An Adaptive Algorithm for Maximization of Non-submodular Function with a Matroid Constraint

Xin Sun[1], Dachuan Xu[1], Dongmei Zhang[2], and Yang Zhou[3(✉)]

[1] Department of Operations Research and Information Engineering, Beijing University of Technology, Beijing 100124, People's Republic of China
athossun@emails.bjut.edu.cn, xudc@bjut.edu.cn

[2] School of Computer Science and Technology, Shandong Jianzhu University, Jinan 250101, People's Republic of China
zhangdongmei@sdjzu.edu.cn

[3] School of Mathematics and Statistics, Shandong Normal University, Jinan 250014, People's Republic of China
zhyg1212@163.com

Abstract. In this paper, we consider the problem of maximizing a non-submodular set function subject to a matroid constraint with the continuous generic submodularity ratio γ. It quantifies how close a monotone function is to being submodular. As our main contribution, we propose a $(1 - e^{-\gamma^2} - O(\varepsilon))$-approximation algorithm when the submodularity ratio is sufficiently large. Our work also can be seen as the first extension of the adaptive sequencing technique in non-submodular case.

Keywords: Non-submodular optimization · Matroid constraint · Submodularity ratio · Adaptive sequencing

1 Introduction

As a classical problem in submodular optimization, maximization of a monotone submodular function subject to a single matroid constraint has been intensively studied in recent years. It is well-known that the standard greedy algorithm [15] gave a 1/2-approximation ratio for this problem. In previous works, Feige [13] showed that there exists no polynomial-time algorithm with an approximation ratio better than $(1 - e^{-1})$. Also, Nemhauser and Wolsely [20] showed that any improvement over $(1 - e^{-1})$ has to make sacrifice for an exponential number of queries to the value oracle. In recent years, Calinescu et al. [6,22] presented a randomized continuous greedy technique based $(1 - e^{-1})$-approximation algorithm for the above problem. This is a breakthrough result that perfectly matches the conjecture given by Feige [13] and it also lays a solid foundation for the technique used in this paper. This approach resembles a common paradigm for designing approximation algorithms and is composed of two steps. In the first step, a fractional solution is found for a relaxation of the problem. In the second step, the

S. Chellappan et al. (Eds.): CSoNet 2020, LNCS 12575, pp. 3–13, 2020.
https://doi.org/10.1007/978-3-030-66046-8_1

fractional solution is rounded to obtain an integral one while incurring only a small loss in the objective. This approach has been used to obtain improved approximations to this problem with various cases, including monotone [6], non-monotone [14] and recently in differencial privacy [21]. Another excellent work of this problem is done by Filmus and Ward [16]. It gave a non-oblivious local search algorithm with an approximation ratio of $(1 - e^{-1})$ utilizing the combinatorial structure of this problem. Lately, Buchbinder et al. [5] made a step forward from deterministic perspective. They gave an 0.5008-approximation algorithm based on greedy-like technique.

However, for many applications in practice, including subset selection [1], experimental design and sparse Gaussian processes [19], the corresponding set function is close to submodular, but not strictly submodular [18]. Naturally the results in submodular function setting do not hold any more. To depict the difference between submodular and non-submodular, a crucial parameter should be introduced to describe the characteristics of the non-submodular functions. Das and Kempe [11] proposed the submodularity ratio, $\hat{\gamma} = \min_{\Omega, S \subseteq N} \frac{\sum_{j \in \Omega \setminus S} f_S(j)}{f_S(\Omega)}$. It is a quantity characterizing how close a set function is to being submodular.

In this context, Bian et al. [4] showed that, under a cardinality constraint, the standard greedy algorithm enjoys an approximation factor of $(1-e^{-\hat{\gamma}})$, where $\hat{\gamma}$ is the submodularity ratio [11] of the set function. Very recently, Harshaw et al. [17] have also shown that there is no polynomial algorithm with better guarantees. Moreover, for the same problem subject to a general matroid constraint, it has only been studied very recently by Chen et al. [10], who offered a randomized version of the standard greedy algorithm with approximation guarantee of $(1 + 1/\hat{\gamma})^{-2}$.

1.1 Our Contribution

In this paper, we propose an approximation algorithm with low adaptive rounds for the problem of maximization of a monotone closely submodular function over matroid constraints. Our technique is based on the *adaptive sequencing* algorithm [3]. It is a powerful method and can be applied to several cases like cardinality constraint, non-monotone set functions, partition matroids and intersection of P matroids. Here, we first generalize this algorithm to the weakly submodular case.

