial R(\psi_{t-1,L})) \leq \left(\frac{\alpha}{k} + 2\right) \cdot OPT_N(G, k).$$

Let  $U \subseteq V$  such that  $U$  has at most  $k$  connected components. Let  $A$  be the set of connected components containing at least one node of degree higher than 2, and let  $B$  be the set of the remaining components, i.e., containing nodes with degree in  $[2]_0$  only. By definition of  $A$  and  $B$ , we necessarily have that  $|\partial A| \leq \sum_{v \in V: deg_v(G) > 2} deg_v(G) \leq \alpha$  and  $|\partial B| \leq 2k$ . Thus  $|\partial U| \leq |\partial A| + |\partial B| \leq \alpha + 2k$ , and the next claim follows.

*Claim 2.* Given a subset  $U \subseteq V$  made of at most  $k$  connected components, then  $|\partial U| \leq \alpha + 2k$ .

Now, we have that

$$\begin{aligned} \sigma(\partial R(\psi_{t-1,L})) &\leq \sigma(\partial R(\psi_{k,L})) \\ &\leq \frac{|\partial R(\psi_{k,L})|}{k} \cdot OPT_N(G, k) \end{aligned} \quad (3.24)$$

$$\leq \frac{\alpha + 2k}{k} \cdot OPT_N(G, k), \quad (3.25)$$

where (3.24) comes from Lemma 3.6, and (3.25) holds since  $R(\psi_{k,L})$  contains at most  $k$  connected components and because of Claim 2. Thus, by (3.25), the claim of the lemma follows.  $\square$

We can now prove Theorem 3.7.

*Proof of Theorem 3.7.* For any  $t \in [k]$ , we have

$$\begin{aligned} &k \cdot (OPT_N(G, t) - OPT_N(G, t-1)) \\ &\geq k \cdot (\mathbb{E}_{L,\rho}[f(\psi_{\rho,t,L})] - \mathbb{E}_{L,\rho}[f(\psi_{t-1,L})]) \\ &= \mathbb{E}_{\psi_{t-1,L}} [k \cdot \mathbb{E}_{L,\rho}[f(\psi_{\rho,t,L}) - f(\psi_{t-1,L}) | \psi_{t-1,L}]] \\ &\geq \mathbb{E}_{\psi_{t-1,L}} [OPT_A(G, k) - \sigma(R(\psi_{t-1,L}))] \end{aligned} \quad (3.26)$$

$$\geq \mathbb{E}_{\psi_{t-1,L}} \left[ OPT_A(G, k) - f(\psi_{t-1,L}) - \left( \frac{\alpha}{k} + 2 \right) OPT_N(G, k) \right] \quad (3.27)$$

$$\begin{aligned} &= \mathbb{E}_{\psi_{t-1,L}} [OPT_A(G, k)] - \mathbb{E}_{\psi_{t-1,L}} [f(\psi_{t-1,L})] - \left( \frac{\alpha}{k} + 2 \right) \cdot \mathbb{E}_{\psi_{t-1,L}} [OPT_N(G, k)] \\ &= OPT_A(G, k) - \sigma(S_{t-1}) - \left( \frac{\alpha}{k} + 2 \right) \cdot OPT_N(G, k) \\ &= OPT_A(G, k) - \left( \frac{\alpha}{k} + 2 \right) \cdot OPT_N(G, k) - OPT_N(G, t-1), \end{aligned} \quad (3.28)$$

where (3.26) comes from Lemma 3.2 and (3.27) comes from Lemma 3.8. Thus, by (3.28), we get the following recursive relation:

$$OPT_N(G, t) \geq \frac{1}{k} \left( OPT_A(G, k) - \left( \frac{\alpha}{k} + 2 \right) OPT_N(G, k) \right) + \left( 1 - \frac{1}{k} \right) OPT_N(G, t-1), \quad (3.29)$$

for any  $t \in [k]$ . By applying iteratively (3.29), we get

$$\begin{aligned} OPT_N(G, k) &\geq \frac{1}{k} \cdot \left( OPT_A(G, k) - \left( \frac{\alpha}{k} + 2 \right) \cdot OPT_N(G, k) \right) \cdot \sum_{t=0}^{k-1} \left( 1 - \frac{1}{k} \right)^j \\ &= \left( OPT_A(G, k) - \left( \frac{\alpha}{k} + 2 \right) \cdot OPT_N(G, k) \right) \cdot \left( 1 - \left( 1 - \frac{1}{k} \right)^k \right), \end{aligned}$$



that, after some manipulations, leads to

$$\frac{OPT_A(G, k)}{OPT_N(G, k)} \leq \frac{\alpha}{k} + 2 + \frac{1}{1 - (1 - 1/k)^k} \leq \frac{\alpha}{k} + 2 + \frac{1}{1 - e^{-1}}. \quad (3.30)$$

By Theorem 3.4, we have that  $\frac{OPT_A(G, k)}{OPT_N(G, k)} \leq k$ , thus, by (3.30), we get

$$\begin{aligned} \frac{OPT_A(G, k)}{OPT_N(G, k)} &\leq \min \left\{ k, \frac{\alpha}{k} + 2 + \frac{1}{1 - (1 - 1/k)^k} \right\} \\ &\leq \min \left\{ k, \frac{\alpha}{k} + 2 + \frac{1}{1 - e^{-1}} \right\} \\ &\leq \frac{\sqrt{4(e-1)^2\alpha + (3e-2)^2} + 3e - 2}{2(e-1)}, \end{aligned} \quad (3.31)$$

where (3.31) is equal to the real value of  $k \geq 0$  such that  $k = \frac{\alpha}{k} + 2 + \frac{1}{1 - e^{-1}}$ . By (3.31) the claim follows.  $\square$

For the particular case of 0-bounded-degree graphs, the following theorem provides a better upper bound on the adaptivity gap.

**Theorem 3.9.** *Let  $\mathcal{G}$  be the class of 0-bounded-degree graphs. Then*

$$AG(\mathcal{G}, k) \leq \min \left\{ k, \frac{3}{1 - (\max\{0, 1 - 3/k\})^k} \right\} \leq \frac{3e^3}{e^3 - 1} \approx 3.16,$$

for any  $k \geq 2$ .

The proof of Theorem 3.9 is similar to that of Theorem 3.1. Let  $G = (V = [n], E, (p_{uv})_{(u,v) \in E})$  be a 0-bounded-degree graph. We recall the notation from Theorem 3.1 and we give the following preliminary lemma, whose proof is analogue to that of Lemma 4.9.

**Lemma 3.10.** *When the input influence graph  $G$  is a 0-bounded-degree graph, we have*

$$\sigma(R(\psi_{t-1, L})) \leq f(\psi_{t-1, L}) + 2 \cdot OPT_N(G, t-1), \quad (3.32)$$

for any  $t \in [k]$  and live-edge graph  $L$ .

*Proof.* As in Lemma 3.3, we show that  $\sigma(\partial R(\psi_{t-1, L})) \leq 2 \cdot OPT_N(G, t-1)$ . First of all, we assume that  $t \geq 2$ , otherwise  $\sigma(R(\psi_{t-1, L}))$  and the claim holds. By Lemma 3.6, we have that  $\sigma(\partial R(\psi_{t-1, L})) \leq \frac{|\partial R(\psi_{t-1, L})|}{t-1} \cdot OPT_N(G, t-1)$ . As  $G$  is a 0-bounded-degree graph, we have that  $|\partial R(\psi_{t-1, L})| \leq 2(t-1)$ . By considering the above inequalities, we get  $\sigma(\partial R(\psi_{t-1, L})) \leq \frac{|\partial R(\psi_{t-1, L})|}{t-1} \cdot OPT_N(G, t-1) \leq \frac{2(t-1)}{t-1} \cdot OPT_N(G, t-1) = 2 \cdot OPT_N(G, t-1)$ , and the claim follows.  $\square$

We are now ready to show Theorem 3.9.

*Proof of Theorem 3.9.* By Theorem 3.4, we have that  $k$  is an upper bound on the  $k$ -adaptivity gap, thus it is sufficient showing that  $\frac{3}{1-(\max\{0,1-3/k\})^k}$  is a further upper bound. If  $k \leq 3$  the claim trivially holds, since  $k$  is an upper bound on the  $k$ -adaptivity gap. Then, we assume that  $k > 3$ , and it is sufficient showing that  $\frac{3}{1-(1-3/k)^k}$  is an upper bound. For any  $t \in [k]$ , we have

$$\begin{aligned}
& k \cdot (OPT_N(G, t) - OPT_N(G, t - 1)) \\
&= k \cdot (\sigma(S_t) - \sigma(S_{t-1})) \\
&= k \cdot (\mathbb{E}_L[f(\psi_{t,L})] - \mathbb{E}_L[f(\psi_{t-1,L})]) \\
&\geq k \cdot (\mathbb{E}_{L,\rho}[f(\psi_{\rho,t,L})] - \mathbb{E}_L[f(\psi_{t-1,L})]) \\
&= k \cdot (\mathbb{E}_{L,\rho}[f(\psi_{\rho,t,L})] - \mathbb{E}_{L,\rho}[f(\psi_{t-1,L})]) \\
&= k \cdot \mathbb{E}_{L,\rho}[f(\psi_{\rho,t,L}) - f(\psi_{t-1,L})] \\
&= \mathbb{E}_{\psi_{t-1,L}} [k \cdot \mathbb{E}_{L,\rho}[f(\psi_{\rho,t,L}) - f(\psi_{t-1,L}) | \psi_{t-1,L}]] \\
&\geq \mathbb{E}_{\psi_{t-1,L}} [OPT_A(G, k) - \sigma(R(\psi_{t-1,L}))] \tag{3.33}
\end{aligned}$$

$$\geq \mathbb{E}_{\psi_{t-1,L}} [OPT_A(G, k) - f(\psi_{t-1,L}) - 2 \cdot OPT_N(G, t - 1)] \tag{3.34}$$

$$= \mathbb{E}_{\psi_{t-1,L}} [OPT_A(G, k)] - \mathbb{E}_{\psi_{t-1,L}} [f(\psi_{t-1,L})] - 2 \cdot \mathbb{E}_{\psi_{t-1,L}} [OPT_N(G, t - 1)]$$

$$= OPT_A(G, k) - \sigma(S_{t-1}) - 2 \cdot OPT_N(G, t - 1)$$

$$= OPT_A(G, k) - 3 \cdot OPT_N(G, t - 1), \tag{3.35}$$

where (3.33) comes from Lemma 3.2 and (3.34) comes from Lemma 3.10. Thus, by (3.35), we get  $k \cdot (OPT_N(G, t) - OPT_N(G, t - 1)) \geq OPT_A(G, k) - 3 \cdot OPT_N(G, t - 1)$ , that after some manipulations leads to the following recursive relation:

$$OPT_N(G, t) \geq \frac{1}{k} \cdot OPT_A(G, k) + \left(1 - \frac{3}{k}\right) \cdot OPT_N(G, t - 1), \quad \forall t \in [k]. \tag{3.36}$$

By applying iteratively (3.36), we get

$$OPT_N(G, k) \geq \frac{1}{k} \cdot \sum_{t=0}^{k-1} \left(1 - \frac{3}{k}\right)^t \cdot OPT_A(G, k) = \frac{1 - (1 - 3/k)^k}{3} \cdot OPT_A(G, k),$$

that leads to

$$\frac{OPT_A(G, k)}{OPT_N(G, k)} \leq \frac{3}{1 - (1 - 3/k)^k} \leq \frac{3}{1 - e^{-3}}, \tag{3.37}$$

and this shows the claim.  $\square$

### 3.4.2 $(\beta, \gamma)$ -bounded-activation graphs

Let  $N(u) := \{v \in V : (u, v) \in E\}$  denote the set of out-neighbors of node  $u$ . Given an integer  $\beta \geq 0$  and a real value  $\gamma \in [0, 1)$ , an influence graph  $G = (V, E, (p_{uv})_{(u,v) \in E})$  is a  $(\beta, \gamma)$ -bounded-activation graph if there exists  $\hat{S} \subseteq V$  with  $|\hat{S}| \leq \beta$  such that  $\sum_{v \in N(u)} p_{u,v} \leq \gamma$  for any  $u \in V \setminus \hat{S}$ . Informally, the class of  $(\beta, \gamma)$ -bounded-activation graphs coincides with all the influence graphs such that all nodes but  $\beta$  influence in expectation has at most  $\gamma$  neighbors each.

In the following theorem, whose proof is partially based on Theorem 3.4 and Lemma 3.6, we provide an upper bound on the adaptivity gap of  $(\beta, \gamma)$ -bounded-activation graphs, for any integer  $\beta \geq 0$  and  $\gamma \in [0, 1)$ .

**Theorem 3.11.** *Given an integer  $\beta \geq 0$  and  $\gamma \in [0, 1)$ , let  $\mathcal{G}$  be the class of  $(\beta, \gamma)$ -bounded-activation graphs. Then*

$$\begin{aligned} AG(\mathcal{G}, k) &\leq \min \left\{ k, \frac{\max\{\beta, k\} \cdot \min\{1, \beta\} + \frac{k}{1-\gamma}}{k} \right\} \\ &\leq \max \left\{ \frac{\sqrt{\left(\frac{1}{1-\gamma}\right)^2 + 4\beta} + \frac{1}{1-\gamma}}{2}, \min\{1, \beta\} + \frac{1}{1-\gamma} \right\} \\ &\leq \sqrt{\beta} + \frac{1}{1-\gamma}. \end{aligned} \tag{3.38}$$

for any  $k \geq 2$ .

Let  $G = (V = [n], E, (p_{uv})_{(u,v) \in E})$  be a  $(\beta, \gamma)$ -bounded-activation graph. To show the claim of Theorem 3.11, we recall the notations from Theorem 3.1 and we give some preliminary lemmas.

Let  $\hat{S}$  be the set of nodes such that  $|\hat{S}| \leq \beta$  and  $\sum_{v \in N(u)} p_{u,v} \leq \gamma$  for any  $u \in V \setminus \hat{S}$ . Let  $G \setminus \hat{S}$  be the graph obtained from  $G$  by removing the nodes in  $\hat{S}$  and their adjacent edges, and let  $OPT_N(G \setminus \hat{S}, 1)$  denote the optimal non-adaptive influence spread  $OPT_N(G \setminus \hat{S}, 1)$  achieved by a unique seed in graph  $G \setminus \hat{S}$ . In the following lemma, we provide an upper bound for the optimal adaptive influence spread  $OPT_A(G, k)$  in  $G$ .

**Lemma 3.12.** *We have that  $OPT_A(G, k) \leq \sigma(\hat{S}) + k \cdot OPT_N(G \setminus \hat{S}, 1)$ .*

*Proof.* Let  $\hat{\pi}$  be an optimal adaptive policy that first selects the nodes in  $\hat{S}$  and then adaptively selects  $k$  nodes. By construction, we have that

$$OPT_A(G, k) \leq \sigma(\hat{\pi}). \tag{3.39}$$

Let  $\psi_{0,\hat{\pi},L}$  denote the partial realization observed by  $\hat{\pi}$  after the selection of  $\hat{S}$ , and let  $\psi_{t,\hat{\pi},L}$  be the partial realization observed after the selection of the  $t$ -th seed node in  $V \setminus \hat{S}$ . By exploiting the adaptive submodularity (Claim 1) similarly as in the proof of Theorem 3.4, one can easily show that

$$\mathbb{E}_L[f(\psi_{t,\hat{\pi},L}) - f(\psi_{t-1,\hat{\pi},L}) | \psi_{t-1,\hat{\pi},L} = \psi] \leq OPT_N(G \setminus \hat{S}, 1) \quad (3.40)$$

for any fixed partial realization  $\psi$  (with  $\mathbb{P}[\psi_{t,\hat{\pi},L} = \psi] > 0$ ) and any  $t \in [k]$ . Thus, we get

$$\begin{aligned} \sigma(\hat{\pi}) &= \mathbb{E}_L[f(\psi_{k,\hat{\pi},L})] \\ &= \mathbb{E}_L[f(\psi_{0,\hat{\pi},L})] + \sum_{t=1}^k \mathbb{E}_L[f(\psi_{t,\hat{\pi},L}) - f(\psi_{t-1,\hat{\pi},L})] \\ &= \mathbb{E}_L[f(\psi_{0,\hat{\pi},L})] + \sum_{t=1}^k \mathbb{E}_{\psi_{t-1,\hat{\pi},L}}[\mathbb{E}_L[f(\psi_{t,\hat{\pi},L}) - f(\psi_{t-1,\hat{\pi},L}) | \psi_{t-1,\hat{\pi},L}]] \\ &\leq \mathbb{E}_L[f(\psi_{0,\hat{\pi},L})] + k \cdot \mathbb{E}_{\psi_{t-1,\hat{\pi},L}}[OPT_N(G \setminus \hat{S}, 1)] \quad (3.41) \\ &= \mathbb{E}_L[f(\psi_{0,\hat{\pi},L})] + k \cdot OPT_N(G \setminus \hat{S}, 1) \end{aligned}$$

$$= \sigma(\hat{S}) + k \cdot OPT_N(G \setminus \hat{S}, 1), \quad (3.42)$$

where (3.41) comes from (3.40). By putting (3.39) and (3.42) together, the claim follows.  $\square$

In the following lemma, we provide an upper bound for  $OPT_N(G \setminus \hat{S}, 1)$  in terms of parameter  $\gamma$ .

