



Adaptive Regularized Submodular Maximization

Shaojie Tang  

Naveen Jindal School of Management, University of Texas at Dallas, TX, USA

Jing Yuan  

Department of Computer Science, University of Texas at Dallas, TX, USA

Abstract

In this paper, we study the problem of maximizing the difference between an adaptive submodular (revenue) function and a non-negative modular (cost) function. The input of our problem is a set of n items, where each item has a particular state drawn from some known prior distribution. The revenue function g is defined over items and states, and the cost function c is defined over items, i.e., each item has a fixed cost. The state of each item is unknown initially and one must select an item in order to observe its realized state. A policy π specifies which item to pick next based on the observations made so far. Denote by $g_{avg}(\pi)$ the expected revenue of π and let $c_{avg}(\pi)$ denote the expected cost of π . Our objective is to identify the best policy $\pi^o \in \arg \max_{\pi} g_{avg}(\pi) - c_{avg}(\pi)$ under a k -cardinality constraint. Since our objective function can take on both negative and positive values, the existing results of submodular maximization may not be applicable. To overcome this challenge, we develop a series of effective solutions with performance guarantees. Let π^o denote the optimal policy. For the case when g is adaptive monotone and adaptive submodular, we develop an effective policy π^l such that $g_{avg}(\pi^l) - c_{avg}(\pi^l) \geq (1 - \frac{1}{e} - \epsilon)g_{avg}(\pi^o) - c_{avg}(\pi^o)$, using only $O(n\epsilon^{-2} \log \epsilon^{-1})$ value oracle queries. For the case when g is adaptive submodular, we present a randomized policy π^r such that $g_{avg}(\pi^r) - c_{avg}(\pi^r) \geq \frac{1}{e}g_{avg}(\pi^o) - c_{avg}(\pi^o)$.

2012 ACM Subject Classification Theory of computation \rightarrow Approximation algorithms analysis; Mathematics of computing \rightarrow Approximation algorithms; Mathematics of computing \rightarrow Submodular optimization and polymatroids

Keywords and phrases Adaptive submodularity, approximation algorithms, active learning

Digital Object Identifier 10.4230/LIPIcs.ISAAC.2021.69

1 Introduction

Maximizing a submodular function subject to practical constraints has attracted increased attention recently [3, 16, 17, 7]. Submodularity encodes a natural diminishing returns property, which can be found in a wide variety of machine learning tasks such as active learning [3], virtual marketing [16, 17], sensor placement [7], and data summarization [8]. Under the non-adaptive setting, where one must select a group of items all at once, [11] shows that a classic greedy algorithm achieves $1 - 1/e$ approximation ratio for the problem of maximizing a monotone and non-negative submodular function subject to a cardinality constraint. For non-monotone and non-negative objectives, [1] obtains an approximation of $1/e + 0.004$.

Very recently, [4] studies the problem of maximizing the difference between a monotone non-negative submodular function and a non-negative modular function. Given that the objective function of the above problem may take both positive and negative values, most existing technologies, which require the objective function to take only non-negative values, can not provide nontrivial approximation guarantees. They overcome this challenge by developing a series of effective algorithms. In this paper, we extend their work to the adaptive setting by considering the problem of adaptive regularized submodular maximization, i.e., our goal is to adaptively select a group of items to maximize the difference between an adaptive submodular (revenue) function and a non-negative modular (cost) function. We next provide



© Shaojie Tang and Jing Yuan;

licensed under Creative Commons License CC-BY 4.0

32nd International Symposium on Algorithms and Computation (ISAAC 2021).

Editors: Hee-Kap Ahn and Kunihiko Sadakane; Article No. 69; pp. 69:1–69:13

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

more details about our adaptive setting. Following the framework of adaptive submodular maximization [3], a natural stochastic variant of the classical non-adaptive submodular maximization problem, we assume that each item is in a particular state drawn from a known prior distribution. The state of each item is unknown initially and one must select an item before observing its state. A policy π specifies which item to pick next based on the observations made so far. Note that the decision on selecting an item is irrevocable, that is, we can not discard any item that is previously selected. The revenue function g is defined over items and states, and the cost function c is defined over items. Note that there are two sources of randomness that make our problem more complicated than its non-adaptive counterpart. One is the random realization of items' states, and the other one is the random decision that is made by the policy. We use $g_{avg}(\pi)$ to denote the expected revenue of π and let $c_{avg}(\pi)$ denote the expected cost of π . Our objective is to identify the best policy:

$$\max_{\pi} g_{avg}(\pi) - c_{avg}(\pi)$$

under a k -cardinality constraint. The above formulation has its applications in many domains [5, 12]. When $g_{avg}(\pi)$ represents the revenue of π and $c_{avg}(\pi)$ encodes the cost of π , the above formulation is to maximize profits. In general, the above formulation may be interpreted as a regularized submodular maximization problem under the adaptive setting. Since our objective function can take on both negative and positive values, the existing results of non-monotone adaptive submodular maximization [14, 15], which require the objective function to take only non-negative values, may not be applicable.

Our contribution is threefold. We first consider the case when the revenue function g is adaptive monotone and adaptive submodular. Letting π^o denote the optimal policy, we develop an effective policy π^d such that $g_{avg}(\pi^d) - c_{avg}(\pi^d) \geq (1 - \frac{1}{e})g_{avg}(\pi^o) - c_{avg}(\pi^o)$, using $O(kn)$ value oracle queries. Our second result is the development of a faster policy π^l such that $g_{avg}(\pi^l) - c_{avg}(\pi^l) \geq (1 - \frac{1}{e} - \epsilon)g_{avg}(\pi^o) - c_{avg}(\pi^o)$, using only $O(n\epsilon^{-2} \log \epsilon^{-1})$ value oracle queries. For the case when g is (non-monotone) adaptive submodular, we present a randomized policy π^r such that $g_{avg}(\pi^r) - c_{avg}(\pi^r) \geq \frac{1}{e}g_{avg}(\pi^o) - c_{avg}(\pi^o)$.

2 Related Work

Submodular maximization is a well-studied topic due to its applications in a wide range of domains including active learning [3], virtual marketing [16, 17], sensor placement [7]. Most of existing studies focus on the non-adaptive setting where one must select a group of items all at once. [11] shows that a classic greedy algorithm, which iteratively selects the item that has the largest marginal revenue on top of the previously selected items, achieves a $1 - 1/e$ approximation ratio when maximizing a monotone non-negative submodular function subject to a cardinality constraint. The problem of maximizing a sum of a non-negative monotone submodular function and an (arbitrary) modular function is first studied in [13]. Notably, their objective function may take on negative values. [2] develops a faster algorithm using a surrogate objective that varies with time. For the case of a cardinality constraint and a non-negative c , [4] develops the first practical algorithm. Their results have been enhanced by [5] for the unconstrained case. Recently, [6, 12] extend this study to streaming and distributed settings. Our work is different from theirs in that we focus on the adaptive setting [3, 14, 15]. Moreover, we consider a more general problem of maximizing the difference of a non-negative *non-monotone* adaptive submodular function and a non-negative modular function.