Besides, we also reach the approximation guarantee with the help of *generic submodularity ratio* γ' and its continuous version. It is a quantity characterizing how close a nonnegative nondecreasing set function is to be submodular. Compared with $\hat{\gamma}$, it is derived from a different equivalent definition of submodular functions and has more flexible properties. Our main theorem is stated as following.

Theorem 1.1. *For any $\varepsilon > 0$, there is an $O\left(\frac{\log n \log \frac{k}{\varepsilon}}{\frac{1}{\varepsilon^2} \log n \log \frac{1}{\gamma} - \frac{1}{\varepsilon}\log(1-\varepsilon)}\right)$ adaptive algorithm that, with probability $1-o(1)$, obtains $(1-e^{-\gamma^2}-O(\varepsilon))$ approximation for maximizing a γ-weakly submodular function under matroid constraints when*

γ is near to 1, where n and k are the ground set size and the matroid rank respectively.

1.2 Technical Overview

The result is inspired by the brilliant work of Balkanski et al. [3] and it can naturally reduce to their primal conclusions when the set function is strictly submodular. They obtained a $(1 - e^{-1} - O(\varepsilon))$-approximation with only requiring $O(\log n \log k)$ adaptive rounds for maximization of a monotone submodular set function over a matroid constraint. When designing a new algorithm for the γ-weakly submodular case, we extend the adaptive sequencing technique. The crucial challenge is making carefully adjustment thanks to the continuous generic submodularity ratio γ such that all selected elements in every step in the algorithm own nearly optimal marginal contribution and satisfy the feasibility constraints.

1.3 Organization

The remainder of the paper is organized as below: Sect. 2 gives preliminary definitions of the paper; Sect. 3 presents the new adaptive algorithm for non-submodular model; Sect. 4 shows the analysis of the algorithms; Sect. 5 finally concludes the paper. In addition, the formal proofs are omitted due to the length limitation but are nevertheless given in the appendix.

2 Preliminaries

This section gives the formal definition of the terms and notations used in the paper. We define the set function $f : 2^N \to \mathbb{R}$ on a *ground set* $N = [n]$, which is *non-decreasing*. Moreover, we say f is *monotone* if $f(S) \leq f(T)$ whenever $S \subseteq T \subseteq N$. Given such a function, the *marginal profit* of adding an element $j \in N$ to S is defined by $f_S(\{j\}) \doteq f(S \cup \{j\}) - f(S)$. For simplicity, we abbreviate $f_S(\{j\})$ and $f(S \cup \{j\})$ as $f_S(j)$ and $f(S \cup j)$, respectively.

Informally, the adaptivity of an algorithm is the number of sequential rounds of queries it makes, where every round allows for polynomially-many parallel queries. We present the formal definition here.

Definition 2.1. *Given a value oracle f, an algorithm is* ***r-adaptive*** *if every query $f(S)$ for the value of a set S occurs at a round $i \in [r]$ s.t. S is independent of the values $f(S')$ of all other queries at round i.*

Also, we consider f is *weakly submodular* characterized by *generic submodularity ratio* γ'. The generic submodularity ratio of f is the largest scalar γ' such that for any subset $S \subseteq T \subseteq N$ and any element $j \in N \backslash T$, $f_S(j) \geq \gamma' \cdot f_T(j)$. It measures how close a non-negative increasing set function is to be submodular. For generic submodularity ratio γ' we have the following properties.

Proposition 2.1. *For an increasing set function $f : 2^N \to \mathbb{R}_{\geq 0}$ with generic submodularity ratio γ', it holds that*

(a) $\gamma' \in (0, 1]$;
(b) $f(\cdot)$ *is submodular* $\iff \gamma' = 1$;
(c) $\sum_{j \in T \setminus S} f_S(j) \geq \gamma' \cdot f_S(T)$, *for any set* $S, T \subseteq N$.

A pair $\mathcal{M} = (N, \mathcal{I})$ is called a *matroid* w.r.t. a ground set N, if and only if the independence system $\mathcal{I}$ is a non-empty collection of subsets of N satisfying the following properties:

(i) If $S \subseteq T \subseteq N$ and $T \in \mathcal{I}$, then $S \in \mathcal{I}$;
(ii) If $S, T \in \mathcal{I}$ and $|S| < |T|$, then there is an element $j \in T \setminus S$ such that $S + j \in \mathcal{I}$.

