



Maximization of Constrained Non-submodular Functions

Ruiqi Yang¹, Dachuan Xu¹, Donglei Du², Yicheng Xu^{3(✉)}, and Xihong Yan⁴

¹ Department of Operations Research and Scientific Computing,
Beijing University of Technology, Beijing 100124, People's Republic of China
yangruiqi@emails.bjut.edu.cn, xudc@bjut.edu.cn

² Faculty of Business, University of New Brunswick,
Fredericton, NB E3B 5A3, Canada
ddu@unb.ca

³ Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences,
Shenzhen 518055, People's Republic of China
yc.xu@siat.ac.cn

⁴ Department of Mathematics, Taiyuan Normal University,
Jinzhong 030619, Shanxi, People's Republic of China
xihong1@e.ntu.edu.sg

Abstract. We investigate a non-submodular maximization problem subject to a p -independence system constraint, where the non-submodularity of the utility function is characterized by a series of parameters, such as submodularity (supmodularity) ratio, generalized curvature, and zero order approximate submodularity coefficient, etc. Inspired by Feldman et al. [15] who consider a non-monotone submodular maximization with a p -independence system constraint, we extend their Repeat-Greedy algorithm to non-submodular setting. While there is no general reduction to convert algorithms for submodular optimization problems to non-submodular optimization problems, we are able to show the extended Repeat-Greedy algorithm has an almost constant approximation ratio for non-monotone non-submodular maximization.

Keywords: Approximation algorithms · Non-submodular maximization · Independence system constraint

1 Introduction

Submodular optimization is widely studied in optimization, computer science, and economics, etc. Submodularity is a very powerful tool in many optimization applications such as viral marketing [9, 18], recommendation system [12, 15, 21], nonparametric learning [1, 16], and document summarization [20], etc.

The greedy algorithm introduced by Nemhauser et al. [22] gave the first $(1 - e^{-1})$ -approximation for monotone submodular maximization with a cardinality constraint (SMC). Feige [13] considered a maximal k -cover problem, which is a special case of SMC, and showed that there is no algorithms with approximation

ratio greater than $(1 - e^{-1} + \epsilon)$ for any $\epsilon > 0$, under the assumption $P \neq NP$. Sviridenko [23] considered a monotone submodular maximization with a knapsack constraint, and provided a tight $(1 - e^{-1})$ -approximation algorithm with time complexity $O(n^5)$. Calinescu et al. [8] provided a $(1 - e^{-1})$ -approximation algorithm for monotone submodular maximization with a matroid constraint. All extant results of constrained submodular maximization assume monotonicity of the objective functions. In this paper, we consider a non-monotonic and non-submodular maximization problem subject to a more general p -independence system constraint.

1.1 Our Contributions

In this work, we consider a non-submodular maximization with an independence system constraint. Specifically, all feasible solutions associated with this model generate a p -independence system and the objective function is characterized by a series of parameters, such as submodularity (supmodularity) ratio, generalized curvature, and zero order approximate submodularity coefficient, etc. Our main results can be summarized as follows.

- We first investigate the efficiency of a greedy algorithm with two scenarios. Firstly, the objective function is non-submodularity and non-monotonic. Secondly, the feasible solution belongs to a p -independence system. We show that some good properties are still retained in the non-submodular setting (Theorem 1).
- Second we study the non-monotone non-submodular maximization problem without any constraint. Based on a simple approximate local search, for any $\epsilon > 0$ we show that there exists a polynomial time $(3/c^2 + \epsilon)$ -approximation algorithm, where c is the zero order approximate submodularity coefficient of objective function (Theorem 2).
- Finally, we apply the first two algorithms as the subroutines to solve a non-monotone non-submodular maximization problem with p -independence system constraint. Our algorithm is an extension of the Repeat-Greedy introduced in [15]. Based on a multiple times rounding of the above subroutine algorithms, we derived a nearly constant approximation ratio algorithm (Theorem 3).

1.2 The Organization

We give a brief summary of related work in Sect. 2. The necessary preliminaries and definitions are presented in Sect. 3. The main algorithms and analyses are provided in Sect. 4. We present a greedy algorithm in Sect. 4.1, an approximate local search in Sect. 4.2, and the core algorithm is provided in Sect. 4.3. In Sect. 5, we offer a conclusion for our work.