3 Preliminaries

In the rest of this paper, we use $[m]$ to denote the set $\{0, 1, \dots, m\}$. We mostly follow [3] and adopt similar notations.

3.1 Items and States

The input of our problem is a set E of n items. Each item $e \in E$ is in a random state $\Phi(e) \in O$ where O represents the set of all possible states. We use a function $\phi : E \rightarrow O$, called a *realization*, to represent the realized states of all items, i.e., $\phi(e)$ represents a realization of $\Phi(e)$. There is a known prior probability distribution $p = \{\Pr[\Phi = \phi] : \phi \in U\}$ over all possible realizations U . The state $\Phi(e)$ of each item $e \in E$ is unknown initially and one must select e before observing its realized state. If we select multiple items $S \subseteq E$, then we are able to observe a *partial realization* $\psi : S \rightarrow O$ and $\text{dom}(\psi) = S$ is called the *domain* of ψ . A partial realization ψ is said to be consistent with a realization ϕ , denoted $\phi \sim \psi$, if they are equal everywhere in $\text{dom}(\psi)$. A partial realization ψ is said to be a *subrealization* of ψ' , denoted $\psi \subseteq \psi'$, if $\text{dom}(\psi) \subseteq \text{dom}(\psi')$ and they are equal everywhere in the domain $\text{dom}(\psi)$ of ψ . Let $p(\phi | \psi)$ denote the conditional distribution over realizations conditioned on a partial realization ψ : $p(\phi | \psi) = \Pr[\Phi = \phi | \Phi \sim \psi]$. In the rest of this paper, we use uppercase letters to denote random variables, and lowercase letters for realizations. For example, Ψ refers to a random variable, and ψ is a realization of Ψ .

3.2 Revenue and Cost

For a set $Y \subseteq E$ of items and a realization ϕ , let $g(Y, \phi)$ represent the revenue of selecting Y conditioned on ϕ , where g is called *revenue function*. Moreover, each item $e \in E$ has a fixed cost c_e . For any set subset of items $Y \subseteq E$, let $c(Y) = \sum_{e \in Y} c_e$ denote the total cost of Y , where c is called *cost function*.

3.3 Problem Formulation

A policy specifies which item to select next based on the partial realization observed so far. Mathematically, we represent a policy using a function π that maps a set of observations to a distribution $\mathcal{P}(E)$ of E : $\pi : 2^E \times O^E \rightarrow \mathcal{P}(E)$.

► **Definition 1** ([3], Policy Concatenation). *Given two policies π and π' , let $\pi @ \pi'$ denote a policy that runs π first, and then runs π' , ignoring the observation obtained from running π .*

► **Definition 2** ([3], Level- t -Truncation of a Policy). *Given a policy π , we define its level- t -truncation π_t as a policy that runs π until it selects t items.*

For each realization ϕ , let $E(\pi, \phi)$ denote the subset of items selected by π under realization ϕ . Note that $E(\pi, \phi)$ is a random variable. The expected revenue $g_{avg}(\pi)$ of a policy π can be written as

$$g_{avg}(\pi) = \mathbb{E}_{\Phi, \Pi}[g(E(\pi, \Phi), \Phi)]$$

where the expectation is taken over possible realizations according to p and the internal randomness of the policy. Similarly, the expected cost $c_{avg}(\pi)$ of a policy π can be written as

$$c_{avg}(\pi) = \mathbb{E}_{\Phi, \Pi}[c(E(\pi, \Phi))]$$

69:4 Adaptive Regularized Submodular Maximization

We next introduce the conditional expected marginal revenue $g(e \mid \psi)$ of e conditioned on a partial realization ψ :

$$g(e \mid \psi) = \mathbb{E}_{\Phi}[g(\text{dom}(\psi) \cup \{e\}, \Phi) - g(\text{dom}(\psi), \Phi) \mid \Phi \sim \psi]$$

where the expectation is taken over Φ with respect to $p(\phi \mid \psi) = \Pr(\Phi = \phi \mid \Phi \sim \psi)$.

► **Definition 3** ([3], Adaptive Submodularity). *For any two partial realizations ψ and ψ' such that $\psi \subseteq \psi'$, we assume that the following holds for each $e \in E \setminus \text{dom}(\psi')$:*

$$g(e \mid \psi) \geq g(e \mid \psi') \quad (1)$$

Let $\Omega = \{\pi \mid \forall \phi \in U^+, |E(\pi, \phi)| \leq k\}$ denote the set of all policies that select at most k items where $U^+ = \{\phi \in U \mid p(\phi) > 0\}$, our objective is listed in below:

$$\max_{\pi \in \Omega} g_{\text{avg}}(\pi) - c_{\text{avg}}(\pi) \quad (2)$$

Before presenting our solutions to the above problem, we introduce some additional notations. By abuse of notation, for any partial realization ψ , we define $g(\psi) = \mathbb{E}_{\Phi \sim \psi}[g(\text{dom}(\psi), \Phi)]$. We next introduce two useful functions: G_i , the distorted objective function, and H_i , which is used to analyze the trajectory of G_i . For any partial realization ψ , and any iteration $i \in [k]$ of our algorithms, we define

$$G_i(\psi) = \left(1 - \frac{1}{k}\right)^{k-i} g(\psi) - c(\text{dom}(\psi))$$

For any partial realization ψ , and any iteration $i \in [k-1]$ of our algorithms, we define

$$H_i(\psi, e) = \left(1 - \frac{1}{k}\right)^{k-(i+1)} g(e \mid \psi) - c_e$$

4 Monotone g : Adaptive Distorted Greedy Policy

We start with the case when g is adaptive submodular and *adaptive monotone* [3], i.e., for any realization ψ , the following holds for each $e \in E \setminus \text{dom}(\psi)$: $g(e \mid \psi) \geq 0$. Our approach is a natural extension of the *Distorted-Greedy* algorithm, the first practical non-adaptive algorithm developed in [4]. Note that there are two factors that make our problem more complicated than its non-adaptive counterpart. First, since the objective function is defined over random realization, the key of analysis is to estimate the expected utility under the distribution of realizations p . Second, the policy itself might produce random outputs even under the same realization, this adds an additional layer of difficulty to the design and analysis of our policy. To address the above complications, we develop an *Adaptive Distorted Greedy Policy* π^d such that $g_{\text{avg}}(\pi^d) - c_{\text{avg}}(\pi^d) \geq \left(1 - \frac{1}{e}\right) g_{\text{avg}}(\pi^o) - c_{\text{avg}}(\pi^o)$, where π^o denotes the optimal policy. We next explain the idea of π^d (Algorithm 1), then analyze its performance bound.

