TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 07/22 pp76–85
DOI: 10.26599/TST.2023.9010026
Volume 29, Number 1, February 2024

Multipass Streaming Algorithms for Regularized Submodular Maximization

Qingin Gong, Suixiang Gao, Fengmin Wang, and Ruiqi Yang*

Abstract: In this work, we study a k-Cardinality Constrained Regularized Submodular Maximization (k-CCRSM) problem, in which the objective utility is expressed as the difference between a non-negative submodular and a modular function. No multiplicative approximation algorithm exists for the regularized model, and most works have focused on designing weak approximation algorithms for this problem. In this study, we consider the k-CCRSM problem in a streaming fashion, wherein the elements are assumed to be visited individually and cannot be entirely stored in memory. We propose two multipass streaming algorithms with theoretical guarantees for the above problem, wherein submodular terms are monotonic and nonmonotonic.

Key words: submodular optimization; regularized model; streaming algorithms; threshold

1 Introduction

Submodular maximization has a lengthy history both in theoretical and applied aspects. It encompasses problems of interest in numerous applications, such as data summarization[2–4], influence maximization in social computing[5–7], sensor placement in environmental monitoring[8–10], system recommendation[11–13], and team formation[14–16], to a name a few.

Representative application data summarization is a fundamental task in machine learning that aims to find a diverse set of elements by maximizing a utility function formally described by submodularity.

The optimization problem involves finding a subset $S \in \mathcal{I}$ that maximizes $f(S)$ given a universe of finite elements of V, a submodular function $f : 2^V \to \mathbf{R}^+$, and k-cardinality constraint $\mathcal{I} = \{S \subseteq V : |S| \leqslant k\}$. The monotonicity of utility functions is encountered in a variety of scenarios and heavily affects algorithmic performances. A set function f is monotone if $f(A) \leqslant f(B)$ for any pair $A \subseteq B \subseteq V$. For the k-Cardinality Constrained Regularized Submodular Maximization (k-CCRSM) problem in an offline setting, a greedy-based $(1 - 1/\mathrm{e})$ approximation[17] exists for the monotonic case. Generally, the greedy-based algorithm chooses an element with the maximum marginal gain during each iteration. Interestingly, most of the designed algorithms for submodular optimizations are heavily based on non-negativity and monotonicity.

The monotonicity of the objective function encourages the revenue of adding elements, but it may lead to the risk of overfitting because adding elements can never decrease the utility values. The regularized utility function $f(\cdot) - c(\cdot)$ was introduced by Ref. [13]. In this function, $f(\cdot)$ denotes a submodular revenue term, and $c(\cdot)$ denotes a modular regular or penalty term. This formula forces us to maintain a balance between the submodular term and the regularized term

• Qinqin Gong and Ruiqi Yang are with Beijing Institute for Scientific and Engineering Computing, Beijing University of Technology, Beijing 100124. China. E-mail: gongqinqin@emails.bjut.edu.cn; yangruiqi@bjut.edu.cn.
• Suixiang Gao is with School of Mathematical Sciences, University of Chinese Academy Sciences, Beijing 100049, China. E-mail: sxgao@ucas.ac.cn.
• Fengmin Wang is with Beijing Jinghang Research Institute of Computing and Communication, Beijing 100074, China. E-mail: casic_wfm@163.com.
• A preliminary version of this paper has been presented at the 15th Ann. Int. Conf. on Combinatorial Optimization and Applications (COCOA 2021)[1].
∗ To whom correspondence should be addressed.
Manuscript received: 2022-03-10; revised: 2023-01-06; accepted: 2023-03-11

© The author(s) 2024. The articles published in this open access journal are distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/).

to select highly cost-effective elements. The regularized submodular maximization is formally defined as to find a subset $S \subseteq V$, such that the utility $f(S) - c(S)$ is maximized. We note that the regularized objective function loses the non-negativity and monotonicity, but retains the submodularity. Indeed, we can conclude that no multiplicative constant approximation factors exist in polynomial time for the discussed regularized problem. What follows is a brief summary of the meaningful weak approximation introduced by Refs. [18, 19].

Definition 1 An algorithm $\mathcal{A}$ is a weak ρ-approximation for the k-CCRSM problem, if it returns a solution S of size at most k in polynomial time, such that

$$f(S) - c(S) \geqslant \rho \cdot f(O) - c(O),$$

for some $\rho \in (0, 1)$, where O is an optimum solution for the discussed problem.

The approximation guarantees of $(1 - 1/\mathrm{e})$ and 0.491 have been shown to be tight with respect to the monotonic and nonmonotonic k-cardinality constrained submodular maximization problem in nonregularized offline settings. With the development of studies on the k-CCRSM problem, a state-of-the-art algorithm retains a weak $(1 - 1/\mathrm{e})$ approximation was found in Ref. [20]. Moreover, Ref. [21] developed an extended bicriteria approximation by further exploiting the regular model's combinatorial structure. We briefly restate this approximation here.

Definition 2 An algorithm $\mathcal{A}$ is a (ρ_1, ρ_2)-bicriteria approximation for the k-CCRSM problem, if it returns a solution S with a size of at most k in polynomial time, such that

$$f(S) - c(S) \geqslant \rho_1 \cdot f(O) - \rho_2 \cdot c(O),$$

for some pair $\rho_1, \rho_2 \in (0, 1)$.

Obviously, any weak ρ-approximation exactly reduces into a $(\rho, 1)$-bicriteria approximation in terms of the bicriteria approximation perspective.

Often, in an online model, we assume that elements are released once at a time. Once an element is released, we need to decide irrevocably whether or not to retain the element in the solution. Indeed, many applications in practical scenarios have motivated the need for space-efficient algorithms, i.e., streaming algorithms for the constrained submodular maximization problems because the entire elements cannot possible be accessed to in advance, and only a fraction of elements can be maintained in the main memory. In the formal streaming style, elements are released individually, and we need to extract a set of elements from the stream with a limited memory at any time during the streaming process, such that the optimizer restricted to the extracted set can be comparable with the optimizer over the entire stream. Elements are released in an arbitrary order, and the algorithm is usually allowed to visit the input stream only once. This algorithm is called a single pass algorithm. In the context of our regularized problem, the input stream consists of the elements are released in a fixed sequence, and the algorithm is allowed to visit the input stream many times.