2 Related Works

Non-monotone Submodular Optimization. Unlike the monotone submodular optimization, there exists a natural obstacle in the study of non-monotonic

case. For example, direct application of the greedy algorithm introduced by Nemhauser et al. [22] to the non-monotonic case does not yield a constant approximation guarantee. Some previous work is summarized below. For the non-monotone submodular maximization problem without any constraint (USM), Feige et al. [14] presented a series of algorithms. They first showed a uniform random algorithm has a $1/4$ -approximation ratio. Second, they gave a deterministic local search $1/3$ -approximation algorithm and a random $2/5$ -approximation algorithm. For symmetric submodular functions, they derived a $1/2$ -approximation algorithm and showed that any $(1/2 + \epsilon)$ -approximation for symmetric submodular functions must need an exponential number of queries for any fixed $\epsilon > 0$. Based on local search technique, Buchbinder et al. [7] provided a random linear time $1/2$ -approximation algorithm.

Optimization of non-monotone submodular with complex constraints are also considered previously. Buchbinder and Feldman [5] gave a deterministic $1/2$ -approximation algorithm for USM with time complexity $O(n^2)$. For non-monotone submodular maximization with cardinality constraint, they derived a deterministic $1/e$ -approximation algorithm, which has a slightly better approximation ratio than the random $(1/e + 0.004)$ -approximation ratio by [6]. Buchbinder and Feldman [4] considered a more general non-monotone submodular maximization problem with matroid constraint and presented the currently best random 0.385 -approximation algorithm. Lee et al. [19] derived a $1/(p + 1 + 1/(p - 1) + \epsilon)$ -approximation algorithm as well as non-monotone submodular maximization algorithm with a constraint of the intersection of p matroids. For a more general p -independence system constraint, Gupta et al. [17] derived a $1/3p$ -approximation, which needs $O(np\ell)$ function value oracles, where ℓ is the maximum size of feasible solutions. Mirzasoleiman et al. [21] improved the approximation ratio to $1/2k$, while the time complexity was still bounded by $O(np\ell)$. Recently, with improved time complexity of $O(n\ell\sqrt{p})$, the approximation ratio was improved to $1/(p + \sqrt{p})$ by Feldman et al. [15].

Non-submodular Maximization. There are also many problems in optimization and machine learning whose utility functions do not possess submodularity. Das and Kempe [11] introduced a definition of submodularity ratio γ to measure the magnitude of submodularity of the utility function. For the maximization of monotone non-submodular function with cardinality constraint (NSMC), they showed the greedy algorithm can achieve a $(1 - e^{-\gamma})$ -approximation ratio. Conforti and Cornuéjols [10] studied the efficiency of the greedy algorithm by defining curvature κ of submodular objective functions for SMC and showed the approximation could be improved it to $1/\kappa(1 - e^{-\kappa})$. Bian et al. [2] introduced a more expressive formulation by providing a definition of the generalized curvature α of any non-negative set function. Combining the submodularity ratio with the generalized curvature, they derived the tight $1/\alpha(1 - e^{1/(\alpha\gamma)})$ -approximation ratio of the greedy algorithm for NSMC. Inspired by these work, Bogunovic et al. [3] introduced further parameters, such as supmodularity ratio, inverse generalized curvature, etc., to characterize the utility function. They derived the first

constant approximation algorithm for monotone robust non-submodular maximization problem with cardinality constraint.

3 Preliminaries

In this section, we present some necessary notations. We are given a ground set $\mathcal{V} = \{u_1, \dots, u_n\}$, and a utility function $f : 2^{\mathcal{V}} \rightarrow R_+$. The function f may not be submodular; namely the following zero order condition of submodular may not hold

$$f(A) + f(B) \geq f(A \cup B) + f(A \cap B), \forall A, B \subseteq \mathcal{V}.$$

For our purpose, we define a parameter c to approximate the submodularity of our utility function.