4.1 Design of π^d

We first add a dummy item d to the ground set, such that, $c_d = 0$, and for any partial realization ψ , we have $g(d \mid \psi) = 0$. Let $E' = E \cup \{d\}$. We add this to ensure that our policy will not select an item that has a negative profit. Note that d can be safely removed from the final solution without affecting its performance. π^d performs in k iterations: It starts with an empty set. In each iteration $i \in [k-1]$, let ψ_i denote the current partial realization, π^d selects an item e_i that maximizes $H_i(\psi_i, \cdot)$:

$$e_i \leftarrow \arg \max_{e \in E'} H_i(\psi_i, e)$$

After observing the state $\Phi(e_i)$ of e_i , we update the current partial realization ψ_{i+1} using $\psi_i \cup \{(e_i, \Phi(e_i))\}$ and enter the next iteration. This process iterates until all k items have been selected.

■ **Algorithm 1** Adaptive Distorted Greedy Policy π^d .

-
- 1: $S_0 = \emptyset; i = 0; \psi_0 = \emptyset$.
 - 2: **while** $i < k$ **do**
 - 3: $e_i \leftarrow \arg \max_{e \in E'} H_i(\psi_i, e)$;
 - 4: $S_{i+1} \leftarrow S_i \cup \{e_i\}$;
 - 5: $\psi_{i+1} \leftarrow \psi_i \cup \{(e_i, \Phi(e_i))\}; i \leftarrow i + 1$;
 - 6: **return** S_k
-

4.2 Performance Analysis

We first present three preparatory lemmas which are used to lower bound the marginal gain in the distorted objective. Recall that Ψ_{i+1} refers to a random variable, and ψ_{i+1} is a realization of Ψ_{i+1} .

► **Lemma 4.** *In each iteration of π^d ,*

$$\mathbb{E}_{\Phi \sim \psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\psi_i)] = H_i(\psi_i, e_i) + \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g(\psi_i)$$

Proof. We start with the case when $e_i \in \text{dom}(\psi_i)$,

$$\begin{aligned} & \mathbb{E}_{\Phi \sim \psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\psi_i)] \\ &= \left(1 - \frac{1}{k}\right)^{k-(i+1)} g(\psi_i) - \left(1 - \frac{1}{k}\right)^{k-i} g(\psi_i) \\ &= \left(1 - \frac{1}{k}\right)^{k-(i+1)} g(\psi_i) - \left(1 - \frac{1}{k}\right) \left(1 - \frac{1}{k}\right)^{k-(i+1)} g(\psi_i) \\ &= \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g(\psi_i) \\ &= H_i(\psi_i, e_i) + \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g(\psi_i) \end{aligned}$$

The last equality is due to $H_i(\psi_i, e_i) = 0$ when $e_i \in \text{dom}(\psi_i)$. We next prove the case when $e_i \notin \text{dom}(\psi_i)$,

$$\begin{aligned} & \mathbb{E}_{\Phi \sim \psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\psi_i)] \\ &= \mathbb{E}_{\Phi \sim \psi_i} \left[\left(1 - \frac{1}{k}\right)^{k-(i+1)} g(\psi_i \cup \{\Phi(e_i)\}) \right. \\ & \quad \left. - c(\text{dom}(\psi_i) \cup \{e_i\}) - \left(\left(1 - \frac{1}{k}\right)^{k-i} g(\psi_i) - c(\text{dom}(\psi_i)) \right) \right] \\ &= \mathbb{E}_{\Phi \sim \psi_i} \left[\left(1 - \frac{1}{k}\right)^{k-(i+1)} g(\psi_i \cup \{\Phi(e_i)\}) \right. \\ & \quad \left. - c(\text{dom}(\psi_i) \cup \{e_i\}) - \left(\left(1 - \frac{1}{k}\right)^{k-i} g(\psi_i) - c(\text{dom}(\psi_i)) \right) \right] \\ &= \mathbb{E}_{\Phi \sim \psi_i} \left[\left(1 - \frac{1}{k}\right)^{k-(i+1)} g(\psi_i \cup \{\Phi(e_i)\}) \right. \\ & \quad \left. - c(\text{dom}(\psi_i) \cup \{e_i\}) - \left(\left(1 - \frac{1}{k}\right)^{k-(i+1)} \left(1 - \frac{1}{k}\right) g(\psi_i) - c(\text{dom}(\psi_i)) \right) \right] \\ &= \mathbb{E}_{\Phi \sim \psi_i} \left[\left(1 - \frac{1}{k}\right)^{k-(i+1)} (g(\psi_i \cup \{\Phi(e_i)\}) - g(\psi_i)) \right] - c_{e_i} + \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g(\psi_i) \end{aligned}$$

69:6 Adaptive Regularized Submodular Maximization

$$\begin{aligned}
&= g(e_i \mid \psi_i) - c_{e_i} + \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g(\psi_i) \\
&= H_i(\psi_i, e_i) + \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g(\psi_i)
\end{aligned}$$

The fourth equality is due to $e_i \notin \text{dom}(\psi_i)$. \blacktriangleleft

► **Lemma 5.** *In each iteration of π^d ,*

$$H_i(\psi_i, e_i) \geq \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} \mathbb{E}_{\Phi \sim \psi_i} [g_{\text{avg}}(\pi^o) - g_{\text{avg}}(\pi_i^d)] - \frac{1}{k} \mathbb{E}_{\Phi \sim \psi_i} [c_{\text{avg}}(\pi^o)]$$

