

# Price of Dependence: Stochastic Submodular Maximization with Dependent Items

Shaojie Tang

The University of Texas at Dallas

In this paper, we study the stochastic submodular maximization problem with dependent items subject to packing constraints such as matroid and knapsack constraints. The input of our problem is a finite set of items, and each item is in a particular state from a set of possible states. After picking an item, we are able to observe its state. We assume a monotone and submodular utility function over items and states, and our objective is to select a group of items adaptively so as to maximize the expected utility. Previous studies on stochastic submodular maximization often assume that items' states are independent, however, this assumption may not hold in general. This motivates us to study the stochastic submodular maximization problem with dependent items. We first introduce the concept of *degree of independence* to capture the degree to which one item's state is dependent on others'. Then we propose a non-adaptive policy that approximates the optimal adaptive policy within a factor of  $\alpha(1 - e^{-\frac{\kappa}{2} + \frac{\kappa}{18m^2}} - \frac{\kappa+2}{3m\kappa})$  where the value of  $\alpha$  is depending on the type of constraints, e.g.,  $\alpha = 1$  for matroid constraint,  $\kappa$  is the degree of independence, e.g.,  $\kappa = 1$  for independent items, and  $m$  is the number of items. We also analyze the adaptivity gap, i.e., the ratio of the values of best adaptive policy and best non-adaptive policy, of our problem with prefix-closed constraints.

---

## 1. Introduction

Stochastic submodular maximization (SSM) and its variants have been extensively studied recently (Asadpour et al. 2008)Asadpour and Nazerzadeh (2015)Adamczyk et al. (2016)Hellerstein et al. (2015). The input of SSM is a set of items, each item belongs to a particular state from a set of possible states. After picking an item, we are able to observe its state. Given a monotone and submodular utility function over all items and their states, our objective is to adaptively select a group of items that maximize the expected utility subject to a variety of constraints. One example is stochastic sensor cover problem. In this example, we are given a set of sensors and the state of each sensor is the subset of targets it covers, this subset may change due to uncertain environmental conditions. After selecting a sensor, we are able to observe its state, i.e., the actual subset of targets that can be covered by this sensor. Then the objective of stochastic sensor cover problem is to adaptively select a group of sensors to cover the largest amount of targets (in expectation).

Majority of existing work assume that items are independent from each other, i.e., one item's state does not depend on others'. However, this assumption does not always hold in reality. Consider the example of stochastic maximum  $k$ -cover, since each sensor's state is affected by the environmental conditions which are shared among all sensors, thus their states are correlated. Another example is from viral marketing (Yuan and Tang 2017a,b) where one customer's influence in the social network is correlated with her neighbors' influence. Golovin and Krause (2011) extended the previous studies to dependent items, however, their results only hold when the utility function is *adaptive submodular*. It is not clear how to generalize their results to more general settings when adaptive submodularity does not hold. Very recently, Fujii and Sakaue (2019) propose a concept named *adaptive submodularity ratio* to study the performance of greedy policy subject to cardinality constraint. However, their analysis cannot be applied to general constraints.

In this paper, we study the SSM with dependent items (SSMDI) subject to general constraints such as downward-closed and prefix-closed constraints. To capture the degree to which one item's state is correlated with others', we introduce the concept of *degree of independence*. A larger degree of independence indicates a weaker correlation among all items' states, i.e., this value is 1 for independent items. Then we propose a non-adaptive policy based on the optimistic continuous greedy algorithm. We say a policy is non-adaptive if it always picks the next item before observing the states of picked items. We show that our non-adaptive policy achieves approximation ratio  $\alpha(1 - e^{-\frac{\kappa}{2} + \frac{\kappa}{18m^2}} - \frac{\kappa+2}{3m\kappa})$  where the value of  $\alpha$  is depending on the type of constraints, e.g.,  $\alpha = 1$  for matroid constraint,  $\kappa$  is the degree of independence, and  $m$  is the number of items. Since our policy is non-adaptive,  $\frac{1}{\alpha(1 - e^{-\frac{\kappa}{2} + \frac{\kappa}{18m^2}} - \frac{\kappa+2}{3m\kappa})}$  is also known as an upper bound on the adaptivity gap, i.e., the ratio of the values of best adaptive and best non-adaptive policies is bounded by  $\frac{1}{\alpha(1 - e^{-\frac{\kappa}{2} + \frac{\kappa}{18m^2}} - \frac{\kappa+2}{3m\kappa})}$ . As a negative result, Asadpour and Nazerzadeh (2015) show that the adaptivity gap with dependent items can be at least  $m/2$ . At the end of this paper, we also analyze the adaptivity gap of SSMDI with prefix-closed constraints.