**Lemma 3.13.** *We have that  $OPT_N(G \setminus \hat{S}, 1) \leq \frac{1}{1-\gamma}$ .*

*Proof.* Let  $v_0$  be the node that maximizes the expected influence spread in  $G \setminus \hat{S}$  when selected as unique seed. For any live-edge graph  $L$  and  $j \in [n-1]_0$ , let  $A(j, L)$  denote the set of nodes activated at the  $j$ -th round of diffusion when  $v_0$  is the initial seed node of  $G \setminus \hat{S}$ , i.e.,  $A(0, L) = \{v_0\}$  and  $A(j, L)$  is the set of neighbors of  $A(j-1, L)$  activated

by  $A(j-1, L)$ . We can easily observe that

$$\begin{aligned}
& \mathbb{E}_L[|A(j, L)|] \\
&= \sum_{v \in G \setminus \hat{S}} \mathbb{P}[v \in A(j, L)] \\
&\leq \sum_{v \in G \setminus \hat{S}} \sum_{\substack{P = (v_0, v_1, \dots, v_j := v): \\ P \text{ is a path of } G \setminus \hat{S} \text{ from } v_0 \text{ to } v \text{ having } j \text{ edges}}} p_{v_0, v_1} \cdot p_{v_1, v_2} \cdots p_{v_{j-2}, v_{j-1}} \cdot p_{v_{j-1}, v_j} \\
&= \sum_{\substack{P = (v_0, v_1, \dots, v_j): \\ P \text{ is a path of } G \setminus \hat{S} \text{ from } v_0 \text{ having } j \text{ edges}}} p_{v_0, v_1} \cdot p_{v_1, v_2} \cdots p_{v_{j-2}, v_{j-1}} \cdot p_{v_{j-1}, v_j} \\
&= \sum_{v_1 \in N(v_0) \cap \hat{S}} p_{v_0, v_1} \sum_{v_2 \in N(v_1) \cap \hat{S}} p_{v_1, v_2} \sum_{v_3 \in N(v_2) \cap \hat{S}} \cdots p_{v_{j-2}, v_{j-1}} \sum_{v_j \in N(v_{j-1}) \cap \hat{S}} p_{v_{j-1}, v_j}
\end{aligned}$$

for any  $j \in [n-1]$ . Thus

$$\begin{aligned}
& OPT_N(G \setminus \hat{S}, 1) \\
&= \mathbb{E}_L \left[ \sum_{j=0}^{n-1} |A(j, L)| \right] \\
&= \mathbb{E}_L[|A(0, L)|] + \sum_{j=1}^{n-1} \mathbb{E}_L[|A(j, L)|] \\
&\leq 1 + \underbrace{\sum_{j=1}^{n-1} \sum_{v_1 \in N(v_0) \cap \hat{S}} p_{v_0, v_1} \sum_{v_2 \in N(v_1) \cap \hat{S}} \cdots \sum_{v_{j-1} \in N(v_{j-2}) \cap \hat{S}} p_{v_{j-2}, v_{j-1}} \sum_{v_j \in N(v_{j-1}) \cap \hat{S}} p_{v_{j-1}, v_j}}_{\leq \gamma^j} \underbrace{\leq \gamma^2}_{\leq \gamma}}_{\leq \gamma^{j-1}} \\
&\leq 1 + \sum_{j=1}^{n-1} \gamma^j \leq \sum_{j=0}^{\infty} \gamma^j = \frac{1}{1-\gamma},
\end{aligned}$$

and this shows the claim.  $\square$

We are ready to show Theorem 3.11.

*Proof.* We first show the upper bound in (3.38). By Theorem 3.4, we have that  $k$  is an upper bound on the  $k$ -adaptivity gap. Thus, it is sufficient to show that  $\frac{\max\{\beta, k\} \cdot \min\{1, \beta\} + \frac{k}{1-\gamma}}{k}$

is an upper bound on the  $k$ -adaptivity gap. We have that

$$\frac{OPT_A(G, k)}{OPT_N(G, k)} \leq \frac{\sigma(\hat{S}) + OPT_N(G \setminus \hat{S}, 1) \cdot k}{OPT_N(G, k)} \quad (3.43)$$

$$\leq \frac{\sigma(\hat{S}) + \frac{k}{1-\gamma}}{OPT_N(G, k)} \quad (3.44)$$

$$\leq \frac{OPT_N(G, \beta) + \frac{k}{1-\gamma}}{OPT_N(G, k)}, \quad (3.45)$$

where (3.43) and (3.44) follows from Lemmas 3.12 and 3.13, respectively.

If  $\beta = 0$ , by using  $OPT_N(G, \beta) = 0$  and  $OPT_N(G, k) \geq k$  in (3.45), we get that (3.45) is at most  $\frac{k}{1-\gamma} = \frac{\max\{\beta, k\} \cdot \min\{1, \beta\} + \frac{k}{1-\gamma}}{k}$ , and this shows (3.38).

If  $1 \leq \beta \leq k$ , by continuing from (3.45), we get

$$\begin{aligned} \frac{OPT_A(G, k)}{OPT_N(G, k)} &\leq \frac{OPT_N(G, \beta) + \frac{k}{1-\gamma}}{OPT_N(G, k)} \\ &\leq \frac{OPT_N(G, k) + \frac{k}{1-\gamma}}{OPT_N(G, k)} \\ &\leq \frac{k + \frac{k}{1-\gamma}}{k}, \end{aligned} \quad (3.46)$$

where (3.46) holds since  $OPT_N(G, k) \geq k$ . As (3.46) is equal to  $\frac{\max\{\beta, k\} \cdot \min\{1, \beta\} + \frac{k}{1-\gamma}}{k}$ , inequality (3.38) holds if  $1 \leq \beta \leq k$ .

Finally, if  $\beta > k$ , by continuing from (3.45), we get

$$\begin{aligned} \frac{OPT_A(G, k)}{OPT_N(G, k)} &\leq \frac{OPT_N(G, \beta) + \frac{k}{1-\gamma}}{OPT_N(G, k)} \\ &\leq \frac{\frac{\beta}{k} \cdot OPT_N(G, k) + \frac{k}{1-\gamma}}{OPT_N(G, k)} \end{aligned} \quad (3.47)$$

$$\leq \frac{\frac{\beta}{k} \cdot k + \frac{k}{1-\gamma}}{k}, \quad (3.48)$$

where (3.47) follows from Lemma 3.6 and (3.48) holds since  $OPT_N(G, k) \geq k$ . As (3.48) is equal to  $\frac{\max\{\beta, k\} \cdot \min\{1, \beta\} + \frac{k}{1-\gamma}}{k}$ , inequality (3.38) holds if  $\beta > k$ .

We conclude that  $\min \left\{ k, \frac{\max\{\beta, k\} \cdot \min\{1, \beta\} + \frac{k}{1-\gamma}}{k} \right\}$  is an upper bound on the  $k$ -adaptivity gap. Furthermore, as  $k$  and  $\frac{\max\{\beta, k\} \cdot \min\{1, \beta\} + \frac{k}{1-\gamma}}{k}$  are respectively increasing and non-increasing in  $k$  (for any fixed integer  $\beta \geq 0$ ), the real value  $k$  such that the two quantities

are equal is a further upper bound on the adaptivity gap, and such value is

$$\begin{aligned}
k &= \max \left\{ \frac{\sqrt{\left(\frac{1}{1-\gamma}\right)^2 + 4\beta + \frac{1}{1-\gamma}}}{2}, \min\{1, \beta\} + \frac{1}{1-\gamma} \right\} \\
&\leq \max \left\{ \frac{\sqrt{\left(\frac{1}{1-\gamma}\right)^2 + \sqrt{4\beta} + \frac{1}{1-\gamma}}}{2}, \min\{1, \beta\} + \frac{1}{1-\gamma} \right\} \\
&= \max \left\{ \sqrt{\beta} + \frac{1}{1-\gamma}, \min\{1, \beta\} + \frac{1}{1-\gamma} \right\} \\
&= \sqrt{\beta} + \frac{1}{1-\gamma}.
\end{aligned}$$

Thus,  $\max \left\{ \frac{\sqrt{\left(\frac{1}{1-\gamma}\right)^2 + 4\beta + \frac{1}{1-\gamma}}}{2}, \min\{1, \beta\} + \frac{1}{1-\gamma} \right\}$  and  $\sqrt{\beta} + \frac{1}{1-\gamma}$  are further upper bounds on the adaptivity gap.

□

## 3.5 Experiments

This section is dedicated to the evaluation of adaptive and non-adaptive greedy algorithms under different network settings. We conduct experiments on the IC and LT to observe the performance generated by the algorithms under these diffusion models. For the sake of simplicity, we have used only undirected graphs. For non-adaptive greedy, we considered the  $TIM^+$  algorithm [33]. Both the codes for  $TIM^+$  and the adaptive greedy algorithm are open source. The adaptive greedy algorithm can be accessed at <https://github.com/debashmitap/Adaptive-TIM>.

### 3.5.1 Testbed

The adaptive greedy algorithm and the adaptive greedy with  $TIM^+$  seeds are executed on a Mac with an Intel core i5 with 2.3GHz CPU and 8GB RAM. The non-adaptive greedy algorithm is executed on a 64 GB Linux machine with an Intel(R) Xeon(R) 3.4GHz CPU.

<sup>3</sup><https://sourceforge.net/projects/timplus/>

The languages used are C++ compiled with g++ 4.2.1 and shell for the algorithms. Python 2.7.16 along with NetworkX package is used to generate the networks (ER,WS,BA graphs). The Facebook graph is a real world network from the SNAP database.

**Erdős-Rényi.** An *Erdős-Rényi (ER)* graph [61] is a random graph, whose nodes and edges are generated at random. Let us denote an ER graph as  $G_{n,p}$ , which represents an undirected graph on  $n$  nodes, where each edge  $(u,v)$  appears independently and identically distributed with probability  $p$ . In expectation, we will normally have  $n*n*p$  edges and these edges are placed at random between the nodes.

ER graphs suffer from low connectivity, when  $p$  is small, and is considered to be a dense graph when the value of  $p$  tends to one. Since we are interested in a connected graph for our experiments, we consider a specific value of  $p$ , denoted as  $p^* = \frac{\ln n}{n}$ . When,  $p > p^*$ , the graph  $G(n,p)$  is very likely to be connected, otherwise not. We use this particular formula for  $p$  to generate a random graph with the *erdos\_renyi\_graph()* library.

**Watts-Strogatz.** The *Watts-Strogatz (WS)* graph [62] is used to model small world networks. Social networks exhibit properties of high clustering coefficient and low path length. Random graphs like the ER, have low path length and low clustering. However, regular graphs, where every node has the same number of neighbours have high clustering and high path length. WS combines a regular and a random graph to extract the high clustering and low path length property to create a generative model of a social network.

Let  $G(n,k,p)$  denote a regular graph with  $n$  nodes, where each node is connected to  $k$  neighbours. A subset of edges are chosen with a probability  $p$  and rewired by replacing them with random edges. The value of  $p$  detects the randomness of the graph. When the value of  $p$  tends to 1, the graph behaves like a random graph, whereas with  $p = 0$ , the graph is a regular graph. To generate a WS graph for our problem, we resort to the *connected\_watts\_strogatz\_graph()* library to ensure the connectedness of the graph.

**Barabási-Albert.** *Barabási-Albert (BA)* graphs [63] are scale-free networks that follow preferential attachments, i.e., the degree distribution obeys the power law. Many real world networks are scale free networks (eg., WWW). These networks include hubs, which are networks with high degree, and these hubs grow over time due to preferential attachments.

Let us denote a BA graph as  $G(n,m)$  where  $n$  represents the number of nodes and  $m$  represents the number of edges that are preferentially attached from a new node to the



existing nodes in the network. For the experiments we set the value of  $m$  as 2, and use the `barabasi_albert_graph()` library to generate the BA graphs for different values of  $n$ .

**Facebook network.** We analyze a real world Facebook<sup>4</sup> dataset from the SNAP database. The network consists of 4039 nodes and 88234 edges.

**Greedy Adaptivity Gap.** The greedy adaptivity gap is the ratio between the greedy adaptive algorithm (section 2.4.3.1) and the non-adaptive greedy. Golovin and Krause [2] showed that the adaptive greedy policy achieves a  $(1 - 1/e)$  approximation ( $\alpha_{adaptive}$  of figure 1.1) to the optimal adaptive policy. Therefore, the greedy adaptivity gap will be at least a fraction  $(1 - 1/e)$  of the adaptivity gap.

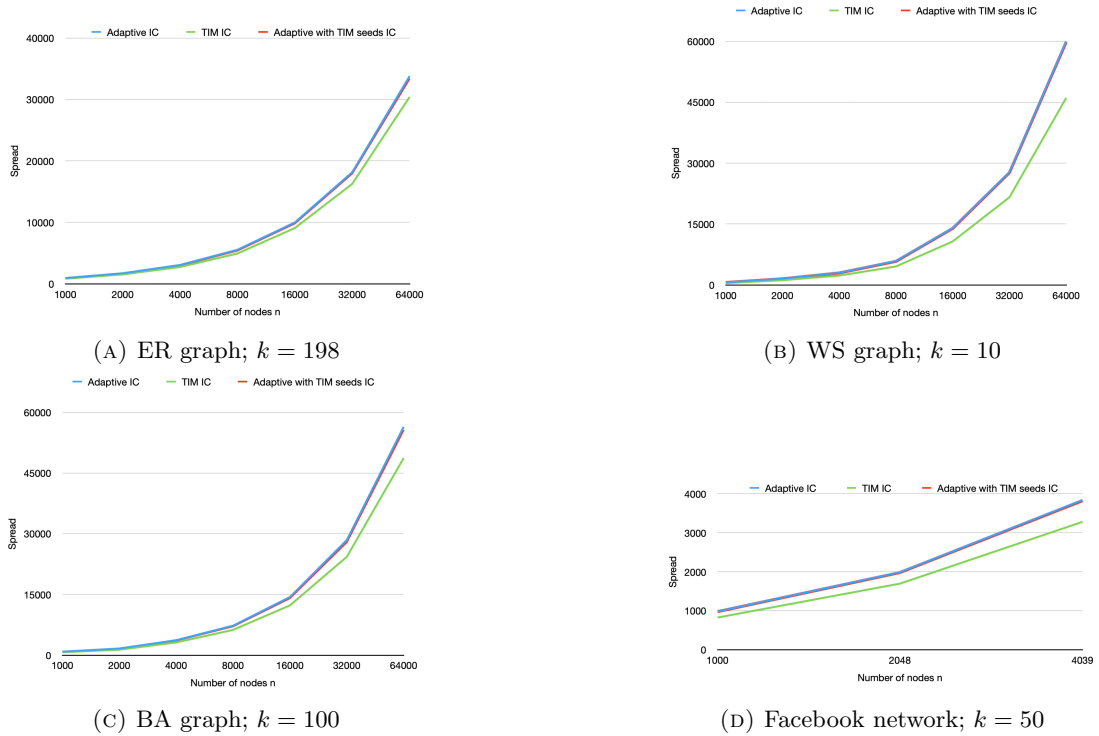
### 3.5.2 Independent Cascade

**Parameters.** For the IC model, we do the evaluation on the ER, WS and BA graphs by generating synthetic networks with  $n = \{1000, 2000, \dots, 32000, 64000\}$ . The edge weights for the aforementioned graphs are assigned at random ranging from  $[0.1, 1.0]$ . In the Facebook network, we use a BFS algorithm to sample subgraphs from the network with  $n = \{1000, 2048, 4039\}$ . For the ER and Facebook, we also analyzed the network behaviour by changing the edge activation probabilities to contain in between  $[0.1, 0.2]$ , and study the effects of low activation. The approximation error parameter  $\epsilon$  is set to 0.05 for all the experiments.

**Results.** The figures in 3.2 represent the 4 different networks with the edge probability chosen at random between  $[0.1, 1.0]$ . In general, the three generated graphs are comparable to each other with Barabasi-Albert having the best spread. From figure 3.2 and table 3.1, we observe that the seeds selected by the  $TIM^+$  algorithm are nearly as good as the ones selected by the adaptive greedy TIM. For comparing the quality of selected seeds that the adaptive greedy TIM and  $TIM^+$  produce, we introduce a new algorithm, called the adaptive greedy with TIM seeds, which take the seeds generated by  $TIM^+$ , but runs them on the same realisations generated by the adaptive greedy TIM.

Table 3.1 represents the ratio of the synthetic networks with the three variants of the TIM algorithm, with the greedy adaptivity gap being the ratio between the adaptive greedy TIM and  $TIM^+$ . Table 3.2 shows the three sampled networks of the Facebook graph and their adaptive ratios of the spread, with  $k = 50$  generated via the TIM

<sup>4</sup><https://snap.stanford.edu/data/ego-Facebook.html>

FIGURE 3.2: Results under the Independent Cascade model with  $p = [0.1, 1.0]$ .

Graph	Greedy Adaptivity Gap	Adaptive vs. greedy adaptive (TIM seeds)	Greedy adaptive (TIM seeds) vs. TIM+
Erdős-Rényi	1.2915	1.00004	1.2914
Watts-Strogatz	1.3024	1.0031	1.2983
Barabasi-Albert	1.3155	1.0068	1.3067

TABLE 3.1: Adaptive ratios with  $n = 64000$  and  $k = 10$  under the IC model for generated graphs.

No. of nodes	Greedy Adaptivity Gap	Adaptive vs. greedy adaptive (TIM seeds)	Greedy adaptive (TIM seeds) vs. TIM+
1000	1.18872	1.00497	1.18285
2048	1.17294	1.00458	1.16759
4039	1.17135	1.00541	1.16504

TABLE 3.2: Adaptive ratios for the Facebook network with  $k = 50$ .

algorithms. The ratios are normally higher than 1 because adaptive algorithms perform strictly better than the non-adaptive ones. The second column is the ratio between the adaptive greedy TIM and the adaptive greedy with the predetermined  $TIM^+$  seeds. The ratios in this column tends to 1 which proves that the seeds selected by  $TIM^+$  are capable of generating a spread close to its adaptive greedy equivalent. The third column is the column that is similar to the first one, where we compare the greedy adaptive with  $TIM^+$  seeds and the  $TIM^+$  algorithm. Notice that the values of this column is similar to the greedy adaptivity gap further proving that in this setting,  $TIM^+$  seeds are almost as good as the seeds selected by the adaptive greedy policy.