Proof. Let A_e be an indicator that e is selected by the optimal solution π^o conditioned on a partial realization ψ_i , then we have

$$\begin{aligned}
H_i(\psi_i, e_i) &= \left(1 - \frac{1}{k}\right)^{k-(i+1)} g(e_i \mid \psi_i) - c_{e_i} \\
&= \max_{e \in E'} \left[\left(1 - \frac{1}{k}\right)^{k-(i+1)} g(e \mid \psi_i) - c_e \right] \\
&\geq \frac{1}{k} \sum_{e \in E'} \Pr[A_e = 1] \left[\left(1 - \frac{1}{k}\right)^{k-(i+1)} g(e \mid \psi_i) - c_e \right] \\
&= \frac{1}{k} \sum_{e \in E'} \Pr[A_e = 1] \left[\left(1 - \frac{1}{k}\right)^{k-(i+1)} g(e \mid \psi_i) \right] - \frac{1}{k} \sum_{e \in E'} \Pr[A_e = 1] \times c_e \\
&= \frac{1}{k} \sum_{e \in E'} \Pr[A_e = 1] \left[\left(1 - \frac{1}{k}\right)^{k-(i+1)} g(e \mid \psi_i) \right] - \frac{1}{k} \mathbb{E}_{\Phi \sim \psi_i} [c_{\text{avg}}(\pi^o)] \\
&\geq \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} \mathbb{E}_{\Phi \sim \psi_i} [g_{\text{avg}}(\pi^o) - g_{\text{avg}}(\pi_i^d)] - \frac{1}{k} \mathbb{E}_{\Phi \sim \psi_i} [c_{\text{avg}}(\pi^o)]
\end{aligned}$$

The second equality is due to the design of π^d , i.e., it selects an item e_i that maximizes $H_i(\psi_i, \cdot)$. The first inequality is due to $\sum_{e \in E'} \Pr[A_e = 1] \leq k$ since π^o selects at most k items. The third equality is due to the assumption that $\Pr[A_e = 1]$ is the probability that e is selected by π^o conditioned on ψ_i . The second inequality is due to g is adaptive submodular. \blacktriangleleft

► **Lemma 6.** *In each iteration of π^d ,*

$$\mathbb{E}_{\Psi_i} [\mathbb{E}_{\Phi \sim \Psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\Psi_i)]] \geq \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g_{\text{avg}}(\pi^o) - \frac{1}{k} c_{\text{avg}}(\pi^o)$$

Proof. We first prove that for any fixed partial realization ψ_i , the following inequality holds:

$$\mathbb{E}_{\Phi \sim \psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\psi_i)] \geq \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} \mathbb{E}_{\Phi \sim \psi_i} [g_{\text{avg}}(\pi^o)] - \frac{1}{k} \mathbb{E}_{\Phi \sim \psi_i} [c_{\text{avg}}(\pi^o)] \quad (3)$$

Due to Lemma 4, we have

$$\begin{aligned}
&\mathbb{E}_{\Phi \sim \psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\psi_i)] \\
&= H_i(\psi_i, e_i) + \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g(\psi_i) \\
&\geq \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} \mathbb{E}_{\Phi \sim \psi_i} [g_{\text{avg}}(\pi^o) - g_{\text{avg}}(\pi_i^d)] \\
&\quad - \frac{1}{k} \mathbb{E}_{\Phi \sim \psi_i} [c_{\text{avg}}(\pi^o)] + \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g(\psi_i) \\
&= \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} \mathbb{E}_{\Phi \sim \psi_i} [g_{\text{avg}}(\pi^o)] - \frac{1}{k} \mathbb{E}_{\Phi \sim \psi_i} [c_{\text{avg}}(\pi^o)]
\end{aligned}$$

The inequality is due to Lemma 5 and the second equality is due to $\mathbb{E}_{\Phi \sim \psi_i} [g_{avg}(\pi_i^d)] = g(\psi_i)$. Now we are ready to prove this lemma.

$$\begin{aligned}
& \mathbb{E}_{\Psi_i} [\mathbb{E}_{\Phi \sim \Psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\Psi_i)]] \\
& \geq \mathbb{E}_{\Psi_i} \left[\frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} \mathbb{E}_{\Phi \sim \Psi_i} [g_{avg}(\pi^o)] - \frac{1}{k} \mathbb{E}_{\Phi \sim \Psi_i} [c_{avg}(\pi^o)] \right] \\
& = \mathbb{E}_{\Psi_i} \left[\frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} \mathbb{E}_{\Phi \sim \Psi_i} [g_{avg}(\pi^o)] \right] - \mathbb{E}_{\Psi_i} \left[\frac{1}{k} \mathbb{E}_{\Phi \sim \Psi_i} [c_{avg}(\pi^o)] \right] \\
& = \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g_{avg}(\pi^o) - \frac{1}{k} c_{avg}(\pi^o)
\end{aligned}$$

The inequality is due to (3). ◀

We next present the first main theorem of this paper.

► **Theorem 7.** $g_{avg}(\pi^d) - c_{avg}(\pi^d) \geq \left(1 - \frac{1}{e}\right) g_{avg}(\pi^o) - c_{avg}(\pi^o)$.

Proof. According to the definition of G_k , we have $\mathbb{E}_{\Psi_k} [G_k(\Psi_k)] = \mathbb{E}_{\Psi_k} \left[\left(1 - \frac{1}{k}\right)^0 g(\Psi_k) - c(\text{dom}(\Psi_k)) \right] = g_{avg}(\pi^d) - c_{avg}(\pi^d)$ and $\mathbb{E}_{\Psi_0} [G_0(\Psi_0)] = \mathbb{E}_{\Psi_0} \left[\left(1 - \frac{1}{k}\right)^k g(\Phi(S_0)) - c(\text{dom}(\Psi_0)) \right] = 0$. Hence,

$$\begin{aligned}
& g_{avg}(\pi^d) - c_{avg}(\pi^d) = \mathbb{E}_{\Psi_k} [G_k(\Psi_k)] - \mathbb{E}_{\Psi_0} [G_0(\Psi_0)] \\
& = \mathbb{E}_{\Psi_{k-1}} [\mathbb{E}_{\Phi \sim \Psi_{k-1}} [G_k(\Psi_k)]] - \mathbb{E}_{\Psi_0} [\mathbb{E}_{\Phi \sim \Psi_0} [G_0(\Psi_0)]] \\
& = \sum_{i \in [k-1]} (\mathbb{E}_{\Psi_i} [\mathbb{E}_{\Phi \sim \Psi_i} [G_{i+1}(\Psi_{i+1})]] - \mathbb{E}_{\Psi_i} [\mathbb{E}_{\Phi \sim \Psi_i} [G_i(\Psi_i)]]) \\
& = \sum_{i \in [k-1]} \mathbb{E}_{\Psi_i} [\mathbb{E}_{\Phi \sim \Psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\Psi_i)]] \\
& \geq \sum_{i \in [k-1]} \left(\frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g_{avg}(\pi^o) - \frac{1}{k} c_{avg}(\pi^o) \right) \\
& = \sum_{i \in [k-1]} \left(\frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g_{avg}(\pi^o) \right) - c_{avg}(\pi^o) \\
& \geq \left(1 - \frac{1}{e}\right) g_{avg}(\pi^o) - c_{avg}(\pi^o)
\end{aligned}$$

