Stochastic Submodular Maximization

Arash Asadpour, Hamid Nazerzadeh, and Amin Saberi

Stanford University, Stanford, CA.
{asadpour,hamidnz,saberi}@stanford.edu

Abstract. We study stochastic submodular maximization problem with respect to a cardinality constraint. Our model can capture the effect of uncertainty in different problems, such as cascade effects in social networks, capital budgeting, sensor placement, etc. We study non-adaptive and adaptive policies and give optimal constant approximation algorithms for both cases. We also bound the adaptivity gap of the problem between 1.21 and 1.59.

1 Introduction

The problem of maximizing submodular functions with respect to known constraints is a very well-studied problem in operations research and computer science. A function $f : \mathbb{R}^n \to \mathbb{R}$ is submodular if for all $x, y \in \mathbb{R}^n$:

$$f(x \vee y) + f(x \wedge y) \leq f(x) + f(y)$$

where $x \vee y$ and $x \wedge y$ denote the component-wise maximum and the component-wise minimum of x and y, respectively. If f is twice differentiable, then submodularity is equivalent to the condition $\frac{\partial^2 f}{\partial x_i \partial x_j} \leq 0$, where x_i and x_j are any two coordinates of x [17]. One may imagine the function f on the domain of 0/1-vectors as a set function where $f(S) = f(x)$, $x_i = 1$ whenever $i \in S$, and $x_i = 0$ otherwise. In other words, a set function $f : 2^N \to \mathbb{R}$ is submodular if for any two subsets $S, T \subseteq N$:

$$f(S \cup T) + f(S \cap T) \leq f(S) + f(T)$$

A wide range of optimization problems that arise in the real world can be modeled as maximizing submodular functions with respect to some (usually cardinality) constraints. One instance is the problem of viral marketing and maximizing influence through the network [9, 14], where the goal is to choose an initial “active” set of people, so as to maximize the spread of an innovation or behavior in a social network. It is well-known that under many models of influence propagation in networks (e.g. decreasing cascade model [9]), the expected size of the final cascade is a submodular function of the set of initially activated people. Also, due to some budget limitations the number of people that we can activate in the beginning is bounded. Hence, the maximizing influence problem can be seen as a maximizing submodular function problem subject to some cardinality constraint.

Another example is the capital budgeting problem which is to find the optimal investment of capital among different projects with a limited budget. There is a set of projects and one wants to invest on a group of them that maximizes his expected profit while not exceeding his budget. This problem has been studied extensively under various assumptions on the utility of the investor and dependencies among the projects [20, 21, 13, 2]. Naturally, the utility functions are non-negative and monotone. Also, the risk-averse investors are characterized by their submodular utility functions. Therefore, such investors need to solve a submodular maximization problem to find their best bet.

Yet another example is the problem of optimal placement of sensors for environmental monitoring [11, 12] where the objective is to place a limited number of sensors in the environment in order to most effectively reduce uncertainty in observations. It is known that the efficiency of a subset of sensors is a submodular set function.

For the problem of maximizing submodular set functions subject to a cardinality constraint, the celebrated result of Cornuejols et al. [4] and Nemhauser et al. [15] shows that for nonnegative monotone submodular functions the greedy algorithm that at each step chooses an element with the maximum marginal value gives a $(1 - \frac{1}{e} \approx 0.632)$-approximation of the optimal solution. This problem has also been studied for more complicated domains. In particular, for maximizing a submodular function over a matroid (note that the cardinality constraint is a special case of this) a recent result by Vondrak [19] shows that it is still possible to get a $1 - \frac{1}{e}$-approximation.

However, in practice one must deal with the stochasticity caused by the uncertain nature of the problem, the incomplete information about the environment, etc. For instance, in viral marketing some people in the initial set might not adopt the behavior. Another example is the capital budgeting problem where some projects taken by an investor may fail (in the beginning) due to some unexpected events in the market. Also, in environmental monitoring some sensors might not work properly because of bad weather or inconsistent connections.

All these possibilities motivate the problem of stochastic submodular maximization. In the stochastic setting, the outcome of the elements in the selected set are not known in advance; when they are picked, with some known probability they might remain in the set or not. One may think of probability p_i corresponding to each element i, and then the expected value of selecting a set S will be the expected value of function f over set $\hat{S}$ derived from S by removing each element i independently with probability $1 - p_i$. In fact, in this paper we will consider a more general version of this problem in which the stochasticity of the problem converts the set S into a vector in the continuous space via some known probability distributions. For the exact definitions see Section 2.