## 2. Preliminaries

### 2.1. Submodular Function and Multilinear Extension

A submodular function is a set function  $f : 2^\Omega \rightarrow \mathbb{R}$ , where  $2^\Omega$  denotes the power set of  $\Omega$ , which satisfies a natural “diminishing returns” property: the marginal gain from adding an element to a set  $X$  is at least as high as the marginal gain from adding the same

element to a superset of  $X$ . Formally, a submodular function satisfies the follow property: For every  $X, Y \subseteq \Omega$  with  $X \subseteq Y$  and every  $x \in \Omega \setminus Y$ , we have that  $f(X \cup \{x\}) - f(X) \geq f(Y \cup \{x\}) - f(Y)$ . We say a submodular function  $f$  is monotone if  $f(X) \leq f(Y)$  whenever  $X \subseteq Y$ . Consider any vector  $\mathbf{x} \in [0, 1]^n$ . The multilinear extension of  $f$  is defined as  $F(\mathbf{x}) = \sum_{X \subseteq \Omega} f(X) \prod_{i \in X} x_i \prod_{i \notin X} (1 - x_i)$ .

### 3. Notations and Problem Formulation

#### 3.1. Items and States

Let  $E$  denote a finite set of  $m$  items, and each item  $e \in E$  is in a particular state from a set  $O$  of possible states. Let  $\phi : E \rightarrow O$  denote a realization of item states. Let  $\Phi = \{\Phi_e | e \in E\}$  be a random realization where  $\Phi_e$  denotes a random realization of  $e$ . After picking an item  $e$ , we are able to observe its realization  $\Phi_e$ . Let  $\mathcal{U}$  denote the set of all realizations, we assume there is a known prior probability distribution  $\mathcal{D}$  over realizations, i.e.,  $\mathcal{D} = \{\Pr[\Phi = \phi] : \phi \in \mathcal{U}\}$ . Given a set of items  $S$ , let  $\Phi_S = \{\Phi_e | e \in S\}$  denote the random realization of  $S$  and  $\mathcal{U}_S$  denote the set of all possible realizations that involve  $S$ . Let  $\mathcal{D}_e(\phi_S) = \{\Pr[\Phi_e = \phi_e | \Phi_S = \phi_S] : \phi_e \in \mathcal{U}_e\}$  denote the prior probability distribution over realizations conditioned on  $\Phi_S = \phi_S$ .

#### 3.2. Utility Function and Problem Formulation

Let  $f : 2^{E \times O} \rightarrow \mathbb{R}_{\geq 0}$  be a monotone and submodular function over all items and their states. A policy  $\pi$  is a function that specifies which item to pick next under the observations made so far:  $\pi : 2^{E \times O} \rightarrow E$ . Note that  $\pi$  can be regarded as some decision tree that specifies a rule for picking items adaptively. Let  $\mathcal{I}$  be a *downward-closed* family of subsets of  $E$ . Let  $E(\pi, \phi)$  denote the subset of items picked by policy  $\pi$  under  $\phi$ . Then the utility of  $\pi$  can be expressed as  $f(\pi) = \sum_{\phi \in \mathcal{U}} \beta_\phi f(\cup_{e \in E(\pi, \phi)} \phi_e)$  where  $\beta_\phi$  denotes the probability that  $\phi$  is realized. We say a policy  $\pi$  is *feasible* if for any  $\phi$ ,  $E(\pi, \phi) \in \mathcal{I}$ . Our goal is to identify the best feasible policy that maximizes its expected utility.

$$\max_{\pi} f(\pi) \text{ subject to } E(\pi, \phi) \in \mathcal{I} \text{ for any } \phi.$$

#### 3.3. More Notations and Degree of Independence

By abuse of notation, define  $f(S) = \mathbb{E}_{\Phi \sim \mathcal{D}}[f(\cup_{v \in S} \Phi_v)]$  as the value of items  $S \subseteq E$  where  $\Phi \sim \mathcal{D}$  means  $\Phi$  follows the distribution of  $\mathcal{D}$ . Let  $f_S(e) = f(S \cup \{e\}) - f(S)$  denote the marginal value of item  $e$  with respect to  $S$ . Define  $f_S(\phi_e) = \mathbb{E}_{\Phi \sim \mathcal{D}}[f((\cup_{v \in S} \Phi_v) \cup \phi_e)] - f(S)$  as the marginal value of item  $e$ 's state  $\phi_e$  with respect to  $S$ .

Given a vector  $\mathbf{x} \in [0, 1]^m$ , let  $R$  be a random set obtained by picking each item  $e$  independently with probability  $x_e$ , then the multilinear extension  $F(\mathbf{x})$  is defined as the expected value of  $f(R)$ :  $F(\mathbf{x}) = \sum_{S \subseteq V} f(S) \prod_{e \in S} x_e \prod_{e \notin S} (1 - x_e)$ . Let  $F_{\mathbf{x}}(e) = \mathbb{E}[f_R(e)]$  denote the marginal value of item  $e$  with respect to  $\mathbf{x}$ . Define  $F_{\mathbf{x}}(\phi_e) = \mathbb{E}[f_R(\phi_e)]$  as the marginal value of item  $e$ 's state  $\phi_e$  with respect to  $\mathbf{x}$ . For notation convenience, denote  $\mathbf{x} \setminus e$  as a new vector by setting  $x_e$  in  $\mathbf{x}$  to be 0.