The first inequality is due to Lemma 6. ◀

5 Monotone g : Linear-time Adaptive Distorted Greedy Policy

We next propose a faster algorithm *Linear-time Adaptive Distorted Greedy Policy*, denoted by π^l , for the case when g is adaptive monotone. As compared with π^d whose running time is $O(nk)$, our new policy π^l achieves nearly the same performance guarantee with $O(n \log \frac{1}{\epsilon})$ value oracle queries. Our design is inspired by the sampling technique developed in [10] for maximizing a monotone and submodular function. Very recently, [14] extends this approach to the adaptive setting to develop a linear-time adaptive policy for maximizing an adaptive submodular and adaptive monotone function. In this work, we apply this technique to design a linear-time adaptive policy for our adaptive regularized submodular maximization problem. Note that our objectives are not adaptive monotone and they may take negative values.

5.1 Design of π^l

We present the details of our algorithm in Algorithm 2. We first add a set D of $k-1$ dummy items to the ground set, such that, each dummy item $d \in D$ has zero cost, i.e., $\forall d \in D, c_d = 0$, and for any $d \in D$, and any partial realization ψ , we have $g(d | \psi) = 0$. Let $E' = E \cup D$. We next explain the idea of π^l : It starts with an empty set. In each iteration $i \in [k-1]$, π^l first samples a set R_i of size $\frac{n}{k} \log \frac{1}{\epsilon}$ uniformly at random, then adds an item e_i with the largest $H_i(\psi_i, \cdot)$ from R_i to the solution. After observing the state $\Phi(e_i)$ of e_i , we update the current partial realization ψ_{i+1} using $\psi_i \cup \{(e_i, \Phi(e_i))\}$ and enter the next iteration. This process iterates until all k items have been selected. It is worth noting that the technique of lazy updates [9] can be used to further accelerate the computation of our algorithms in practice.

■ **Algorithm 2** Linear-time Adaptive Distorted Greedy Policy π^l .

```

1:  $S_0 = \emptyset; i = 0; \psi_0 = \emptyset.$ 
2: while  $i < k$  do
3:    $R_i \leftarrow$  a random set sampled uniformly at random from  $E'$ ;
4:    $e_i \leftarrow \arg \max_{e \in R_i} H_i(\psi_i, e);$ 
5:    $S_{i+1} \leftarrow S_i \cup \{e_i\};$ 
6:    $\psi_{i+1} \leftarrow \psi_i \cup \{(e_i, \Phi(e_i))\}; i \leftarrow i + 1;$ 
7: return  $S_k$ 

```

5.2 Performance Analysis

We first present three preparatory lemmas.

► **Lemma 8.** *In each iteration of π^l ,*

$$\mathbb{E}_{\Phi \sim \psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\psi_i)] = \mathbb{E}_{e_i} [H_i(\psi_i, e_i)] + \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g(\psi_i)$$

The above lemma immediately follows from Lemma 4.

► **Lemma 9.** *In each iteration of π^l , $\mathbb{E}_{e_i} [H_i(\psi_i, e_i)] \geq \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} \mathbb{E}_{\Phi \sim \psi_i} [g_{avg}(\pi^o) - g_{avg}(\pi_i^l)] - \frac{1}{k} \mathbb{E}_{\Phi \sim \psi_i} [c_{avg}(\pi^o)].$*

Proof. Let A_e be an indicator that e is selected by the optimal solution π^o conditioned on a partial realization ψ_i . Let B_e be an indicator that e is selected by π^l in iteration i conditioned on a partial realization ψ_i . Let $M(\psi_i)$ denote the top k items with the largest marginal contribution to ψ_i in terms of $H_i(\psi_i, \cdot)$, i.e., $M(\psi_i) \leftarrow \arg \max_{S \subseteq E', |S|=k} \sum_{e \in S} H_i(\psi_i, e)$. Then we have

$$\begin{aligned} & \mathbb{E}_{e_i} [H_i(\psi_i, e_i)] \\ &= \sum_{e \in E'} \Pr[B_e = 1] \left(\left(1 - \frac{1}{k}\right)^{k-(i+1)} g(e | \psi_i) - c_e \right) \\ &\geq \Pr[R_i \cap M(\psi_i) \neq \emptyset] \frac{1}{k} \sum_{e \in M(\psi_i)} \left(\left(1 - \frac{1}{k}\right)^{k-(i+1)} g(e | \psi_i) - c_e \right) \\ &\geq (1 - \epsilon) \frac{1}{k} \sum_{e \in M(\psi_i)} \left(\left(1 - \frac{1}{k}\right)^{k-(i+1)} g(e | \psi_i) - c_e \right) \\ &\geq (1 - \epsilon) \frac{1}{k} \sum_{e \in E'} \Pr[A_e = 1] \left(\left(1 - \frac{1}{k}\right)^{k-(i+1)} g(e | \psi_i) - c_e \right) \\ &\geq (1 - \epsilon) \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} \mathbb{E}_{\Phi \sim \psi_i} [g_{avg}(\pi^o) - g_{avg}(\pi_i^l)] - (1 - \epsilon) \frac{1}{k} \mathbb{E}_{\Phi \sim \psi_i} [c_{avg}(\pi^o)] \end{aligned}$$

The second inequality is due to Lemma 4 in [14], where they show that $\Pr[R_i \cap M(\psi_i) \neq \emptyset] \geq 1 - \epsilon$ given that R_i has size of $\frac{n}{k} \log \frac{1}{\epsilon}$. The third inequality is due to $\sum_{e \in E'} \Pr[A_e = 1] \leq k$. The last inequality is due to g is adaptive submodular and c is modular. \blacktriangleleft

► **Lemma 10.** *In each iteration of π^l ,*

$$\begin{aligned} & \mathbb{E}_{\Psi_i} [\mathbb{E}_{\Phi \sim \Psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\Psi_i)]] \\ & \geq (1 - \epsilon) \frac{1}{k} (1 - \frac{1}{k})^{k-(i+1)} g_{avg}(\pi^o) - (1 - \epsilon) \frac{1}{k} c_{avg}(\pi^o) \end{aligned}$$