The main difference between the non-stochastic and stochastic problems is that the latter can benefit from adaptivity. An *adaptive* policy can use the outcome of the steps taken so far to optimize the decisions it is going to make in the future. On the other hand, the actions chosen by a *non-adaptive* policy are independent of the outcome of the other actions. Therefore, a non-adaptive policy is equivalent to a predetermined subset of elements. Although non-adaptive policies may not perform as well as adaptive ones, they are particularly useful when it is difficult or time-consuming to discover the outcome of an action.

Our Results: In Section 3 we first show that the expected value of a submodular set function in the stochastic setting is still a submodular set function. This immediately leads to a $(1-\frac{1}{e})$-approximation of the optimal non-adaptive policy. Then, we consider the adaptive policy that at each step chooses an element with the maximum *expected* marginal value, conditioned on the outcome of the previous elements. We show that the approximation ratio of this greedy policy with respect to the optimal adaptive policy is $1-\frac{1}{e}$. [1] We also give a lower bound and an upper bound on the adaptivity gap of the problem. The *adaptivity gap* is defined as the maximum ratio between the expected value of the optimal adaptive and non-adaptive policies [5]. As a lower bound, we prove that the adaptivity gap of stochastic submodular maximization problem is at least 1.21 (see Section 2.1). On the flip side, in Section 4 we show that the adaptivity gap is bounded from above by $\frac{e}{e-1} \approx 1.59$, i.e. there exists a non-adaptive policy which achieves at least $\frac{1}{1.59}$ fraction of the value of best adaptive policy. We also show that a non-adaptive policy within a factor of $(\frac{e-1}{e})^2 \approx \frac{1}{2.51}$ of the optimum adaptive policy can be found in polynomial time. In order to prove this bound, we generalize some of the techniques developed by Vondrák [18]. These extensions could be of independent interest.

1.1 Related Work

We first briefly overview some parts of the literature on (non-stochastic) submodular optimization. Then, we explain some of the works that study stochastic settings similar to ours.

Cornuejols et al. [4] proved that a simple greedy algorithm gives a $(1-\frac{1}{e})$-approximation for the problem of maximizing monotone submodular set functions subject to capacity constraints. Later, Feige [6] proved that it is not possible to improve this ratio unless $NP \subset TIME(n^{O(\log\log n)})$. For maximizing non-monotone submodular, recently Feige et al. [7] gave a constant approximation algorithm. Another well-studied submodular maximization problem is the problem of allocating resources to agents with submodular utilities, for which several interesting approximation algorithms have been developed, see [18]. In this paper, we use some of these techniques to bound the adaptivity gap.

Goemans and Vondrák [8] consider the problem of stochastic covering. In this problem the goal is to cover all elements of a target set using minimum number of subsets. The subsets are random variables and their probability distributions are given. They propose adaptive and non-adaptive policies for the problem. They also observe that the adaptivity gap is not constant.

Chan and Farias [3] study a generalization of the stochastic maximum k-cover problem where the sequence of elements arrive according to a stochastic

[1] This is also independently observed by Chan and Farias [3].

process and utility functions may vary over time. They show that under some conditions, a myopic policy is a 2-approximation of the optimal adaptive policy.

In a recent work, Streeter and Golovin [16] study an online job scheduling problem in a setting that the cost of jobs is given by a submodular function. The goal is to cover as many jobs as possible subject to a budget constraints. They take the regret minimization approach and present approximate optimal policy.

2 Problem Definition

We define the following abstraction for the stochastic submodular maximization problem. A set $\mathcal{A} = \{X_1, \cdots, X_n\}$ of independent random variables is given. After choosing X_i, its actual value (outcome of an element), denoted by x_i, is discovered. We assume that $x_i \in [0,1]$. Let $S \subseteq \mathcal{A}$ be a subset of variables. Also, let vector $s = < \hat{x}_1, \cdots, \hat{x}_n >$ denote the *realization* of set S, where $\hat{x}_i = x_i$ for $i \in S$ and $\hat{x}_i = 0$ for $i \notin S$. The value obtained by choosing the set S after the realization is equal to $f(s)$, where $f : [0,1]^N \to \mathbb{R}^+$ is a submodular function.