Definition 1. *Given an integer k , let $\{A_i\}_{i=1}^k$ be a collection of subsets of \mathcal{V} . The zero order approximate submodularity coefficient is the largest $c_k \in [0, 1]$ such that*

$$\sum_{i=1}^k f(A_i) \geq c_k \cdot [f(\cup_i A_i) + f(\cap_i A_i)].$$

For any $A, B \subseteq \mathcal{V}$, we set $f_A(B) = f(A \cup \{B\}) - f(A)$ as the amount of change by adding B to A . For the sake of brevity and readability, we set $f_A(u) = f(A + u) - f(A)$ for any singleton element $u \in \mathcal{V}$. Then we restate the submodularity ratio γ in the following definition. The submodularity ratio measures how close of f being submodular.

Definition 2 ([2, 3, 11]). *Given an integer k , the submodularity ratio of a non-negative set function f with respect to \mathcal{V} is*

$$\gamma_{\mathcal{V},k}(f) = \min_{A \subseteq \mathcal{V}, B: |B| \leq k, A \cap B = \emptyset} \frac{\sum_{u \in B} f_A(u)}{f_A(B)}.$$

Let k be the maximum size of any feasible solution, and omit signs k, \mathcal{V} and f for clarity. Bian et al. [2] introduced an equivalent formulation of submodularity ratio γ by the largest γ such that

$$\sum_{u \in B} f_A(u) \geq \gamma \cdot f_A(B), \forall A, B \subseteq \mathcal{V}, A \cap B = \emptyset.$$

Bogunovic et al. [3] defined *supmodularity ratio* $\tilde{\gamma}$ that measures how close a utility function is supermodular.

Definition 3 ([3]). *The supmodularity ratio of a non-negative set function f is the largest $\tilde{\gamma}$ such that*

$$f_A(B) \geq \tilde{\gamma} \cdot \sum_{u \in B} f_A(u), \forall A, B \subseteq \mathcal{V}, A \cap B = \emptyset.$$

For a monotone submodular function, Conforti and Cornuéjols [10] introduced the definition of *total curvature* κ_f and *curvature* $\kappa_f(S)$ w.r.t. a set $S \subseteq \mathcal{V}$ as follows. Denote $\kappa_f = 1 - \min_{u \in \mathcal{V}} f_{\mathcal{V} \setminus \{u\}}(u)/f(u)$, and $\kappa_f(S) = 1 - \min_{u \in S} f_{S \setminus \{u\}}(u)/f(u)$. Sviridenko et al. [24] provided a definition of curvature from submodular to non-submodular functions. The expanded *curvature* is defined as $\kappa^o = 1 - \min_{u \in \mathcal{V}} \min_{A, B \subseteq \mathcal{V} \setminus \{u\}} f_A(u)/f_B(u)$. Bian et al. [2] presented a more expressive formulation of curvature, which measures how close a set function is to being supmodular.

Definition 4 ([2]). *The generalized curvature of a non-negative function f is the smallest scalar α such that*

$$f_{A \setminus \{u\} \cup B}(u) \geq (1 - \alpha) \cdot f_{A \setminus \{u\}}(u), \forall A, B \subseteq \mathcal{V}, u \in A \setminus B.$$

Recently, Bogunovic et al. [3] introduced the concept of *inverse generalized curvature* $\check{\alpha}$, which can be described as follows:

Definition 5 ([3]). *The inverse generalized curvature of a non-negative function f is the smallest scalar $\check{\alpha}$ such that*

$$f_{A \setminus \{u\}}(u) \geq (1 - \check{\alpha}) f_{A \setminus \{u\} \cup B}(u), \forall A, B \subseteq \mathcal{V}, u \in A \setminus B.$$

The above parameters are used to characterize a non-negative set function from different points of view. We provide a lower bound of zero order approximate submodularity coefficient c by inverse generalized curvature $\check{\alpha}$. I.e., $c \geq 1 - \check{\alpha}$. The proof is referred to the full version. We omit the relation of the other parameters as they can be found in [2, 3]. In the rest of this part, we restate the concept of the p -independence system.