For a matroid $\mathcal{M} = (N, \mathcal{I})$ [12], a subset S of N is called *independent* if and only if S belongs to $\mathcal{I}$. The common size of all maximal independent subset is called the *rank* of $\mathcal{M}$ and denoted by $r(\mathcal{M})$. Also, we assume that the algorithms have access to matroids only through an independence oracle that for a given set $S \subseteq N$ answers whether S is independent or not. The matroid polytope $\mathcal{P}(\mathcal{M})$ [12] is the collection of points $\mathbf{x} = (x_1, \ldots, x_n) \in [0, 1]^n$ in the convex hull of the independent sets of $\mathcal{M}$, or equivalently the points $\mathbf{x}$ such that $\sum_{i \in A} x_i \leq r(\mathcal{M})$ for all $S \in N$.

The multilinear extension of a set function f is defined as $F : [0, 1]^n \to \mathbb{R}_{\geq 0}$, which maps a point $\mathbf{x} \in [0, 1]^n$ to the expected value of a random set $R \sim \mathbf{x}$ containing each element $j \in [n]$ with probability x_j independently, i.e. $F(\mathbf{x}) = \mathbb{E}_{R \sim \mathbf{x}}[f(R)]$. We note that given an oracle for f, one can estimate $F(\mathbf{x})$ arbitrarily well in one round by querying in parallel a sufficiently large number of samples $R_1, \ldots, R_m$ draw i.i.d from $\mathbf{x}$ and taking the average value of $f(R_i)$ over $i \in [m]$ [7,8]. For ease of presentation, we assume throughout the paper that we are given access to an exact value oracle for F in addition to f. The results which rely on F then extend to the case where the algorithm is only given an oracle for f with an arbitrarily small loss in the approximation, no loss in the adaptivity, and additional $O(n \log n)$ factor in the query complexity With $O(2n \log n)$ samples, $F(\mathbf{x})$ is estimated within a $(1 \pm \epsilon)$ multiplicative factor with high probability [8].

Besides, we also define *continuous generic submodularity ratio*, which is an extended version of generic submodularity ratio. It is more flexible in the analysis of the multilinear extension.

Definition 2.2 (Continuous generic submodularity ratio). *Given any normalized set function f, the continuous generic submodular ratio is defined as the largest scalar $\gamma \in [0, 1]$ subject to*

$$F_{\mathbf{x} \setminus i}(i) \geq \gamma F_{\mathbf{y} \setminus i}(i), \ \mathbf{x} \leq \mathbf{y}.$$

It is obvious that $\gamma \leq \gamma'$ by comparing their definitions.

In this paper, we are interested in the problem of maximizing a weakly submodular function $f : 2^N \to \mathbb{R}_{\geq 0}$ subject to a matroid $\mathcal{M} = (N, \mathcal{I})$ constraint in an adaptive model. The value of the optimal solution O for this problem is denoted by $\texttt{OPT}$, i.e. $O := \arg\max_{A \in \mathcal{M}} f(A)$ and $\texttt{OPT} := f(O)$.

Definition 2.3 (Chernoff bound [2]). *Let X_i, $i = 1, \ldots, k$, be mutually independent random variables such that $\mathbb{E}[X_i] = 0$ and $|X_i| \leq 1$ for any i. Set $S = X_1 + \ldots + X_k$ and denote by a positive real number. Then*

$$\Pr[|S| > a] \leq 2e^{-a^2/2k}.$$

3 The Adaptive Algorithm for Non-submodular Set Function

In this section we show the new adaptive algorithm in the case of weakly-submodular set function maximization problem over a matroid constraint. This algorithm is used as a subroutine in the main algorithm which achieves an approximation arbitrarily close to $1 - e^{\gamma^2}$ with $O(\log(n)\log(k))$ adaptivity when the continuous generic submodularity ratio γ is sufficiently large. It points out an update direction $\mathbf{1}_S$ for the current continuous solution. Comparing with [3], the procedure of locating this direction S is more complicated. The cause is how to make sure that all chosen elements in every rounds preserve nearly optimal marginal contribution and fulfill the matroid constraints.

For the convenience of readers, we restate an important definition which defined a random generalized feasible elements set for matroid $\mathcal{M}$.