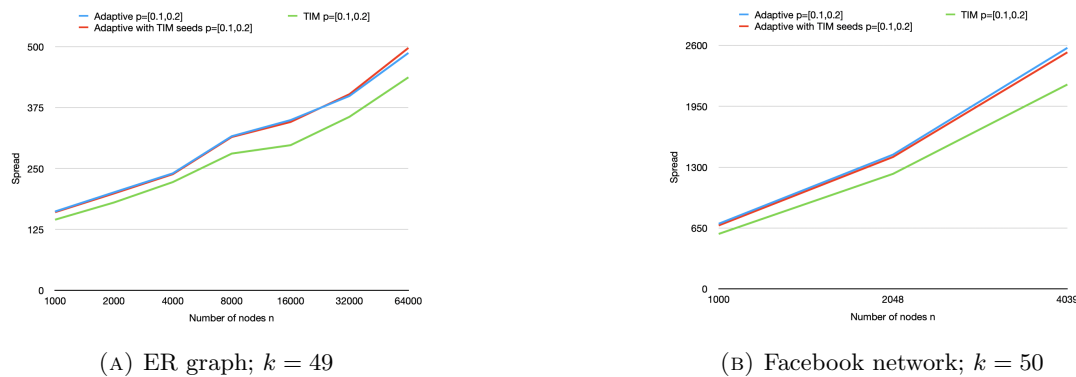


FIGURE 3.3: Results under the Independent Cascade model with  $p = [0.1, 0.2]$ .

We move on to Figure 3.3 where we restrain the probability on edges between 0.1 and 0.2 to observe the spread obtained by the three algorithms. From the graph, we see that the seeds selected by  $TIM^+$  has a coverage as good as the ones selected by the greedy adaptive policy for both, a synthetic ER graph and a real world Facebook network. We consider these two particular networks to show the difference in activation between a generated graph (ER has a low clustering coefficient which helps to understand the result for low activation probabilities), and the Facebook network represents a real world network.

In the end, since the results generated by the adaptive greedy TIM and  $TIM^+$  almost converges, if we want a time efficient seed selection process, we will select  $TIM^+$  but at a cost of huge memory overhead. On the other hand, if there is a constraint on the memory, and we need to use a personal computer to run huge networks, an adaptive greedy algorithm is what we want to use, but the time required to run such networks should be calculated in weeks, even when the realisations are run in parallel.

### 3.5.3 Linear Threshold

Since the BA graph and the Facebook network have high clustering coefficients, we considered just the same ER and WS graphs we generated for observing the behaviour of LT w.r.t IC. Again, the values of  $n$  range from 1000 to 64000. However, the generation of realisations for LT to run the adaptive greedy TIM differs from the IC.

**Parameters.** The BA graph and most other real world networks are power-law networks, and we observed that the LT model doesn't perform well with these graphs under the adaptive setting. So we consider just ER and WS graphs for the LT. The parameters are the same as in independent cascade with the error parameter  $\epsilon = 0.05$ . The probability on edges is in between  $[0.1, 1.0]$  and  $k = 198$  for ER and  $k = 10$  for WS,

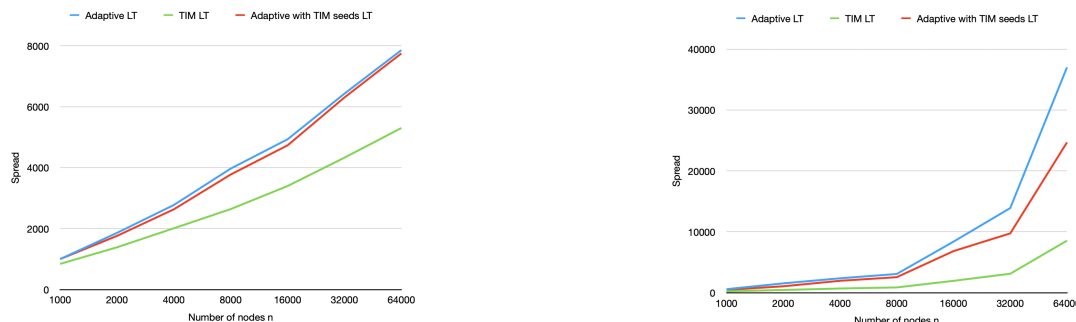
since the WS graph has a high clustering coefficient, which helps in activating almost the entire network with a lesser  $k$  value than that of ER. Separate realisations for adaptive TIM are generated according to the underlying LT diffusion model.

**Results.** From figure 3.4b we immediately notice in the WS graph plot that the  $TIM^+$  seeds do not perform as well as it was doing in the IC model. Table 3.3 compares the greedy adaptivity gap under both the diffusion models with the same parameters. We observe from the table that the greedy adaptivity gap on average increases by  $\approx 65\%$  for the WS network under LT with 10 seeds.

The ER graph plot shows a slight convergence between the red and blue lines representing the spread of adaptive greedy and the adaptive greedy with  $TIM^+$  seeds, but the seeds selected by  $TIM^+$  still perform quite well when compared to that of WS. To understand why we notice such a deviation can be attributed to the fact that the adaptive greedy TIM is no longer submodular in the LT model, but the theory is not yet proven. The question remains, that experimentally, we see that both models with a high and low clustering coefficient performs better in the LT model as compared to its IC counterpart, so is there a way to pass the submodularity requirement in order to reach better theoretical bounds on the adaptivity gap of the linear threshold model with the full-adoption feedback?

<i>No. of nodes</i>	<i>Greedy Adaptivity Gap (IC)</i>	<i>Greedy Adaptivity Gap (LT)</i>
1000	1.41086	2.45217
2000	1.34130	3.37854
4000	1.30168	3.38854
8000	1.30027	3.51286
16000	1.30840	4.28206
32000	1.28733	4.45013
64000	1.30238	4.32444

TABLE 3.3: Greedy adaptivity gap for the WS network under IC and LT with  $k = 10$ .

(A) ER graph;  $k = 198$ (B) WS graph;  $k = 10$ FIGURE 3.4: Results under the Linear Threshold model with  $p = [0.1, 1.0]$ .

### 3.6 Summary

This chapter studies the adaptivity gap of the independent cascade model with full-adoption feedback. We introduce a new technique to bridge the non-adaptive optimal and the adaptive optimal solution. Our technique directly builds the connection through defining the marginals related to the size  $t - 1$  non-optimal solution.

First, we improve on the upper bound of the adaptivity gap for in-arborescences from  $2e/(e - 1)$  to  $2e^2/(e^2 - 1)$ . Second, we give an upper bound of  $\lceil n^{1/3} \rceil$  for general graphs, which is the first sub-linear result on general graphs. Third, we show that for a class of  $\alpha$ -bounded graphs, the adaptivity gap is upper bounded by  $\sqrt{\alpha} + O(1)$ . Fourth, we prove that for a class of  $(\beta, \gamma)$ -bounded-activation graphs, we have an adaptivity gap of  $\sqrt{\beta} + \frac{1}{1-\gamma}$ . All the above results are based on our new technique.

We conduct some experiments on different type of synthetic and real world networks to understand the effect of adaptive greedy with the non-adaptive greedy algorithms. We give an overview of the greedy adaptivity gap to realize that even though the greedy adaptive algorithm is strictly better than the greedy non-adaptive policy, the adaptive greedy policy is not much better than the latter in both the IC and LT diffusion model. The adaptive greedy is memory efficient, whereas the non-adaptive greedy has a better efficiency in time.

## Chapter 4

# Improved Approximation Factor for Adaptive Influence Maximization via Simple greedy Strategies

In this chapter, we focus on the myopic model and analyze the approximation factor of the non-adaptive greedy algorithm without passing through the adaptivity gap [3.1]. We show that the algorithm achieves at least a fraction of  $\frac{1}{2} \left(1 - \frac{1}{e}\right) \approx 0.316$  of the adaptive optimum (Theorem 4.5). By definition, this implies that the adaptivity gap is at most  $\frac{2e}{e-1} \approx 3.164$  (Remark 4.12). For both approximation ratio and adaptivity gap we obtain a substantial improvement with respect to the upper bounds obtained in [39], which are  $\frac{1}{4} \left(1 - \frac{1}{e}\right) \approx 0.158$  and 4, respectively.

Non-adaptive policies are strictly weaker than adaptive ones, since the latter can implement the former by simply ignoring any kind of feedback. On the other hand, adaptive policies are difficult to implement as they require to probe suitable seeds and to observe the corresponding feedback, which can be expensive and error-prone. Moreover, they may consist of exponentially-large decision trees that are hard to compute and store. In contrast, non-adaptive policies are easy to design and implement and are independent from the feedback. In particular, the non-adaptive greedy algorithm has been extensively studied and successfully applied in the field of influence maximization. For the non-adaptive setting, several efficient implementations of the greedy algorithm have been devised that allow us to use it in large real-world networks [27, 28, 33–35, 64]. Our results show that the simple non-adaptive greedy algorithm performs well, even in the adaptive setting where we compare it with the adaptive optimum.

To show our bounds, we introduce a new approach that relate the non-adaptive greedy policy to an optimal adaptive solution. The new approach is not based on multilinear extensions and Poisson processes (like, e.g. [15, 51, 53, 65]) neither on random walks on the optimal decision trees (like, e.g. [39, 49, 50, 66]), which are the main tools used so far to relate adaptive and non-adaptive policies, and to bound the adaptivity gaps. Previous techniques derive adaptive approximation factors by combining non-adaptive approximation ratios with a bound on the adaptivity gap which is obtained by showing the existence of a “good” non-adaptive policy. However such a policy is hard to compute as it is usually constructed by using an optimal adaptive policy. Our approach, instead, directly analyzes a non-adaptive policy and therefore provides the exact policy that gives the desired adaptivity gap and adaptive approximation factor. We believe that our approach is of independent interest and may be used to bound approximation factors and adaptivity gaps of different adaptive optimization problems.

Our new approach consists in defining a simple randomized non-adaptive policy whose performance is not higher than that guaranteed by the greedy algorithm, and to relate such randomized non-adaptive policy with the optimal adaptive policy. In order to recover good properties of the objective function (like, e.g. submodularity) that usually guarantee good approximations when adopting greedy strategies, we introduce an artificial diffusion process in which each seed has two chances to influence its neighbours. A similar process was introduced by Peng and Chen [39], who consider a diffusion model in which the seeds appear in multiple copies of the influence graphs, so that, roughly speaking, each node has several chances to influence the neighbours, and the main machinery they consider to relate the optimal adaptive strategies with optimal non-adaptive ones are from Bradac et al. [49]. Our direct and more refined analysis of the non-adaptive greedy algorithm improves, at the same time, both the approximation ratio and the adaptivity gap.

To illustrate our approach, in Section 4.1 we first apply our machinery to the simpler setting of adaptive monotone submodular maximization under cardinality constraint. As observed by Asadpour and Nazerzadeh [51], a constant factor approximation of the non-adaptive greedy policy applied to such setting can be obtained by combining the approximation ratio over the non-adaptive optimum and the adaptivity gap (which is equal to  $(1 - \frac{1}{e})$  [51]); this leads to an approximation guarantee of  $(1 - \frac{1}{e})^2 \approx 0.399$ . We give a more refined analysis of the non-adaptive greedy policy and show that its approximation ratio is at least  $\frac{1}{2} (1 - \frac{1}{e^2}) \approx 0.432$ . Asadpour and Nazerzadeh [51] also showed that a so-called continuous greedy policy [53, 65] achieves an approximation ratio of  $1 - \frac{1}{e} - \epsilon$  (in polynomial time w.r.t.  $\frac{1}{\epsilon}$ ). Since the continuous greedy policy

is a non-adaptive policy, this bound is strictly better than ours. However, the non-adaptive greedy policy is simpler and deterministic, while the continuous greedy policy is randomized and more complex.

Finally, in Section 4.4, we analyze the adaptive version of the greedy algorithm applied to the adaptive influence maximization problem. Again by resorting to an artificial diffusion process, we show that such adaptive algorithms guarantees an approximation ratio of  $1 - \frac{1}{\sqrt{e}} \approx 0.393$ . Thus, we further improve the upper bound shown for the non-adaptive greedy algorithm, and we also give a more refined analysis of the adaptive greedy algorithm than that of Peng and Chen [39], who showed a  $\frac{1}{4} (1 - \frac{1}{e}) \approx 0.158$  approximation factor.

### Related Work.

Golovin and Krause [2] conjectured that the influence maximization problem under the myopic feedback model admits a constant approximation algorithm and a constant adaptivity gap, despite the missing adaptive submodularity under such a feedback model. Since then, several studies have been conducted on the myopic feedback model. Some recent works include that of Salha et al. [67], in which they consider a modified version of the independent cascade model which gives multiple chances to the seeds to activate their neighbours, and consider a different utility function which needs to be maximized. They demonstrate that the myopic feedback model is adaptive submodular under such modified diffusion model, and provide an adaptive greedy policy that achieves a  $1 - 1/e$  approximation ratio for the problem of finding the best adaptive policy. The work of Peng and Chen [39] is the first to provide a constant upper bound on the adaptivity gap under the myopic feedback model. They introduce a policy in which each seed appears in multiple copies of the original graph; furthermore, this hybrid policy connects the adaptive and the non-adaptive policies via a machinery used by [49, 50, 66] in the context of stochastic probing. Peng and Chen [39] prove that the adaptivity gap lies in between  $[e/(e-1), 4]$ , and by extension has a  $(1 - 1/e)/4$  approximation factor.

Singer et al. [56–59], in their line of research on adaptivity gaps, studied a two-stage process called adaptive seeding, that exhibits some similarities with the influence maximization problem under the myopic feedback model. Beyond influence maximization problems, adaptive optimization and adaptivity gaps has been generally studied for many other stochastic settings [2, 49, 49–51, 66, 68–76].

A general adaptive optimization framework deals with the fact that an item will reveal its actual state only when it has been irrevocably included in the final solution, and the main goal is to optimize an objective function under such uncertainty. Stochastic



variants of packing integer programs, 0/1 knapsack, and covering problems, have been studied under the perspective of adaptive optimization in [71–73], respectively.

Asadpour et al. [51, 69] study the adaptivity gap of the stochastic submodular maximization problem under a matroid constraint, in which the goal is to select a subset of items satisfying a matroid constraint, that maximizes the value of a monotone submodular value function defined on the random states of the selected items. In [51], the authors consider an adaptive greedy policy (often denoted as the myopic policy) to approximate the optimal value of the best adaptive policy. They show that it achieves a  $1/2$  approximation ratio under general matroid constraint, and a  $1 - \frac{1}{e}$  approximation ratio under cardinality constraint. Interesting variants or extensions of the above optimization problem have been considered in [49, 50, 66].

Chan and Farias [70] study the efficiency of adaptive greedy policies applied to a general class of stochastic optimization problems, called stochastic depletion, in which the adaptive policy, at each step, chooses an action that generates a reward and depletes some of the items. They show that, under certain structural properties, a simple adaptive greedy policy guarantees a constant factor approximation of the best adaptive policy. Hellerstein et al. [75] use an optimal decision tree to build a connection between the adaptive and the non-adaptive policies, and show that the adaptivity gap of stochastic submodular maximization under cardinality constraint is  $1 - \frac{1}{e^\tau}$ , where  $\tau$  is the minimum value of the probability that an item is in some state.

### Contribution.

The main contributions of this chapter lies in the new technique that we use to bound the approximation factor of the non-adaptive greedy algorithm, resulting in an improved bound of  $\frac{1}{2} \left(1 - \frac{1}{e}\right) \approx 0.316$ . The result is extended to also improve on the adaptivity gap to 3.164 from 4 for the myopic feedback under the independent cascade model. Finally, we also analyze the adaptive greedy algorithm, and show that it guarantees an improved approximation factor of  $1 - \frac{1}{\sqrt{e}} \approx 0.393$ .

### Organization of the Chapter.

In section Section 4.1 we introduce our new approach by applying it to a simpler setting. We apply our technique to the stochastic submodular maximization problem and introduce the necessary notation and definitions. Section 4.2 defines the AIM problem under the IC model with myopic feedback. In Section 4.3 we give the main results of this chapter, that is the improved approximation factor for the non-adaptive greedy algorithm

and the upper bound on adaptivity gap for the adaptive influence maximization problem under myopic feedback. Finally, in Section 4.4 we show an improved approximation ratio for the adaptive greedy algorithm.

## 4.1 Stochastic Submodular Maximization

In this section, we illustrate our machinery by applying part of it to the problem of maximizing a stochastic submodular set function under cardinality constraints [51].

For two integers  $h$  and  $k$ ,  $h \leq k$ , let  $[k]_h := \{h, h+1, \dots, k\}$  and  $[k] := [k]_1$ . A function  $f : \mathbb{R}_{\geq 0}^n \rightarrow \mathbb{R}_{\geq 0}$  is a *monotone submodular value function* if, for any vectors  $x, y \in \mathbb{R}_{\geq 0}^n$ , we get  $f(x \vee y) + f(x \wedge y) \leq f(x) + f(y)$ , where  $x \wedge y$  denotes the componentwise minimum and  $x \vee y$  denotes the componentwise maximum.

Let  $[n]$  be a finite set of  $n$  items, and let  $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_n)$  be a vector of  $n$  real, non-negative, and independent *state* random variables, where each  $\boldsymbol{\theta}_i$  returns the state  $\theta_i \in \mathbb{R}_{\geq 0}$  associated to each item  $i$  following a certain probability distribution. Let  $f : \mathbb{R}_{\geq 0}^n \rightarrow \mathbb{R}_{\geq 0}$  be the *objective function*, that is a monotone submodular value function. For any  $S \subseteq [n]$ , let  $\boldsymbol{\theta}(S) := (\boldsymbol{\theta}_1(S), \dots, \boldsymbol{\theta}_n(S))$  be the *partial state* random variable, that is a random vector defined as  $\boldsymbol{\theta}_i(S) = \boldsymbol{\theta}_i$  if  $i \in S$ ,  $\boldsymbol{\theta}_i(S) = 0$  otherwise. With a little abuse of notation, we assume that vector  $\boldsymbol{\theta}(S)$  gives also information on the set  $S$  which  $\boldsymbol{\theta}(S)$  is based on.