We next introduce the concept of *degree of independence* which refers to the degree to which one item's state is correlated the others'.

**DEFINITION 1.** [Degree of Independence] The degree of independence  $\kappa(\mathcal{D}) \in [0, 1]$  of a known prior probability distribution  $\mathcal{D}$  is defined as follows

$$\kappa(\mathcal{D}) := \min_{e \in E, S \subseteq E \setminus \{e\}, V \subseteq E \setminus \{e\}, \phi_V \in \mathcal{U}_V} \frac{f_S(e)}{\mathbb{E}_{\Phi_e \sim \mathcal{D}_e(\phi_V)}[f_S(\Phi_e)]} \quad (1)$$

If the numerator and denominator are both 0, the ratio is considered to be 1.

Notice that if all items' states are realized independently from each other, we have  $\forall e \in E, \forall S \subseteq E \setminus \{e\}, \forall V \subseteq E \setminus \{e\}, \forall \phi_V \in \mathcal{U}_V : f_S(e) = \mathbb{E}_{\Phi_e \sim \mathcal{D}_e(\phi_V)}[f_S(\Phi_e)]$  due to the state of  $e$  does not dependent on other items' states. It follows that  $\kappa(\mathcal{D}) = 1$  when items are independent. We omit  $\mathcal{D}$  from  $\kappa(\mathcal{D})$  if it is clear from the context.

## 4. Algorithm Design

In this section, we present a **non-adaptive policy**  $\pi^g$ , later, we show that the ratio of the utilities of  $\pi^g$  to **best adaptive policy** is bounded. This result is also known as adaptivity gap. The general idea of  $\pi^g$  is to first find a fractional solution using the optimistic continuous greedy algorithm (Algorithm 1) and then round it to an integral solution.

*Stage 1: Optimistic Continuous Greedy Algorithm* We first explain the design of the optimistic continuous greedy algorithm. Algorithm 1 maintains a fractional solution  $\mathbf{y}(t)$ , starting with  $\mathbf{y}(0) = (0, 0, \dots, 0)$ . Let  $R(t)$  be a random set which contains each  $e$  independently with probability  $y_e(t)$ . In each round  $t$ , it updates the weight  $F_{\mathbf{y}(t) \setminus e}(e)$  of each item  $e \in E$  as follows,

$$F_{\mathbf{y}(t) \setminus e}(e) = \mathbb{E}[f(R_{\bar{e}}(t) \cup \{e\})] - \mathbb{E}[f(R_{\bar{e}}(t))] \quad (2)$$

where  $R_{\bar{e}}(t) = R(t) \setminus e$  is a subset of  $R(t)$  by excluding  $e$ . As compared with the standard continuous greedy algorithm (Calinescu et al. 2011), we define the weight of each item in

a different way, i.e., Calinescu et al. (2011) adopt the following weight function  $F_{\mathbf{y}(t)}(e) = \mathbb{E}[f(R(t) \cup \{e\})] - \mathbb{E}[f(R(t))]$ . We call our algorithm optimistic continuous greedy algorithm due to  $F_{\mathbf{y}(t) \setminus e}(e) \geq F_{\mathbf{y}(t)}(e)$ , i.e., the expected marginal contribution of any item  $e$  defined in our algorithm is no less than the one defined in the standard continuous greedy algorithm. Eq. (2) is also different from the one defined in Asadpour and Nazerzadeh (2015). They define the weight of  $e$  as  $\mathbb{E}[f(\Theta_{R(t)}^{\uparrow e})] - \mathbb{E}[f(\Theta_{R(t)})]$  which only makes sense when the state of each item belongs to  $\mathbb{R}_+$  (See Section 2 in Asadpour and Nazerzadeh (2015)).

Since we are not able to obtain the exact value of  $F_{\mathbf{y}(t) \setminus e}(e)$ , we estimate this value by averaging over  $\frac{10}{\delta^2}(1 + \ln m)$  independent samples of  $R_{\bar{e}}(t)$ . Let  $\tilde{F}_{\mathbf{y}(t) \setminus e}(e)$  denote the estimated value of  $F_{\mathbf{y}(t) \setminus e}(e)$ . Assume  $P_{\mathcal{I}} = \text{conv}\{\mathbf{1}_I : I \in \mathcal{I}\}$ , the convex relaxation for  $\mathcal{I}$ , is a down-monotone solvable polytope, we solve the following optimization problem.

$$\begin{array}{ll} \mathbf{P1:} & \text{Maximize } \sum_{e \in E} \tilde{F}_{\mathbf{y}(t) \setminus e}(e) y_e \\ & \text{subject to: } \mathbf{y} \in P_{\mathcal{I}} \end{array}$$

After solving **P1** at round  $t$  and obtain an optimal solution  $\bar{\mathbf{y}}(t)$ , we update the fractional solution at round  $t$  as follows  $y_e(t + \delta) = y_e(t) + \delta \bar{y}_e(t)$ . After  $1/\delta$  rounds where  $\delta = \frac{1}{9m^2}$ ,  $\mathbf{y}(1)$  is returned as the final solution.