Let g_i be the probability distribution of random variable X_i. For every subset $S \subset \mathcal{A}$, it defines a probability measure $g_S : [0,1]^n \to \mathbb{R}$, which represents the probability density function of observing s while selecting S:

$$g_S(ds) = \int_{x \in ds} \prod_{i \in S} g_i(dx_i)$$

Also, $g_S(ds)$ is defined to be zero if there exist i such that $s_i \neq 0, i \notin S$. Now define function $F_g : [0,1]^n \to \mathbb{R}^+$ as the expected value obtained by choosing set S, i.e.,

$$F_g(S) = \int_{s \in [0,1]^n} f(s) g_S(ds) \tag{1}$$

Our goal is to choose a set S of size at most k which maximizes $F_g(S)$.

$$\max_{S \subset \mathcal{A}: |S| \leq k} F_g(S).$$

For simplicity, we assume one cannot choose an element of $\mathcal{A}$ multiple times. [2] For this problem, we study two types of policies: adaptive and non-adaptive. A *non-adaptive policy* is represented by a fixed subset of $\mathcal{A}$. An *adaptive policy* uses the realized value of the previously chosen elements to determine the next element in the subset. In order to compare the value of these optimal policies, we study the adaptivity gap of the problem. The *adaptivity gap* is defined as the ratio between the expected value of optimal adaptive and non-adaptive policies.

[2] This assumption is not necessary for our results, and is made for sake of simplicity. One can create k independent copies of each random variable to simulate multiplicity. This is in contrast with the set cover problem where allowing to chose multiple copies of an element significantly reduces the adaptivity gap [8].

One issue that arises in the algorithmic discussions of this paper and many of the related works is computing the value of functions similar to F_g. We assume that we are given an oracle which computes these values up to a desired degree of accuracy. In fact, it can be shown that in many interesting cases such an oracle can be built efficiently. Two most important cases are when the probability distribution functions (g_j's) are constant Lipschitz continuous, or when their support is a polynomial size set of discrete values. Therefore, from now on, all of our results will involve an arbitrary small error term of ϵ that we will not mention explicitly.

In the next section, we illustrate the problem by giving an example. We also present a non-adaptive and an adaptive policy for this example.

2.1 An Example: Stochastic Maximum k-Cover

A special case of stochastic submodular maximization is the stochastic maximum k-cover problem. Given a collection $\mathcal{F}$ of subsets of $\{1, 2, \cdots, n\}$, the max k-cover problem is defined as finding k subsets from $\mathcal{F}$ such that their union has the maximum cardinality [6]. In the stochastic version, the subset that an element of $\mathcal{F}$ would cover becomes known after choosing the element. In this section, we define an instance of this problem. We also use this example to give a lower bound on the adaptivity gap.

The instance we consider in this section is as follows: A ground set $G = \{1, 2, \cdots, 2n\}$ and a collection $F = \{C_1, C_2, \cdots, C_{2n}\}$ of its subsets are given. For $1 \leq i \leq n$, $C_i = \{1, 2, \cdots, n\}$ with probability $\frac{1}{n}$ and is the empty set with probability $1 - \frac{1}{n}$. For $n + 1 \leq i \leq 2n$, $C_i = \{i\}$ with probability $\frac{1}{e}$ and is the empty set with the remaining probability. The goal is to cover the maximum number of elements in G by selecting at most n subsets in C.

Lemma 1. *For large enough values of n, the optimal non-adaptive policy is to select $S = \{C_1, C_2, \cdots, C_n\}$. Also, the expected value of this policy is $n(1 - \frac{1}{e})$.*

Proof. Consider a subset S' selected by a non-adaptive policy. Let q the fraction of elements of S' that are in $\{C_1, C_2, \cdots, C_n\}$, i.e., $q = |S' \cap \{C_1, C_2, \cdots, C_n\}|/n$. Such a policy covers the elements of $\{1, 2, \cdots, n\}$ with probability $1 - (1 - \frac{1}{n})^{nq}$. Also, in expectation, S' covers at most $\frac{n}{e}(1-q)$ elements from set $\{n+1, \cdots, 2n\}$. Therefore, the expected number of covered elements is

$$n(1 - (1 - \frac{1}{n})^{nq}) + \frac{n}{e}(1 - q)$$

We can approximate the expression above by $n(1 + \frac{1}{e} - (\frac{1}{e})^q - \frac{q}{e})$ with arbitrary high precision for large enough n. This expression is increasing in q. Therefore, for the optimum non-adaptive policy we have $q = 1$ or equivalently $S = \{C_1, C_2, \cdots, C_n\}$.

Now consider the following adaptive policy that at each step chooses the elements with maximum marginal value: At step i, $1 \leq i \leq n$, choose set C_i until one of these sets covers $\{1, \cdots, n\}$. After that, pick a set from $\{C_{n+1}, \cdots, C_{2n}\}$ until the number of chosen sets reaches n.