Let $\mathcal{I} = \{A_i\}_i$ be a finite collection of subsets chosen from \mathcal{V} . We say the tuple $(\mathcal{V}, \mathcal{I})$ is an *independence system* if for any $A \in \mathcal{I}$, $A' \subseteq A$ implies that $A' \in \mathcal{I}$. The sets of \mathcal{I} are called the *independent sets* of the independence system. An independent set B contained in a subset $X \subseteq \mathcal{V}$ is a *base* (basis) of X if no other independent set $A \subseteq X$ strictly contains B . By the above terminologies we restate the definition of p -independence system as follows.

Definition 6 ([15]). *An independence system $(\mathcal{V}, \mathcal{I})$ is a p -independence system if, for every subset $X \subseteq \mathcal{V}$ and for any two bases B_1, B_2 of X , we have $|B_1|/|B_2| \leq p$.*

In our model, assume the utility function is characterized by these parameters, and the collection of all feasible subsets constructs a p -independence system. We also assume that there exist a utility function value oracle and an independence oracle; i.e., for any $A \subseteq \mathcal{V}$, we can obtain the value of $f(A)$ and know if A in \mathcal{I} or not. The model can be described as follows:

$$OPT \leftarrow \arg \max_{S \subseteq \mathcal{V}, S \in \mathcal{I}} f(S), \tag{1}$$

where $(\mathcal{V}, \mathcal{I})$ is a p -independence system.

4 Algorithms

In this section, we present some algorithms in dealing with non-submodular maximization. Before providing our main algorithm, we first investigate the efficiency of two sub-algorithms. In Subsect. 4.1, we restate the greedy algorithm for submodular maximization with p -independence system constraint, and show that some good properties are still retained in the non-submodular setting. In Subsect. 4.2, we present a local search for non-monotone non-submodular maximization without any constraint. Finally, we provide the core algorithm in Subsect. 4.3.

4.1 Greedy Algorithm Applied to Non-submodular Optimization

The pseudo codes of the greedy algorithm are presented in Algorithm 1. Let $S^G = \{u_1, \dots, u_\ell\}$ be the returned set by Algorithm 1. We start with $S^G = \emptyset$. In each iteration, we choose the element u with maximum gain, and add it to the current solution if it satisfies $S^G + u \in \mathcal{I}$. For clarity, we let OPT be any optimum solution set of maximizing the utility function under p -independence system constraint. Then we can derive a lower bound of $f(S^G)$ by the following theorem.

Theorem 1. *Let S^G be the returned set of Algorithm 1, then we have*

$$f(OPT \cup S^G) \leq \left(\frac{p}{\gamma^2 \tilde{\gamma}(1 - \tilde{\alpha})} + 1 \right) f(S^G).$$

Proof. Refer to the full version of this paper.

Algorithm 1. Greedy($\mathcal{V}, f, \mathcal{I}$)

- 1: $S^G \leftarrow \emptyset, A \leftarrow \emptyset$
 - 2: **repeat**
 - 3: $A \leftarrow \{e | S^G \cup \{e\} \in \mathcal{I}\}$
 - 4: **if** $A \neq \emptyset$ **then**
 - 5: $e \leftarrow \arg \max_{e' \in A} f_{S^G}(e')$
 - 6: $S^G \leftarrow S^G + e$
 - 7: **end if**
 - 8: **until** $A = \emptyset$
 - 9: Return S^G
-

Let $B \in \mathcal{I}$ be any independent set and set $S_i^G = \{u_1, \dots, u_i\}$ be the set of the first i elements added by Algorithm 1. We can iteratively construct a partition of B according to S^G . We start with $B_0 = B$ and set $B_i = \{u \in B \setminus S_i^G | S_i^G + u \in \mathcal{I}\}$ for iteration $i \in [\ell] = \{1, \dots, \ell\}$, where ℓ denotes the size of S^G in the end. Then the collection of $\{B_{i-1} \setminus B_i\}_{i=1}^\ell$ derives a partition of B . Let $C_i = B_{i-1} \setminus B_i$ for any $i \in [\ell]$. The construction can be summarized as Algorithm 2. The properties of the above partition are presented in the following lemma.