*Stage 2: Rounding Fractional Solution* In the second stage, we round the fractional solution  $\mathbf{y}(1)$  to an integral solution. As shown in Chekuri et al. (2014), if there exists a  $\alpha$ -balanced contention resolution scheme for  $\mathcal{I}$  where  $\alpha \in [0, 1]$ , then we can find a feasible solution  $I \in \mathcal{I}$  such that  $\mathbb{E}[f(I)] \geq \alpha F(\mathbf{y}(1))$ . It turns out many useful constraints admit good  $\alpha$ -balanced contention resolution schemes, including matroid constraints and knapsack constraints. For example, when  $\mathcal{I}$  specifies an intersection of  $k$  matroids and  $O(1)$  knapsacks, we have  $\alpha = \frac{0.38}{k}$ . Notice that when  $\mathcal{I}$  specifies a matroid constraint, we can apply pipage rounding technique (Ageev and Sviridenko 2004) to find a feasible solution with expected utility  $F(\mathbf{y}(1))$ , i.e.,  $\alpha = 1$ .

## 5. Performance Analysis

In this section, we prove the following main results.

**THEOREM 1.** *Assume  $\pi^\diamond$  is the optimal policy and there exists a  $\alpha$ -balanced contention resolution scheme for  $\mathcal{I}$ , we have  $f(\pi^g) \geq \alpha(1 - e^{-\frac{\kappa}{2} + \frac{\kappa}{18m^2}} - \frac{\kappa+2}{3m\kappa})f(\pi^\diamond)$  where  $\kappa$  is the degree of independence.*

**Algorithm 1** Optimistic Continuous Greedy

- 1: Set  $\delta = \frac{1}{9m^2}$ ,  $t = 0$ ,  $\mathbf{y}(0) = (0, 0, \dots, 0)$ .
- 2: **while**  $t < 1$  **do**
- 3:   Let  $R(t)$  be a random set which contains each  $e$  independent with probability  $y_e(t)$ .
- 4:   For each  $e \in E$ , let  $R_{\bar{e}}(t) = R(t) \setminus e$ , estimate

$$F_{\mathbf{y}(t) \setminus e}(e) = \mathbb{E}[f(R_{\bar{e}}(t) \cup \{e\})] - \mathbb{E}[f(R_{\bar{e}}(t))]$$

- 5:   Solve **P1** with estimated weight  $\tilde{F}_{\mathbf{y}(t) \setminus e}(e)y_e$  for each  $e$  and obtain an optimal solution  $\bar{\mathbf{y}}(t)$

- 6:   

**P1:** Maximize  $\sum_{e \in E} \tilde{F}_{\mathbf{y}(t) \setminus e}(e)y_e$   
**subject to:**  $\mathbf{y} \in P_{\mathcal{I}}$  where  $P_{\mathcal{I}} = \text{conv}\{\mathbf{1}_I : I \in \mathcal{I}\}$  is a down-monotone solvable polytope.

- 7:   Let  $y_e(t + \delta) = y_e(t) + \delta \bar{y}_e(t)$ ;
- 8:   Increment  $t = t + \delta$ ;

*Proof:* Observing that if there exists a  $\alpha$ -balanced contention resolution scheme for  $\mathcal{I}$ , we can find a feasible solution  $I \in \mathcal{I}$  such that  $\mathbb{E}[f(I)] \geq \alpha F(\mathbf{y}(1))$ , it suffice to prove that  $F(\mathbf{y}(1)) \geq (1 - e^{-\frac{\kappa}{2} + \frac{\kappa}{18m^2}} - \frac{\kappa+2}{3m\kappa})f(\pi^\diamond)$ . Thus, in the rest of the proof, we focus on proving this inequality.

For every  $e \in E$ , let  $y_e^\diamond$  denote the probability that  $e$  is picked by  $\pi^\diamond$ . Due to  $\pi^\diamond$  is a feasible policy,  $\mathbf{y}^\diamond$  is a convex combination of feasible solutions in  $\mathcal{I}$ , thus,  $y_e^\diamond \in P_{\mathcal{I}}$ . We first prove that  $f(\pi^\diamond) \leq F(\mathbf{x}) + \kappa \sum_{e \in E} y_e^\diamond F_{\mathbf{x} \setminus e}(e)$  for any vector  $\mathbf{x} \in [0, 1]^m$ .