Streaming-based optimization is now a very hot topic, and recent progresses has resulted in a good understanding of streaming algorithms for the submodular maximization in non-regularized scenarios. Threshold-based streaming algorithms with an optimal approximation guarantee of $1/2$ exist for streaming models[22–24]. The above approximation can be boosted to $(1 - 1/\mathrm{e})$ by making multipasses over the stream. The threshold-based method is a popular technique for dealing with the streaming scenario, in which one intuitively sieves arrived elements with a lower bound setting of the threshold value (which is usually dependent on the optimum for analysis). Other techniques have also been developed for the submodular maximization under streaming models, which can be found in Refs. [25–27]. We believe that these new techniques are of independent interest to streaming submodular optimization and should be further studied.

We consider the k-CCRSM problem in the streaming fashion and allow the element stream to be read numerous times. Assume that a weak $\tilde{\rho}$-approximation algorithm exists a priori for the discussed regularized problem. Given some parameter $\varepsilon \in (0, 1)$, we develop a multipass threshold based $(\tilde{\rho}/\lambda, 1/\lambda)$-bicriteria approximation, which makes over $O(\log(\lambda/\tilde{\rho})/\varepsilon)$ passes, but only consumes $O(k)$ memory and uses $O(n \log(\lambda/\tilde{\rho})/\varepsilon)$ queries, where the parameter λ can be optimized by $\tilde{\rho}$. The above results can also be found in our conference version[1]. In this version, we further study the k-CCRSM problem, wherein the submodular terms are monotonic and nonmonotonic. Indeed, we present two boosting multipass streaming algorithms for the k-CCRSM problem with some theoretical guarantees. Additionally, we assume that a weak ρ-approximation exists a priori for the discussed regularized problem. The improved results are summarized as follows:

- For any given $\varepsilon \in (0, 1)$, we obtain a deterministic multipass stream algorithm for the k-CCRSM problem in the monotonic scenario. The algorithm makes at most $O(\log(1/(\varepsilon\rho))/\varepsilon)$ passes, needs $O(n\log(1/(\varepsilon\rho))/\varepsilon)$ queries, consumes $O(k)$ memory, and produces a solution S satisfying $f(S)-\mathrm{e}^{-(1-\varepsilon)}c(S)\geqslant(1-\mathrm{e}^{-1}-O(\varepsilon))(f(O)-c(O))$, where O denotes any optimal solution of the stated problem. The results are formally summarized in Theorem 1 in Section 4 in the following.

• We further acquire a deterministic multipass stream algorithm for the k-CCRSM problem under the nonmonotonic scenario. The algorithm makes at most $O(\log((1-\varepsilon)^2/(2\varepsilon-\varepsilon^2))/\varepsilon)$ passes, needs $O(n\log((1-\varepsilon)^2/(2\varepsilon-\varepsilon^2))/\varepsilon)$ queries, consumes the same memory $O(k)$ as the monotonic case, and produces a solution S obeying $(f(S)-c(S))(2-\varepsilon)/(1-\varepsilon)\geqslant\min\{1/2-\varepsilon, 2(1-\varepsilon)\rho\}f(O)-2(1-\varepsilon)c(O)$ for some given accuracy parameter $\varepsilon\in(0,1/2)$. The main results are summarized in Theorem 2 in Section 5 in the following.

The remainder of this paper is structured as follows: In Section 2, we briefly introduce related works. In Section 3, we present some basic concepts of submodular functions and formally state the k-CCRSM problem. In Section 4, we discuss the setting of the k-CCRSM problem when the submodular term is monotonic. We present a threshold-based multipass streaming algorithm in Section 4.1, and discuss the analysis to obtain a weak approximation in Section 4.2. We then study the above problem in the non-monotonic case, as discussed in Section 5. We introduce an extended multipass streaming algorithm in Section 5.1 and provide the performance guarantees of the stated algorithm in Section 5.2. Finally, in Section 6, we give a conclusion to our work.

2 Related Work

In this section, we mainly review the studies in centralized and streaming settings that are most closely related to our work.

2.1 Centralized algorithms

Often, maximizing submodular functions with various constraints is computationally NP-hard. Hence, most previous works focused on approximation algorithms for such problems. Considering the k-CCSM problem in the nonregularity setting, the $(1-1/\mathrm{e})$ approximation and $1/\mathrm{e}$ approximation[28] existing with respect to the submodular term are monotonic and nonmonotonic, respectively. Submodular maximization problems with numerous kinds of constraints have been further studied, such as knapsack[19], matroid[29–31], and highly complex independence systems[32, 33]. The aforementioned studies investigated submodular maximization in nonregularity scenarios.

The k-CCRSM model has been recently formally introduced by Ref. [18]. They presented a distorted greedy-based algorithm with a weak approximation of $1-1/\mathrm{e}$, that is, $(1-1/\mathrm{e}, 1)$-bicriteria approximation. We let $\zeta(O)=f(O)/c(O)$ for convenience. An equivalent formula of the above approximation guarantee can be stated as the $(1, 1+\zeta(O)/\mathrm{e})$-bicriteria approximation. Reference [18] also showed that the weak approximation ratio is tight under the assumption of $P\neq NP$. The "return on investment (ROI)" greedy algorithm[34], achieves an improved $(1, 1+\ln(\zeta(O)))$-bicriteria approximation for the URSM. Considering a more general matroid constrained regularized submodular maximization, a distorted continuous greedy-based algorithm[35], attaining a deterministic weak $(1-1/\mathrm{e})$ approximation exists. A weak $1/\mathrm{e}$ approximation is further provided in expectation in the case wherein the submodular term is nonmonotonic. A large body of literature[19, 21, 36–38] on regularized submodular maximization problems in the centralized setting exists.