Algorithm 2. Construct($\mathcal{V}, f, \mathcal{I}$)

```

1:  $S_0 \leftarrow \emptyset, B_0 \leftarrow B$ 
2: for  $i = 1 : \ell$  do
3:    $B_i \leftarrow \{u \in B \setminus S_i \mid S_i + u \in \mathcal{I}\}$ 
4: end for
5: Return  $\{B_{i-1} \setminus B_i\}_{i=1}^\ell$ 

```

Lemma 1. Let $\{C_i\}_{i=1}^\ell$ be the returned partition of Algorithm 2, then

- for each $i \in [\ell]$, we have $\sum_{j=1}^i p_j \leq p \cdot i$ where $p_j = |C_j|$; and
- for each $i \in [\ell]$, we have $p_i \cdot \delta_i \geq \gamma(1 - \check{\alpha})f_{S^c}(C_i)$.

Proof. Refer to the full version of this paper.

4.2 Local Search Applied to Non-submodular Optimization

In this subsection, we present a local search algorithm for the non-monotone non-submodular maximization problem without any constraint. The main pseudo codes are provided by Algorithm 3. Feige et al. [14] introduced the local search approach to deal with the non-monotone submodular optimization problem. We extend their algorithm to the non-submodular setting, and show that the algorithm still keeps a near constant approximation ratio by increasing a factor.

In order to implement our algorithm in polynomial time, we relax the local search approach and find an approximate local solution. Let S^{LS} be the returned set of Algorithm 3 and let OPT^o be any optimum solution without any constraint. We restate the definition of approximate local optimum as follows.

Definition 7 ([14]). Given a set function $f : 2^{\mathcal{V}} \rightarrow R$, a set $A \subseteq \mathcal{V}$ is called a $(1 + \lambda)$ -approximate local optimum if, $f(A - u) \leq (1 + \lambda) \cdot f(A)$ for any $u \in A$ and $f(A + u) \leq (1 + \lambda) \cdot f(A)$ for any $u \notin A$.

By the definition of the approximate local optimum solution, we show that there exists a similar performance guarantee in the non-submodular case. The details are summarized in the following theorem.

Theorem 2. Given $\epsilon > 0, c$ and $\check{\alpha} \in [0, 1)$. Let S^{LS} be the returned set of Algorithm 3 by setting set $\lambda = \frac{c^2 \epsilon}{(1 - \check{\alpha})n}$. We have

$$f(OPT^o) \leq \left(\frac{3}{c^2} + \epsilon \right) \cdot f(S^{LS}).$$

Proof. Refer to the full version of this paper.

Before proving the above theorem, we need the following lemma.

Lemma 2. If S is a $(1 + \lambda)$ -approximate local optimum for a non-submodular function f , then for any set T such that $T \subseteq S$ or $T \supseteq S$, we have

$$f(T) \leq [1 + \lambda n(1 - \check{\alpha})] \cdot f(S).$$

Algorithm 3. Local Search(\mathcal{V}, f, λ)

```

1:  $S \leftarrow \arg \max_{u \in \mathcal{V}} f(u), \mathcal{U} \leftarrow \mathcal{V}$ 
2: repeat
3:   if there exists  $u \in \mathcal{U} \setminus S$  such that  $f(S + u) \geq (1 + \lambda)f(S)$  then
4:      $S \leftarrow S + u$ 
5:      $\mathcal{U} \leftarrow \mathcal{U} - u$ 
6:   end if
7: until
8: if there exists  $u \in S$  such that  $f(S - u) \geq (1 + \lambda)f(S)$  then
9:    $S \leftarrow S - u$ , and go back to Repeat loop.
10: end if
11: Return  $S^{LS} \leftarrow \arg \max\{f(S), f(\mathcal{V} \setminus S)\}$ 