For a given integer  $k \geq 0$ , we aim at selecting a subset  $S \subseteq [n]$  subject to cardinality constraint  $|S| = k$ , that maximizes the expected value  $\mathbb{E}_{\boldsymbol{\theta}}[f(\boldsymbol{\theta}(S))]$ . To guarantee a (possibly) better solution we can resort to an *adaptive policy*, that, at each step, observes the partial state  $\xi \sim \boldsymbol{\theta}(U)$ , where  $U$  denotes the set of items previously selected, and selects another further item  $\pi(\xi) \in [n] \setminus U$ ; after  $k$  iterations, the policy returns a set  $U_{\boldsymbol{\theta},k}(\pi) \subseteq [n]$  with  $|U_{\boldsymbol{\theta},k}(\pi)| = k$ , which is a random set depending on the state of  $\boldsymbol{\theta}$ . The *stochastic monotone submodular maximization problem* (SMSM) takes as input a set of items  $[n]$ , a random vector  $\boldsymbol{\theta}$ , a monotone submodular value function  $f$ , and an integer  $k \in [n]$ , and asks to find an adaptive policy  $\pi$  that maximizes the expected value  $\mathbb{E}_{\boldsymbol{\theta}}[f(\boldsymbol{\theta}(U_{\boldsymbol{\theta},k}(\pi)))]$ .

In general, computing an optimal adaptive strategy is a computationally hard problem. Furthermore, in many contexts it is difficult to implement adaptive strategies, and non-adaptive strategies (in which the solution is chosen without observing the states of the random variables) is a more feasible choice. Asadpour and Nazerzadeh [51] show that a non-adaptive randomized continuous greedy algorithm guarantees a  $(1 - \frac{1}{e} - \epsilon)$  approximation for the SMSM problem (in polynomial time w.r.t.  $1/\epsilon$ ); however, the

proposed approach resorts to a quite sophisticated randomized algorithm. They also consider a simpler and deterministic *non-adaptive greedy algorithm* as approximation algorithm, that starts from an empty set  $S := \emptyset$  and, at each iteration  $t \in [k]$ , adds in  $S$  the item  $i \in [n] \setminus S$  maximizing  $\mathbb{E}_{\theta}[f(\theta(S \cup \{i\}))]$ . They show that the greedy algorithm guarantees an approximation factor of  $\frac{1}{2} \left(1 - \frac{1}{e}\right) \approx 0.316$  if the chosen subsets are subject to a matroid constraint, that can be reduced to  $\left(1 - \frac{1}{e}\right)^2 \approx 0.399$  when considering a cardinality constraint  $|S| \leq k$  only (i.e., uniform matroid constraint).

In the following theorem, we give a better analysis of the greedy algorithm under cardinality constraints, and we show that the approximation factor increases to  $\frac{1}{2} \left(1 - \frac{1}{e^2}\right) \approx 0.432$ .

**Theorem 4.1.** *The non-adaptive greedy algorithm is a  $\frac{1}{2} \left(1 - \frac{1}{e^2}\right)$  approximation algorithm for the SMSM problem.*

For the proof of Theorem 4.1 we relate the non-adaptive greedy solution with the optimal adaptive solution. We assume w.l.o.g that  $k \geq 2$ , otherwise the approximation ratio is equal to 1. For any  $t \in [k]_0$ , let  $S_t$  be the set of  $t$  items computed by the greedy algorithm at iteration  $t$  (where  $S_0 := \emptyset$ ). Let  $\pi$  be an optimal adaptive policy, and let  $x = (x_1, \dots, x_n) \in [0, 1]^n$  be the vector such that  $x_i$  is the probability that node  $i \in [n]$  is selected by policy  $\pi$ . Let  $OPT_A(k)$  denote the value of policy  $\pi$  (i.e., the optimal value of the problem). Given  $i \in [n]$ , let  $e^i := (e_1^i, \dots, e_n^i)$  be a random vector where  $e_j^i$  has the same distribution of  $\theta_j$  if  $i = j$ , and  $e_j^i = 0$  otherwise; given a partial state  $\xi$ , let  $\Delta(i|\xi) := \mathbb{E}_{e^i}[f(\xi \vee e^i) - f(\xi)]$ , i.e., the expected increment of the objective function when adding an item  $i$  under partial state  $\xi$ .

For any  $t \in [k-1]_0$ , as intermediate step of our analysis, we consider a *randomized non-adaptive policy*  $\text{Rand}_t$  that, starting from the greedy solution  $S_t$ , computes a random set  $S_{\rho,t} := S_t \cup \{\rho\}$ , where  $\rho \in [n]$  is a random item such that  $\mathbb{P}[\rho = i] = x_i/k$  for any  $i \in [n]$  and selected independently from any other event. Observe that the above random variable is well-defined, as  $\sum_{v \in V} (x_v/k) = k/k = 1$ ; furthermore, we observe that the expected value of  $f$  under  $\text{Rand}_t$  is  $\mathbb{E}_{\theta, \rho}[f(\theta(S_{\rho,t}))]$ . Furthermore, for any  $t \in [k-1]_0$ , we consider a *hybrid adaptive policy*  $\text{Hyb}_t$ , that first runs the adaptive policy  $\pi$ , and then merges the items of  $S_t$  with the items of  $U_{\hat{\theta},k}(\pi)$  selected by  $\pi$ , where  $\hat{\theta}$  is a random state that follows the same distribution of  $\theta$  but is independent from  $\theta$ ; finally, the expected value of  $f$  under  $\text{Hyb}_t$  is defined as  $\mathbb{E}_{\theta, \hat{\theta}}[f(\theta(S_t) \vee \hat{\theta}(U_{\hat{\theta},k}(\pi) \cup S_t))]$ .

A similar hybrid adaptive policy has been also considered by Asadpour and Nazerzadeh [51], but in place of the randomized non-adaptive policy considered in our work, they resort to a non-adaptive strategy defined by a Poisson process based on the multilinear extension of the expected value function  $\mathbb{E}_{\theta}[f(*)]$ . Before showing the theorem, we give

some preliminary lemmas, which relate the randomized non-adaptive policy with the hybrid adaptive policy.

**Lemma 4.2.** *We have that  $\mathbb{E}_{\theta, \rho}[f(\theta(S \cup \{\rho\}))] - \mathbb{E}_{\theta}[f(\theta(S))] = \sum_{i \in [n] \setminus S} x_i \cdot \mathbb{E}_{\theta}[\Delta(i|\theta(S))]$  for any  $S \subseteq [n]$ .*

*Proof.* As  $\mathbb{P}[\rho = i] = \frac{x_i}{k}$  for any  $i \in [n]$ , we have that

$$\begin{aligned} & \mathbb{E}_{\theta, \rho}[f(\theta(S \cup \{\rho\}))] - \mathbb{E}_{\theta}[f(\theta(S))] \\ &= k \cdot \mathbb{E}_{\rho}[\mathbb{E}_{\theta}[f(\theta(S \cup \rho)) - f(\theta(S))]] \\ &= k \cdot \sum_{i \in [n] \setminus S} \mathbb{P}[\rho = i] \cdot \mathbb{E}_{\theta}[\Delta(i|\theta(S))] \\ &= k \cdot \sum_{i \in [n] \setminus S} \frac{x_i}{k} \cdot \mathbb{E}_{\theta}[\Delta(i|\theta(S))] \\ &= \sum_{i \in [n] \setminus S} x_i \cdot \mathbb{E}_{\theta}[\Delta(i|\theta(S))]. \end{aligned}$$

□

**Lemma 4.3.** *For any  $S \subseteq [n]$ , we have  $\mathbb{E}_{\theta, \hat{\theta}}[f(\theta(S) \vee \hat{\theta}(U_{\hat{\theta}, k}(\pi) \cup S))] \leq \mathbb{E}_{\theta, \hat{\theta}}[f(\theta(S) \vee \hat{\theta}(S))] + \sum_{i \in [n] \setminus S} x_i \cdot \mathbb{E}_{\theta}[\Delta(i|\theta(S))]$ .*

*Proof.* It is sufficient to show that  $\mathbb{E}_{\hat{\theta}}[f(\xi \vee \hat{\theta}(U_{\hat{\theta}, k}(\pi) \cup S))] \leq \mathbb{E}_{\hat{\theta}}[f(\xi \vee \hat{\theta}(S))] + \sum_{i \in [n] \setminus S} x_i \cdot \Delta(i|\xi)$  for any partial state  $\xi \sim \theta(S)$ , so that, by doing the expectation on  $\xi \sim \theta(S)$ , the claim follows.

We observe that  $\mathbb{E}_{\hat{\theta}}[f(\xi \vee \hat{\theta}(U_{\hat{\theta}, k}(\pi) \cup S))]$  can be rewritten as the expected value  $\mathbb{E}_{\hat{\theta}}[f(\xi \vee \hat{\theta}(S))]$  coming from the selection of  $S$ , plus the sum of expected increments of  $f$  caused by all the nodes  $i \in [n] \setminus S$  selected by policy  $\pi$ , when observing partial states from  $\hat{\theta}$ . We have that the second sum can be written as

$$\sum_{i \in [n] \setminus S} \sum_{\hat{\xi}} \chi(i|\hat{\xi}) \cdot \Delta(i|\xi \vee \hat{\xi}),$$

where  $\chi(i|\hat{\xi}) \in \{0, 1\}$  is the indicator random variable that is equal to 1 if policy  $\pi$  visits state  $\hat{\xi}$  at some step of the execution and then selects node  $i$  (i.e.,  $i = \pi(\hat{\xi})$ ), and  $\chi(i|\hat{\xi}) = 0$  otherwise. Thus,

$$\begin{aligned} & \mathbb{E}_{\hat{\theta}}[f(\xi \vee \hat{\theta}(U_{\hat{\theta}, k}(\pi) \cup S))] \\ &= \mathbb{E}_{\hat{\theta}}[f(\xi \vee \hat{\theta}(S))] + \sum_{i \in [n] \setminus S} \sum_{\hat{\xi}} \chi(i|\hat{\xi}) \cdot \Delta(i|\xi \vee \hat{\xi}). \end{aligned} \tag{4.1}$$

For any partial state  $\hat{\xi}$  we have that

$$\begin{aligned}
& \Delta(i|\xi \vee \hat{\xi}) \\
&= \mathbb{E}_{e^i}[f(\xi \vee \hat{\xi} \vee e^i) - f(\xi \vee \hat{\xi})] \\
&= \mathbb{E}_{e^i}[f((\xi \vee \hat{\xi}) \vee (\xi \vee e^i)) - f(\xi \vee \hat{\xi})] \\
&\leq \mathbb{E}_{e^i}[f(\xi \vee e^i) - f((\xi \vee \hat{\xi}) \wedge (\xi \vee e^i))] \tag{4.2} \\
&\leq \mathbb{E}_{e^i}[f(\xi \vee e^i) - f(\xi)] \\
&= \Delta(i|\xi), \tag{4.3}
\end{aligned}$$

where (4.2) holds since  $f$  is a monotone submodular value function: indeed, we have that  $f(x \vee y) + f(x \wedge y) \leq f(x) + f(y)$  with  $x := (\xi \vee \hat{\xi})$  and  $y := (\xi \vee e^i)$ , thus showing (4.2). Furthermore, by exploiting the fact that  $x_i$  is the probability that an item  $i$  is selected in some step of policy  $\pi$ , we get

$$\sum_{\hat{\xi}} \chi(i|\hat{\xi}) = x_i. \tag{4.4}$$

Finally, by combining (4.1), (4.3), and (4.4), we get

$$\begin{aligned}
& \mathbb{E}_{\hat{\theta}}[f(\xi \vee \hat{\theta}(U_{\hat{\theta},k}(\pi) \cup S))] \\
&= \mathbb{E}_{\hat{\theta}}[f(\xi \vee \hat{\theta}(S))] + \sum_{i \in [n] \setminus S} \sum_{\hat{\xi}} \chi(i|\hat{\xi}) \cdot \Delta(i|\xi \vee \hat{\xi}) \\
&\leq \mathbb{E}_{\hat{\theta}}[f(\xi \vee \hat{\theta}(S))] + \sum_{i \in [n] \setminus S} \sum_{\hat{\xi}} \chi(i|\hat{\xi}) \cdot \Delta(i|\xi) \\
&= \mathbb{E}_{\hat{\theta}}[f(\xi \vee \hat{\theta}(S))] + \sum_{i \in [n] \setminus S} \left( \sum_{\hat{\xi}} \chi(i|\hat{\xi}) \right) \cdot \Delta(i|\xi) \\
&= \mathbb{E}_{\hat{\theta}}[f(\xi \vee \hat{\theta}(S))] + \sum_{i \in [n] \setminus S} x_i \cdot \Delta(i|\xi),
\end{aligned}$$

thus showing the claim.  $\square$

**Lemma 4.4.** *We have that  $OPT_A(k) \leq \mathbb{E}_{\theta, \hat{\theta}}[f(\theta(S) \vee \hat{\theta}(U_{\hat{\theta},k}(\pi) \cup S))]$  for any  $S \subseteq [n]$ .*

*Proof.* The claim easily follows from the fact that  $\hat{\theta}(U_{\hat{\theta},k}(\pi))$  is componentwise non-higher than  $\theta(S) \vee \hat{\theta}(U_{\hat{\theta},k}(\pi) \cup S)$ , thus, as  $f$  is monotone we get  $OPT_A(k) = \mathbb{E}_{\hat{\theta}}[f(\hat{\theta}(U_{\hat{\theta},k}(\pi)))] \leq \mathbb{E}_{\theta, \hat{\theta}}[f(\theta(S) \vee \hat{\theta}(U_{\hat{\theta},k}(\pi) \cup S))]$ .  $\square$

Armed with the above lemmas, we can prove Theorem 4.1.

*Proof of Theorem 4.1.* For any  $t \in [k]_0$ , let  $GR_N(t) := \mathbb{E}_{\theta}[f(\theta(S_t))]$  denote the expected value of  $f$  obtained at the  $t$ -th iteration of the non-adaptive greedy algorithm. We have that

$$GR_N(t+1) - GR_N(t) = \max_{v \in V} [\mathbb{E}_{\theta}[f(\theta(\{v\} \cup S_t))] - \mathbb{E}_{\theta}[f(\theta(S_t))] \quad (4.5)$$

$$\begin{aligned} &\geq \overbrace{\mathbb{E}_{\theta, \rho}[f(\theta(S_t \cup \{\rho\}))]}^{\text{Exp. value of Rand}_t} - \mathbb{E}_{\theta}[f(\theta(S_t))] \\ &\geq \frac{1}{k} \cdot \sum_{i \in [n] \setminus S_t} x_i \cdot \mathbb{E}_{\theta}[\Delta(i|\theta(S_t))] \end{aligned} \quad (4.6)$$

$$\geq \frac{1}{k} \cdot \left( \overbrace{\mathbb{E}_{\theta, \hat{\theta}}[f(\theta(S_t) \vee \hat{\theta}(U_{\hat{\theta}, k}(\pi) \cup S_t))] - \mathbb{E}_{\theta, \hat{\theta}}[f(\theta(S_t) \vee \hat{\theta}(S_t))]}^{\text{Exp. value of Hyb}_t} \right) \quad (4.7)$$

$$\geq \frac{1}{k} \cdot \left( OPT_A(k) - \mathbb{E}_{\theta, \hat{\theta}}[f(\theta(S_t) \vee \hat{\theta}(S_t))] \right) \quad (4.8)$$

$$\geq \frac{1}{k} \cdot OPT_A(k) - \frac{1}{k} \cdot \left( \mathbb{E}_{\theta, \hat{\theta}}[f(\theta(S_t)) + f(\hat{\theta}(S_t))] \right) \quad (4.9)$$

$$\geq \frac{1}{k} \cdot OPT_A(k) - \frac{1}{k} \cdot \left( \mathbb{E}_{\theta}[f(\theta(S_t))] + \mathbb{E}_{\hat{\theta}}[f(\hat{\theta}(S_t))] \right)$$

$$= \frac{1}{k} \cdot OPT_A(k) - \frac{2}{k} \cdot \mathbb{E}_{\theta}[f(\theta(S_t))]$$

$$= \frac{1}{k} \cdot OPT_A(k) - \frac{2}{k} \cdot GR_N(t), \quad (4.10)$$

where (4.5) comes from the fact that the greedy strategy, at iteration  $t+1$ , adds to  $S_t$  the item  $i$  guaranteeing the best expected value of  $f$ , (4.6) comes from Lemma 4.2, (4.7) comes from Lemma 4.3, (4.8) comes from Lemma 4.4, and (4.9) holds since  $f$  is a monotone submodular value function (as  $f(\theta(S_t) \vee \hat{\theta}(S_t)) \leq f(\theta(S_t) \vee \hat{\theta}(S_t)) + f(\theta(S_t) \wedge \hat{\theta}(S_t)) \leq f(\theta(S_t)) + f(\hat{\theta}(S_t))$ ). Thus, by (4.10) and some manipulations, we get  $GR_N(t+1) \geq \frac{1}{k} \cdot OPT_A(k) + (1 - \frac{2}{k}) \cdot GR_N(t)$  for any  $t \in [k-1]_0$ . By applying iteratively the above inequality, we get  $GR_N(k) \geq \frac{1}{k} \cdot \sum_{t=0}^{k-1} (1 - \frac{2}{k})^t \cdot OPT_A(k) = \frac{1}{2} \left( 1 - (1 - \frac{2}{k})^k \right) \cdot OPT_A(k)$ , that leads to  $\frac{GR_N(k)}{OPT_A(k)} \geq \frac{1}{2} \left( 1 - (1 - \frac{2}{k})^k \right) \geq \frac{1}{2} \left( 1 - \frac{1}{e^2} \right)$ , and this shows the claim.  $\square$

## 4.2 Adaptive Influence Maximization under the Myopic Feedback Model: Preliminaries

**Non-adaptive Influence Maximization.** Defined in Chapter 3. Refer to Chapter 2, Sections 2.1–2.3 for a more detailed overview.