2.2 Streaming algorithms

We then investigate the development of the algorithms for submodular maximization in a streaming fashion. As mentioned previously, the state-of-the-art algorithm has an approximation ratio of $1/2$ when the stream input can be visited only once. Moreover, many works[24, 39–41] with approximation results can be boosted to the tight ratio $(1-1/\mathrm{e})$ by making multipasses over the stream. A single pass streaming algorithm[13] develops a distorted threshold-based algorithm with a multiplicative approximation of $(\zeta(O)-\sqrt{2\zeta(O)-1})/(2\zeta(O)-2)$. A weak $1/2$ approximation and 0.382 approximation for the k-CCRSM exist in online and streaming fashion[6].

3 Preliminary

We provide a brief overview of the models, necessary notations, and assumptions that we will use in this paper.

(1) Element stream model

We denote V as the ground set of elements and assume the elements to be released in a streaming style. In the element stream model, the input is visited in a sequential manner, and the algorithm is allowed to read it in an arbitrary order. Let $V=\{v_1, v_2, \ldots, v_n\}$, ordered by the visiting order of elements from the input stream. At any time t, the algorithm acquires access to v_t and performs computation on the basis of encountered element v_t, albeit without knowledge of future unreleased elements. Often, the stream input is allowed to be released only once, and the algorithm accordingly visits the stream only one time. This algorithm is formally called the single-pass algorithm. We consider that the input stream can be released in multiple times, and accordingly the algorithm is allowed to visit the stream in multipasses.

(2) Submodular function

Here, we recall the definition of a submodular function $f : 2^V \rightarrow \mathbf{R}^+$ for any two subsets $A, B \subseteq V$, $f(A) + f(B) \geqslant f(A \cup B) + f(A \cap B)$ holds. Note that the function $f(\cdot)$ reduces to modular if the above inequality holds the equation. Moreover, the set function $f(\cdot)$ is monotonic if for any pair of subsets $A \subseteq B \subseteq V$, $f(A) \leqslant f(B)$ holds. Given $A, B \subseteq V$ and an element $v \in V$, we denote $A + v$ and $A + B$ as the expression $A \cup \{v\}$ and $A + B$, respectively. Additionally, we denote $f(v|A)$ as the marginal contribution of adding v to A with respect to $f(\cdot)$, that is, $f(v|A) \triangleq f(A+v) - f(A)$. Similarly, we denote $f(B|A) \triangleq f(A + B) - f(A)$ as the marginal contribution of adding a set $B \subseteq V$ to A with respect to $f(\cdot)$. From the perspective of diminishing marginal contribution, a set function f is submodular if the marginal contribution satisfying $f(v|A) \geqslant f(v|B)$ for any $v \notin B, A \subseteq B \subseteq V$.

(3) *k*-CCRSM problem

We state the k-CCRSM problem to find a subset $S \subseteq V$ of size at most k, such that the regularized utility $f(S) - c(S)$ is maximized , i.e.,

$$\arg \max_{S \subseteq V, |S| \leqslant k} f(S) - c(S) \tag{1}$$

where the function $f(\cdot)$ is non-negative and submodular, and the regularizer function $c(\cdot)$ is non-negative and modular.

Throughout this paper we assume that an oracle can obtain the regularized submodular utility for any given set of elements.

4 *k*-CCRSM in a Monotonic Setting

In this section, we present a boosting multipass streaming algorithm for the k-CCRSM when the submodular term is monotonic. Recall that in this problem, we are given a non-negative monotonic submodular function $f(\cdot)$ and a modular function $c(\cdot)$. We are aim to maximize the regularized utility $f(\cdot) - c(\cdot)$ subject to a k-cardinality constraint under the streaming model.

4.1 Description of the Algorithm 1

The formal algorithm that we use to prove Theorem 1 is presented as Algorithm 1, which provides an accuracy parameter $\varepsilon > 0$ and takes an approximate value Γ of the instance of k-CCRSM, such that $\rho \cdot f(O) - c(O) \leqslant \Gamma \leqslant f(O) - c(O)$.

Algorithm 1 starts by initializing the threshold value $\tau = \Gamma/(\rho k)$. By introducing the parameter λ, we build the lower bound for the threshold, which is specifically denoted as $\tau < (1-\varepsilon)\Gamma/(\lambda k)$. We leave the variable in this part and determine it as discussed in the following section. During each iteration of the "while" loop, the threshold value τ reduces to $(1 - \varepsilon)$ times the former iteration. Meanwhile, the element stream is released in an arbitrary order over time, and Algorithm 1 accordingly visits the stream and decides if the visited element v should be stored as a marginal contribution denoted by $f(v|S) - c(v) > \tau$, where τ is the value of the threshold when the element v is considered. The solution set S starts with $S = \varnothing$.

Algorithm 1 returns S with $|S| = k$ or terminates at most $O(\varepsilon^{-1} \log(1/\varepsilon))$ when the lower bound of the threshold value is encountered.

4.2 Theoretical performance guarantees

In this section, we show that Algorithm 1 implies Theorem 1. The following two lemmas prove together that Algorithm 1 obtains the approximation guarantee of Theorem 1. Lemma 1 handles the case of $|S| = k$. We state Lemma 1 below. Without loss of generality, we set $S = \{v_1, v_2, \ldots, v_k\}$ and let $S_i = \{v_1, v_2, \ldots, v_i\}$

Algorithm 1 Multipass algorithm for monotone *k*-CCRSM

Input: Given an accuracy parameter $\varepsilon \in (0, 1)$, approximation value Γ satisfying $\rho \cdot f(O) - c(O) \leqslant \Gamma \leqslant f(O) - c(O)$ for some $\rho \in (0, 1)$, integer k, and parameter $\lambda = 1/\varepsilon$

Output: Solution set S

Set $S \leftarrow \varnothing, \tau \leftarrow \Gamma/(\rho k)$;
while $\tau \geqslant (1-\varepsilon)\Gamma/(\lambda k)$ **do**
  $\tau \leftarrow (1-\varepsilon)\tau$;
  for each element $v \in V$ **do**
    if $f(v|S) - c(v) \geqslant \tau$ **then**
      $S \leftarrow S + v$;
    end if
    if $|S| = k$ **then**
      return S;
    end if
  end for
end while

to be the first i elements added to S from the stream by the Algorithm 1 for any $i \in \{1, 2, \ldots, k\}$. Additionally, we let $S_0 = \varnothing$.