$$\begin{aligned}
f(\pi^\diamond) &= \sum_{\phi \in \mathcal{U}} \beta_\phi f(\cup_{e \in E(\pi^\diamond, \phi)} \phi_e) \leq \sum_{\phi \in \mathcal{U}} \beta_\phi (F(\mathbf{x}) + \sum_{e \in E(\pi^\diamond, \phi)} F_{\mathbf{x}}(\phi_e)) \\
&\leq \sum_{\phi \in \mathcal{U}} \beta_\phi (F(\mathbf{x}) + \sum_{e \in E(\pi^\diamond, \phi)} F_{\mathbf{x} \setminus e}(\phi_e)) \\
&\leq F(\mathbf{x}) + \sum_{\phi \in \mathcal{U}} \sum_{e \in E(\pi^\diamond, \phi)} (\beta_\phi F_{\mathbf{x} \setminus e}(\phi_e)) \\
&\leq F(\mathbf{x}) + \frac{1}{\kappa} \sum_{e \in E} y_e^\diamond F_{\mathbf{x} \setminus e}(e)
\end{aligned} \tag{3}$$

The first two inequalities are due to the submodularity of  $f$ . The third inequality is due to  $\sum_{\phi \in \mathcal{U}} \beta_\phi = 1$ . The last inequality is due to the definition of degree of independence (Definition 1).

We next provide a lower bound on the increased utility of  $F(\mathbf{y}(t))$  during one round of Algorithm 1. To simplify the notation, we use  $\bar{\mathbf{y}}$  to denote the optimal solution to **P1** in

round  $t$ .

$$\begin{aligned} & F(\mathbf{y}(t + \delta)) - F(\mathbf{y}(t)) \\ & \geq \sum_{e \in E} \delta \bar{y}_e \prod_{e' \neq e} (1 - \delta \bar{y}_{e'}) F_{\mathbf{y}(t)}(e) \end{aligned} \quad (4)$$

$$\geq \sum_{e \in E} \delta \bar{y}_e \prod_{e' \neq e} (1 - \delta \bar{y}_{e'}) (1 - y_e(t)) F_{\mathbf{y}(t) \setminus e}(e) \quad (5)$$

$$\begin{aligned} & \geq \sum_{e \in E} \delta \bar{y}_e (1 - \delta)^{m-1} (1 - y_e(t)) F_{\mathbf{y}(t) \setminus e}(e) \\ & = \delta (1 - \delta)^{m-1} \sum_{e \in E} \bar{y}_e (1 - y_e(t)) F_{\mathbf{y}(t) \setminus e}(e) \\ & \geq \delta (1 - \delta)^{m-1} \sum_{e \in E} \bar{y}_e (1 - t\delta) F_{\mathbf{y}(t) \setminus e}(e) \end{aligned} \quad (6)$$

$$\begin{aligned} & = (1 - t\delta) \delta (1 - \delta)^{m-1} \sum_{e \in E} \bar{y}_e F_{\mathbf{y}(t) \setminus e}(e) \\ & \geq (1 - t\delta) \delta (1 - \delta)^{m-1} \left( \sum_{e \in E} y_e^\diamond F_{\mathbf{y}(t) \setminus e}(e) - 2m\delta f(\pi^\diamond) \right) \end{aligned} \quad (7)$$

$$\begin{aligned} & \geq (1 - t\delta) \delta (1 - \delta)^{m-1} (\kappa(f(\pi^\diamond) - F(\mathbf{y}(t))) - 2m\delta f(\pi^\diamond)) \\ & = (1 - t\delta) \delta (1 - \delta)^{m-1} \kappa \left( \left(1 - \frac{2m\delta}{\kappa}\right) f(\pi^\diamond) - F(\mathbf{y}(t)) \right) \end{aligned} \quad (8)$$

$$\begin{aligned} & \geq (1 - t\delta) \delta (1 - \delta m) \kappa \left( \left(1 - \frac{2m\delta}{\kappa}\right) f(\pi^\diamond) - F(\mathbf{y}(t)) \right) \\ & \geq (1 - t\delta) \delta \kappa \left( \left(1 - \frac{(\kappa + 2)m\delta}{\kappa}\right) f(\pi^\diamond) - F(\mathbf{y}(t)) \right) \end{aligned} \quad (9)$$

Inequality (4) is due to Lemma 3.3 in (Calinescu et al. 2011). Inequality (5) is due to  $F_{\mathbf{y}(t)}(e) = (1 - y_e(t)) F_{\mathbf{y}(t) \setminus e}(e)$ . Inequality (6) is due to  $\min_{e \in E} (1 - y_e(t)) \geq 1 - t\delta$ . Inequality (7) is due to  $\bar{\mathbf{y}}$  is an optimal solution to  $\mathbf{P1}$  and  $\mathbf{y}^\diamond \in P_{\mathcal{I}}$  is a feasible solution to  $\mathbf{P1}$ , and the rest proof is similar to the one of Lemma 3.2 in (Calinescu et al. 2011). Inequality (8) is due to Inequality (3).