Definition 3.1 (Random Feasible Sequence [3]). *Given a matroid $\mathcal{M}$ we say $(a_1, \ldots, a_{r(\mathcal{M})})$ is a* ***random feasible sequence*** *if for all $i \in [r(\mathcal{M})]$, a_i is an element chosen u.a.r. from $\{a : \{a_1, \ldots, a_{i-1}, a\} \in \mathcal{M}\}$.*

And, we put the technical algorithm in [3] below for producing a random sequence of elements defined above without any adaptive cost.

Algorithm 1. Random Sequence

Input: matroid $\mathcal{M}$
Output: $a_1, \ldots, a_{r(\mathcal{M})}$

1: **for** $i = 1$ to $r(\mathcal{M})$ **do**
2: $\quad X \leftarrow \{a : \{a_1, \ldots, a_{i-1}, a\} \in \mathcal{M}\}$
3: $\quad a_i \sim a$ uniformly random element from X
4: **end for**

Algorithm 1 selects an element in each iteration uniformly at random on the condition that the output set constructed by all elements is feasible. Also, it is noticeable that the adaptivity of this algorithm is zero since the generated elements set is irrelevant with set function f.

Algorithm 2. Adaptive Sequencing for Non-submodular Function

Input: function f, feasibility constraint $\mathcal{M}$
Output: S

$S \leftarrow \emptyset$, $t \leftarrow \max_{a \in N} f(a)$
for Δ iterations **do**
  $X \leftarrow N$
  while $X \neq \emptyset$ **do**
    $a_1, \ldots, a_{r(\mathcal{M}(S,X))} \leftarrow$ Random Sequence$(\mathcal{M}(S, X))$
    $X_i \leftarrow \{a \in X : S \cup \{a_1, \ldots, a_i, a\} \in \mathcal{M}$ and $f_{S \cup \{a_1,\ldots,a_i\}} \geq t\}$
    $i^\star \leftarrow \min\{i : |X_i| \leq (1-\varepsilon)|X|\}$
    $S \leftarrow S \cup \{a_i, \ldots, a_{i^\star}\}$
    $X \leftarrow X_{i^\star}$
    if $X \neq \emptyset$ **then**
      $t \rightarrow t/\gamma'$
    end if
  end while
  $t \leftarrow (1-\varepsilon)t$
end for

The fact that the sequence is randomly generated is very important. It ensures that the final conclusion of this paper is established with a high probability.

Similar with the adaptive sequencing algorithm, the new adaptive algorithm for non-submodular model utilizes the random feasible sequence produced by Algorithm 1 in each adaptive loop. It verifies which fragment of the sequence ought to be inserted to the solution and abandons the rest part of the output automatically. The algorithm starts from a empty set and selects part of random sequence so as to allocate the chosen elements into the current solution. All sequence components are invented from a specific set, which is conveyed from last iteration. We describe an element is *good* if the following two critical conditions can be satisfied. One is appending an element a to the current solution and a portion of the sequence w.r.t. an index i meets the matroid constraint, i.e. $S \cup \{a_1, \ldots, a_i\} \cup a \in \mathcal{M}$; the other is the marginal contribution of the fresh set above is beyond the threshold. After constructing all good set regarding to entire index i from 0 to n, we need to find a suitable location $i^\star$ which ensure the number of the remaining good elements set X in the next round is over $1-\varepsilon$ and the rest elements are thrown away certainly. This discarding guarantees that there are at most logarithmically many iterations until X is empty. As a result, the algorithm adds $\{a_1, \ldots, a_{i^\star}\}$ to S when we finish this iteration. In fact, the reason that adding those elements to the current solution S is the marginal profit of anyone in the set is nearly optimal in expectation. We will characterize this result in the analysis.

A conspicuous thing in weakly-submodular setting is that the threshold t is sluggish growth instead of maintaining unchanged in the inner loop. This means that the threshold no longer drops monotonically throughout the execution of the algorithm. The reason is the threshold value needs to arbitrarily close to the optional marginal contribution all the time and it can not be without submodularity. Therefore we need to give a modification to the algorithm by the aid of submodularity ratio γ' after elements appending. This trick makes adaptive sequencing algorithm possible in the problem of weakly submodular.

Additionally, the term $\mathcal{M}(S, X) := \{T \subseteq X : S \cup T \in \mathcal{M}\}$ in Algorithm 2 also denotes a matroid related to sets S and X. A subset of X is feasible in the new matroid if its union with S is feasible in $\mathcal{M}$.