Proof. We first show that for any fixed partial realization ψ_i ,

$$\begin{aligned} & \mathbb{E}_{\Phi \sim \psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\psi_i)] \\ & \geq (1 - \epsilon) \left(\frac{1}{k} (1 - \frac{1}{k})^{k-(i+1)} \mathbb{E}_{\Phi \sim \psi_i} [g_{avg}(\pi^o)] - \frac{1}{k} \mathbb{E}_{\Phi \sim \psi_i} [c_{avg}(\pi^o)] \right) \end{aligned} \quad (4)$$

Due to Lemma 8, we have

$$\begin{aligned} & \mathbb{E}_{\Phi \sim \psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\psi_i)] \\ & = \mathbb{E}_{e_i} [H_i(\psi_i, e_i)] + \frac{1}{k} (1 - \frac{1}{k})^{k-(i+1)} g(\psi_i) \\ & \geq (1 - \epsilon) \frac{1}{k} (1 - \frac{1}{k})^{k-(i+1)} \mathbb{E}_{\Phi \sim \psi_i} [g_{avg}(\pi^o) - g_{avg}(\pi_i^l)] \\ & \quad - (1 - \epsilon) \frac{1}{k} \mathbb{E}_{\Phi \sim \psi_i} [c_{avg}(\pi^o)] + \frac{1}{k} (1 - \frac{1}{k})^{k-(i+1)} g(\psi_i) \\ & = (1 - \epsilon) \left(\frac{1}{k} (1 - \frac{1}{k})^{k-(i+1)} \mathbb{E}_{\Phi \sim \psi_i} [g_{avg}(\pi^o)] - \frac{1}{k} \mathbb{E}_{\Phi \sim \psi_i} [c_{avg}(\pi^o)] \right) \\ & \quad + \epsilon \frac{1}{k} (1 - \frac{1}{k})^{k-(i+1)} g(\psi_i) \\ & \geq (1 - \epsilon) \left(\frac{1}{k} (1 - \frac{1}{k})^{k-(i+1)} \mathbb{E}_{\Phi \sim \psi_i} [g_{avg}(\pi^o)] - \frac{1}{k} \mathbb{E}_{\Phi \sim \psi_i} [c_{avg}(\pi^o)] \right) \end{aligned}$$

The first inequality is due to Lemma 9, the second equality is due to $\mathbb{E}_{\Phi \sim \psi_i} [g_{avg}(\pi_i^l)] = g(\psi_i)$, and the last inequality is due to g is non-negative. Now we are ready to prove this lemma.

$$\begin{aligned} & \mathbb{E}_{\Psi_i} [\mathbb{E}_{\Phi \sim \Psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\Psi_i)]] \\ & \geq (1 - \epsilon) \mathbb{E}_{\Psi_i} \left[\frac{1}{k} (1 - \frac{1}{k})^{k-(i+1)} \mathbb{E}_{\Phi \sim \Psi_i} [g_{avg}(\pi^o)] - \frac{1}{k} \mathbb{E}_{\Phi \sim \Psi_i} [c_{avg}(\pi^o)] \right] \\ & = (1 - \epsilon) \mathbb{E}_{\Psi_i} \left[\frac{1}{k} (1 - \frac{1}{k})^{k-(i+1)} \mathbb{E}_{\Phi \sim \Psi_i} [g_{avg}(\pi^o)] \right] - \mathbb{E}_{\Psi_i} \left[\frac{1}{k} \mathbb{E}_{\Phi \sim \Psi_i} [c_{avg}(\pi^o)] \right] \\ & = (1 - \epsilon) \frac{1}{k} (1 - \frac{1}{k})^{k-(i+1)} g_{avg}(\pi^o) - (1 - \epsilon) \frac{1}{k} c_{avg}(\pi^o) \end{aligned}$$

The first inequality is due to (4). \blacktriangleleft

We next present the second main theorem of this paper.

► **Theorem 11.** $g_{avg}(\pi^l) - c_{avg}(\pi^l) \geq (1 - \frac{1}{e} - \epsilon) g_{avg}(\pi^o) - c_{avg}(\pi^o)$.

Proof. According to the definition of G_k , we have $\mathbb{E}_{\Psi_k} [G_k(\Psi_k)] = \mathbb{E}_{\Psi_k} [(1 - \frac{1}{k})^0 g(\Psi_k) - c(\text{dom}(\Psi_k))] = g_{avg}(\pi^l) - c_{avg}(\pi^l)$ and $\mathbb{E}_{\Psi_0} [G_0(\Psi_0)] = \mathbb{E}_{\Psi_0} [(1 - \frac{1}{k})^k g(\Phi(S_0)) - c(\text{dom}(\Psi_0))] = 0$. Hence,

$$\begin{aligned}
 & g_{avg}(\pi^l) - c_{avg}(\pi^l) \\
 &= \mathbb{E}_{\Psi_k}[G_k(\Psi_k)] - \mathbb{E}_{\Psi_0}[G_0(\Psi_0)] \\
 &= \mathbb{E}_{\Psi_{k-1}}[\mathbb{E}_{\Phi \sim \Psi_{k-1}}[G_k(\Psi_k)]] - \mathbb{E}_{\Psi_0}[\mathbb{E}_{\Phi \sim \Psi_0}[G_0(\Psi_0)]] \\
 &= \sum_{i \in [k-1]} (\mathbb{E}_{\Psi_i}[\mathbb{E}_{\Phi \sim \Psi_i}[G_{i+1}(\Psi_{i+1})]] - \mathbb{E}_{\Psi_i}[\mathbb{E}_{\Phi \sim \Psi_i}[G_i(\Psi_i)]]) \\
 &= \sum_{i \in [k-1]} \mathbb{E}_{\Psi_i}[\mathbb{E}_{\Phi \sim \Psi_i}[G_{i+1}(\Psi_{i+1}) - G_i(\Psi_i)]] \\
 &\geq \sum_{i \in [k-1]} \left((1-\epsilon) \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g_{avg}(\pi^o) - (1-\epsilon) \frac{1}{k} c_{avg}(\pi^o) \right) \\
 &= \sum_{i \in [k-1]} \left((1-\epsilon) \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g_{avg}(\pi^o) \right) - (1-\epsilon) c_{avg}(\pi^o) \\
 &\geq (1-\epsilon) \left(1 - \frac{1}{e}\right) g_{avg}(\pi^o) - (1-\epsilon) c_{avg}(\pi^o) \\
 &\geq \left(1 - \frac{1}{e} - \epsilon\right) g_{avg}(\pi^o) - c_{avg}(\pi^o)
 \end{aligned}$$

The first inequality is due to Lemma 10. ◀

6 Non-monotone g : Adaptive Random Distorted Greedy Policy

We next discuss the case when g is non-monotone adaptive submodular. We present an *Adaptive Random Distorted Greedy Policy* π^r for this case.