By induction, we have  $F(\mathbf{y}(1)) \geq (1 - e^{-\sum_{t=1}^{1/\delta} (1-t\delta)\delta\kappa}) (1 - \frac{(\kappa+2)m\delta}{\kappa}) f(\pi^\diamond)$ . It follows that  $F(\mathbf{y}(1)) \geq (1 - e^{-\frac{\kappa(1-\delta)}{2}}) (1 - \frac{(\kappa+2)m\delta}{\kappa}) f(\pi^\diamond)$ . Finally, since  $\delta = \frac{1}{9m^2}$ , we have  $F(\mathbf{y}(1)) \geq (1 - e^{-\frac{\kappa}{2} + \frac{\kappa}{18m^2}}) (1 - \frac{\kappa+2}{3m\kappa}) f(\pi^\diamond) \geq (1 - e^{-\frac{\kappa}{2} + \frac{\kappa}{18m^2}} - \frac{\kappa+2}{3m\kappa}) f(\pi^\diamond)$ .  $\square$

As a byproduct of our main theorem, we have the following corollary.

**COROLLARY 1.** *The adaptivity gap of SSMDI is  $\frac{1}{\alpha(1 - e^{-\frac{\kappa}{2} + \frac{\kappa}{18m^2}} - \frac{\kappa+2}{3m\kappa})}$ .*

## 6. Adaptivity Gap of SSMDI with Prefix-closed Constraints

In this section, we analyze the adaptivity gap of SSMDI with prefix-closed constraints. Notice that the class of prefix-closed constraints contains any downward-closed/packing



constraint (Gupta et al. 2017). To facilitate our study, we first introduce a new form of degree of independence.

DEFINITION 2. [A Second Form of Degree of Independence] The degree of independence  $\gamma(\mathcal{D}) \in [0, 1]$  of a known prior probability distribution  $\mathcal{D}$  is defined as follows

$$\gamma(\mathcal{D}) := \min_{e \in E, V \subseteq E \setminus \{e\}, \phi_V, \phi'_V \in \mathcal{U}_V} \frac{\mathbb{E}_{\Phi_e}[f_{\phi_V \cup \phi'_V}(\Phi_e)]}{\mathbb{E}_{\Phi'_e}[f_{\phi_V \cup \phi'_V}(\Phi'_e)]} \quad (10)$$

where  $\Phi'_e$  follows the distribution of  $\mathcal{D}_e(\phi'_V)$ ,  $\Phi_e$  follows the distribution of  $\mathcal{D}_e(\phi_V)$ , and  $f_{\phi_V \cup \phi'_V}(\cdot) = f(\phi_V \cup \phi'_V \cup \cdot) - f(\phi_V \cup \phi'_V)$ . If the numerator and denominator are both 0, the ratio is considered to be 1.

Clearly, when all items are realized independently,  $\gamma(\mathcal{D}) = 1$ . We next present the main theorem. We omit  $\mathcal{D}$  from  $\gamma(\mathcal{D})$  if it is clear from the context.

THEOREM 2. *The adaptivity gap of SSMDI with prefix-closed constraints is  $\frac{1+\gamma}{\gamma}$ .*

*Proof:* The basic idea of our proof is to show that given any adaptive policy, we can build a non-adaptive policy whose utility is at least  $\frac{\gamma}{1+\gamma}$  times the utility of the given adaptive policy. The way we construct such a non-adaptive policy is inspired by the one proposed in (Bradac et al. 2019). However, our setting is more complicated than theirs due to the dependency among items.

DEFINITION 3 (VIRTUAL NON-ADAPTIVE POLICY). Given any adaptive policy  $\pi$ , we define a non-adaptive policy  $\sigma$  as follows: randomly draw a realization  $\phi$  according to  $\mathcal{D}$ , then run  $\pi$  on  $\phi$  virtually to pick a group of items.

Consider the first  $|S|$  items  $S$  picked by  $\pi$ , denote by  $A$  the realization of  $S$  when picked by  $\pi$  and we also use  $A$  to denote the virtual realization when picked by  $\sigma$ . Let  $B$  be the true realization of  $S$  when picked by  $\sigma$ . Let  $\pi_A$  denote the sub-decision tree  $\pi$  goes to when  $\Phi_S = A$ . Similarly, let  $\sigma_A$  denote the sub-decision tree<sup>1</sup>  $\sigma$  goes to when  $\Phi_S = A$ . Assume the first item picked by  $\pi_A$  and  $\sigma_A$  is  $v$ . Denote by  $\Phi_v$  the realization of  $v$  when picked by  $\sigma_A$  and also the virtual realization when picked by  $\sigma_A$ , while  $\Phi'_v$  be the true realization of  $v$  when picked by  $\sigma_A$ . Thus  $\Phi_v$  follows the distribution of  $\mathcal{D}_v(A)$  and  $\Phi'_v$  follows the distribution of  $\mathcal{D}_v(B)$ .

Given two realizations  $A$  and  $B$  of  $S$ , we redefine the expected utility of a policy based on a new utility function  $f_{A \cup B}(\cdot) = f(A \cup B \cup \cdot) - f(A \cup B)$ . Let  $\mathcal{R}(\pi_A, f_{A \cup B} | A) =$

<sup>1</sup> Any adaptive policy can be represented as a decision tree: Each node in the decision tree represents an item, we first pick the root item and observe its state, and then choose a subtree to go to.