The following lemma gives a lower bound on the number of elements covered by the adaptive policy.

Lemma 2. *For large enough n, the expected number of elements covered by the adaptive policy describe above is $n(1 - \frac{1}{e} + \frac{1}{e^2})$.*

Proof. The probability that C_i covers the first n elements is $\frac{1}{n}(1 - \frac{1}{n})^{i-1}$. If C_i covers the first n elements, the policy will choose $n-i$ subsets from $C_{n+1}, \cdots, C_{2n}$, each covers a single element with probability $\frac{1}{e}$. Therefore, the expected number of covered elements is:

$$\begin{aligned}
\sum_{i=1}^{n}[\frac{1}{n}(1-\frac{1}{n})^{i-1} \times (n+(n-i)\frac{1}{e})] = (1+\frac{1}{e})\sum_{i=1}^{n}(1-\frac{1}{n})^{i-1} - \frac{1}{en}\sum_{i=1}^{n} i(1-\frac{1}{n})^{i-1} \\
\approx\ (1+\frac{1}{e})n(1-\frac{1}{e}) - \frac{1}{en}\left[n^2(1-(1-\frac{1}{n})^n) - n^2(1-\frac{1}{n})^n\right] \\
\approx\ n\left[(1-\frac{1}{e^2}) - \frac{1}{e}(1-\frac{2}{e})\right] \\
=\ n(1-\frac{1}{e}+\frac{1}{e^2}).
\end{aligned}$$

which completes the proof of the lemma.

By combining the results of Lemmas 1 and 2 we have the corollary below.

Corollary 3. *The adaptivity gap of stochastic maximum k-cover is at least:*

$$\tfrac{1-e^{-1}+e^{-2}}{1-e^{-1}} > 1.21$$

3 Near-Optimal Non-Adaptive and Adaptive Policies

In this section we first present non-adaptive policy for the stochastic submodular maximization problem. Later, we give an adaptive policy. A non-adaptive policy is represented by a fixed subset $S \subseteq \mathcal{A}$. The expected value of the policy is equal to $F_g(S)$. Therefore, finding the optimal non-adaptive policy is equivalent to finding set S which maximizes $F_g(S)$. Note that the maximum k-cover problem is a special case of our problem. Therefore, it is not possible to find an approximation ratio better than $1 - \frac{1}{e}$ for the optimal adaptive policy unless $NP \subset TIME(n^{O(\log\log n)})$ [6]. In this section, we show that there exists a policy that is implementable in polynomial time and its value is within a $1 - \frac{1}{e}$ ratio of the optimal non-adaptive policy. For the ease of notation, when it is clear from the context, we use $F(S)$ instead of $F_g(S)$. Note that $F(S)$ is a convex combination of a set of monotone submodular functions. Therefore, we have the following lemma.

Lemma 4. *The function $F(S)$ is monotone and submodular in S.*

Submodularity of F, immediately leads to the following result [4, 15].

Corollary 5. *Consider the non-adaptive greedy policy that at each step chooses the element with maximum marginal increase in value. The approximation ratio of this policy with respect to the optimal non-adaptive policy is at least $1-\frac{1}{e}$.*

Now we present an adaptive greedy policy with approximation ratio $1-\frac{1}{e}$, with respect to the optimal adaptive policy. It is easy to see that finding maximum k-cover can be reduced to designing an adaptive policy. Therefore, it is not possible to improve this ratio unless $NP \subset TIME(n^{O(\log\log n)})$.

Theorem 6. *Consider the adaptive greedy policy that at each step selects an element with the maximum marginal value, conditioned on the realized value of the previously chosen elements. The approximation ratio of the adaptive greedy policy with respect to the optimal adaptive policy is $1-\frac{1}{e}$.*

Before stating the proof, we describe some notations. For $1 \leq i \leq k$, let S_i be the set of elements chosen by the greedy adaptive policy up to (and including) step i . Define S_0 to be the empty set. Also, let s_i denote the realization of S_i. The adaptive greedy policy at each step i chooses an element in

$$\text{argmax}_{j\in\mathcal{A}\setminus S_{i-1}} E[F(S_{i-1}\cup j)|s_{i-1}]$$

Proof. The proof presented here is similar to the proof of Kleinberg et al.[10] for submodular set functions. Let T_j be the set chosen by the optimal adaptive policy up to step j. Also, denote the expected marginal value of the i^{th} element chosen by the greedy policy by Δ_i, i.e.,

$$\Delta_i = E[F(S_i)|s_{i-1}] - f(s_{i-1}) = E[F(S_i) - F(S_{i-1})|s_{i-1}]$$

Consider a realization s_i of S_i. Because the realization of each element of T_j is independent from other elements, and f is submodular, we can write $F(T_j\cup S_i|s_i)$ as the sum of a set of monotone submodular functions. Therefore, $F(T_j\cup S_i|s_i)$ is monotone submodular with respect to j. Hence, for $T=T_k$ we have:

$$E[F(T)|s_i] \leq E[F(T\cup S_i)|s_i] \leq E[F(S_i) + k(F(T_1\cup S_i) - F(S_i))|s_i]$$