```

Proof. Let $S = \{u_1, \dots, u_q\}$ be a $(1 + \lambda)$ -approximate local optimum solution returned by Algorithm 3. W.l.o.g., we assume $T \subseteq S$, then we construct T_i such that $T = T_1 \subseteq T_2 \subseteq \dots \subseteq T_r = S$ and $u_i = T_i \setminus T_{i-1}$. For each $i \in \{2, \dots, q\}$, we have

$$f(T_i) - f(T_{i-1}) \geq (1 - \check{\alpha})(f(S) - f(S - u_i)) \geq -\lambda(1 - \check{\alpha})f(S),$$

where the first inequality follows by the definition of the inverse generalized curvature and the second inequality follows by the definition of the approximate local optimum. Summing up the above inequalities, we have

$$f(S) - f(T) = \sum_{i=2}^q [f(T_i) - f(T_{i-1})] \geq -\lambda q(1 - \check{\alpha})f(S),$$

implying that $f(T) \leq [1 + \lambda q(1 - \check{\alpha})]f(S) \leq [1 + \lambda n(1 - \check{\alpha})]f(S)$, where the second inequality follows from $q \leq n$. Simultaneously, the case of $T \supseteq S$ can be similarly derived by the above process.

4.3 The Core Algorithm

In this subsection, we present the main algorithm, which is an extension of the Repeat-Greedy algorithm introduced in [15]. The pseudo codes are presented as Algorithm 4. We run the main algorithm in r rounds. Let \mathcal{V}_i be the set of candidate elements set at the start of round $i \in [r]$. We first run the greedy step of Algorithm 1. Then, we proceed with the local search step of Algorithm 3 on the set returned from the first step. Simultaneously, we update the candidate ground set as $\mathcal{V}_i = \mathcal{V} \setminus \mathcal{V}_{i-1}$. Finally, we output the best solution among all returned sets. We can directly obtain two estimations of the utility function by Theorems 1 and 2, respectively. The results are summarized as follows.

Lemma 3. For any iteration $i \in [r]$ of Algorithm 4, we have

1. $f(S_i \cup (OPT \cap \mathcal{V}_i)) \leq \left(\frac{p}{\gamma^{2\check{\gamma}}(1-\check{\alpha})} + 1\right) f(S_i)$, and
2. $f(S_i \cap OPT) \leq \left(\frac{3}{c^2} + \epsilon\right) f(S'_i)$.

Buchbinder et al. [6] derived an interesting property in dealing with non-monotone submodular optimization problems. Now, we extend this property to the non-submodular case, as summarized in the following lemma.

Lemma 4. *Let $g : 2^{\mathcal{V}} \rightarrow R_+$ be a set function with inverse generalized curvature $\check{\alpha}_g$, and S be a random subset of \mathcal{V} where each element appears with probability at most pr (not necessarily independent). Then $\mathbf{E}[g(S)] \geq [1 - (1 - \check{\alpha}_g)\text{pr}] \cdot g(\emptyset)$.*

Proof. Refer to the full version of this paper.

Using this result, we can derive an estimation of $f(OPT)$. Let S be a random set of $\{S_i\}_{i=1}^r$ with probability $\text{pr} = \frac{1}{r}$ and set $g(S) = f(OPT \cup S)$ for any $S \subseteq \mathcal{V}$. Then we have $\check{\alpha} = \check{\alpha}_f = \check{\alpha}_g$. By Lemma 4, we yield

$$\begin{aligned} \frac{1}{r} \sum_{i=1}^r f(S_i \cup OPT) &= \mathbf{E}[f(S \cup OPT)] = \mathbf{E}[g(S)] \geq [1 - (1 - \check{\alpha}_g)\text{pr}] \cdot g(\emptyset) \\ &= \left(1 - \frac{1 - \check{\alpha}}{r}\right) \cdot f(OPT). \end{aligned}$$

Multiplying both sides of the last inequality by r , we get

$$\sum_{i=1}^r f(S_i \cup OPT) \geq [r - (1 - \check{\alpha})] \cdot f(OPT). \tag{2}$$

The following lemma presents a property based on zero order approximate submodularity coefficient of the objective function.