$\sum_{\phi \in \mathcal{U}} \beta_{\phi}^A f_{A \cup B}(\cup_{e \in E(\pi_A, \phi)} \phi_e)$ , where  $\beta_{\phi}^A$  is the probability that  $\phi$  is realized conditioned on  $\Phi_S = A$ , denote the expected utility of  $\pi_A$  conditioned on  $\Phi_S = A$ . Similarly, we use  $\mathcal{R}(\sigma_A, f_{A \cup B} | B) = \sum_{\phi \in \mathcal{U}} \beta_{\phi}^B f_{A \cup B}(\cup_{e \in E(\sigma_A, \phi)} \phi_e)$ , where  $\beta_{\phi}^B$  is the probability that  $\phi$  is realized conditioned on  $\Phi_S = B$ , to denote the expected utility of  $\sigma_A$  conditioned on  $\Phi_S = B$ . To prove this theorem, it suffice to prove that when  $S = \emptyset$ ,  $\mathcal{R}(\sigma_{\emptyset}, f_{\emptyset} | \emptyset) \geq \frac{\gamma}{1+\gamma} \mathcal{R}(\pi_{\emptyset}, f_{\emptyset} | \emptyset)$ . Next we prove a even stronger result: for any  $S$  and its realizations  $A$  and  $B$ ,

$$\mathcal{R}(\sigma_A, f_{A \cup B} | B) \geq \frac{\gamma}{1+\gamma} \mathcal{R}(\pi_A, f_{A \cup B} | A) \quad (11)$$

We prove (11) through induction on the size of  $S$ . Assume the deepness of  $\pi$  is  $T$ , i.e.,  $\pi$  picks at most  $T$  items. The proof of the base case when  $|S| = T - 1$  is as follows. Note that when  $|S| = T - 1$ ,  $\pi_A$  (and also  $\sigma_A$ ) picks at most one item. To avoid trivial cases, assume  $\pi_A$  (and also  $\sigma_A$ ) picks  $v$ , it follows that

$$\mathcal{R}(\sigma_A, f_{A \cup B} | B) = \mathbb{E}_{\Phi_v} [f_{A \cup B}(\Phi_v)]$$

$$\mathcal{R}(\pi_A, f_{A \cup B} | A) = \mathbb{E}_{\Phi'_v} [f_{A \cup B}(\Phi_v)]$$

Because  $\mathbb{E}_{\Phi_v} [f_{A \cup B}(\Phi_v)] \geq \gamma \mathbb{E}_{\Phi'_v} [f_{A \cup B}(\Phi'_v)]$  due to Definition 2, we have

$$\mathcal{R}(\sigma_A, f_{A \cup B} | B) \geq \gamma \mathcal{R}(\pi_A, f_{A \cup B} | A) \geq \frac{\gamma}{1+\gamma} \mathcal{R}(\pi_A, f_{A \cup B} | A)$$

We next prove the induction step, assume the first item picked by  $\pi_A$  (and also  $\sigma_A$ ) is  $v$ . We first derive an uppder bound on  $\mathcal{R}(\pi_A, f_{A \cup B} | A)$ .

$$\begin{aligned} \mathcal{R}(\pi_A, f_{A \cup B} | A) &= \mathbb{E}_{\Phi_v} [f_{A \cup B}(\Phi_v) + \mathcal{R}(\pi_{A \cup \Phi_v}, f_{A \cup B \cup \Phi_v} | A \cup \Phi_v)] \\ &\leq \mathbb{E}_{\Phi_v, \Phi'_v} [f_{A \cup B}(\Phi_v \cup \Phi'_v) + \mathcal{R}(\pi_{A \cup \Phi_v}, f_{A \cup B \cup \Phi_v \cup \Phi'_v}^i | A \cup \Phi_v)] \end{aligned} \quad (12)$$

$$\begin{aligned} &\leq \mathbb{E}_{\Phi_v, \Phi'_v} [f_{A \cup B}(\Phi_v) + f_{A \cup B}(\Phi'_v) + \mathcal{R}(\pi_{A \cup \Phi_v}, f_{A \cup B \cup \Phi_v \cup \Phi'_v}^i | A \cup \Phi_v)] \\ &\leq (1 + \frac{1}{\gamma}) \mathbb{E}_{\Phi'_v} [f_{A \cup B}(\Phi'_v)] + \mathbb{E}_{\Phi_v, \Phi'_v} [\mathcal{R}(\pi_{A \cup \Phi_v}, f_{A \cup B \cup \Phi_v \cup \Phi'_v}^i | A \cup \Phi_v)] \end{aligned} \quad (13)$$

Inequality (12) is due to  $f_{A \cup B}$  is submodular. Inequality (13) is due to  $\gamma \mathbb{E}_{\Phi_v} [f_{A \cup B}(\Phi_v)] \leq \mathbb{E}_{\Phi'_v} [f_{A \cup B}(\Phi'_v)]$ . Next we derive a lower bound on  $\mathcal{R}(\sigma_A, f_{A \cup B} | B)$ .