Now the main algorithm can be unveiled on the stage naturally. Similar with the standard continuous greedy algorithm [22], the accelerated continuous greedy algorithm follows a guidance of an output update direction $\mathbf{1}_S \in \mathcal{M}$ too. What makes this algorithm "accelerated" is the manner of how to choose and use the direction. The solution $\mathbf{x} \in [0,1]^n$ moves along $\mathbf{1}_S$ given by Algorithm 2 in a measurement of the surrogate function g. The function g can be seen as the marginal profit value of the multilinear extension when the solution $\mathbf{x}$ marches a step size λ in the direction $\mathbf{1}_S$. That is, $g(S) := F_{\mathbf{x}}(\lambda \mathbf{1}_S) = F(\mathbf{x} + \lambda \mathbf{1}_S) - F(\mathbf{x})$, where S actually means $\mathbf{1}_S$. In this way, the continuous solution can be improved in a constant step size and returned also in a constant rounds. In our setting of non-submodular set function, the adaptivity of each round is the same comparing with [3], i.e. $O(\log(n)\log(k))$, which is much faster than the linear-time required by the standard continuous greedy algorithm. Algorithm 3 finally yields a continuous solution whose approximate ratio is with high probability arbitrarily close to $1 - e^{\gamma^2}$. Then, the technique of dependent rounding [9] or contention resolution schemes [23] can help it reduce to a feasible discrete solution almost without any loss in approximation guarantee and any additional burden in adaptivity.

Algorithm 3. Accelerated Continuous Greedy

Input: matroid $\mathcal{M}$, step size λ
Output: $\mathbf{x}$
1: **for** $1/\lambda$ iterations **do**
2: define $g : 2^N \to \mathbb{R}$ to be $g(T) = F_{\mathbf{x}}(\lambda T)$
3: $S \leftarrow$ Adaptive Sequencing For Non-submodular $(g, \mathcal{M})$
4: $\mathbf{x} \leftarrow \mathbf{x} + \lambda S$
5: **end for**

4 Analysis

Using the above denotations, we can now give the formal analysis for Algorithm 2 and Algorithm 3. The following lemma proves that at any time the threshold t in Adaptive Sequencing For Non-submodular is a good imitator to the optimal marginal profit of the current solution S.

Lemma 4.1. *Assume that f is weakly submodular with submodularity ratio γ'. Then, at any iteration of the algorithm, the lower bound of the threshold value is close to the near-optimal marginal profit of the current solution, i.e.* $t \geq (1 - \varepsilon) \max_{a : S \cup a \in \mathcal{M}} f_S(a)$.

Now we prove the conclusion that for any element in the returned solution $S = \{a_1, \ldots, a_l\}$, the marginal profit of inserting a_i for $i \leq l$ to $\{a_1, \ldots, a_{i-1}\}$ is close to optimal in expectation comparing with all possible element a such that $\{a_1, \ldots, a_{i-1}, a\} \in \mathcal{M}$. Before given the next lemma, we denote that $X_i^{\mathcal{M}} := \{a \in X : S \cup \{a_1, \ldots, a_i\} \cup a \in \mathcal{M}\}$.

Lemma 4.2. *Assume that $a_1, \ldots, a_{r(\mathcal{M}(S,X))}$ is a generated random feasible sequence, then for all $i \leq i^*$, the expectation value of the marginal profits for each element is nearly optimal*

$$\mathbb{E}_{a_i}[f_{S\cup\{a_1,\ldots,a_{i-1}\}}(a_i)] \geq (1-\varepsilon)^2 \max_{a:S\cup\{a_1,\ldots,a_{i-1}\}\cup a\in\mathcal{M}} f_{S\cup\{a_1,\ldots,a_{i-1}\}}(a_i).$$

Then we give the analysis of the approximate ratio of Algorithm 2.

Lemma 4.3. *Assume that the output $S = \{a_1, \ldots, a_k\}$ of Algorithm 2 gives the result $\mathbb{E}_{a_i}[f_{S_i}(a_i)] \geq (1-\varepsilon)\max_{a:S_{i-1}\cup a\in\mathcal{M}} f_{S_{i-1}}(a)$ where $S_i = \{a_1, \ldots, a_i\}$. Then, for the expectation value of S, we have $\mathbb{E}[f(S)] \geq \left(1 - \frac{1}{1+(1-\varepsilon)\gamma'^2}\right)\mathtt{OPT}$.*

At the end, we give the analysis of the adaptivity of this algorithm.