Because $\Delta_i \geq E[F(T_1\cup S_i) - F(S_i)|s_i]$ we get,

$$E[F(T)|s_i] \leq E[F(S_i) + k\Delta_{i+1}|s_i]$$

Since the inequality above holds for every history, adding up all such inequalities, for all i, $0\leq i\leq k-1$, we have:

$$\begin{aligned} E[F(T)] &\leq E[F(S_i)] + kE[\Delta_{i+1}] \\ &= E[\Delta_1 + \cdots + \Delta_i] + kE[\Delta_{i+1}] \end{aligned}$$

We multiply the i^{th} inequality, $0 \leq i \leq k-1$, by $(1-\frac{1}{k})^{k-1-i}$, and add them all up. The sum of the coefficients of $E[F(T)]$ is equal to

$$\sum_{i=0}^{k-1}(1-\frac{1}{k})^{k-1-i} = \sum_{i=0}^{k-1}(1-\frac{1}{k})^{i} = \frac{1-(1-\frac{1}{k})^k}{1-(1-\frac{1}{k})} = k(1-(1-\frac{1}{k})^k) \tag{2}$$

One the right hand side, the sum of the coefficient of $E[\Delta_i]$, $1 \leq i \leq k$, is equal to

$$\begin{aligned} k(1-\frac{1}{k})^{k-i} + \sum_{j=i}^{k-1}(1-\frac{1}{k})^{k-1-j} &= k(1-\frac{1}{k})^{k-i} + \sum_{j=0}^{k-i-1}(1-\frac{1}{k})^{j} \\ &= k(1-\frac{1}{k})^{k-i} + k(1-(1-\frac{1}{k})^{k-i}) \\ &= k \end{aligned} \tag{3}$$

Therefore, by inequalities (2) and (3) we get

$$E[F(T)] \leq (1-(1-\frac{1}{k})^k)\sum_{i=1}^{k} E[\Delta_i] = (1-(1-\frac{1}{k})^k)E[F(S_k)]$$

Therefore, the approximation ratio of the greedy policy is at least $1-\frac{1}{e}$.

It is easy to see from the proof above that if at every step, a policy chooses an element which is an α approximation of the maximum marginal value, then it achieves approximation ratio $1-(\frac{1}{e})^{\alpha}$.

4 Adaptivity Gap: An Upper Bound of 1.59

Our concern in this section will be to set an upper bound for the adaptivity gap. In other words, we want to have a lower bound on the approximation ratio of non-adaptive policies against the best adaptive policy. We establish such a bound through the following theorem:

Theorem 7. *There exists a non-adaptive policy that approximates the optimal adaptive policy within a factor of* $\frac{e-1}{e} \approx \frac{1}{1.59}$. *Moreover, There exists a polynomial time non-adaptive policy with the approximation ratio at least* $(\frac{e-1}{e})^2 \approx \frac{1}{2.51}$.

The proof of the above theorem is inspired by the techniques in Section 3.5 of [18]. For the sake of consistency, we will use the same notation as [18] wherever possible. We generalize these techniques by extending the domain of the function F_g to real vectors. We will define a function f^+ which sets an upper bound on the performance of all adaptive policies and also lies within a constant factor (at most $\frac{e}{e-1}$) of the maximum value of F_g. As we will see, this implies that for every adaptive policy Adapt there exists a non-adaptive policy which gains at least a fraction of $\frac{e-1}{e}$ of the expected value gained by ADAPT. Also, Corollary 5 shows that the greedy non-adaptive policy approximates the optimal non-adaptive by a factor of $\frac{e-1}{e}$. Hence, it will be within a factor of $(\frac{e-1}{e})^2$ of the best adaptive policy.