**Adaptive Influence Maximization.** Differently from the non-adaptive setting, in which all the seeds are selected at the beginning, an *adaptive policy* activates the seeds

sequentially in  $k$  steps, one seed at each step, and the decision on the next seed to select is based on the feedback resulting from the observed spread of previously selected nodes. The feedback model considered in this work is *myopic*: when a node is selected, the adaptive policy observes the state of its neighbours.

An adaptive policy under the myopic feedback model is formally defined as follows. Given  $L \subseteq E$ , the *realisation*  $\phi_L : V \rightarrow 2^V$  associated to  $L$  assigns to each node  $v \in V$  the value  $\{z \in V : (v, z) \in L\} \cup \{v\}$ , i.e., the set containing  $v$  and the neighbours activated by seed  $v$  when  $\mathbf{L} = L$ . Let  $\Phi$  denote the *random realisation*, i.e., the random variable such that  $\mathbb{P}[\Phi = \phi_L] = \mathbb{P}[\mathbf{L} = L]$  for any  $L \subseteq E$ . Given a set  $S \subseteq V$ , a *partial realisation*  $\psi : S \rightarrow 2^V$  is the restriction to  $S$  of the domain of some realisation, i.e., there exists  $L \subseteq E$  such that  $\psi(v) = \phi_L(v)$  for any  $v \in S$ . Given a partial realisation  $\psi : S \rightarrow 2^V$ , let  $\text{dom}(\psi) := S$ , i.e.,  $\text{dom}(\psi)$  is the domain of partial realisation  $\psi$ , and let  $\text{Im}(\psi) := \bigcup_{v \in \text{dom}(\psi)} \psi(v)$ . A partial realisation  $\psi'$  is a *sub-realisation* of a partial realisation  $\psi$  (or, equivalently,  $\psi' \subseteq \psi$ ), if  $\text{dom}(\psi') \subseteq \text{dom}(\psi)$  and  $\psi'(v) = \psi(v)$  for any  $v \in \text{dom}(\psi')$ . We observe that any partial realisation  $\psi$  can be equivalently represented as  $\{(v, \phi_L(v)) : v \in \text{dom}(\psi)\}$  for some  $L \subseteq E$ .

An adaptive policy  $\pi$  takes as input a partial realisation  $\psi$  and, either returns a node  $\pi(\psi) \in V$  and activates it as seed, or interrupts the activation of new seeds, e.g., by returning a string  $\pi(\psi) := \text{STOP}$ . In particular, an adaptive policy  $\pi$  can be run as follows: (i) start from an empty realisation  $\psi := \emptyset$ ; (ii) if  $\pi(\psi) \neq \text{STOP}$  set  $\psi \leftarrow \psi \cup \{(v, \phi_L(v))\}$  and repeat (ii) until  $\pi(\psi) = \text{STOP}$ ; (iii) at the end, return  $\psi_\pi := \psi$ . Let  $\Psi_\pi$  be the random partial realisation returned by the execution of policy  $\pi$ . The *expected influence spread* of an adaptive policy  $\pi$  is defined as  $\sigma(\pi) := \mathbb{E}_{\mathbf{L}}[\sigma_{\mathbf{L}}(\text{dom}(\Psi_\pi))]$ , i.e., it is the expected value of the number of nodes reached by the diffusion process at the end of the execution of policy  $\pi$ . We say that  $|\pi| = k$  if policy  $\pi$  always return a partial realisation  $\psi_\pi$  with  $|\text{dom}(\psi_\pi)| = k$ . The *adaptive influence maximization problem (under IC and myopic feedback)* is the computational problem that, given an influence graph  $G$  and  $k \in [n]$ , asks to find an adaptive policy  $\pi$  subject to  $|\pi| = k$  that maximizes the expected influence spread  $\sigma(\pi)$ .

The adaptive influence maximization defined above is analogous to the definition in chapter 3 section 3.1, however, we need new notations to cope with the new proof arguments for the myopic feedback. We redefine the AIM scenario again in this chapter for the sake of clarity and completeness.

**Adaptivity Gap.** Defined formally in Section 3.1 Chapter 3.

### 4.3 Efficiency of Non-Adaptive Greedy algorithm

In this section, we show that a simple non-adaptive algorithm guarantees an approximation ratio of  $\frac{1}{2} \left(1 - \frac{1}{e}\right) \approx 0.316$  for the adaptive influence maximization problem, thus improving the approximation ratio of  $\frac{1}{4} \left(1 - \frac{1}{e}\right) \approx 0.158$  given in [39]. The algorithm provided by [39] is the usual *non-adaptive greedy algorithm* given in [9] and reported in the previous section.

We observe that such algorithm is non-adaptive, i.e., despite it is used for adaptive optimization, does not resort to the use of any adaptive policy and all the seeds are selected without observing any partial realisation.

**Theorem 4.5.** *Given an influence graph  $G$  with  $n$  nodes and  $k \in [n]_2$ , the non-adaptive greedy algorithm is a  $\frac{1}{2} \left(1 - \left(1 - \frac{1}{k}\right)^k\right) \geq \frac{1}{2} \left(1 - \frac{1}{e}\right)$  approximation algorithm for the adaptive influence maximization problem (under IC and myopic feedback) applied to  $(G, k)$ .*

In the proof of Theorem 4.5 (see Subsection 4.3.4) we relate the expected influence spread coming from the non-adaptive greedy algorithm with that of the optimal adaptive policy. We first need some notation and preliminary results. Let  $G = (V = [n], E, (p_{uv})_{(u,v) \in E})$  be an influence graph, and let  $k \in [n]_2$ . For any  $t \in [k]_0$ , let  $S_t$  denote the set of the first  $t$  seeds selected by the greedy algorithm, so that  $\sigma(S_k)$  is the expected influence spread of the solution returned by the algorithm. Let  $\pi$  be an optimal adaptive policy, and let  $x = (x_1, \dots, x_n)$  be the vector such that  $x_i$  is the probability that node  $i$  is selected by  $\pi$ . As in Section 4.1, we resort again to a randomized non-adaptive policy to relate the expected influence spread of the greedy algorithm with that of the adaptive policy.

**Randomized Non-adaptive Policy.** For any  $t \in [k-1]_0$ , as intermediate step of our analysis, we consider again the randomized non-adaptive policy  $\text{Rand}_t$  defined in Section 4.1: starting from the greedy solution  $S_t$ , we compute a random set  $S_{\rho,t} := S_t \cup \{\rho\}$ , where  $\rho \in [n]$  is a random item such that  $\mathbb{P}[\rho = i] = x_i/k$  for any  $i \in [n]$  and selected independently from any other event. We observe that the expected value of  $f$  under  $\text{Rand}_t$  is  $\mathbb{E}_{\rho}[\sigma(S_{\rho,t})]$ .

As a support of our analysis, we also define a new diffusion model and a hybrid adaptive policy. In particular, the new diffusion model will be used to recover certain properties connected with the submodularity that, as in Section 4.1, will allow us to relate the expected influence spread of the randomized non-adaptive policy with that of the hybrid adaptive policy. Furthermore, following again the approach of Section 4.1, the hybrid adaptive policy is obtained by combining the greedy (non-adaptive) solution and the



optimal adaptive one, and it will be used in our analysis to get an upper bound on the optimal adaptive spread.

**2-level Diffusion Model.** In the 2-level diffusion model each selected seed  $u \in V$  has two chances to influence its neighbours, and all the non-seeds have one chance only, i.e., the activation probability for all the edges  $(u, v)$  is  $1 - (1 - p_{u,v})^2$  if  $u$  is a seed, and  $p_{u,v}$  otherwise. More formally, let  $\hat{\mathbf{L}}$  be a live-edge graph distributed as  $\mathbf{L}$  and independent from  $\mathbf{L}$ . Given a set of seeds  $S$ , the *2-level live-edge graph* can be defined as  $\mathbf{L}^2(S) := \mathbf{L} \cup \{\hat{\mathbf{L}} \cap \{(u, v) \in E : u \in S\}\}$ . Given a set of nodes  $S$ , let  $\sigma_{\mathbf{L}, \hat{\mathbf{L}}}^2(S) := \sigma_{\mathbf{L}^2(S)}(S)$  denote the influence spread induced by  $S$  in live-edge graph  $\mathbf{L}^2(S)$ , and let  $\sigma^2(S) := \mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}}[\sigma_{\mathbf{L}, \hat{\mathbf{L}}}^2(S)]$  denote the expected influence spread induced by  $S$  under the 2-level diffusion model. Let  $\Phi$  and  $\hat{\Phi}$  denote the random realisations associated to live-edge graphs  $\mathbf{L}$  and  $\hat{\mathbf{L}}$ , respectively.

**2-Level Hybrid Adaptive Policy.** For any  $t \in [k-1]_0$ , let  $\text{Hyb}_t^2$  be a hybrid adaptive policy defined as follows: (i)  $\text{Hyb}_t^2$  selects all the nodes in  $S_t$  as seeds; (ii) then,  $\text{Hyb}_t^2$  adds to  $S_t$  all the nodes that the optimal adaptive policy  $\pi$  would have select when starting from the empty realisation, and observing, at each step, partial realisations coming from the live-edge graph  $\hat{\mathbf{L}}$  only; in other words, for any seed  $v \in V$  selected by the policy, the new set of nodes that the policy observes (to choose the next seed) is  $\hat{\Phi}(v)$ ; (iii) finally, denoting with  $\hat{\Psi}_\pi$  the random realisation returned by  $\pi$ , the expected influence spread of  $\text{Hyb}_t^2$  is defined as  $\mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}}[\sigma_{\mathbf{L}, \hat{\mathbf{L}}}^2(\text{dom}(\hat{\Psi}_\pi) \cup S_t)]$ , i.e., it is the expected influence spread determined  $\text{dom}(\hat{\Psi}_\pi) \cup S_t$ , according to the 2-level live-edge graph  $\mathbf{L}^2(\text{dom}(\hat{\Psi}_\pi) \cup S_t)$ .

In what follows we give some technical results which are based on the above definitions and will be used in Subsection 4.3.4 to show the main theorem. In particular, we show that the 2-level diffusion model satisfies certain properties connected with submodular set functions (see Subsection 4.3.1); then, we use such properties to relate the expected influence spread in the ordinary diffusion model to that of the 2-level diffusion model (see Subsection 4.3.2); finally, by using a similar approach as in Section 4.1, we use the above relations to relate the efficiency of the randomized non-adaptive policy with that of the optimal adaptive policy, via the 2-level hybrid adaptive policy (see Subsection 4.3.3).

### 4.3.1 Adaptive Submodularity in the 2-Level Diffusion Model

The adaptive submodularity [2] is a property that extends the well-known concept of submodularity to the adaptive framework, and allows us to design efficient adaptive

approximation algorithms. In particular, the adaptive submodularity states that, given two subrealisations  $\psi \subseteq \psi'$  and a node  $v \in V$ , adding node  $v$  under partial realisation  $\psi'$  causes an expected increment of the influence spread that is not higher than that caused under partial realisation  $\psi$ . Unfortunately, as shown in the arXiv version of [2], the myopic feedback model, in general, does not satisfy the adaptive submodularity.

Anyway, by resorting to the 2-level diffusion model, we can recover a similar property as the adaptive submodularity. Given  $S \subseteq V$ , a partial realisation  $\hat{\psi}$ , and  $v \in V$ , let

$$\Delta_S^2(v|\hat{\psi}) := \mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}} \left[ \sigma_{\mathbf{L}^2(\{v\} \cup \text{dom}(\hat{\psi}) \setminus S)}(\{v\} \cup S \cup \text{dom}(\hat{\psi})) - \sigma_{\mathbf{L}^2(\text{dom}(\hat{\psi}) \setminus S)}(S \cup \text{dom}(\hat{\psi})) | \hat{\psi} \subseteq \hat{\Phi} \right]$$

denote the expected increment of the influence spread w.r.t. the 2-level diffusion model when adding seed  $v$  to the set of nodes  $S \cup \text{dom}(\hat{\psi})$ , but assuming that the nodes in  $S$  have a unique chance to influence their neighbors, and that partial realisation  $\hat{\psi}$  has been observed. We say that the 2-level diffusion model is *adaptive submodular* if, for any  $S \subseteq V$ , any partial realisations  $\hat{\psi}, \hat{\psi}'$  with  $\hat{\psi} \subseteq \hat{\psi}'$ , and any  $v \in V$ , we have that  $\Delta_S^2(v|\hat{\psi}) \geq \Delta_S^2(v|\hat{\psi}')$ .

**Lemma 4.6.** *The 2-level diffusion model is adaptive submodular.*

*Proof.* To show the lemma, we connect the adaptive submodularity of the 2-level diffusion model, with the (non-adaptive) submodularity of the ordinary diffusion model shown by Kempe et al. [9]. They observed that, for any live-edge graph  $L$ , the expected influence spread function  $\sigma_L$  is a submodular set function: for any  $U, Y, Z \subseteq V$  such that  $U \subseteq Y$  we have that

$$\sigma_L(U \cup Z) - \sigma_L(U) \geq \sigma_L(Y \cup Z) - \sigma_L(Y). \quad (4.11)$$

Fix  $S \subseteq V$ , two partial realisations  $\hat{\psi}, \hat{\psi}'$  with  $\hat{\psi} \subseteq \hat{\psi}'$ , and  $v \in V$ . Given a partial realisation  $\bar{\psi}$ , let  $\bar{\psi}_{\setminus S}$  denote the subrealisation of  $\bar{\psi}$  with  $\text{dom}(\bar{\psi}_{\setminus S}) = \text{dom}(\bar{\psi}) \setminus S$ . We get

$$\begin{aligned} & \Delta_S^2(v|\hat{\psi}) \\ &= \mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}} \left[ \sigma_{\mathbf{L}^2(\{v\} \cup \text{dom}(\hat{\psi}) \setminus S)}(\{v\} \cup S \cup \text{dom}(\hat{\psi})) - \sigma_{\mathbf{L}^2(\text{dom}(\hat{\psi}) \setminus S)}(S \cup \text{dom}(\hat{\psi})) | \hat{\psi} \subseteq \hat{\Phi} \right] \\ &= \mathbb{E}_{\hat{\Phi}(v)} \left[ \mathbb{E}_{\mathbf{L}} [\sigma_{\mathbf{L}}(\hat{\Phi}(v) \cup S \cup \text{Im}(\hat{\psi}_{\setminus S})) - \sigma_{\mathbf{L}}(S \cup \text{Im}(\hat{\psi}_{\setminus S}))] \right] \\ &\geq \mathbb{E}_{\hat{\Phi}(v)} \left[ \mathbb{E}_{\mathbf{L}} [\sigma_{\mathbf{L}}(\hat{\Phi}(v) \cup S \cup \text{Im}(\hat{\psi}'_{\setminus S})) - \sigma_{\mathbf{L}}(S \cup \text{Im}(\hat{\psi}'_{\setminus S}))] \right] \\ &= \Delta_S^2(v|\hat{\psi}'), \end{aligned} \quad (4.12)$$

where (4.12) holds by non-adaptive submodularity: for  $U := S \cup \text{Im}(\hat{\psi}_{\setminus S})$ ,  $Y := S \cup \text{Im}(\hat{\psi}'_{\setminus S})$ , and  $Z := \hat{\Phi}(v)$ , we have that  $U \subseteq Y$ , thus we can apply (4.11) to derive (4.12).  $\square$

### 4.3.2 From the Ordinary to the 2-level Diffusion Model

In this subsection, we see how to relate the 2-level diffusion model to the ordinary one.

**Lemma 4.7.** *We have that  $\sigma^2(S) \leq 2 \cdot \sigma(S)$  for any  $S \subseteq V$ .*

*Proof.* By using the fact that the random graphs  $L$  and  $\hat{L}$  are independent and identically distributed, we have that

$$\begin{aligned} \sigma^2(S) &= \mathbb{E}_{L^2(S)}[\sigma_{L^2(S)}(S)] \\ &\leq \mathbb{E}_{L, \hat{L}}[\sigma_L(S) + \sigma_{\{L \setminus \{(u,v) \in E: u \in S\}\} \cup \{\hat{L} \cap \{(u,v) \in E: u \in S\}\}}(S)] \\ &\leq \mathbb{E}_L[\sigma_L(S)] + \mathbb{E}_{\hat{L}}[\sigma_{\hat{L}}(S)] = 2 \cdot \sigma(S). \end{aligned}$$

$\square$

Now, given a set  $S \subseteq V$  and  $v \in V$ , let  $\Delta_S(v) := \sigma(S \cup \{v\}) - \sigma(S)$ , that is the expected increment under the ordinary diffusion model when adding a new node to a set  $S$ , and let  $\Delta_S^2(v) := \mathbb{E}_{L, \hat{L}}[\sigma_{L^2(\{v\})}(S \cup \{v\}) - \sigma_L(S)]$ , that is the above expected increment, with the further assumption that  $v$  has two chances to influence its neighbors.