Lemma 5 ([15]). *For any subsets $A, B, C \subseteq \mathcal{V}$, we have*

$$f(A \cup B) \leq \frac{1}{c} \cdot [f(A \cup (B \cap C)) + f(B \setminus C)].$$

Algorithm 4. Repeat Greedy($\mathcal{V}, f, \mathcal{I}, r$)

- 1: $i = 1, \mathcal{V}_1 \leftarrow \mathcal{V}$
 - 2: **repeat**
 - 3: $S_i \leftarrow$ output of Greedy($\mathcal{V}_i, f, \mathcal{I}$)
 - 4: $S'_i \leftarrow$ output of Local Search(S_i, f, λ_i)
 - 5: $\mathcal{V}_i \leftarrow \mathcal{V}_i \setminus S_i$
 - 6: $i \leftarrow i + 1$
 - 7: **until** $i = r$
 - 8: Return $S \leftarrow \arg \max\{f(S_i), f(S'_i)\}_{i \in [r]}$
-

Proof. To prove this lemma, we have the following

$$\begin{aligned} f(A \cup B) &= f(A \cup (B \cap C) \cup (B \setminus C)) \\ &\leq f(A \cup (B \cap C) \cup (B \setminus C)) + f(A \cup (B \cap C) \cap (B \setminus C)) \\ &\leq \frac{1}{c} \cdot [f(A \cup (B \cap C)) + f(B \setminus C)], \end{aligned}$$

where the first inequality follows from the nonnegativity of the objective function and the second inequality is derived by the definition of the zero order approximate submodularity coefficient c .

From these lemmas and choosing properly the number of rounds, we conclude that if the parameters of the utility function are fixed, or have a food estimation, then Algorithm 4 yields a near constant performance guarantee for problem (1). The details are presented in the following theorem.

Theorem 3. *Give an objective function $f : 2^V \rightarrow R_+$ with parameters $c, \gamma, \check{\gamma}, \check{\alpha}$, and a real number $\epsilon > 0$, let S be the returned set of Algorithm 4. Set $r = \lceil \Delta \rceil$. Then we have*

$$\frac{f(OPT)}{f(S)} \leq \left[\left(\frac{p}{\gamma^2 \check{\gamma} c (1 - \check{\alpha})} + \frac{3\Delta}{2c^4} + \frac{1}{c} \right) + \frac{\epsilon \Delta}{2c^4} \right] \cdot [1 - (1 - \check{\alpha}) \cdot (\Delta + 1)]^{-1},$$

where

$$\Delta = (1 - \check{\alpha}) + \sqrt{(1 - \check{\alpha})^2 + (1 - \check{\alpha}) \left(\frac{2c^3}{3 + c^2\epsilon} + 1 \right) + \frac{p}{\gamma^2 \check{\gamma}} \cdot \frac{2c^3}{3 + c^2\epsilon}}.$$

Proof. Refer to the full version of this paper.

5 Conclusion

We consider the non-submodular and non-monotonic maximization problem with a p -independence system constraint, where the objective utility function is characterized by a set of parameters such as submodularity (supmodularity) ratio, inverse generalized curvature, and zero order approximate submodularity coefficient. We study a greedy algorithm applied to non-submodular optimization with p -independence system constraint, and show the algorithm preserves some good properties even though the objective function is non-submodularity. Then, we investigate the unconstrained non-submodular maximization problem. Utilizing an approximate local search technique, we derive an $O(3/c^2 + \epsilon)$ -approximation algorithm, where c is the zero order approximate submodularity coefficient. Finally, combining these two algorithms, we obtain an almost constant approximation algorithm for the non-monotone non-submodular maximization problem with p -independence system constraint.

Acknowledgments. The first two authors are supported by Natural Science Foundation of China (Nos. 11531014, 11871081). The third author is supported by Natural Sciences and Engineering Research Council of Canada (No. 283106). The fourth author is supported by China Postdoctoral Science Foundation funded project (No. 2018M643233) and Natural Science Foundation of China (No. 61433012). The fifth author is supported by Natural Science Foundation of Shanxi province (No. 201801D121022).