Theorem 4.1. *With $\Delta = O\left(\frac{\log\frac{k}{\varepsilon}}{\frac{1}{\varepsilon}\log n \log\frac{1}{\gamma} - \log(1-\varepsilon)}\right)$,* Adaptive Sequencing For Non-submodular *has adaptivity $O\left(\frac{\log n \log\frac{k}{\varepsilon}}{\frac{1}{\varepsilon^2}\log n \log\frac{1}{\gamma} - \frac{1}{\varepsilon}\log(1-\varepsilon)}\right)$.*

Therefore, we have the following similar result of Adaptive Sequencing For Non-submodular and the proof can be adopted the same approach in [3]

Theorem 4.2. *For any $\varepsilon > 0$, Algorithm 2 is an $O\left(\frac{\log n \log\frac{k}{\varepsilon}}{\frac{1}{\varepsilon^2}\log n \log\frac{1}{\gamma} - \frac{1}{\varepsilon}\log(1-\varepsilon)}\right)$ adaptive algorithm that has $1 - \frac{1}{1+(1-\varepsilon)\gamma'^2}$ approximation guarantee with probability $1 - o(1)$ for maximizing a monotone weakly submodular function under a matroid constraint.*

In [3], they discussed the relationship between the value $g(S)$ and the residual value $\mathtt{OPT} - F(\mathbf{x})$ on the premise $F(\mathbf{x}) < (1-e^{-1})\mathtt{OPT}$. This bound is the tight approximation ratio for the problem of maximizing submodular function over matroid constraints. Then, they derived that $\mathtt{OPT} \leq e(\mathtt{OPT} - F(\mathbf{x}))$. Therefore, in Lemma 7 of [3] they can have such assumption of the direction value on the surrogate function and the residual value. However, for the problem of maximizing a weakly submodular set function subject to a matroid constraint, there is no such bound found yet. Instead, we assume $F(\mathbf{x}) \leq (1 - 1/\zeta)\cdot\mathtt{OPT}$ for weakly submodular case. Then

$$\mathtt{OPT} \leq \zeta \cdot (\mathtt{OPT} - F(\mathbf{x})).$$

So we could make assumption like [3] and have the following functional lemma, which concludes the sum of the whole marginal profits on g of the optimal elements to S is arbitrarily close to $\gamma\lambda(1-\lambda)\cdot(\mathtt{OPT} - \mathtt{F}(\mathbf{x}))$.

Lemma 4.4. *Assume that* $g(S) \le \lambda(\texttt{OPT} - F(\mathbf{x}))$*, then*

$$\sum_i g_{S\setminus O_{i:k}}(o_i) \ge \gamma\lambda(1-\lambda)\cdot(\texttt{OPT} - \texttt{F}(\mathbf{x})).$$

Combining Lemma 4.2, that all elements picked into the direction S have near-optimal marginal profits, with Lemma 4.4, we can obtain the following result. It characterizes the relationship between the expectation value of $g(S)$ and the residual value $\texttt{OPT} - F(\mathbf{x})$ in every iteration.

Lemma 4.5. *Let* $\Delta = O\left(\frac{\log\frac{k}{\varepsilon}}{\frac{1}{\varepsilon}\log n \log\frac{1}{\gamma} - \log(1-\varepsilon)}\right)$ *and* $\lambda = O(\varepsilon)$*. For any* $\mathbf{x}$ *s.t.* $\texttt{OPT} \le \zeta(\texttt{OPT} - F(\mathbf{x}))$*, the set* S *returned by* Adaptive Sequencing$(g, \mathcal{M})$ *has the following result when* γ *is near to* 1*:*

$$\mathbb{E}[F_{\mathbf{x}}(\lambda S)] \ge \left(\gamma^2 - O(\varepsilon)\zeta\right)\cdot\lambda(\texttt{OPT} - \texttt{F}(\mathbf{x})).$$

So far, we get the lower bound in expectation of the marginal contribution for current solution $\mathbf{x}$ of the main algorithm in the update direction S. The lower bound is portrayed by the residual value $\texttt{OPT} - F(\mathbf{x})$. Therefore, we can use inductive method in greedy-like algorithms to obtain the approximate ratio in expectation.