Lemma 1 If the size of solution S reaches the upper bound of the k-cardinality, i.e., $|S| = k$, let $\varepsilon \in (0, 1)$, then the following holds,

$$\frac{f(S) - \mathrm{e}^{-(1-\varepsilon)} c(S)}{f(O) - c(O)} \geqslant 1 - \mathrm{e}^{-(1-\varepsilon)}.$$

Proof Given any $i \in \{1, 2, \ldots, k\}$ and an accuracy $\varepsilon > 0$, we first consider the condition that v_i is added to S_{i-1} during an iteration with $\tau < (1-\varepsilon)(f(O) - c(O))/k$. For any $o \in O \setminus S_{i-1}$, it follows that $f(o|S_{i-1}) - c(o) \leqslant \dfrac{\tau}{1-\varepsilon}$. Therefore, we obtain

$$\begin{aligned}&f(S_i) - f(S_{i-1}) - c(v_i) \geqslant \tau \geqslant \\ &\frac{1-\varepsilon}{k} \sum_{o \in O \setminus S_{i-1}} f(o|S_{i-1}) - c(o) \geqslant \\ &\frac{1-\varepsilon}{k}(f(O \cup S_{i-1}) - f(S_{i-1}) - c(O)) \geqslant \\ &\frac{1-\varepsilon}{k}(f(O) - f(S_{i-1}) - c(O)).\end{aligned}$$

Rearranging the above inequality yields the following recursive form:

$$\begin{aligned}f(S_i) \;\geqslant\; &\left(1 - \frac{1-\varepsilon}{k}\right) f(S_{i-1}) + c(v_i) + \\ &\frac{1-\varepsilon}{k}(f(O) - c(O)).\end{aligned}$$

We now consider if the element v_i is added to S_{i-1} during an iteration with $\tau \geqslant (1-\varepsilon)(f(O) - c(O))/k$, Then $f(v_i|S_{i-1}) - c(v_i) \geqslant \tau \geqslant (1-\varepsilon)(f(O) - c(O))/k$. Hence the recursive inequality also holds. Let $q = 1 - \dfrac{1-\varepsilon}{k}$ and $i = k$, we further obtain

$$\begin{aligned}&f(S_k) \geqslant \\ &q^{k-1} c(S_k) + (1 - q^k)(f(O) - c(O)) \geqslant \\ &q^{k-1} c(S_k) + \mathrm{e}^{-(1-\varepsilon)}(f(O) - c(O)).\end{aligned}$$

Given that $\varepsilon \in (0, 1)$ and $(1 - \frac{1-\varepsilon}{k})^{k-1} \geqslant \mathrm{e}^{-(1-\varepsilon)}$, we acquire

$$f(S) - \mathrm{e}^{-(1-\varepsilon)} c(S) \geqslant (1 - \mathrm{e}^{-(1-\varepsilon)})(f(O) - c(O)).$$

Therefore, the claim is proven. ■

The case of $|S| < k$ is handled as shown below, and we conclude it by using the following Lemma 2.

Lemma 2 If the size of solution S does not reach the upper bound of the k-cardinality, i.e., $|S| < k$, then the following holds:

$$\frac{f(S) - \mathrm{e}^{-(1-\varepsilon)} c(S)}{f(O) - c(O)} \geqslant (1-\varepsilon)\,(1 - \mathrm{e}^{-(1-\varepsilon)}).$$

Proof Observe that the initial value of the threshold τ must be at least $(1-\varepsilon)(f(O) - c(O)/\rho)/k$ and the last value of τ must be at most $(f(O) - c(O))/(\lambda k)$. Consider an arbitrary element $o \in O \setminus S$. Given that o is not selected for S and the solution set S with $|S| < k$, we obtain $f(o|S') - c(o) \leqslant \tau$, where S' denotes the state of S in the last iteration of encountering o and τ represents the threshold value during the last iteration. Then, by the submodularity of $f(\cdot)$, we acquire $f(o|S) - c(o) \leqslant f(o|S') - c(o) \leqslant \tau$. We further gain the following inequality by adding the inequalities over all elements $o \in O \setminus S$ as

$$\begin{aligned}&f(O) - f(S) - c(O) \leqslant \\ &\sum_{o \in O \setminus S} f(o|S) - c(o) \leqslant \\ &\sum_{o \in O \setminus S} \frac{\Gamma}{\lambda k} \leqslant \frac{1}{\lambda}(f(O) - c(O)).\end{aligned}$$

Rearranging the above inequality provides

$$f(S) \geqslant \left(1 - \frac{1}{\lambda}\right)(f(O) - c(O)) \tag{2}$$

In addition, $f(S) - c(S) \geqslant 0$, as discussed in Algorithm 1. Some $\varepsilon \in (0, 1)$ is fixed, and adding an $\mathrm{e}^{-(1-\varepsilon)}$ fraction of the above inequality to a $(1 - \mathrm{e}^{-(1-\varepsilon)})$ fraction of Formula (2) implies

$$\begin{aligned}&\frac{f(S) - \mathrm{e}^{-(1-\varepsilon)} c(S)}{f(O) - c(O)} \geqslant \\ &\left(1 - \frac{1}{\lambda}\right)(1 - \mathrm{e}^{-(1-\varepsilon)}) = \\ &(1-\varepsilon)(1 - \mathrm{e}^{-(1-\varepsilon)}),\end{aligned}$$

where the equality is obtained by setting $\lambda = 1/\varepsilon$. Then we finish the proof of the second case. ■

Our result for the monotonic k-CCRSM, i.e., Theorem 1, now follows from Lemmas 1 and 2. A lower bound $f(S) - \mathrm{e}^{-(1-\varepsilon)} c(S)$ comparing to $f(O) - c(O)$ exists.

The properties of our proposed algorithm are summarized by the following theorem.