**Lemma 4.8.** *We have that  $\Delta_S^2(v) \leq 2 \cdot \Delta_S(v)$  for any  $S \subseteq V$  and  $v \in V$ .*

*Proof.* Let  $S \subseteq V$  and  $v \in V$ . Given a live-edge graph  $L$  and a set  $S \subseteq [n]$ , let  $L_{\setminus S}$  be another live edge graph obtained from  $L$  after removing all the nodes that would have been activated if  $S$  was the set of seeds, and after removing all the edges adjacent to such activated nodes (we observe that, if  $v \notin L_{\setminus S}$ , then  $\sigma_L(\{v\}) = 0$ ). We have that

$$\begin{aligned} \Delta_S^2(v) &= \mathbb{E}_{L^2(\{v\})_{\setminus S}}[\sigma_{L^2(\{v\})_{\setminus S}}(\{v\})] \\ &\leq 2 \cdot \mathbb{E}_{L_{\setminus S}}[\sigma_{L_{\setminus S}}(\{v\})] \\ &= 2 \cdot \Delta_S(v), \end{aligned} \tag{4.13}$$

where (4.13) can be shown by using analogous arguments as in the proof of Lemma 4.7.  $\square$

### 4.3.3 From the Adaptive to the Randomized Non-adaptive Policy

The following three lemmas (Lemma 4.9, Lemma 4.10, and 4.11) relate the randomized non-adaptive policy with the 2-level hybrid adaptive policy, and then with the optimal adaptive policy of the ordinary diffusion model. In particular, the proofs of Lemma 4.9, Lemma 4.10, and 4.11, resort to a similar approach of the proofs of Lemma 4.2, Lemma 4.3, and Lemma 4.4 of Section 4.1.

**Lemma 4.9.** *For any  $S \subseteq V$ , we have that  $k \cdot (\mathbb{E}_\rho[\sigma(\{\rho\} \cup S)] - \sigma(S)) \geq \sum_{v \in V \setminus S} x_v \cdot \Delta_S(v)$ .*

*Proof.* We have

$$\begin{aligned}
& k \cdot (\mathbb{E}_\rho[\sigma(\{\rho\} \cup S)] - \sigma(S)) \\
&= k \cdot (\mathbb{E}_\rho[\mathbb{E}_L[\sigma(\{\rho\} \cup S) - \sigma(S)]]) \\
&= k \cdot \mathbb{E}_\rho[\Delta_S(\rho)] \\
&= k \cdot \sum_{v \in V \setminus S} \frac{x_v}{k} \cdot \Delta_S(v) \\
&= \sum_{v \in V \setminus S} x_v \cdot \Delta_S(v).
\end{aligned}$$

□

The following lemma relates the adaptive setting with the non-adaptive one, and its proof resorts to the adaptive submodularity defined in Subsection 4.3.1.

**Lemma 4.10.** *We have  $\mathbb{E}_{L, \hat{L}}[\sigma_{L, \hat{L}}^2(\text{dom}(\hat{\Psi}_\pi) \cup S)] \leq \sigma^2(S) + \sum_{v \in V \setminus S} x_v \cdot \Delta_S^2(v)$  for any  $S \subseteq V$ .*

*Proof.* We observe that  $\mathbb{E}_{L, \hat{L}}[\sigma_{L, \hat{L}}^2(\text{dom}(\hat{\Psi}_\pi) \cup S)]$  can be rewritten as the expected influence spread  $\sigma^2(S)$  coming from the selection of  $S$  under the 2-level diffusion model, plus the sum of the expected increments of the influence spread under the 2-level diffusion model caused by all the nodes  $v \in V \setminus S$  selected by policy  $\pi$ , when observing partial realisations distributed according to  $\hat{\Phi}$ . We observe that the second sum is at most equal to

$$\sum_{v \in V \setminus S} \sum_{\hat{\psi}'} \chi(v|\hat{\psi}') \cdot \Delta_S^2(v|\hat{\psi}'),$$

where  $\chi(v|\hat{\psi}') \in \{0, 1\}$  is the indicator random variable that is equal to 1 if and only if policy  $\pi$  visits partial realisation  $\hat{\psi}'$  at some step of the execution and then selects node

$i$  (i.e.,  $i = \pi(\hat{\psi}')$ ). Thus, we have that

$$\mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}}[\sigma_{\mathbf{L}, \hat{\mathbf{L}}}^2(\text{dom}(\hat{\Psi}_\pi) \cup S)] \leq \sigma^2(S) + \sum_{v \in V \setminus S} \sum_{\hat{\psi}'} \chi(v|\hat{\psi}') \cdot \Delta_S^2(v|\hat{\psi}'). \quad (4.14)$$

As the probability that each node  $v \in V$  is selected by policy  $\pi$  is  $x_v$ , we have that

$$\sum_{\hat{\psi}'} \chi(v|\hat{\psi}') = x_v. \quad (4.15)$$

We have that,

$$\begin{aligned} & \sum_{v \in V \setminus S} \sum_{\hat{\psi}'} \chi(v|\hat{\psi}') \cdot \Delta_S^2(v|\hat{\psi}') \\ & \leq \sum_{v \in V \setminus S} \sum_{\hat{\psi}'} \chi(v|\hat{\psi}') \cdot \Delta_S^2(v|\emptyset) \end{aligned} \quad (4.16)$$

$$\begin{aligned} & \leq \sum_{v \in V \setminus S} \left( \sum_{\hat{\psi}'} \chi(v|\hat{\psi}') \right) \cdot \Delta_S^2(v|\emptyset) \\ & = \sum_{v \in V \setminus S} x_v \cdot \Delta_S^2(v), \end{aligned} \quad (4.17)$$

where (4.16) follows from Lemma 4.6 (i.e., from the adaptive submodularity, and  $\hat{\psi} := \emptyset$  is the empty partial realisation), and (4.17) holds by (4.15). By applying (4.17) to (4.14), we get

$$\begin{aligned} & \mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}}[\sigma_{\mathbf{L}, \hat{\mathbf{L}}}^2(\text{dom}(\hat{\Psi}_\pi) \cup S)] \\ & \leq \sigma^2(S) + \sum_{v \in V \setminus S} \sum_{\hat{\psi}'} \chi(v|\hat{\psi}') \cdot \Delta_S^2(v|\hat{\psi}') \\ & \leq \sigma^2(S) + \sum_{v \in V \setminus S} x_v \cdot \Delta_S^2(v), \end{aligned}$$

and this concludes the proof.  $\square$

The following lemma shows that the optimal adaptive influence spread is upper bounded by that expected influence spread of the 2-level hybrid adaptive policy.

**Lemma 4.11.** *We have  $\text{OPT}_A(G, k) \leq \mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}}[\sigma_{\mathbf{L}, \hat{\mathbf{L}}}^2(\text{dom}(\hat{\Psi}_\pi) \cup S)]$  for any  $S \subseteq V$ .*

*Proof.* We have that  $\text{OPT}_A(G, k) = \mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}}[\sigma_{\mathbf{L}, \hat{\mathbf{L}}}^2(\text{dom}(\hat{\Psi}_\pi))] \leq \mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}}[\sigma_{\mathbf{L}, \hat{\mathbf{L}}}^2(\text{dom}(\hat{\Psi}_\pi) \cup S)]$ .  $\square$

### 4.3.4 Proof of Theorem 4.5

Armed with the above results, we can now prove Theorem 4.5. For any  $t \in [k]_0$ , let  $GR_N(G, t) := \sigma(S_t)$  denote the expected influence spread  $\sigma(S_t)$  obtained when the first  $t$  seeds have been selected by the greedy algorithm. We have that

$$\begin{aligned} & GR_N(G, t+1) - GR_N(G, t) \\ &= \max_{v \in V} [\sigma(\{v\} \cup S_t)] - \sigma(S_t) \\ &\geq \overbrace{\mathbb{E}_{\rho}[\sigma(\{\rho\} \cup S_t)]}^{\text{Exp. value of Rand}_t} - \sigma(S_t) \end{aligned} \quad (4.18)$$

$$\geq \frac{1}{k} \sum_{v \in V \setminus S_t} x_v \cdot \Delta_{S_t}(v) \quad (4.19)$$

$$\geq \frac{1}{2k} \sum_{v \in V \setminus S_t} x_v \cdot \Delta_{S_t}^2(v) \quad (4.20)$$

$$\geq \frac{1}{2k} \overbrace{(\mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}}[\sigma_{\mathbf{L}, \hat{\mathbf{L}}}^2(\text{dom}(\hat{\Psi}_\pi) \cup S_t)] - \sigma^2(S_t))}^{\text{Exp. value of Hyb}_t^2} \quad (4.21)$$

$$\geq \frac{1}{2k} (OPT_A(G, k) - \sigma^2(S_t)) \quad (4.22)$$

$$\geq \frac{1}{2k} (OPT_A(G, k) - 2 \cdot \sigma(S_t)) \quad (4.23)$$

$$= \frac{1}{2k} OPT_A(G, k) - \frac{1}{k} GR_N(G, t), \quad (4.24)$$

where (4.18) holds since the greedy strategy adds to  $S_t$  the node  $v$  maximizing  $\sigma(S_t \cup \{v\})$ , (4.19) comes from Lemma 4.9, (4.20) comes from Lemma 4.8, (4.21) comes from Lemma 4.10, (4.22) comes from Lemma 4.11, and (4.23) comes from Lemma 4.7. Thus, by (4.24) and some manipulations we get the following recursive relation:  $GR_N(G, t+1) \geq \frac{1}{2k} \cdot OPT_A(G, k) + (1 - \frac{1}{k}) \cdot GR_N(G, t)$  for any  $t \in [k-1]_0$ . By applying iteratively the above inequality, we get  $GR_N(G, k) \geq \frac{1}{2k} \cdot \sum_{t=0}^{k-1} (1 - \frac{1}{k})^t \cdot OPT_A(G, k) = \frac{1}{2} \left(1 - (1 - \frac{1}{k})^k\right) \cdot OPT_A(G, k)$ , that leads to  $\frac{GR_N(G, k)}{OPT_A(G, k)} \geq \frac{1}{2} \left(1 - (1 - \frac{1}{k})^k\right) \geq \frac{1}{2} \left(1 - \frac{1}{e}\right)$ , and this shows the claim.  $\square$

*Remark 4.12.* By Theorem 4.5, we can easily show that, for any influence graph  $G$  with  $n$  nodes, the  $k$ -adaptivity gap of  $G$  is at most  $2 \left(1 - (1 - \frac{1}{k})^k\right)^{-1} \leq \frac{2e}{e-1} \approx 3.164$ .

## 4.4 The Efficiency of the Adaptive Greedy Algorithm

We show that the adaptive version of the greedy algorithm guarantees an even better approximation ratio of  $1 - \frac{1}{\sqrt{e}} \approx 0.393$  for the adaptive influence maximization problem. The *adaptive greedy algorithm* is an adaptive policy  $\pi_k^{GR}$  that selects  $k$  seeds in  $k$  steps,

and at each step  $t$  selects the  $t$ -th seed that maximizes the expected influence spread conditioned by the observed realisation.

**Theorem 4.13.** *Given an influence graph  $G$  with  $n$  nodes and  $k \in [n]_2$ , the adaptive greedy algorithm is a  $1 - (1 - \frac{1}{2k})^k \geq 1 - \frac{1}{\sqrt{e}}$  approximation algorithm for the adaptive influence maximization problem (under IC and myopic feedback) applied to  $(G, k)$ .*

In the proof of Theorem 4.13 we relate the expected influence spread coming from the adaptive greedy algorithm with that of the optimal adaptive policy, passing through a new hybrid adaptive policy. In the same spirit of Theorem 4.5, we also consider a new diffusion model and a new notion of adaptive submodularity (slightly different from that of Subsection 4.3.1). To show the main theorem (see Subsection 4.4.1) we need some notation and preliminary results. Let  $G = (V = [n], E, (p_{uv})_{(u,v) \in E})$  be an influence graph, and let  $k \in [n]_2$ . Let  $\pi$  be an optimal adaptive policy, and let  $x = (x_1, \dots, x_n)$  be the vector such that  $x_i$  is the probability that node  $i$  is selected by  $\pi$ .

**Strong 2-level Diffusion Model.** We define  $\mathbf{L}, \hat{\mathbf{L}}, \mathbf{L}^2(S), \Phi, \hat{\Phi}$  as in the 2-level diffusion model considered in Section 3.2. Given a partial realisation  $\psi$  and a set  $S \subseteq V$ , let  $\sigma_{\mathbf{L}, \hat{\mathbf{L}}, \psi}^2(S) := \sigma_{\mathbf{L}^2(S \setminus \text{dom}(\psi))}(S)$  denote the influence spread induced by  $S$  in live-edge graph  $\mathbf{L}^2(S \setminus \text{dom}(\psi))$ , and let  $\sigma_{\psi}^2(S) := \mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}}[\sigma_{\mathbf{L}, \hat{\mathbf{L}}, \psi}^2(S) | \psi \subseteq \Phi]$  denote the above influence spread in expectation, conditioned by partial realisation  $\psi$ .

**Strong 2-level Hybrid Adaptive Policy.** Given a partial realisation  $\psi$ , let  $\text{Hyb}_{\psi}^2$  be a hybrid adaptive policy defined as follows: (i)  $\text{Hyb}_{\psi}^2$  selects all the nodes in  $\text{dom}(\psi)$  as seeds; (ii) then,  $\text{Hyb}_{\psi}^2$  adds to  $\text{dom}(\psi)$  all the nodes that policy  $\pi$  would have selected when starting from the empty realisation, and observing, at each step, partial realisations coming from the live-edge graph  $\hat{\mathbf{L}}$  only; (iii) finally, denoting with  $\hat{\Psi}_{\pi}$  the random realisation returned by policy  $\pi$ , the expected influence spread of  $\text{Hyb}_{\psi}^2$  is defined as  $\mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}}[\sigma_{\mathbf{L}, \hat{\mathbf{L}}, \psi}^2(\text{dom}(\psi) \cup \text{dom}(\hat{\Psi}_{\pi})) | \psi \subseteq \Phi]$ , i.e., it is the expected influence spread determined  $\text{dom}(\psi) \cup \text{dom}(\hat{\Psi}_{\pi})$  in the strong 2-level diffusion model, conditioned by partial realisation  $\psi$ . Differently from the 2-level hybrid policy defined in Section 3.2, the expected influence spread of the hybrid policy defined here is conditioned by partial realisation  $\psi$  and the unique seeds that have two chances to influence their neighbors are those in  $\text{dom}(\hat{\Psi}_{\pi}) \setminus \text{dom}(\psi)$ .

**Strong Adaptive Submodularity of the Strong 2-level Diffusion Model.** Given two partial realisations  $\psi, \hat{\psi}$  and  $v \in V$ , let

$$\Delta_{\psi}^2(v|\hat{\psi}) = \mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}} \left[ \sigma_{\mathbf{L}, \hat{\mathbf{L}}, \psi}^2(\{v\} \cup \text{dom}(\psi) \cup \text{dom}(\hat{\psi})) - \sigma_{\mathbf{L}, \hat{\mathbf{L}}, \psi}^2(\text{dom}(\psi) \cup \text{dom}(\hat{\psi})) | \psi \subseteq \Phi, \hat{\psi} \subseteq \hat{\Phi} \right],$$

i.e.,  $\Delta_{\psi}^2(v|\hat{\psi})$  is the expected increment of the influence spread in the strong 2-level diffusion model when adding seed  $v$  to the set of nodes  $\text{dom}(\hat{\psi}) \cup \text{dom}(\psi)$ , conditioned by the observation of partial realisations  $\hat{\psi}$  and  $\psi$ . We say that the strong 2-level diffusion model is *strongly adaptive submodular* if, for any partial realisations  $\hat{\psi}, \hat{\psi}'$  with  $\hat{\psi} \subseteq \hat{\psi}'$ , any partial realisation  $\psi$ , and any  $v \in V$ , we have that  $\Delta_{\psi}^2(v|\hat{\psi}) \geq \Delta_{\psi}^2(v|\hat{\psi}')$ .

**Lemma 4.14.** *The strong 2-level diffusion model is strongly adaptive submodular.*

*Proof.* This lemma can be shown analogously to Lemma 4.6. Fix two partial realisations  $\hat{\psi}, \hat{\psi}'$  with  $\hat{\psi} \subseteq \hat{\psi}'$ , a partial realisation  $\psi$ , and  $v \in V$ . Given a partial realisation  $\bar{\psi}$ , let  $\bar{\psi}_{\setminus \psi}$  denote the subrealisation of  $\bar{\psi}$  with  $\text{dom}(\bar{\psi}_{\setminus \psi}) = \text{dom}(\bar{\psi}) \setminus \text{dom}(\psi)$ . We get

$$\begin{aligned} & \Delta_{\psi}^2(v|\hat{\psi}) \\ &= \mathbb{E}_{\hat{\Phi}(v)} \left[ \mathbb{E}_{\mathbf{L}} [\sigma_{\mathbf{L}}(\hat{\Phi}(v) \cup \text{Im}(\hat{\psi}_{\setminus \psi}) \cup \text{dom}(\psi)) - \sigma_{\mathbf{L}}(\text{Im}(\hat{\psi}_{\setminus \psi}) \cup \text{dom}(\psi)) | \psi \subseteq \Phi] \right] \\ &\geq \mathbb{E}_{\hat{\Phi}(v)} \left[ \mathbb{E}_{\mathbf{L}} [\sigma_{\mathbf{L}}(\hat{\Phi}(v) \cup \text{Im}(\hat{\psi}'_{\setminus \psi}) \cup \text{dom}(\psi)) - \sigma_{\mathbf{L}}(\text{Im}(\hat{\psi}'_{\setminus \psi}) \cup \text{dom}(\psi)) | \psi \subseteq \Phi] \right] \quad (4.25) \\ &= \Delta_{\psi}^2(v|\hat{\psi}'), \end{aligned}$$

where (4.25) holds since  $\sigma_{\mathbf{L}}$  is a submodular set function.  $\square$

**From the Ordinary to the Strong 2-level Diffusion Model.** Given a partial realisation  $\psi$ , and  $v \in V$ , let  $\Delta_{\psi}(v) := \mathbb{E}_{\mathbf{L}} [\sigma_{\mathbf{L}}(\{v\} \cup \text{dom}(\psi)) - \sigma_{\mathbf{L}}(\text{dom}(\psi)) | \psi \subseteq \Phi]$ , that is the expected increment under the ordinary diffusion model when adding a new node to the nodes in  $\text{dom}(\psi)$  conditioned by partial realisation  $\psi$ , and let  $\Delta_{\psi}^2(v) := \Delta_{\psi}^2(v|\emptyset) = \mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}} [\sigma_{\mathbf{L}, \hat{\mathbf{L}}, \psi}(\{v\} \cup \text{dom}(\psi)) - \sigma_{\mathbf{L}, \hat{\mathbf{L}}, \psi}(\text{dom}(\psi)) | \psi \subseteq \Phi]$ , that is the above conditional expectation, but w.r.t. the strong 2-level diffusion model. The following lemma can be shown analogously to Lemma 4.8.