$$\begin{aligned} \mathcal{R}(\sigma_A, f_{A \cup B} | B) &= \mathbb{E}_{\Phi_v, \Phi'_v} [f_{A \cup B}(\Phi'_v) + \mathcal{R}(\sigma_{A \cup \Phi_v}, f_{A \cup B \cup \Phi_v} | B \cup \Phi'_v)] \\ &\geq \mathbb{E}_{\Phi_v, \Phi'_v} [f_{A \cup B}(\Phi'_v) + \mathcal{R}(\sigma_{A \cup \Phi_v}, f_{A \cup B \cup \Phi_v \cup \Phi'_v} | B \cup \Phi'_v)] \\ &= \mathbb{E}_{\Phi'_v} [f_{A \cup B}(\Phi'_v)] + \mathbb{E}_{\Phi_v, \Phi'_v} [\mathcal{R}(\sigma_{A \cup \Phi_v}, f_{A \cup B \cup \Phi_v \cup \Phi'_v} | B \cup \Phi'_v)] \\ &\geq \mathbb{E}_{\Phi'_v} [f_{A \cup B}(\Phi'_v)] + \frac{\gamma}{1+\gamma} \mathbb{E}_{\Phi_v, \Phi'_v} [\mathcal{R}(\sigma_{A \cup \Phi_v}, f_{A \cup B \cup \Phi_v \cup \Phi'_v} | A \cup \Phi_v)] \end{aligned} \quad (14)$$

Inequalities (13) and (14) imply that  $\mathcal{R}(\sigma_A, f_{A \cup B} \mid B) \geq \frac{\gamma}{1+\gamma} \mathcal{R}(\pi_A, f_{A \cup B} \mid A)$ .  $\square$

When  $\gamma = 1$ , i.e., items are independent, Theorem 2 reduces to the following corollary.

**COROLLARY 2.** (*Bradac et al. 2019*) *The adaptivity gap is 2 when items are independent.*

## 7. Conclusion

Previous studies on SSM often assume that items are independent, however, this assumption may not always hold. In this paper, we study SSM with dependent items. To capture the impact of item dependency, we first introduce the concept of degree of independence. Then we propose a non-adaptive policy whose performance is close to the optimal adaptive policy. In particular, we prove that our non-adaptive policy achieves an approximation ratio whose value is depending on the degree of independence, the number of items, and the performance loss due to rounding. We also extend our study to the case with prefix-closed constraints and derive an adaptivity gap that depends on the (second form of) degree of independence.

## References

- Adamczyk, Marek, Maxim Sviridenko, Justin Ward. 2016. Submodular stochastic probing on matroids. *Mathematics of Operations Research* **41**(3) 1022–1038.
- Ageev, Alexander A, Maxim I Sviridenko. 2004. Pipage rounding: A new method of constructing algorithms with proven performance guarantee. *Journal of Combinatorial Optimization* **8**(3) 307–328.
- Asadpour, Arash, Hamid Nazerzadeh. 2015. Maximizing stochastic monotone submodular functions. *Management Science* **62**(8) 2374–2391.
- Asadpour, Arash, Hamid Nazerzadeh, Amin Saberi. 2008. Stochastic submodular maximization. *International Workshop on Internet and Network Economics*. Springer, 477–489.
- Bradac, Domagoj, Sahil Singla, Goran Zuzic. 2019. (near) optimal adaptivity gaps for stochastic multi-value probing. *CoRR* **abs/1902.01461**. URL <http://arxiv.org/abs/1902.01461>.
- Calinescu, Gruia, Chandra Chekuri, Martin Pál, Jan Vondrák. 2011. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM Journal on Computing* **40**(6) 1740–1766.
- Chekuri, Chandra, Jan Vondrák, Rico Zenklusen. 2014. Submodular function maximization via the multi-linear relaxation and contention resolution schemes. *SIAM Journal on Computing* **43**(6) 1831–1879.
- Fujii, Kaito, Shinsaku Sakaue. 2019. Beyond adaptive submodularity: Approximation guarantees of greedy policy with adaptive submodularity ratio. *arXiv preprint arXiv:1904.10748*.

- Golovin, Daniel, Andreas Krause. 2011. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *Journal of Artificial Intelligence Research* 427–486.
- Gupta, Anupam, Viswanath Nagarajan, Sahil Singla. 2017. Adaptivity gaps for stochastic probing: Submodular and xos functions. *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 1688–1702.
- Hellerstein, Lisa, Devorah Kletenik, Patrick Lin. 2015. Discrete stochastic submodular maximization: Adaptive vs. non-adaptive vs. offline. *International Conference on Algorithms and Complexity*. Springer, 235–248.
- Yuan, Jing, Shaojie Tang. 2017a. Adaptive discount allocation in social networks. *Proceedings of the eighteenth ACM international symposium on Mobile ad hoc networking and computing*. ACM.
- Yuan, Jing, Shaojie Tang. 2017b. No time to observe: adaptive influence maximization with partial feedback. *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. AAAI Press, 3908–3914.