Lemma 4.6. *Assume that* Adaptive Sequencing For Non-submodular *outputs* $S \in \mathcal{M}$ *s.t.*

$$\mathbb{E}[F_{\mathbf{x}}(\lambda S)] \ge \Phi\cdot\lambda(\texttt{OPT} - F(\mathbf{x})),$$

where $\Phi = \gamma^2 - O(\varepsilon)\zeta$ *at every iteration of* Accelerated Continuous Greedy*. Then* Accelerated Continuous Greedy *outputs* $\mathbf{x} \in P(\mathcal{M})$ *s.t.*

$$\mathbb{E}[F(\mathbf{x})] \ge \left(1 - e^{-\Phi}\right)\texttt{OPT}.$$

At last we also need a technical lemma which exists for proving that the guarantee of Accelerated Continuous Greedy holds with high probability in the very end. The statement is below and it can be easily followed with the same idea in [3].

Lemma 4.7. *Assume that* Adaptive Sequencing For Non-submodular *outputs* $S \in \mathcal{M}$ *s.t.* $F_{\mathbf{x}}(\lambda S) \ge \alpha_i\lambda(\texttt{OPT} - F(\mathbf{x}))$ *at every iteration* i *of* Accelerated Continuous Greedy *and that* $\lambda\sum_{i=1}^{\lambda^{-1}}\alpha_i \ge \Phi$*, where* $\Phi = \gamma^2 - O(\varepsilon)\zeta$*. Then* Accelerated Continuous Greedy *outputs* $\mathbf{x} \in P(\mathcal{M})$ *s.t.* $F(\mathbf{x}) \ge \left(1 - e^{-\Phi}\right)\texttt{OPT}$*.*

Finally we can present the proof of Theorem 1.1.

Proof *(Proof of Theorem 1.1).* The adaptivity can be easily obtained due to Theorem 4.1. For the approximation result, from Lemma 4.5 we have $F_{\mathbf{x}}(\delta S) \ge$

$\alpha_i \lambda(\texttt{OPT} - F(\mathbf{x}))$ at every iteration i with $\mathbb{E}[\alpha_i] \geq \Phi$. By a Chernoff bound with $\mathbb{E}[\lambda \sum_{i \in [\lambda^{-1}]} \alpha_i] \geq \Phi$,

$$\Pr\left[\lambda \sum_{i \in [\lambda^{-1}]} \alpha_i < (1-\varepsilon)\Phi\right] \leq e^{-\varepsilon^2 \Phi \lambda^{-1}/2}.$$

Thus, with probability $p = 1 - e^{-\varepsilon^2 \Phi \lambda^{-1}/2}$, $\lambda^{-1}\alpha_i \geq \Phi - \varepsilon$. By Lemma 4.7, we conclude that w.p. p, $F(\mathbf{x} \geq (1 - e^{-\Phi}))\texttt{OPT}$. With step size $\lambda = O(\varepsilon^2/\log(1/\delta))$, we get that with probability $1 - \delta$,

$$F(\mathbf{x}) \geq \left(1 - e^{-\Phi}\right)\texttt{OPT} \geq \left((1 - e^{\gamma^2}) - O(\varepsilon)\right)\texttt{OPT},$$

where $\Phi = \gamma^2 - O(\varepsilon)\zeta$. □

5 Conclusions

In this paper, we first generalize the adaptive sequencing algorithm in the problem of maximizing a weakly submodular set function subject to a matroid constraint. This technique provides a continuous solution with $1 - e^{\gamma^2} - O(\varepsilon)$ approximation guarantee when the continuous generic submodularity ratio is sufficiently large. This result can be easily rounded to a feasible discrete solution almost without any loss by using either dependent rounding [9] or contention resolution schemes [23]. Besides, the generalized algorithm maintains few set function evaluations like [3] and the adaptivity is $O\left(\frac{\log n \log \frac{k}{\varepsilon}}{\frac{1}{\varepsilon^2}\log n \log \frac{1}{\gamma} - \frac{1}{\varepsilon}\log(1-\varepsilon)}\right)$.

Acknowledgements. The first and second authors are supported by Beijing Natural Science Foundation Project (No. Z200002) and National Natural Science Foundation of China (No. 11871081). The third author is supported by National Natural Science Foundation of China (No. 11871081). The fourth author is supported by Natural Science Foundation of Shandong Province of China (No. ZR2019PA004) and National Natural Science Foundation of China (No. 12001335).