**Lemma 4.15.** *We have that  $\Delta_{\psi}^2(v) \leq 2 \cdot \Delta_{\psi}(v)$  for any partial realisation  $\psi$  and  $v \in V$ .*

**From the Optimal to the Greedy Adaptive Policy.** The following lemma will be used to relate the strong 2-level hybrid adaptive policy with the adaptive greedy policy; its proof is similar to that of Lemma 4.10, but uses the concept of strong adaptive submodularity.



**Lemma 4.16.** *For any partial realisation  $\psi$ , we have,*

$$\begin{aligned} \mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}}[\sigma_{\mathbf{L}, \hat{\mathbf{L}}, \psi}^2(\text{dom}(\psi) \cup \text{dom}(\hat{\Psi}_\pi)) | \psi \subseteq \Phi] &\leq \mathbb{E}_{\mathbf{L}}[\sigma_{\mathbf{L}}(\text{dom}(\psi)) | \psi \subseteq \Phi] \\ &+ \sum_{v \in V \setminus \text{dom}(\psi)} x_v \cdot \Delta_\psi^2(v). \end{aligned} \quad (4.26)$$

*Proof.* We observe that  $\mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}}[\sigma_{\mathbf{L}, \hat{\mathbf{L}}, \psi}^2(\text{dom}(\psi) \cup \text{dom}(\hat{\Psi}_\pi)) | \psi \subseteq \Phi]$  can be rewritten as the expected influence spread  $\mathbb{E}_{\mathbf{L}}[\sigma_{\mathbf{L}}(\text{dom}(\psi)) | \psi \subseteq \Phi]$  conditioned by  $\psi$ , plus the sum of expected increments of the influence spread under the strong 2-level diffusion model caused by all the nodes  $v \in V \setminus \text{dom}(\psi)$  selected by policy  $\pi$ , conditioned by  $\psi$ . We observe that the second sum can be written as

$$\sum_{v \in V \setminus \text{dom}(\psi)} \sum_{\hat{\psi}'} \chi(v | \hat{\psi}') \cdot \Delta_\psi^2(v | \hat{\psi}'),$$

where  $\chi(v | \hat{\psi}') \in \{0, 1\}$  is the indicator random variable defined as in the proof of Lemma 4.10 (i.e., equal to 1 if and only if policy  $\pi$  visits partial realisation  $\hat{\psi}'$  at some step of the execution and then selects node  $v$ ). Thus, we have that

$$\begin{aligned} &\mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}}[\sigma_{\mathbf{L}, \hat{\mathbf{L}}, \psi}^2(\text{dom}(\psi) \cup \text{dom}(\hat{\Psi}_\pi)) | \psi \subseteq \Phi] \\ &= \mathbb{E}_{\mathbf{L}}[\sigma_{\mathbf{L}}(\text{dom}(\psi)) | \psi \subseteq \Phi] + \sum_{v \in V \setminus \text{dom}(\psi)} \sum_{\hat{\psi}'} \chi(v | \hat{\psi}') \cdot \Delta_\psi^2(v | \hat{\psi}'). \end{aligned} \quad (4.27)$$

As the probability that each node  $v \in V$  is selected by policy  $\pi$  is  $x_v$ , we have that

$$\sum_{\hat{\psi}'} \chi(v | \hat{\psi}') = x_v. \quad (4.28)$$

Thus, we have that

$$\begin{aligned} &\sum_{v \in V \setminus \text{dom}(\psi)} \sum_{\hat{\psi}'} \chi(v | \hat{\psi}') \cdot \Delta_\psi^2(v | \hat{\psi}') \\ &\leq \sum_{v \in V \setminus \text{dom}(\psi)} \sum_{\hat{\psi}'} \chi(v | \hat{\psi}') \cdot \Delta_\psi^2(v | \emptyset) \end{aligned} \quad (4.29)$$

$$\begin{aligned} &= \sum_{v \in V \setminus \text{dom}(\psi)} \left( \sum_{\hat{\psi}'} \chi(v | \hat{\psi}') \right) \cdot \Delta_\psi^2(v | \emptyset) \\ &= \sum_{v \in V \setminus \text{dom}(\psi)} x_v \cdot \Delta_\psi^2(v), \end{aligned} \quad (4.30)$$

where (4.29) follows from Lemma 4.14 (i.e., from the strong adaptive submodularity), and (4.30) holds by (4.28). By applying (4.30) to (4.27), we get

$$\begin{aligned}
& \mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}}[\sigma_{\mathbf{L}, \hat{\mathbf{L}}, \psi}^2(\text{dom}(\psi) \cup \text{dom}(\hat{\Psi}_\pi)) | \psi \subseteq \Phi] \\
&= \mathbb{E}_{\mathbf{L}}[\sigma_{\mathbf{L}}(\text{dom}(\psi)) | \psi \subseteq \Phi] + \sum_{v \in V \setminus \text{dom}(\psi)} \sum_{\hat{\psi}'} \chi(v | \hat{\psi}') \cdot \Delta_{\hat{\psi}'}^2(v | \hat{\psi}') \\
&\leq \mathbb{E}_{\mathbf{L}}[\sigma_{\mathbf{L}}(\text{dom}(\psi)) | \psi \subseteq \Phi] + \sum_{v \in V \setminus \text{dom}(\psi)} x_v \cdot \Delta_{\psi}^2(v),
\end{aligned}$$

thus the claim follows.  $\square$

The following lemma shows that the optimal adaptive influence spread is upper bounded by that expected influence spread of the strong 2-level hybrid adaptive policy, and its proof is completely analogous to that of Lemma 4.11.

**Lemma 4.17.** *We have  $\text{OPT}_A(G, k) \leq \mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}}[\sigma_{\mathbf{L}, \hat{\mathbf{L}}, \psi}^2(\text{dom}(\psi) \cup \text{dom}(\hat{\Psi}_\pi)) | \psi \subseteq \Phi]$  for any partial realisation  $\psi$ .*

#### 4.4.1 Proof of Theorem 4.13

Armed with the above lemmas, and by using a similar approach as in the proof of Theorem 4.5, we can prove Theorem 4.13. Given  $t \in [k]_0$ , let  $\mathbf{S}_t$  denote the (random) set of the first  $t$  seeds selected by the adaptive greedy policy  $\pi_k^{GR}$ , and let  $\Psi_t$  be the (random) partial realisation such that  $\text{dom}(\Psi_t) = \mathbf{S}_t$  (i.e., the partial realisation observed by policy  $\pi_k^{GR}$  at the end of step  $t$ ); let  $GR_A(G, t) := \mathbb{E}_{\mathbf{L}}[\sigma_{\mathbf{L}}(\mathbf{S}_t)]$  denote the expected influence spread of the adaptive greedy policy after selecting the first  $t$  seeds. For any  $t \in [k-1]_0$ , we have that

$$\begin{aligned}
& GR_A(G, t+1) - GR_A(G, t) \\
&= \mathbb{E}_{\mathbf{L}}[\sigma(\mathbf{S}_{t+1}) - \sigma(\mathbf{S}_t)] \\
&= \mathbb{E}_{\Psi_t} \left[ \max_{v \in V \setminus \text{dom}(\Psi_t)} \mathbb{E}_{\mathbf{L}}[\sigma_{\mathbf{L}}(\{v\} \cup \text{dom}(\Psi_t)) - \sigma_{\mathbf{L}}(\text{dom}(\Psi_t)) | \Psi_t] \right] \tag{4.31} \\
&\geq \mathbb{E}_{\Psi_t} \left[ \sum_{v \in V \setminus \text{dom}(\Psi_t)} \frac{x_v}{k} \cdot \mathbb{E}_{\mathbf{L}}[\sigma_{\mathbf{L}}(\{v\} \cup \text{dom}(\Psi_t)) - \sigma_{\mathbf{L}}(\text{dom}(\Psi_t)) | \Psi_t] \right] \\
&= \frac{1}{k} \cdot \mathbb{E}_{\Psi_t} \left[ \sum_{v \in V \setminus \text{dom}(\Psi_t)} x_v \cdot \mathbb{E}_{\mathbf{L}}[\sigma_{\mathbf{L}}(\{v\} \cup \text{dom}(\Psi_t)) - \sigma_{\mathbf{L}}(\text{dom}(\Psi_t)) | \Psi_t] \right]
\end{aligned}$$

$$\begin{aligned}
&= \frac{1}{k} \cdot \mathbb{E}_{\Psi_t} \left[ \sum_{v \in V \setminus \text{dom}(\Psi_t)} x_v \cdot \Delta_{\Psi_t}(v) \right] \\
&\geq \frac{1}{2k} \cdot \mathbb{E}_{\Psi_t} \left[ \sum_{v \in V \setminus \text{dom}(\Psi_t)} x_v \cdot \Delta_{\Psi_t}^2(v) \right] \tag{4.32}
\end{aligned}$$

$$\geq \frac{1}{2k} \cdot \mathbb{E}_{\Psi_t} \left[ \mathbb{E}_{\mathbf{L}, \hat{\mathbf{L}}}[\sigma_{\mathbf{L}, \hat{\mathbf{L}}, \Psi_t}^2(\text{dom}(\Psi_t) \cup \text{dom}(\hat{\Psi}_\pi)) | \Psi_t] - \mathbb{E}_{\mathbf{L}}[\sigma_{\mathbf{L}}(\text{dom}(\Psi_t)) | \Psi_t] \right] \tag{4.33}$$

$$\geq \frac{1}{2k} \cdot \mathbb{E}_{\Psi_t} [OPT_A(G, k) - \mathbb{E}_{\mathbf{L}}[\sigma_{\mathbf{L}}(\text{dom}(\Psi_t)) | \Psi_t]] \tag{4.34}$$

$$\geq \frac{1}{2k} \cdot OPT_A(G, k) - \frac{1}{2k} \cdot \mathbb{E}_{\Psi_t} [\mathbb{E}_{\mathbf{L}}[\sigma_{\mathbf{L}}(\text{dom}(\Psi_t)) | \Psi_t]]$$

$$= \frac{1}{2k} \cdot OPT_A(G, k) - \frac{1}{2k} \cdot \mathbb{E}_{\mathbf{L}}(\sigma_{\mathbf{L}}(\mathbf{S}_t))$$

$$= \frac{1}{2k} \cdot OPT_A(G, k) - \frac{1}{2k} \cdot GR_A(G, t), \tag{4.35}$$

where (4.31) holds since the adaptive greedy strategy at each step adds the node maximizing the expected influence spread (conditioned by the partial realisation coming from the previous selected seeds), (4.32) comes from Lemma 4.15, (4.33) comes from Lemma 4.16, (4.34) comes from Lemma 4.17. Thus, by (4.35) and some manipulations we get the following recursive relation:

$$GR_A(G, t+1) \geq \frac{1}{2k} \cdot OPT_A(G, k) + \left(1 - \frac{1}{2k}\right) \cdot GR_A(G, t), \quad \forall t \in [k-1]_0. \tag{4.36}$$

By applying iteratively (4.36), we get

$$GR_A(G, k) \geq \frac{1}{2k} \cdot \sum_{t=0}^{k-1} \left(1 - \frac{1}{2k}\right)^t \cdot OPT_A(G, k) = 1 - \left(1 - \frac{1}{2k}\right)^k \cdot OPT_A(G, k),$$

that leads to

$$\frac{GR_A(G, k)}{OPT_A(G, k)} \geq 1 - \left(1 - \frac{1}{2k}\right)^k \geq 1 - \frac{1}{\sqrt{e}},$$

and this shows the claim.

## 4.5 Summary

In this chapter, under the context of adaptive optimization, we have introduced a new approach to relate the solution provided by a simple non-adaptive greedy policy with the adaptive optimum. Our setting is based on the techniques used by Asadpour and Nazerzadeh [51]. Even though their results are strictly better than ours due to the usage

of continuous non-adaptive randomized policy achieved by Poisson process and multi-linear extension, our non-adaptive policy is simpler to design due to its deterministic nature and bypasses such complex calculations.

This new technique allowed us to establish better bounds for the adaptive influence maximization problem under the myopic feedback, specifically we improve both the approximation ratio of the non-adaptive greedy policy and the adaptivity gap from 4 to 3.164. We also show the approximation factor for the adaptive greedy under the same setting using our technique.

## Chapter 5

# Conclusion and Open Problems

In this thesis, we study the two feedback models of the Adaptive influence maximization problem focusing mainly on independent cascade. The adaptivity gap gives us a performance quantification of an optimal non-adaptive strategy when compared to an optimal adaptive one. The focus of the thesis is to bound the AG for different graph classes, feedbacks, diffusion models and to tighten that bound. Chapter 2 exhaustively explains the background and problem definitions, along with the necessary literature review related to this thesis.

In this chapter, we conclude the thesis by giving a brief overview on the contents and the results. Section 5.1 discusses the applications related to the scope of the thesis to further motivate the study of adaptivity gaps and its implications. We move on to discussing the open problems and future directions in Section 5.2. For a more concise idea about the gaps that are still open and needs to be found or tightened, refer to Table 5.1.

Chapter 3 of this thesis studies the full adoption feedback under the independent cascade model. Bypassing the requirements of multilinear extension and Poisson process, which are the commonly used techniques to link the non-adaptive spread with the adaptive one [15], we instead opt for introducing a hybrid policy and then use it to link the optimal non-adaptive and the adaptive policies. The hybrid adaptive policy ensures that at each step  $t$  of the diffusion process, the number of selected seed node is  $t$ . Whereas, the non-adaptive policy in [15] selects  $X$  number of seed nodes and evaluates the expected spread on the random variable  $X$ . However, instead of the total number of selected seeds being equal to  $k$ , it is  $k$  under expectation thus increasing the upper bound of AG for in-arborescences. We use this hybrid technique to improve the upper bound of in-arborescences, and bring it down to 2.31 from 3.16. We move on to give the first results on general graph and bound it to  $\lceil n^{1/3} \rceil$ , where  $n$  is the number of nodes. We also study other graph classes such as the  $\alpha$ -bounded-degree, upper bounded to  $\sqrt{\alpha} + O(1)$ , and

$(\beta, \gamma)$ -bounded-activation graphs, whose AG is bounded to  $\sqrt{\beta} + \frac{1}{1-\gamma}$ . In section 5.2.1, we state the open problems related to the results of this chapter. The experimental part of is in Section 3.5, where we use synthetic networks and a real world network. The purpose of the experiments is to study the state-of-the-art non-adaptive greedy algorithm called  $TIM^+$  and compare it using its adaptive greedy equivalent using a measure called the greedy adaptivity gap. We use different parameters and graph types under the IC and LT models with full-adoption to understand the behaviour of the greedy algorithms. The experimental part, shows that the greedy adaptivity gap is close to 1, thus ensuring the efficiency of  $TIM^+$ .

Chapter 4 focuses on the myopic model under IC, and its future directions are listed in Section 5.2.2. Since the myopic feedback model is not adaptive submodular, bounding the AG becomes non-trivial. To recover from the adaptive non-submodularity, we resort to an artificial 2-level diffusion model, which gives a seed node two chances to activate its neighbours. We introduce a new technique to arrive at an improved upper bound for the AG, and omit the use of random walks on optimal decision trees and multilinear extension to relate the non-adaptive and adaptive policies. Instead, by using a simple randomized non-adaptive policy, we relate it with the optimal adaptive policy, and derive the approximation ratio of the non-adaptive greedy algorithm which improved from 0.158 [39] to 0.316. We also notice an improvement in the adaptivity gap from 4 to 3.164 by extension. By using the same techniques, we also calculate the approximation factor for the adaptive greedy algorithm which is  $\approx 0.393$ .

TABLE 5.1: Current bounds on AG and GAG for different models and feedback

Model	Adaptivity Gap	Greedy Adaptivity Gap
Independent Cascade Full-Adoption	<b>Lower:</b> $e/(e-1)$ [15] General: $\lceil n^{1/3} \rceil$ [19] <b>Upper:</b> In-arborescence: $2e^2/(e^2-1)$ [19] Out-arborescence: $2$ [15] $\alpha$ -bounded degree: $\sqrt{\alpha} + O(1)$ [19] 1-D Bipartite: $e/(e-1)$ (tight) [15]	<b>Lower:</b> $1 - (1/e)$ [17] <b>Upper:</b> Open
Independent Cascade Myopic	<b>Lower:</b> $e/(e-1)$ [39] <b>Upper:</b> $2e/(e-1)$ [16]	<b>Lower:</b> $1 - (1/e)$ [17] <b>Upper:</b> $2e^2/(e-1)^2$ [16]
Linear Threshold Full-Adoption	<b>Lower:</b> Open <b>Upper:</b> Open	<b>Lower:</b> $1 - (1/e)$ [17] <b>Upper:</b> Open
Linear Threshold Myopic	<b>Lower:</b> Open <b>Upper:</b> Open	<b>Lower:</b> $1 - (1/e)$ [17] <b>Upper:</b> Open
Triggering Full-Adoption	<b>Lower:</b> $\infty$ [17] <b>Upper:</b> $\infty$ [17]	<b>Lower:</b> $1 - (1/e)$ [17] <b>Upper:</b> $\infty$ [17]
Triggering Myopic	<b>Lower:</b> $e/(e-1)$ [39] <b>Upper:</b> Open	<b>Lower:</b> $1 - (1/e)$ [17] <b>Upper:</b> Open

## 5.1 Applications of the Thesis

**In-arborescences.** Networks exhibiting hierarchical structures are one of the examples of in-arborescences. Many companies have inter networking as a way of information passing, and these are normally hierarchical networks with the number of terminals depending on the size of network. The entire inter network can be seen as an arborescence, with the root being the top level server, and the leaves being the individual employee terminals. The information propagation is usually bidirectional. One specific case, when an in-arborescence is an essential graph topology to consider is, when a middle tier or even an end terminal connection fails. Introducing uncertainty among the nodes based on their probability of failure makes the model stochastic. The example can be analogous to the problem of sensor placement and detecting failures in them.