Theorem 1 For any accuracy parameter $\varepsilon \in (0, 1)$, assuming that a weak ρ approximation exists for the k-CCRSM problem. By making at most $O(\log(1/(\varepsilon\rho))/\varepsilon)$ passes, querying $O(n \log(1/(\varepsilon\rho))/\varepsilon)$ function value oracles, and consuming $O(k)$ memory, Algorithm 1 produces a solution S satisfying

$$\frac{f(S) - \mathrm{e}^{-(1-\varepsilon)} c(S)}{f(O) - c(O)} \;\geqslant\; (1 - \mathrm{e}^{-1}) - O(\varepsilon).$$

Proof We have the following formula as

$$\begin{aligned}&\frac{f(S) - \mathrm{e}^{-(1-\varepsilon)} c(S)}{f(O) - c(O)} \geqslant \\ &(1-\varepsilon)(1 - \mathrm{e}^{-1} - \varepsilon) \geqslant \\ &1 - \mathrm{e}^{-1} - O(\varepsilon).\end{aligned}$$

Memory and query complexities. Memory complexity is readily bounded by $O(k)$. Given that in each iteration, the threshold decreases by a $(1-\varepsilon)$ fraction from the initial threshold $\Gamma/(\rho k)$ to the lower bound $(1-\varepsilon)\Gamma/(\lambda k)$, we acquire that the running passes is $O(\log(\lambda/\rho)/\varepsilon)$ (i.e., $O(\log(1/(\varepsilon\rho))/\varepsilon)$) by the geometric guesses. Therefore, the total query complexity can be bounded by $O(n\log(1/(\varepsilon\rho))/\varepsilon)$.

5 *k*-CCRSM in a Nonmonotonic Setting

In this section, we present a multipass streaming algorithm for the k-CCRSM problem when the submodular term is nonmonotonic.

5.1 Description of Algorithm 2

The algorithm we use to demonstrate Theorem 2 is summarized as Algorithm 2. This algorithm obtains an accuracy parameter $\varepsilon > 0$ and starts by initializing an approximate value $\rho\cdot f(O)-c(O) \leqslant \Gamma \leqslant f(O)-c(O)$ for some $\rho > 0$.

Let A and B be two disjointed maintained sets during Algorithm 2 and initially set $A, B = \varnothing$. We similarly introduce the parameter of λ to instantiate a lower bound threshold values as $\tau < (1-\varepsilon)\Gamma/(\lambda k)$. During the "while" loop, a visited element is added to $S \in \{A, B\}$, which provides a large distorted marginal contribution, if the distorted marginal contribution is no less than the threshold τ and the size of the solution S does not reach its maximum allowed size k before the stream finishes.

5.2 Theoretical performance guarantees

In this section, we demonstrate that Algorithm 2 implies Theorem 2. We utilize a multipass method that is similar to the method in Ref. [42] for non-monotone streaming k-cardinality constrained submodular maximization under nonregularity conditions. Thus, our analysis is performed fairly similarly to that in Ref. [42]. We begin with a case of C with $|C| = k$, where C is any output of A and B after the first round. We readily obtain the following Lemma 3.

Lemma 3 If the size of any output set $C \in \{A, B\}$ is exactly the same as that of k, the following holds:

$$f(C)-c(C) \geqslant \frac{f(O)}{\eta} - \frac{c(O)}{\eta\rho}.$$

Proof Without loss of generality, let $C = \{c_1, c_2, \ldots, c_k\}$ be the set of elements added to C in accordance with their visited order as denoted by $C_i = \{c_1, c_2, \ldots, c_i\}$ and $S_0 = \varnothing$. Then

$$f(C) - \beta\cdot c(C) = \sum_{i=1}^{k} f(c_i|C_{i-1}) - \beta\cdot c(c_i) \geqslant \sum_{i=1}^{k} \Gamma/(\eta\rho k) \geqslant f(O)/\eta f(O) - c(O)/(\eta\rho).$$

Given that $\beta \geqslant 1$, the following holds,

$$f(C)-c(C) \geqslant \frac{f(O)}{\eta} - \frac{c(O)}{\eta\rho} \tag{3}$$

The proof follows. ■

In the following, we show that the case of both $|A| < k$ and $|B| < k$ after the first round of the "while" loop. For analysis, let $C, D \in \{A, B\}$, such that $C \neq D$ have their values after the stream or the lower bound of the threshold is encountered during the loop. Additionally, we set D' as the state of D at the time of the k-th element is added to C if $|C| = k$; otherwise, we set $D' = D$. Then, we build an upper bound on the sum of gains added to $O \setminus (C \cup D')$ in terms of the optimum and the gains of elements added to $C \setminus O$.

Lemma 4 We have

$$\sum_{o\in O\setminus(C\cup D')} f(o|C) - \beta\cdot c(o) \leqslant \frac{1}{1-\varepsilon}\sum_{i:c_i\notin O} \Delta C_i + \frac{f(O)-c(O)}{\lambda}.$$

Proof We first consider the subcase of $|C| = k$. Additionally, we denote $\Delta C_i = f(c_i|C_{i-1}) - \beta\cdot c(c_i)$ the regularized marginal contribution of adding c_i to C_{i-1} for any $i \in \{1, 2, \ldots, k\}$. We denote τ' as the threshold value during the round that the last k-th element is added to C. For any $o \in O \setminus (C \cup D')$, $f(o|C) - \beta\cdot c(0) \leqslant \tau'/(1-\varepsilon)$ must exist because

---

Algorithm 2 Multipass algorithm for non-monotone *k*-CCRSM

Input: Approximation value Γ satisfying $\rho\cdot f(O) - c(O) \leqslant \Gamma \leqslant f(O) - c(O)$ for some $\rho \in (0,1)$, integer k, and parameters $\lambda = 1/\varepsilon$, $\beta = 2-\varepsilon$, and $\eta = (2-\varepsilon)/2\rho(1-\varepsilon)^2$