References

1. Badanidiyuru, A., Mirzasoleiman, B., Karbasi, A., Krause, A.: Streaming submodular maximization: massive data summarization on the fly. In: 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 671–680. ACM (2014)
2. Bian, A.-A., Buhmann, J.-M., Krause, A., Tschitschek, S.: Guarantees for greedy maximization of non-submodular functions with applications. In: 34th International Conference on Machine Learning, pp. 498–507. JMLR (2017)
3. Bogunovic, I., Zhao, J., Cevher, V.: Robust maximization of non-submodular objectives. In: 21st International Conference on Artificial Intelligence and Statistics, pp. 890–899. Playa Blanca, Lanzarote (2018)
4. Buchbinder, N., Feldman, M.: Constrained submodular maximization via a non-symmetric technique [arXiv:1611.03253](https://arxiv.org/abs/1611.03253) (2016)
5. Buchbinder, N., Feldman, M.: Deterministic algorithms for submodular maximization problems. *ACM Trans. Algorithms* **14**(3), 32 (2018)
6. Buchbinder, N., Feldman, M., Naor, J.-S., Schwartz, R.: Submodular maximization with cardinality constraints. In: 25th Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 1433–1452. Society for Industrial and Applied Mathematics (2014)
7. Buchbinder, N., Feldman, M., Seffi, J., Schwartz, R.: A tight linear time 1/2-approximation for unconstrained submodular maximization. *SIAM J. Comput.* **44**(5), 1384–1402 (2015)
8. Calinescu, G., Chekuri, C., Pál, M., Vondrák, J.: Maximizing a monotone submodular function subject to a matroid constraint. *SIAM J. Comput.* **40**(6), 1740–1766 (2011)
9. Chen, W., Zhang, H.: Complete submodularity characterization in the comparative independent cascade model. *Theor. Comput. Sci.* (2018)
10. Conforti, M., Cornuéjols, G.: Submodular set functions, matroids and the greedy algorithm: tight worst-case bounds and some generalizations of the Rado-Edmonds theorem. *Discrete Appl. Math.* **7**(3), 251–274 (1984)
11. Das, A., Kempe, D.: Submodular meets spectral: greedy algorithms for subset selection, sparse approximation and dictionary selection. In: 28th International Conference on Machine Learning, pp. 1057–1064. Omnipress (2011)
12. El-Arini, K., Veda, G., Shahaf, D., Guestrin, C.: Turning down the noise in the blogosphere. In: 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 289–298. ACM (2009)
13. Feige, U.: A threshold of $\ln n$ for approximating set cover. *J. ACM* **45**(4), 634–652 (1998)
14. Feige, U., Mirrokni, V.-S., Vondrák, J.: Maximizing non-monotone submodular functions. *SIAM J. Comput.* **40**(4), 1133–1153 (2011)

15. Feldman, M., Harshaw, C., Karbasi, A.: Greed is good: near-optimal submodular maximization via greedy optimization. In: 30th Annual Conference on Learning Theory, pp. 758–784. Springer (2017)
16. Gomes, R., Krause, A.: Budgeted nonparametric learning from data streams. In: 27th International Conference on Machine Learning, pp. 391–398. Omnipress (2010)
17. Gupta, A., Roth, A., Schoenebeck, G., Talwar, K.: Constrained non-monotone submodular maximization: offline and secretary algorithms. In: Saberi, A. (ed.) WINE 2010. LNCS, vol. 6484, pp. 246–257. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-17572-5_20
18. Kempe, D., Kleinberg, J., Tardos, É.: Maximizing the spread of influence through a social network. In: 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 137–146. ACM (2003)
19. Lee, J., Sviridenko, M., Vondrák, J.: Submodular maximization over multiple matroids via generalized exchange properties. *Math. Oper. Res.* **35**(4), 795–806 (2010)
20. Lin, H., Bilmes, J.: A class of submodular functions for document summarization. In: 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, vol. 1, pp. 510–520. Association for Computational Linguistics (2011)
21. Mirzasoleiman, B., Badanidiyuru, A., Karbasi, A.: Fast constrained submodular maximization: personalized data summarization. In: 33rd International Conference on Machine Learning, pp. 1358–1366. JMLR (2016)
22. Nemhauser, G.-L., Wolsey, L.-A., Fisher, M.-L.: An analysis of approximations for maximizing submodular set functions–I. *Math. Program.* **14**(1), 265–294 (1978)
23. Sviridenko, M.: A note on maximizing a submodular set function subject to a knapsack constraint. *Oper. Res. Lett.* **32**(1), 41–43 (2004)
24. Sviridenko, M., Vondrák, J., Ward, J.: Optimal approximation for submodular and supermodular optimization with bounded curvature. *Math. Oper. Res.* **42**(4), 1197–1218 (2017)