References

1. Altschuler, J., Bhaskara, A., Fu, G., Mirrokni, V., Rostamizadeh, A., Zadimoghaddam, M.: Greedy column subset selection: new bounds and distributed algorithms. In: Proceedings of ICML, pp. 2539–2548 (2016)
2. Alon, N., Spencer, J.H.: The Probabilistic Method, vol. 3, pp. 307–314. Wiley, New York (2008)
3. Balkanski, E., Rubinstein, A., Singer, Y.: An optimal approximation for submodular maximization under a matroid constraint in the adaptive complexity model. In: Proceedings of STOC, pp. 66–77 (2019)
4. Bian, A.A., Buhmann, J.M., Krause, A., Tschiatschek, S.: Guarantees for greedy maximization of non-submodular functions with applications. In: Proceedings of ICML, pp. 498–507 (2017)

5. Buchbinder, N., Feldman, M., Garg, M.: Deterministic $(1/2+\varepsilon)$-approximation for submodular maximization over a matroid. In: Proceedings of SODA, pp. 241–254 (2019)
6. Călinescu, G., Chekuri, C., Pál, M., Vondrák, J.: Maximizing a monotone submodular function subject to a matroid constraint. SIAM J. Comput. **40**, 1740–1766 (2011)
7. Chekuri, C., Jayram, T.S., Vondrák, J.: On multiplicative weight updates for concave and submodular function maximization. In: Proceedings of ITCS, pp. 201–210 (2015)
8. Chekuri, C., Quanrud, K.: Submodular function maximization in parallel via the multilinear relaxation. In: Proceedings of SODA, pp. 303–322 (2019)
9. Chekuri, C., Vondrák, J., Zenklusen, R.: Dependent randomized rounding for matroid polytopes and applications. In: Proceedings of FOCS, pp. 575–584 (2010)
10. Chen, L., Feldman, M., Karbasi, A.: Weakly submodular maximization beyond cardinality constraints: does randomization help greedy? In: Proceedings of ICML, pp. 804–813 (2018)
11. Das, A., Kempe, D.: Submodular meets spectral: greedy algorithms for subset selection, sparse approximation and dictionary selection. In: Proceedings of ICML, pp. 1057–1064 (2011)
12. Edmonds, J.: Submodular functions, matroids, and certain polyhedra. Comb. Optim. **2570**, 11–26 (2003)
13. Feige, U.: A threshold of $\ln n$ for approximating set cover. J. ACM **45**, 634–652 (1998)
14. Feldman, M., Naor, J., Schwartz, R.: A unified continuous greedy algorithm for submodular maximization. In: Proceedings of FOCS, pp. 570–579 (2011)
15. Fisher, M.L., Nemhauser, G.L., Wolsey, L.A.: An analysis of approximations for maximizing submodular set functions-II. Math. Program. Stud. **8**, 73–87 (1978)
16. Filmus, Y., Ward, J.: Monotone submodular maximization over a matroid via non-oblivious local search. SIAM J. Comput. **43**, 514–542 (2014)
17. Harshaw, C., Feldman, M., Ward, J., Karbasi, A.: Submodular maximization beyond non-negativity: guarantees, fast algorithms, and applications. In: Proceedings of ICML, pp. 2634–2643 (2019)
18. Krause, A., Singh, A., Guestrin, C.: Nearoptimal sensor placements in Gaussian processes: theory, efficient algorithms and empirical studies. J. Mach. Learn. Res. **9**, 235–284 (2008)
19. Lawrence, N., Seeger, M., Herbrich, R.: Fast sparse Gaussian process methods: the informative vector machine. Adv. Neural Inf. Process. Syst. **1**, 625–632 (2003)
20. Nemhauser, G.L., Wolsey, L.A.: Best algorithms for approximating the maximum of a submodular set function. Math. Oper. Res. **3**, 177–188 (1978)
21. Rafiey, A., Yoshida, Y.: Fast and private submodular and k-submodular functions maximization with matroid constraints. arXiv:2006.15744 (2020)
22. Vondrák, J.: Optimal approximation for the submodular welfare problem in the value oracle model. In: Proceedings of STOC, pp. 67–74 (2008)
23. Vondrák, J., Chekuri, C., Zenklusen, R.: Submodular function maximization via the multilinear relaxation and contention resolution schemes. In: Proceedings of STOC, pp. 783–792 (2011)