Output: Solution set S

```
Let S ← ∅, τ ← Γ/(ηρk);
while τ ⩾ (1 − ε)Γ/(λk) do
    τ ← (1 − ε)τ;
    for each element v ∈ V do
        S ← arg max{f(v|X) − β · c(v) : X ∈ {A, B} and |X| < k};
        if f(v|S) − β · c(v) ⩾ τ then
            S ← S + v;
        end if
    end for
end while
return S ← arg max_{X∈{A,B}} {f(X) − c(X)}
```

element $o \in O \setminus (C \cup D')$ is not added to C or D' during this round of the "while" loop. We obtain $f(c_i|C_{i-1}) - \beta \cdot c(c_i) \geqslant \tau$. Therefore, we acquire

$$\sum_{o \in O \setminus (C \cup D')} f(o|C) - \beta \cdot c(o) \leqslant \frac{1}{1-\varepsilon} \sum_{i: c_i \notin O} f(c_i|C_{i-1}) - \beta \cdot c(c_i) = \frac{1}{1-\varepsilon} \sum_{i: c_i \notin O} \Delta C_i.$$

We then consider the subcase of $|C| < k$. Following by the fact that the definition of the last threshold τ implies $f(o|C) - \beta \cdot c(o) < \Gamma/(\lambda k)$, we obtain $\sum_{o \in O \setminus (C \cup D')} f(o|C) - \beta \cdot c(o) < \Gamma/\lambda \leqslant (f(O) - c(O))/\lambda$. Combining the above two subcases yields $\sum_{o \in O \setminus (C \cup D')} f(o|C) - \beta \cdot c(o) \leqslant \frac{1}{1-\varepsilon} \sum_{i: c_i \notin O} \Delta C_i + (f(O) - c(O))/\lambda$. ■

In addition, we build an upper bound on $f(O|C) - c(O)$ in terms of the gains of the elements added to C and D.

Lemma 5 We have

$$f(O|C) - \beta \cdot c(O) - \frac{f(O) - c(O)}{\lambda} \leqslant \frac{1}{1-\varepsilon} \left(\sum_{i: d_i \in O} \Delta D_i + \sum_{i: c_i \notin O} \Delta C_i \right).$$

Proof By submodularity, we obtain

$$f(O|C) - \beta \cdot c(O) \leqslant \sum_{o \in O \cap D'} f(o|C) - \beta \cdot c(o) + \sum_{o \in O \setminus (C \cup D')} f(o|C) - \beta \cdot c(o) \leqslant \sum_{i: d_i \in O} \Delta D_i + \sum_{o \in O \setminus (C \cup D')} f(o|C) - \beta \cdot c(o) \leqslant \frac{1}{1-\varepsilon} \left(\sum_{i: d_i \in O} \Delta D_i + \sum_{i: c_i \notin O} \Delta C_i \right) + \frac{f(O) - c(O)}{\lambda}.$$

The last inequality directly follows by Lemma 4. Then the claim is proven. ■

On the basis of Lemma 5, we set $C = A$ and $C = B$ respectively. We derive the following Lemma 6.

Lemma 6 We acquire the lower bound for $f(S) - c(S)$ as follows:

$$\frac{f(S) - c(S)}{(1-\varepsilon)/(4-2\varepsilon)} \geqslant \left(1 - \frac{2}{\lambda}\right) f(O) - \left(4 - 2\varepsilon - \frac{2}{\lambda}\right) c(O).$$

Proof We have the following to prove the above claim:

$$f(O) - 2\beta c(O) \leqslant f(O \cup A) + f(O \cup B) - 2\beta c(O) \leqslant \frac{1}{1-\varepsilon} \left(\sum_i \Delta A_i + \sum_i \Delta B_i \right) + \frac{2(f(O) - c(O))}{\lambda} + f(A) + f(B) \leqslant \frac{2(2-\varepsilon)}{1-\varepsilon} [f(S) - c(S)] + \frac{2}{\lambda} (f(O) - c(O)).$$

The third inequality is obtained by setting $\beta = 2 - \varepsilon$.

Rearranging the above inequality implies

$$\frac{f(S) - c(S)}{(1-\varepsilon)/(2-\varepsilon)} \geqslant \left(\frac{1}{2} - \frac{1}{\lambda}\right) f(O) \left(2 - \varepsilon - \frac{1}{\lambda}\right) c(O).$$

The proof is completed. ■

Our main result for nonmonotonic k-CCRSM, i.e., Theorem 2, is from Lemmas 3 and 6, and our choice of parameters β and λ. Hence, we establish a lower bound $f(S) - c(S)$ in terms of a distorted optimum.

We summarize the main results with the following theorem.

Theorem 2 Given some $\varepsilon \in (0, 1/2)$ and assuming that a weak ρ approximation for the nonmonotonic k-CCRSM problem exists, by making at most $O(\log((1-\varepsilon)^2/(2\varepsilon - \varepsilon^2))/\varepsilon)$ passes, querying $O(n \log((1-\varepsilon)^2/(2\varepsilon - \varepsilon^2))/\varepsilon)$ function value oracles, and consuming $O(k)$ memory, Algorithm 2 produces a solution S satisfying

$$\frac{f(S) - c(S)}{(1-\varepsilon)/(2-\varepsilon)} \geqslant \min\left\{\frac{1}{2} - \varepsilon, 2(1-\varepsilon)\rho\right\} f(O) - 2(1-\varepsilon)c(O) \quad (4)$$

Proof The following calculation implies the claim. Let

$$\frac{1}{\eta\rho} = \frac{1-\varepsilon}{2-\varepsilon} \left(2 - \varepsilon - \frac{1}{\lambda}\right),$$

Then we obtain

$$\frac{f(S) - c(S)}{(1-\varepsilon)/(2-\varepsilon)} \geqslant \min\left\{\frac{1}{2} - \frac{1}{\lambda}, \left(2 - \varepsilon - \frac{1}{\lambda}\right)\rho\right\} f(O) - \left(2 - \varepsilon - \frac{1}{\lambda}\right) c(O) = \min\left\{\frac{1}{2} - \varepsilon, 2(1-\varepsilon)\rho\right\} f(O) - 2(1-\varepsilon)c(O),$$

which holds by setting $\lambda = 1/\varepsilon$.

Query complexity. We conclude that the Algorithm 2 terminates at most $O(\log(1/(\varepsilon\eta\rho)))$ (i.e., $O(\log((1-\varepsilon)^2/(2\varepsilon-\varepsilon^2))/\varepsilon))$ passes, and thus the query complexity is bounded by $O(n\log((1-\varepsilon)^2/(2\varepsilon-\varepsilon^2))/\varepsilon)$.