Here comes our basic observation about adaptive policies. Consider an arbitrary adaptive policy ADAPT. Any such policy decides to choose a sequence of elements, where the decision about which element to choose at any step might depend on the realized values of the previously chosen elements. Therefore, for any realization of outcomes (i.e. realized values of elements) a distribution on the sequence of elements will be implied by ADAPT[3]. This distribution corresponds to what ADAPT does if it observes that specific realization. Any adaptive policy can be described by a (possibly randomized) decision tree in which at each step an element is being added to the current selection. Due to the constraints of the problem, the height of the tree is k. Each path from the root to a leaf of this tree corresponds to a subset with k elements and occurs with some certain probability. Clearly, these probabilities sum up to one. Let y_i be the probability that element i is chosen by ADAPT. Also, let β_s be the probability density function for the outcome s. Then, we have the following properties:

1. $\displaystyle\int_s \beta_s = 1$.
2. $\forall s : \beta_s \geq 0$.
3. $\displaystyle\forall i, dx_i : \int_{s, s_i \in dx_i} \beta_s ds = y_i g_i(x_i) dx_i$.

The first two properties hold because β defines a probability measure on the space of all outcomes. The third property is due to the fact that the left hand side is in fact computing the probability that the element i is chosen and its observed value is x_i.

Now, we are ready to define the function $f^+ : [0,1]^n \rightarrow \mathbb{R}$ which establishes an upper bound on the performance of any adaptive policy. The definition of f^+ is motivated by the above observation about all the possible outcomes of an arbitrary adaptive policy. It can be seen as the generalization of function f^+ in [18] to the continuous domain. For any vector $y \in [0,1]^n$ we define $f^+(y)$ as follows:

$$\sup_\alpha \left\{ \int_s \alpha_s f(s) \right\}$$

where the supremum is taken over all probability measures α defined on $[0,1]^n$ for which

$$\forall i, dx_i : \int_{s, s_i \in dx_i} \alpha_s ds = y_i g_i(x_i) dx_i.$$

Now, we can bound the performance of ADAPT using function f^+. Consider all possible realizations of elements under ADAPT. Let y_i and β_s be defined as

[3] The reason that we mentioned a distribution (and not just a specific sequence) is that ADAPT may be a randomized policy by itself. But as we will see, it does not affect our arguments.

before. Then, the expected value of Adapt is $\int_s \beta_s f(s)$. On the other hand, by the construction of β, it is one of the possibilities that can be used as α in the "sup" term in the definition of $f^+(y)$. Therefore, the performance of the policy is bounded by $f^+(y)$ and we have the following lemma.

Lemma 8. *The expected value of the adaptive policy* Adapt *is at most* $f^+(y)$.

Now, let P_k be the matroid polytope $\{v : 1 \cdot v \leq k, v \geq 0\}$. Any valid policy is limited to select at most k elements, i.e., $y \in P_k$. Therefore, the desired upper bound on the optimal adaptive policy can be obtained as a corollary of Lemma 8.

Corollary 9. *The expected value of the optimal adaptive policy is bounded from above by* $\max_{y \in P_k}\{f^+(y)\}$.

We also define the extension $F_g : [0,1]^n \to \mathbb{R}$. For any vector $y \in [0,1]^n$, $F_g(y)$ is the expected value of its realized outcome $\hat{y}$ when y_i is set to be 1 with probability y_i and 0 otherwise. More formally, if $N = \{1, 2, \cdots, n\}$ then

$$F_g(y) = \mathrm{E}[F_g(\hat{y})] = \sum_{R \in N} \prod_{i \in R} y_i \prod_{i \notin R} (1 - y_i) F_g(R).$$

To complete the proof of Theorem 7 we need to prove that there exists a 0/1 vector $w \in P_k$ such that the value of $F_g(w)$ is a good approximation of the optimum value of f^+. We will do that in two steps. First, we show that for any vector y, the values of $F_g(y)$ and $f^+(y)$ are within a constant of each other. Then, in Lemma 11 we will show that there exists a proper 0/1 vector w so that $F_g(w) \geq F_g(y)$. Remember that in fact $F_g(w) = F_g(S)$ for the subset S corresponding to entries equal to 1 in w. It proves that the ratio of the best non-adaptive and adaptive policy is at least $1 - \frac{1}{e}$. For the second part of the result we show that a 0/1 vector w' such that $F_g(w') \geq (1 - \frac{1}{e})F_g(w)$ can be found in polynomial time which provides an efficient way to find a non-adaptive policy within a factor $(1 - \frac{1}{e})^2$ of the optimal adaptive policy.

The following lemma proves that the defined extension f^+ cannot be too far from F.