Our objective is to maximize the number of activated systems in between, to ensure that there is maximum information propagation to the root, also the detection of failure is reported and handled as early as possible.

**Adaptive Stochastic Submodular Maximization.** Chapter 4 focuses mainly on the usage of the new technique in case of adaptive influence maximization, however, there are other applications which require analysis of non-adaptive policies with adaptive ones that can be considered under this new setting. As shown in Section 4.1, we believe that our new approach could be efficiently used to analyze non-adaptive greedy algorithms in other adaptive optimization problems.

One application is the stochastic probing problem studied by [49, 50, 66, 77], which in turn has further sub problems like maximum cardinality matching, path planning for robots, set cover problems [73, 74]. The stochastic probing problem is defined as follows: given an universal set of elements  $V$ , to know the state of an element  $v \in V$ , where each  $v$  is associated with an independent probability  $p_v$  determining the activation probability, we need to probe  $v$  in order to reveal if it is active or not. This problem is naturally adaptive because in order to probe the next best element, we look at the feedback generated by the previously probed elements that are now a part of the seed set  $S$ .

## 5.2 Discussions and Open problems

Adaptive seeding is a notoriously challenging problem. Due to the randomness on the edge activation, the problem becomes non-trivial, and bounding the adaptivity gap

becomes incredibly tricky. One challenge in the Independent Cascade model with full-adoption feedback is that the feedback generated from two nodes are not independent. This correlated feedback arises due to the propagation overlap from two nodes, which causes a substantial loss in the upper bound of the model. This problem is tackled in Chapter 3, where we try to lower the upper bound by directly building the connections through defining the marginals on the  $t - 1$  non-optimal solution.

Moving to Chapter 4, we deal with the non-submodular myopic feedback model under IC. The fact that the model is non-submodular makes it incredibly challenging to bound. We resort to artificial diffusion models which recovers the adaptive submodularity, however we suffer a substantial loss in the upper bound of the adaptivity gap.

### 5.2.1 Full-Adoption Feedback

The first problem that is left open by our results is the gap between the constant lower bound provided by Chen and Peng [15] and our upper bound on the adaptivity gap for general graphs. Besides trying to lower the upper bound, a possible direction could be that of increasing the lower bound by finding instances with a non constant adaptivity gap. Since the lower bound given in [15] holds even when the graph is a directed path, one direction could be to exploit different graph topologies.

Although in this work we have improved the upper bound on the adaptivity gap of in-arborescence, there is still a gap between the upper and lower bounds, thus another open problem is to close it. It would be also interesting to find better bounds on the adaptivity gap of other graph classes, like e.g. out-arborescences. A further interesting research direction is to study the adaptivity gap of some graph classes modelling real-world networks, both theoretically and experimentally.

Finally, most of the work on adaptive IM has been done for the independent cascade model, and other diffusion models (e.g., the linear threshold and the triggering models) have been less investigated. We observe that in many diffusion models different from the independent cascade (e.g., the linear threshold and the triggering models) the objective function is not adaptive submodular under both myopic and full-adoption feedback and the standard analysis of the greedy approach does not guarantee an efficient approximation. For the general triggering model, [60] overcome this problem by exploiting submodularity ratio, but constant bounds on both the adaptivity gap and the approximation ratio are guaranteed for bipartite graphs only, and the study of other graph topologies is still open.



### 5.2.2 Myopic Feedback

Our results open several research directions in the context of influence maximization and in more general adaptive optimization settings. The approximation factor of the non-adaptive (resp. adaptive) greedy algorithm is between our lower bound of  $\frac{1}{2} \left(1 - \frac{1}{e}\right) \approx 0.316$  (resp.  $1 - \frac{1}{\sqrt{e}} \approx 0.393$ ) and the upper bound of  $\frac{e^2+1}{(e+1)^2} \approx 0.606$  [39], and the adaptivity gap is between the lower bound of  $\frac{e}{e-1} \approx 1.582$  [39] and our upper bound of  $\frac{2e}{e-1} \approx 3.164$ . The first problem left open by our result is to close these gaps. Furthermore, the techniques introduced in this thesis to relate non-adaptive policies with adaptive ones might be useful to find better bounds in several variants of the adaptive influence maximization problem, like a combination of the following settings: different feedback models (e.g., the full-adoption feedback), different diffusion models (e.g., the general triggering model [1]), and different graph classes.



# Bibliography

- [1] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2003.
- [2] Daniel Golovin and Andreas Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *Journal of Artificial Intelligence Research*, 2011.
- [3] Stanley Milgram. The Small-World Problem. *Psychology Today*, 1967.
- [4] Jacqueline Johnson Brown and Peter H. Reingen. Social Ties and Word-of-Mouth Referral Behavior. *Journal of Consumer Research*, 1987.
- [5] Dawn Iacobucci and David Krackhardt. Structural Leverage in Marketing, 2014.
- [6] Pedro Domingos and Matt Richardson. Mining the network value of customers. *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2001.
- [7] 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*, 2002.
- [8] David Kempe, Jon Kleinberg, and Éva Tardos. Influential nodes in a diffusion model for social networks. In *Proceedings of the 32nd International Conference on Automata, Languages and Programming, ICALP*, 2005.
- [9] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. *Theory of Computing*, 2015.
- [10] G.L. Nemhauser and L.A. Wolsey. An Analysis of Approximations for Maximizing Submodular Set Functions-I. *Mathematical Programming*, 1978.
- [11] G. Nemhauser G. Cornuejols, M. Fisher. Location of bank accounts to optimize float. *Management Science*, 1977.

- 
- [12] Goldenberg J., Libai B., and Muller E. Talk of the Network: A Complex Systems Look at the Underlying Process of Word-of-Mouth. *Marketing Letters*, 2001.
- [13] Sharon Goldberg and Zhenming Liu. The diffusion of networking technologies. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, 2013.
- [14] Daniel Golovin and Andreas Krause. Adaptive submodularity: A new approach to active learning and stochastic optimization. *CoRR*, abs/1003.3967, 2017.
- [15] Wei Chen and Binghui Peng. On adaptivity gaps of influence maximization under the independent cascade model with full-adoption feedback. *30th International Symposium on Algorithms and Computation, ISAAC*, 2019.
- [16] Gianlorenzo D’Angelo, Debashmita Poddar, and Cosimo Vinci. Improved approximation factor for adaptive influence maximization via simple greedy strategies. In *48th International Colloquium on Automata, Languages, and Programming, ICALP*, 2021.
- [17] Wei Chen, Binghui Peng, Grant Schoenebeck, and Biaoshuai Tao. Adaptive Greedy versus Non-adaptive Greedy for Influence Maximization. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI*, 2020.
- [18] Cosimo Vinci Gianlorenzo D’Angelo, Debashmita Poddar. Better bounds on the adaptivity gap of influence maximization under full-adoption feedback. *Artificial Intelligence (AI) Journal*, To appear.
- [19] Gianlorenzo D’Angelo, Debashmita Poddar, and Cosimo Vinci. Better bounds on the adaptivity gap of influence maximization under full-adoption feedback. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI*, 2021.
- [20] Eyal Even-Dar and Asaf Shapira. A Note on Maximizing the Spread of Influence. *Proceedings of the 3rd international conference on Internet and network economics, WINE*, 2007.
- [21] Peter Clifford and Aidan Sudbury. A model for spatial conflict. *Biometrika*, 1973.
- [22] Kazumi Saito, Ryohei Nakano, and Masahiro Kimura. Prediction of information diffusion probabilities for independent cascade model. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2008.
- [23] Amit Goyal, Francesco Bonchi, and Laks V.S. Lakshmanan. Learning influence probabilities in social networks. *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining, WSDM*, 2010.

- 
- [24] Mark Granovetter. Threshold Models of Collective Behavior. *American Journal of Sociology*, 1978.
- [25] Wei Chen, Chi Wang, and Yajun Wang. Scalable influence maximization for prevalent viral marketing in large-scale social networks. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2010.
- [26] 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.
- [27] Jure Leskovec, Andreas Krause, Carlos Guestrin, Christos Faloutsos, Jeanne M. VanBriesen, and Natalie S. Glance. Cost-effective outbreak detection in networks. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2007.
- [28] Amit Goyal, Wei Lu, and Laks V. S. Lakshmanan. CELF++: optimizing the greedy algorithm for influence maximization in social networks. In *Proceedings of the 20th International Conference on World Wide Web, WWW*, 2011.
- [29] Suqi Cheng, Huawei Shen, Junming Huang, Guoqing Zhang, and Xueqi Cheng. Staticgreedy: solving the scalability-accuracy dilemma in influence maximization. In *22nd ACM International Conference on Information and Knowledge Management, CIKM*, 2013.
- [30] Naoto Ohsaka, Takuya Akiba, Yuichi Yoshida, and Ken-ichi Kawarabayashi. Fast and accurate influence maximization on large networks with pruned monte-carlo simulations. In *Proceedings of the Twenty-Eighth Conference on Artificial Intelligence, AAAI*, 2014.
- [31] Wei Chen, Yajun Wang, and Siyu Yang. Efficient influence maximization in social networks. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2009.
- [32] Vijay V. Vazirani. Approximation algorithms. *Springer*, 2002.
- [33] Youze Tang, Xiaokui Xiao, and Yanchen Shi. Influence Maximization : Near-Optimal Time Complexity Meets Practical Efficiency. *SIGMOD '14: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, 2014.
- [34] 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*, 2015.

- 
- [35] Hung T. Nguyen, My T. Thai, and Thang N. Dinh. Stop-and-stare: Optimal sampling algorithms for viral marketing in billion-scale networks. In *Proceedings of the 2016 International Conference on Management of Data, SIGMOD*, 2016.
- [36] Keke Huang, Sibor Wang, Glenn Bevilacqua, Xiaokui Xiao, and Laks V. S. Lakshmanan. Revisiting the stop-and-stare algorithms for influence maximization. *Proc. VLDB Endow.*, 2017.
- [37] Jing Tang, Laks V.S. Lakshmanan, Keke Huang, Xueyan Tang, Aixin Sun, Xiaokui Xiao, and Andrew Lim. Efficient approximation algorithms for adaptive seed minimization. *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 2019.
- [38] Sharan Vaswani and Laks V. S. Lakshmanan. Adaptive influence maximization in social networks: Why commit when you can adapt? *CoRR*, abs/1604.08171, 2016.
- [39] Binghui Peng and Wei Chen. Adaptive influence maximization with myopic feedback. *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing System, NeurIPS*, 2019.
- [40] Jing Yuan and Shaojie Tang. No time to observe: Adaptive influence maximization with partial feedback. *IJCAI International Joint Conference on Artificial Intelligence*, 2017.
- [41] Guangmo Tong and Ruiqi Wang. On adaptive influence maximization under general feedback models. *IEEE Transactions on Emerging Topics in Computing*, 2022.
- [42] Lichao Sun, Weiran Huang, Philip S. Yu, and Wei Chen. Multi-round influence maximization. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2018.
- [43] Alkis Gotovos, Amin Karbasi, and Andreas Krause. Non-monotone adaptive submodular maximization. In *Proceedings of the 24th International Conference on Artificial Intelligence, IJCAI*, 2015.
- [44] Niv Buchbinder, Moran Feldman, Joseph (Seffi) Naor, and Roy Schwartz. Submodular maximization with cardinality constraints. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, 2014.
- [45] Shaojie Tang. Beyond pointwise submodularity: Non-monotone adaptive submodular maximization in linear time. *Theoretical Computer Science*, 2021.
- [46] Georgios Amanatidis, Federico Fusco, Philip Lazos, Stefano Leonardi, and Rebecca Reiffenhäuser. Fast adaptive non-monotone submodular maximization subject to a

- knapsack constraint. In *Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS*, 2020.
- [47] Jianxiong Guo and Weili Wu. Adaptive influence maximization: If influential node unwilling to be the seed. *ACM Trans. Knowl. Discov. Data*, 2021.
- [48] Wei Chen, Ruihan Wu, and Zheng Yu. Scalable lattice influence maximization. *IEEE Transactions on Computational Social Systems*, 2020.
- [49] Domagoj Bradac, Sahil Singla, and Goran Zuzic. (near) optimal adaptivity gaps for stochastic multi-value probing. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM*, 2019.
- [50] Anupam Gupta, Viswanath Nagarajan, and Sahil Singla. Adaptivity gaps for stochastic probing: Submodular and XOS functions. *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, 2017.
- [51] Arash Asadpour and Hamid Nazerzadeh. Maximizing stochastic monotone submodular functions. *Management Science*, 2016.
- [52] James R. Norris. *Markov Chains*. Cambridge University Press, 1997.
- [53] Gruia Călinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM J. Comput.*, 2011.
- [54] Marek Adamczyk, Fabrizio Grandoni, and Joydeep Mukherjee. Improved approximation algorithms for stochastic matching. In *Proceedings of the 23rd European Symposium on Algorithms (ESA)*, pages 1–12, 2015.
- [55] Anupam Gupta, Viswanath Nagarajan, and Sahil Singla. Algorithms and adaptivity gaps for stochastic probing. *Proceedings of the twenty-seventh annual ACM-SIAM symposium on Discrete algorithms, SODA*, 2016.
- [56] Ashwinkumar Badanidiyuru, Christos Papadimitriou, Aviad Rubinfeld, Lior Seeman, and Yaron Singer. Locally adaptive optimization: Adaptive seeding for monotone submodular functions. *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, 2016.
- [57] Aviad Rubinfeld, Lior Seeman, and Yaron Singer. Approximability of adaptive seeding under knapsack constraints. *Proceedings of the 2015 ACM Conference on Economics and Computation, EC*, 2015.
- [58] Lior Seeman and Yaron Singer. Adaptive seeding in social networks. *Proceedings - Annual IEEE Symposium on Foundations of Computer Science, FOCS*, 2013.

- [59] Yaron Singer. Influence maximization through adaptive seeding. *ACM SIGecom Exchanges*, 2016.
- [60] Kaito Fujii and Shinsaku Sakaue. Beyond Adaptive Submodularity: Approximation Guarantees of Greedy Policy with Adaptive Submodularity Ratio. In *Proceedings of the 36th International Conference on Machine Learning, ICML*, 2019.
- [61] P. Erdős and A. Rényi. On random graphs, i. *Publicationes Mathematicae Debrecen*, 6, 1959.
- [62] Duncan J. Watts and Steven H. Strogatz. Collective dynamics of ‘small-world’ networks. *Nature*, 1998.
- [63] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *Science*, 1999.
- [64] 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.
- [65] Chandra Chekuri, Jan Vondrák, and Rico Zenklusen. Submodular function maximization via the multilinear relaxation and contention resolution schemes. *SIAM J. Comput.*, 2014.
- [66] Anupam Gupta, Viswanath Nagarajan, and Sahil Singla. Algorithms and Adaptivity Gaps for Stochastic Probing. *Proceedings of the twenty-seventh annual ACM-SIAM symposium on Discrete algorithms, SODA*, 2016.
- [67] Guillaume Salha, Nikolaos Tziortziotis, and Michalis Vazirgiannis. Adaptive submodular influence maximization with myopic feedback. *Proceedings of the 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM*, 2018.
- [68] Marek Adamczyk, Maxim Sviridenko, and Justin Ward. Submodular stochastic probing on matroids. In *31st International Symposium on Theoretical Aspects of Computer Science, STACS*, 2014.
- [69] Arash Asadpour, Hamid Nazerzadeh, and Amin Saberi. Stochastic submodular maximization. In *Internet and Network Economics, 4th International Workshop, WINE*, 2008.
- [70] Carri W. Chan and Vivek F. Farias. Stochastic depletion problems: Effective myopic policies for a class of dynamic optimization problems. *Math. Oper. Res.*, 2009.



- 
- [71] Brian C. Dean, Michel X. Goemans, and Jan Vondrák. Adaptivity and approximation for stochastic packing problems. In *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, 2005.
- [72] Brian C. Dean, Michel X. Goemans, and Jan Vondrák. Approximating the stochastic knapsack problem: The benefit of adaptivity. *Math. Oper. Res.*, 2008.
- [73] Michel X. Goemans and Jan Vondrák. Stochastic covering and adaptivity. In *Theoretical Informatics, 7th Latin American Symposium, LATIN*, 2006.
- [74] Andrew Guillory and Jeff A. Bilmes. Interactive submodular set cover. In *Proceedings of the 27th International Conference on Machine Learning, ICML*, 2010.
- [75] Lisa Hellerstein, Devorah Kletenik, and Patrick Lin. Discrete stochastic submodular maximization: Adaptive vs. non-adaptive vs. offline. In *Algorithms and Complexity - 9th International Conference, CIAC*, 2015.
- [76] Srinivasan Parthasarathy. An analysis of the greedy algorithm for stochastic set cover. *CoRR*, abs/1803.07639, 2018.
- [77] Anupam Gupta and Viswanath Nagarajan. A stochastic probing problem with applications. In *Proceedings of the 16th International Conference on Integer Programming and Combinatorial Optimization, IPCO*, 2013.