6 Conclusion

We consider the problem of maximizing a k-cardinality constrained submodular function with a nonnegative modular regularizer under the streaming model, denoted by k-CCRSM. Under the assumption that we have access to a weak ρ-approximation in advance for the regularized problem, we establish a threshold interval based on the approximate value, and then sieve elements by visiting the element stream in an arbitrary order during iterations. We obtain a boosting multipass threshold-based streaming algorithm for the k-CCRSM problem wherein the submodular term is monotonic. We further present an extended multipass distorted threshold-based streaming algorithm for the regularized model wherein the submodular term of the objective function is nonmonotonic.

Acknowledgment

This work was supported by the Beijing Natural Science Foundation Project (No. Z220004), the National Natural Science Foundation of China (Nos. 11901544 and 12101587), and the China Postdoctoral Science Foundation (No. 2022M720329).

References

[1] Q. Gong, S. Gao, F. Wang, and R. Yang, A multipass streaming algorithm for regularized submodular maximization, in *Proc. 15th Ann. Int. Conf. on Combinatorial Optimization and Applications*, Tianjin, China, 2021, pp. 701–711.

[2] A. Badanidiyuru, B. Mirzasoleiman, A. Karbasi, and A. Krause, Streaming submodular maximization: Massive data summarization on the fly, in *Proc. 20th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, New York, NY, USA, 2014, pp. 671–680.

[3] H. Lin and J. Bilmes, A class of submodular functions for document summarization, in *Proc. 49th Ann. Meeting of the Association for Computational Linguistics*: *Human Language Technologies*, Portland, OR, USA, 2011, pp. 510–520.

[4] B. Mirzasoleiman, A. Badanidiyuru, and A. Karbasi, Fast constrained submodular maximization: Personalized data summarization, in *Proc. 33nd Int. Conf. on Machine Learning*, New York, NY, USA, 2016, pp. 1358–1367.

[5] D. Kempe, J. Kleinberg, and É. Tardos, Maximizing the spread of influence through a social network, in *Proc. 9th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, Washington, DC, USA, 2003, pp. 137–146.

[6] S. M. Nikolakaki, A. Ene, and E. Terzi, An efficient framwork for balancing submodularity and cost, in *Proc. 27th ACM SIGKDD Conf. on Knowledge Discovery and Data Mining*, Virtual Event, 2021, pp. 1256–1266.

[7] J. Wu and N. Wang, Approximating special social influence maximization problems, *Tsinghua Science and Technology*, vol. 25, no. 6, pp. 703–711, 2020.

[8] I. Bogunovic, S. Mitrović, J. Scarlett, and V. Cevher, Robust submodular maximization: A non-uniform partitioning approach, in *Proc. 34th Int. Conf. on Machine Learning*, Sydney, Australia, 2017, pp. 508–516.

[9] A. Krause, H. Brendan, C. Guestrin, and A. Gupta, Robust submodular observation selection, *J. Mach. Learn. Res.*, vol. 9, no. 93, pp. 2761–2801, 2008.

[10] J. B. Orlin, A. S. Schulz, and R. Udwani, Robust monotone submodular function maximization, *Math. Program.*, vol. 172, no. 1, pp. 505–537, 2018.

[11] A. Dasgupta, R. Kumar, and S. Ravi, Summarization through submodularity and dispersion, in *Proc. 51st Ann. Meeting of the Association for Computational Linguistics*, Sofia, Bulgaria, 2013, pp. 1014–1022.

[12] S. Cui, K. Han, T. Zhu, J. Tang, B. Wu, and H. Huang, Randomized algorithms for submodular function maximization with a k-system constraint, in *Proc. 38th Int. Conf. on Machine Learning*, Virtual Event, 2021, pp. 2222–2232.

[13] E. Kazemi, S. Minaee, M. Feldman, and A. Karbasi, Regularized submodular maximization at scale, in *Proc. 38th Int. Conf. on Machine Learning*, Virtual Event, 2021, pp. 5356–5366.

[14] A. Anagnostopoulos, L. Becchetti, C. Castillo, A. Gionis, and S. Leonardi, Online team formation in social networks, in *Proc. 21st Int. World Wide Web Conferences*, Lyon, France, 2012, pp. 839–848.

[15] M. Kargar, M. Zihayat, and A. An, Finding affordable and collaborative teams from a network of experts, in *Proc. 13th SIAM Int. Conf. on Data Mining*, Austin, TX, USA, 2013, pp. 587–595.

[16] T. Lappas, K. Liu, and E. Terzi, Finding a team of experts in social networks, in *Proc. 15th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, Paris, France, 2009, pp. 467–476.

[17] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions-I, *Math. Program.*, vol. 14, no. 1, pp. 265–294, 1978.

[18] C. Harshaw, M. Feldman, J. Ward, and A. Karbasi, Submodular maximization beyond non-negativity: Guarantees, fast algorithms, and applications, in *Proc. 36th Int. Conf. on Machine Learning*, Long Beach, CA, USA, 2019, pp. 2634–2643.

[19] M. Sviridenko, J. Vondrák, and J. Ward, Optimal approximation for submodular and supermodular optimization with bounded curvature, *Math. Oper. Res.*, vol. 42, no. 4, pp. 1197–1218, 2017.

[20] M. Feldman, J. Naor, R. Schwartz, and J. Ward, Improved approximations for k-exchange systems, in *Proc. 19th Ann. European Symposium on Algorithms*, Saarbrücken, Germany, 2011, pp. 784–798.

[21] Y. Wang, D. Xu, D. Du, and R. Ma, Bicriteria algorithms to balance coverage and cost in team formation under online model, *Theor. Comput. Sci.*, vol. 854, pp. 68–76, 2021.

[22] M. Feldman, A. Norouzi-Fard, O. Svensson, and R. Zenklusen, The one-way communication complexity of submodular maximization with applications to streaming and robustness, in *Proc. 52nd Ann. ACM SIGACT Symp. on Theory on Computing*, Chicago, IL, USA, 2020, pp. 1363–1374.