Lemma 10. *For any monotone submodular function f and any vector y, $F_g(y) \geq (1 - \frac{1}{e})f^+(y)$.*

Proof. The proof might be viewed as of a generalization of the proof of Lemmas 3.7 and 3.8 in [18] to the continuous space. We define an auxiliary function $f^* : [0,1]^n \to \mathbb{R}$ as the following:

$$f^*(y) = \inf_z \{f(z) + \sum_{j \in N} \int_{s_j > z_j} g_j(s)(f(z_s(j)) - f(z)) ds_j,$$

where $z_s(j)$ is the vector z with its j-th entry changed to s_j whenever $s_j > z_j$.

We prove that for any vector y, $F_g(y) \le f^+(y) \le f^*(y)$. The first inequality follows directly from the definition of F_g and f^+. To prove the second inequality, note that for any feasible measure α and any vector z,

$$\begin{aligned}\int_s \alpha_s f(s)ds &\le \int_s \alpha_s[f(z) + \sum_{j\in N}(f(z_s(j)) - f(z))]ds \\ &\le f(z) + \sum_{j\in N}\int_{s_j>z_j} g_j(s)(f(z_s(j)) - f(z))ds_j.\end{aligned}$$

The first inequality above holds due to submodularity of f and the second one is a consequence of the definition of α. Also, observe that by plugging $\alpha = \beta$ in this inequality we have $f^+(y) \le f^*(y)$.

Now, it is enough to prove that for all y, $F_g(y) \ge (1-\frac{1}{e})f^*(y)$. Similar to the proof of Lemma 3.8 [18], for each j we define a Poisson clock $\mathcal{C}_j$ with rate y_j. We start with a vector $z = 0$. Once clock $\mathcal{C}_j$ sends a signal, a random variable x is produced from distribution g_j. Then, if $z_j < x$, the value of z_j will change to x. By abuse of notation, we denote this new value by $z_x(j)$ and the value of vector z at time t by $z(t)$. One can observe that $\mathrm{E}[f(z(1))] \le F_g(y)$, using monotonicity of f. On the other hand,

$$\mathrm{E}[f(z(t+dt)) - f(z(t))|z(t) = z] = \sum_j y_j dt[\int_{x>z_j} g_j(x)(f(z_x(j)) - f(z))dx].$$

But the R.H.S. is at least $(f^*(y) - f(z))dt$, by the definition of f^*. Therefore, the following bound can be derived on the derivative of $E[f(z(t))]$:

$$\begin{aligned}\frac{1}{dt}E[f(z(t+dt)) - f(z)|z(t) = z] &\ge (f^*(y) - f(z))dt \\ \Rightarrow \frac{d}{dt}E[f(z(t))] &\ge (f^*(y) - E[f(z(t))])dt.\end{aligned}$$

Solving the differential equation above, shows that $E[f(z(t)) \ge (1-e^{-t})f^*(y)$. Combining this with the fact that $f^+(y) \le f^*(y)$ and also that $\mathrm{E}[f(z(1)] \le F_g(y)$ completes the proof of lemma.

The next lemma shows how to round the vector y to a proper 0/1 vector w.

Lemma 11. *There exists a 0/1 vector $w \in P_k$ such that $\forall y : F_g(w) \ge F_g(y)$.*

Proof. The essential rounding tool for the proof is pipage rounding introduced by [1]. In order to be able to use pipage rounding we need to prove some convexity property on F_g. Define $F^y_{ij} = F_g(y_{ij}(\lambda))$ where $y_{ij}(\lambda)$ is a vector obtained by adding λ to y_i, subtracting λ from y_j and leaving all other entries of y unchanged. First, we show that F^y_{ij} is a convex function of λ. For any y, the function $F_g(y)$ can be written as below.

$$\begin{aligned}F_g(y) = \sum_{R\in N\setminus\{i,j\}}\prod_{k\in R} y_k \prod_{k\notin R\cup\{i,j\}}(1-y_k) \times [(1-y_i)(1-y_j)F_g(R) + \\ (1-y_i)y_jF_g(R+j) + y_i(1-y_j)F_g(R+i) + y_iy_jF_g(R+i+j)].\end{aligned}$$

Hence, we can write the second derivative of F_{ij}^y in an explicit form:

$$\frac{\partial^2 F_{ij}^y}{\partial \lambda^2} = \sum_{R \in N \setminus \{i,j\}} \prod_{k \in R} y_k \prod_{k \notin R \cup \{i,j\}} (1 - y_k) \times \\ [-F_g(R) + F_g(R+i) + F_g(R+j) - F_g(R+i+j)],$$

which is clearly non-negative due to submodularity of F_g.