6.1 Design of π^r

The detailed implementation of π^r is listed in Algorithm 3. We first add a set D of $k-1$ dummy items to the ground set, such that, for any $d \in D$, and any partial realization ψ , we have $c_d = 0$ and $g(d \mid \psi) = 0$. Let $E' = E \cup D$. π^r runs round by round: Starting with an empty set. In each iteration $i \in [k-1]$, π^r randomly selects an item from the set $M(\psi_i)$. Recall that $M(\psi_i)$ is a set of k items that have the largest $H_i(\psi_i, \cdot)$, i.e.,

$$M(\psi_i) \leftarrow \arg \max_{S \subseteq E'; |S|=k} \sum_{e \in S} H_i(\psi_i, e)$$

After observing the state $\Phi(e_i)$ of e_i , we update the current partial realization ψ_{i+1} using $\psi_i \cup \{(e_i, \Phi(e_i))\}$ and enter the next iteration. This process iterates until all k items have been selected.

■ **Algorithm 3** Adaptive Random Distorted Greedy Policy π^r .

-
- 1: $S_0 = \emptyset; i = 0; \psi_0 = \emptyset$.
 - 2: **while** $i < k$ **do**
 - 3: $M(\psi_i) \leftarrow \arg \max_{S \subseteq E'; |S|=k} \sum_{e \in S} H_i(\psi_i, e)$;
 - 4: sample e_i uniformly at random from $M(\psi_i)$;
 - 5: $S_{i+1} \leftarrow S_i \cup \{e_i\}$;
 - 6: $\psi_{i+1} \leftarrow \psi_i \cup \{(e_i, \Phi(e_i))\}$; $i \leftarrow i + 1$;
 - 7: **return** S_k
-

6.2 Performance Analysis

We first present three preparatory lemmas. The first lemma immediately follows from Lemma 4.

► **Lemma 12.** *In each iteration of π^r ,*

$$\mathbb{E}_{\Phi \sim \psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\psi_i)] = \mathbb{E}_{e_i} [H_i(\psi_i, e_i)] + \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g(\psi_i)$$

► **Lemma 13.** *In each iteration of π^r , $\mathbb{E}_{e_i} [H_i(\psi_i, e_i)] \geq \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} \mathbb{E}_{\Phi \sim \psi_i} [g_{avg}(\pi^o @ \pi_i^r)] - g_{avg}(\pi_i^r)] - \frac{1}{k} \mathbb{E}_{\Phi \sim \psi_i} [c_{avg}(\pi^o)]$.*

Proof. Recall that A_e is an indicator that e is selected by the optimal solution π^o conditioned on a partial realization ψ_i ,

$$\begin{aligned} & \mathbb{E}_{e_i} [H_i(\psi_i, e_i)] \\ &= \frac{1}{k} \sum_{e \in M(\psi_i)} \left(\left(1 - \frac{1}{k}\right)^{k-(i+1)} g(e \mid \psi_i) - c_e \right) \\ &\geq \frac{1}{k} \sum_{e \in E'} \Pr[A_e = 1] \left(\left(1 - \frac{1}{k}\right)^{k-(i+1)} g(e \mid \psi_i) - c_e \right) \\ &\geq \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} \mathbb{E}_{\Phi \sim \psi_i} [g_{avg}(\pi^o @ \pi_i^r)] - g_{avg}(\pi_i^r)] - \frac{1}{k} \mathbb{E}_{\Phi \sim \psi_i} [c_{avg}(\pi^o)] \end{aligned}$$

The equality is due to the design of π^r , i.e., it selects an item e_i uniformly at random from $M(\psi_i)$. The first inequality is due to $\sum_{e \in E'} \Pr[A_e = 1] \leq k$ since π^o selects at most k items, and $M(\psi_i)$ contains a set of k items that have the largest $H_i(\psi_i, \cdot)$. The second inequality is due to g is adaptive submodular and c is modular. ◀

► **Lemma 14.** *In each iteration of π^r ,*

$$\mathbb{E}_{\Psi_i} [\mathbb{E}_{\Phi \sim \Psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\Psi_i)]] \geq \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-1} g_{avg}(\pi^o) - \frac{1}{k} c_{avg}(\pi^o)$$

Proof. We first show that for any fixed partial realization ψ_i ,

$$\mathbb{E}_{\Phi \sim \psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\psi_i)] \geq \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} \mathbb{E}_{\Phi \sim \psi_i} [g_{avg}(\pi^o @ \pi_i^r)] - \frac{1}{k} \mathbb{E}_{\Phi \sim \psi_i} [c_{avg}(\pi^o)] \quad (5)$$

Due to Lemma 12, we have

$$\begin{aligned} & \mathbb{E}_{\Phi \sim \psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\psi_i)] \\ &= \mathbb{E}_{e_i} [H_i(\psi_i, e_i)] + \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g(\psi_i) \\ &\geq \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} \mathbb{E}_{\Phi \sim \psi_i} [g_{avg}(\pi^o @ \pi_i^r)] - g_{avg}(\pi_i^r)] \\ &\quad - \frac{1}{k} \mathbb{E}_{\Phi \sim \psi_i} [c_{avg}(\pi^o)] + \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g(\psi_i) \\ &= \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} \mathbb{E}_{\Phi \sim \psi_i} [g_{avg}(\pi^o @ \pi_i^r)] - \frac{1}{k} \mathbb{E}_{\Phi \sim \psi_i} [c_{avg}(\pi^o)] \end{aligned}$$

69:12 Adaptive Regularized Submodular Maximization

The first inequality is due to Lemma 13. The second equality is due to $\mathbb{E}_{\Phi \sim \psi_i} [g_{avg}(\pi_i^r)] = g(\psi_i)$. The last inequality is due to g is non-negative. Now we are ready to prove this lemma.

$$\begin{aligned}
& \mathbb{E}_{\Psi_i} [\mathbb{E}_{\Phi \sim \Psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\Psi_i)]] \\
& \geq \mathbb{E}_{\Psi_i} \left[\frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} \mathbb{E}_{\Phi \sim \Psi_i} [g_{avg}(\pi^o @ \pi_i^r)] - \frac{1}{k} \mathbb{E}_{\Phi \sim \Psi_i} [c_{avg}(\pi^o)] \right] \\
& = \mathbb{E}_{\Psi_i} \left[\frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} \mathbb{E}_{\Phi \sim \Psi_i} [g_{avg}(\pi^o @ \pi_i^r)] \right] - \mathbb{E}_{\Psi_i} \left[\frac{1}{k} \mathbb{E}_{\Phi \sim \Psi_i} [c_{avg}(\pi^o)] \right] \\
& = \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} g_{avg}(\pi^o @ \pi_i^r) - \frac{1}{k} c_{avg}(\pi^o) \\
& \geq \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-(i+1)} \left(1 - \frac{1}{k}\right)^i g_{avg}(\pi^o) - \frac{1}{k} c_{avg}(\pi^o) \\
& = \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-1} g_{avg}(\pi^o) - \frac{1}{k} c_{avg}(\pi^o)
\end{aligned}$$

The first inequality is due to (5), and the second inequality is due to Lemma 1 in [14], where they show that $g_{avg}(\pi^o @ \pi_i^r) \geq \left(1 - \frac{1}{k}\right)^i g_{avg}(\pi^o)$. ◀

We next present the third main theorem of this paper.