[23] E. Kazemi, M. Mitrovic, M. Zadimoghaddam, S. Lattanzi, and A. Karbasi, Submodular streaming in all its glory: Tight approximation, minimum memory and low adaptive complexity, in *Proc. 38th Int. Conf. on Machine Learning*, Long Beach, CA, USA, 2019, pp. 3311–3320.

[24] A. Norouzi-Fard, J. Tarnawski, S. Mitrovic, A. Zandieh, A. Mousavifar, and O. Svensson, Beyond 1/2-approximation for submodular maximization on massive data streams, in *Proc. 35th Int. Conf. on Machine Learning*, Stockholmsmässan, Sweden, 2018, pp. 3826–3835.

[25] N. Buchbinder, M. Feldman, and M. Garg, Deterministic $(1/2+\varepsilon)$-approximation for submodular maximization over a matroid, in *Proc. 30th Ann. ACM-SIAM Symp. on Discrete Algorithms*, San Diego, CA, USA, 2019, pp. 241–254.

[26] C. C. Huang and F. Sellier, Semi-streaming algorithms for submodular function maximization under b-matching, matroid, and matchoid constraints, in *Proc. 25th Int. Workshop on Randomization and Computation* (*RANDOM 2021*) *and 24th Int. Workshop on Approximation Algorithms for Combinatorial Optimization Problems*, Seattle, WA, USA, 2021, pp. 14:1–14:18.

[27] R. Levin and D. Wajc, Streaming submodular matching meets the primal-dual method, in *Proc. 32nd ACM-SIAM Symp. on Discrete Algorithms*, Alexandria, VA, USA, 2021, pp. 1914–1933.

[28] N. Buchbinder, M. Feldman, J. Naor, and R. Schwartz, Submodular maximization with cardinality constraints, in *Proc. 25th Ann. ACM-SIAM Symp. on Discrete Algorithms*, Portland, OR, USA, 2014, pp. 1433–1452.

[29] G. Calinescu, C. Chekuri, M. Pál, and J. Vondrák, Maximizing a monotone submodular function subject to a matroid constraint, *SIAM J. Comput.*, vol. 40, no. 6, pp. 1740–1766, 2011.

[30] U. Feige, A threshold of $\ln(n)$ for approximating set cover, *J. ACM*, vol. 45, no. 4, pp. 634–652, 1998.

[31] J. Lee, M. Sviridenko, and J. Vondrák, Submodular maximization over multiple matroids via generalized exchange properties, *Math. Oper. Res.*, vol. 35, no. 4, pp. 795–806, 2010.

[32] M. Feldman, C. Harshaw, and A. Karbasi, Greed is good: Near-optimal submodular maximization via greedy optimization, in *Proc. 30th Conf. on Learning Theory*, Amsterdam, The Netherlands, 2017, pp. 758–784.

[33] K. K. Sarpatwar, B. Schieber, and H. Shachnai, Constrained submodular maximization via greedy local search, *Oper. Res. Lett.*, vol. 41, no. 1, pp. 1–6, 2019.

[34] T. Jin, Y. Yang, R. Yang, J. Shi, K. Huang, and X. Xiao, Unconstrained submodular maximization with modular costs: Tight approximation and applicatiion to profit maximization. in *Proc. 47th Int. Conf. on Very Large Databases*, Copenhagen, Denmark, 2021, pp. 1756–1768.

[35] C. Lu, W. Yang, and S. Gao, Regularized non-monotone submodular maximization, arXiv:2103.10008, 2021.

[36] S. Chen, W. Yang, S. Gao, and R. Jin, Novel algorithms for maximum DS decomposition, *Theor. Comput. Sci.*, vol. 857, pp. 87–96, 2021.

[37] M. Feldman, Guess free maximization of submodular and linear sums, *Algorithmica*, vol. 83, no. 3, pp. 853–878, 2021.

[38] S. Tang and J. Yuan, Adaptive regularized submodular maximization, in *Proc. 32nd Int. Symp. on Algorithms and Computation*, Fukuoka, Japan, 2021, pp. 69:1–79:13.

[39] C. C. Huang, and N. Kakimura, Multipass streaming algorithms for monotone submodular function maximization, *Theor. Comput. Syst.*, vol. 66, no. 1, pp. 354–394, 2022.

[40] A. Kuhnle, Quick streaming algorithms for the maximizing of monotone submodular functions in linear time, in *Proc. 24th Int. Conf. on Artificial Intelligence and Statistics*, Virtual Event, 2021, pp. 1360–1368.

[41] A. McGregor and H. T. Vu, Better streaming algorithms for the maximum coverage problem, *Theor. Comput. Syst.*, vol. 63, no. 7, pp. 1595–1619, 2019.

[42] A. Kuhnle, Quick streaming algorithms for cardinality-constrained maximization of non-monotone submodular functions in linear time, arXiv:2104.06873, 2021.

Qinqin Gong received the MEng degree from Beijing University of Technology, China in 2018. She is currently a PhD candidate at Beijing Institute for Scientific and Engineering Computing, Beijing University of Technology, China. Her research interests include combinatorial optimization, submodular optimization, and supply chain management.

Suixiang Gao received the PhD degree from the Institute of Applied Mathematics, Chinese Academy of Sciences, China in 1998. He is currently a professor at the School of Mathematical Sciences, University of Chinese Academy of Sciences, China. His research focuses on combinatorial optimization, network optimization, and approximation algorithms.

Fengmin Wang received the PhD degree from Beijing University of Technology, China in 2016. From May 2016 to May 2018, she worked as a postdoctoral researcher at the Academy of Mathematics and Systems Science, Chinese Academy of Sciences, China. She is currently a senior engineer at Beijing Jinghang Research Institute of Computing and Communication, China. Her research focuses on combinatorial optimization, approximation algorithm, distributed collaborative optimization, and data modeling.

Ruiqi Yang received the PhD degree from Beijing University of Technology, China in 2020. From June 2020 to June 2022, he worked as a postdoctoral researcher at the University of Chinese Academy of Sciences, China. He currently works as a teacher at Beijing University of Technology, China. His research interests include combinatorial optimization, approximation algorithms, and submodular optimization.