As a result of convexity of F_{ij}^y, for any vector $y \in P_k$, the main result of [1] ensures that Pipage rounding yields a 0/1 vector w inside P_k such that $F_g(w) \geq F_g(y)$. Hence, there exists such a vector w for which $F_g(w) \geq F_g(y)$ holds for all $y \in P_k$.

Now, we are ready to prove Theorem 7.

Proof. [**Theorem 7**]. Lemma 8 shows that $\mathrm{OPT} = \max_{y \in P_k} f^+(y)$ is an upper-bound on the performance of the best adaptive policy. But from Lemma 10 we know that there exists a vector y^* such that $F_g(y^*) \geq (1 - \frac{1}{e})\mathrm{OPT}$. On the other hand, Lemma 11 implies that there exists a 0/1 vector $w \in P$ such that $F_g(w) \geq F_g(y^*)$ and hence, $F_g(w) \geq (1 - \frac{1}{e})\mathrm{OPT}$. Notice that $F_g(w)$ is in fact the expected value gained by a non-adaptive policy that selects the set S corresponding to the vector w. Also, due to Corollary 5 greedy non-adaptive policy obtains a value at least $(1 - \frac{1}{e})F_g(w)$ that will be at least $(1 - \frac{1}{e})^2\mathrm{OPT}$.

References

1. A. A. Ageev and M. Sviridenko. Pipage rounding: A new method of constructing algorithms with proven performance guarantee. *J. Comb. Optim.*, 8(3):307–328, 2004.
2. S. Ahmed and A. Atamtürk. Maximizing a class of submodular utility functions. Research Report BCOL.08.02, IEOR, University of California-Berkeley, March 2008.
3. C. Chan and V. Farias. Stochastic depletion problems: Effective myopic policies for a class of dynamic optimization problems. *manuscript*, 2008.
4. G. Cornuejols, M. Fisher, and G. Nemhauser. Location of bank accounts to optimize float. *Management Science*, 23:789–810, 1977.
5. B. C. Dean, M. X. Goemans, and J. Vondrák. Approximating the stochastic knapsack problem: The benefit of adaptivity. In *FOCS*, pages 208–217, 2004.
6. U. Feige. A threshold of ln for approximating set cover. *J. ACM*, 45(4):634–652, 1998.
7. U. Feige, V. S. Mirrokni, and J. Vondrák. Maximizing non-monotone submodular functions. In *FOCS*, pages 461–471, 2007.
8. M. X. Goemans and J. Vondrák. Stochastic covering and adaptivity. In *LATIN*, pages 532–543, 2006.
9. D. Kempe, J. M. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In L. Getoor, T. E. Senator, P. Domingos, and C. Faloutsos, editors, *KDD*, pages 137–146. ACM, 2003.

10. J. M. Kleinberg, C. H. Papadimitriou, and P. Raghavan. Segmentation problems. *J. ACM*, 51(2):263–280, 2004.
11. A. Krause and C. Guestrin. Near-optimal nonmyopic value of information in models. In *AAAI*, pages 324–331, 2005.
12. A. Krause and C. Guestrin. Near-optimal observation selection using submodular functions. In *AAAI*, pages 1650–1654. AAAI Press, 2007.
13. A. Mehrez and Z. Sinuany-Stern. Resource allocation to interrelated risky projects using a multiattribute utility function. *Management Science*, (29):490439, 1983.
14. E. Mossel and S. Roch. On the submodularity of influence in social networks. In *Approx*, pages 128–134, 2007.
15. G. Nemhauser, L. Wolsey, and M. Fisher. An analysis of the approximations for maximizing submodular set functions. *Mathematical Programming*, 14:265–294, 1978.
16. M. Streeter and D. Golovin. An online algorithm for maximizing submodular functions. Tech Report CMU-CS-07-171, 2008.
17. D. M. Topkis. Minimizing a submodular function on a lattice. *Operations Research*, 26(2):305–321, 1978.
18. J. Vondrák. *Submodularity in combinatorial optimization.* PhD thesis, Charles University, Prague, 2007.
19. J. Vondrák. Optimal approximation for the submodular welfare problem in the value oracle model. In *STOC*, 2008.
20. H. Weingartner. *Mathematical Programming and the Analysis of Capital Budgeting Problems.* Prentice-Hall, 1963.
21. H. Weingartner. Capital budgeting of interrelated projects: Survey and synthesis. *Management Science*, 1966.