► **Theorem 15.** $g_{avg}(\pi^r) - c_{avg}(\pi^r) \geq \frac{1}{e} g_{avg}(\pi^o) - c_{avg}(\pi^o)$.

Proof. According to the definition of G_k , we have $\mathbb{E}_{\Psi_k} [G_k(\Psi_k)] = \mathbb{E}_{\Psi_k} [(1 - \frac{1}{k})^0 g(\Psi_k) - c(\text{dom}(\Psi_k))] = g_{avg}(\pi^r) - c_{avg}(\pi^r)$ and $\mathbb{E}_{\Psi_0} [G_0(\Psi_0)] = \mathbb{E}_{\Psi_0} [(1 - \frac{1}{k})^k g(\Phi(S_0)) - c(\text{dom}(\Psi_0))] = 0$. Hence,

$$\begin{aligned}
g_{avg}(\pi^r) - c_{avg}(\pi^r) &= \mathbb{E}_{\Psi_k} [G_k(\Psi_k)] - \mathbb{E}_{\Psi_0} [G_0(\Psi_0)] \\
&= \mathbb{E}_{\Psi_{k-1}} [\mathbb{E}_{\Phi \sim \Psi_{k-1}} [G_k(\Psi_k)]] - \mathbb{E}_{\Psi_0} [\mathbb{E}_{\Phi \sim \Psi_0} [G_0(\Psi_0)]] \\
&= \sum_{i \in [k-1]} (\mathbb{E}_{\Psi_i} [\mathbb{E}_{\Phi \sim \Psi_i} [G_{i+1}(\Psi_{i+1})]] - \mathbb{E}_{\Psi_i} [\mathbb{E}_{\Phi \sim \Psi_i} [G_i(\Psi_i)]]) \\
&= \sum_{i \in [k-1]} \mathbb{E}_{\Psi_i} [\mathbb{E}_{\Phi \sim \Psi_i} [G_{i+1}(\Psi_{i+1}) - G_i(\Psi_i)]] \\
&\geq \sum_{i \in [k-1]} \left(\frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-1} g_{avg}(\pi^o) - \frac{1}{k} c_{avg}(\pi^o) \right) \\
&= \sum_{i \in [k-1]} \frac{1}{k} \left(1 - \frac{1}{k}\right)^{k-1} g_{avg}(\pi^o) - c_{avg}(\pi^o) \\
&\geq \frac{1}{e} g_{avg}(\pi^o) - c_{avg}(\pi^o)
\end{aligned}$$

The first inequality is due to Lemma 14. ◀

7 Conclusion

In this paper, we study the adaptive regularized submodular maximization problem. Because our objective function may take both negative and positive values, most existing technologies of submodular maximization do not apply to our setting. We develop a series of effective policies for this problem.

References

- 1 Niv Buchbinder, Moran Feldman, Joseph Naor, and Roy Schwartz. Submodular maximization with cardinality constraints. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 1433–1452. SIAM, 2014.
- 2 Moran Feldman. Guess free maximization of submodular and linear sums. *Algorithmica*, pages 1–26, 2020.
- 3 Daniel Golovin and Andreas Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *Journal of Artificial Intelligence Research*, 42:427–486, 2011.
- 4 Chris Harshaw, Moran Feldman, Justin Ward, and Amin Karbasi. Submodular maximization beyond non-negativity: Guarantees, fast algorithms, and applications. In *International Conference on Machine Learning*, pages 2634–2643. PMLR, 2019.
- 5 Tianyuan Jin, Yu Yang, Renchi Yang, Jieming Shi, Keke Huang, and Xiaokui Xiao. Unconstrained submodular maximization with modular costs: Tight approximation and application to profit maximization. *Proceedings of the VLDB Endowment*, 14(10), 2021.
- 6 Ehsan Kazemi, Shervin Minaee, Moran Feldman, and Amin Karbasi. Regularized submodular maximization at scale. In *International Conference on Machine Learning*, pages 5356–5366. PMLR, 2021.
- 7 Andreas Krause and Carlos Guestrin. Near-optimal observation selection using submodular functions. In *AAAI*, volume 7, pages 1650–1654, 2007.
- 8 Hui Lin and Jeff Bilmes. A class of submodular functions for document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 510–520, 2011.
- 9 Michel Minoux. Accelerated greedy algorithms for maximizing submodular set functions. In *Optimization techniques*, pages 234–243. Springer, 1978.
- 10 Baharan Mirzasoleiman, Ashwinkumar Badanidiyuru, Amin Karbasi, Jan Vondrák, and Andreas Krause. Lazier than lazy greedy. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.
- 11 George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions-i. *Mathematical programming*, 14(1):265–294, 1978.
- 12 Sofia Maria Nikolakaki, Alina Ene, and Evimaria Terzi. An efficient framework for balancing submodularity and cost. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 1256–1266, 2021.
- 13 Maxim Sviridenko, Jan Vondrák, and Justin Ward. Optimal approximation for submodular and supermodular optimization with bounded curvature. *Mathematics of Operations Research*, 42(4):1197–1218, 2017.
- 14 Shaojie Tang. Beyond pointwise submodularity: Non-monotone adaptive submodular maximization in linear time. *Theoretical Computer Science*, 850:249–261, 2021.
- 15 Shaojie Tang. Beyond pointwise submodularity: Non-monotone adaptive submodular maximization subject to knapsack and k -system constraints. In *International Conference on Modelling, Computation and Optimization in Information Systems and Management Sciences*. Springer, 2021.
- 16 Shaojie Tang and Jing Yuan. Influence maximization with partial feedback. *Operations Research Letters*, 48(1):24–28, 2020.
- 17 Jing Yuan and Shaojie Tang. No time to observe: adaptive influence maximization with partial feedback. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 3908–3914, 2017.