# Maximization of Monotone $k$-Submodular Functions with Bounded Curvature and Non-$k$-Submodular Functions

**Tatsuya Matsuoka** TA.MATSUOKA@NEC.COM
*NEC Corporation*

**Naoto Ohsaka** OHSAKA@NEC.COM
*NEC Corporation*

**Editors:** Vineeth N Balasubramanian and Ivor Tsang

## Abstract

The concept of $k$-submodularity is an extension of submodularity, of which maximization has various applications, such as influence maximization and sensor placement. In such situations, to model complicated real problems, we want to deal with multiple factors, such as, more detailed parameter representing a property of a given function or a constraint which should be imposed for a given function, simultaneously. Besides, it is preferable that an algorithm for the modeling problem is simple. In this paper, for both monotone $k$-submodular function maximization with bounded curvature and monotone weakly $k$-submodular function maximization, we give approximation ratio analysis on greedy-type algorithms on the problem with the matroid constraint and that with the individual size constraint. Furthermore, we give an approximation ratio analysis on another type of the relaxation of $k$-submodular functions, approximately $k$-submodular functions, with the matroid constraint.

**Keywords:** $k$-submodular functions, Greedy algorithm, Curvature

## 1. Introduction

Many real-world functions often have a structure, which is called *submodularity* (*diminishing returns property*). A set function $f\colon 2^V \to \mathbb{R}$ is called *submodular* if for any $A, B \subseteq V$, $f(A) + f(B) \geq f(A \cup B) + f(A \cap B)$ hold. For submodular maximization problems, the purpose is to find a subset of a ground set whose function value is maximized satisfying the input constraints. Submodular maximization has been studied actively (Nemhauser and Wolsey, 1978; Nemhauser et al., 1978; Sviridenko, 2004) and has a variety of applications, such as influence maximization (Kempe et al., 2003), sensor placement (Krause et al., 2008), summarization (Lin and Bilmes, 2011), and image segmentation (Jegelka and Bilmes, 2011). For maximization of monotone nonnegative submodular functions, Nemhauser et al. (1978) gave a $(1 - 1/\mathrm{e})$-approximation algorithm and Nemhauser and Wolsey (1978) showed that any polynomial-time algorithm cannot achieve an approximation ratio better than $1 - 1/\mathrm{e}$.

For modeling real problems with submodular structure, to analyze approximation ratios more in detail or to relax the condition, for submodular maximization, there have been some approaches; paying attention to some parameter of functions or relaxing the submodularity by allowing deviation. One of the useful concepts is *curvature* for monotone submodular functions introduced by Conforti and Cornuéjols (1984). The curvature $c$ satisfies $0 \leq c \leq 1$

and represents "how much the function curves". By using curvature $c$, Conforti and Cornuéjols (1984) gave an approximation ratio $1/(1+c)$ for the greedy algorithm for monotone nonnegative submodular maximization with the matroid constraint. On relaxing submodularity, *approximate submodularity* (Horel and Singer, 2016) and *weak submodularity* (deeply related to *submodularity ratio* (Das and Kempe, 2011)) have been introduced.

In real situations, multiple kinds of goods are often dealt with and people want to optimize some index for those goods. The class of $k$-submodular functions can deal with this kind of domain. As an extension of submodularity, *$k$-submodularity* was introduced by this name by Huber and Kolmogorov (2012), the notion itself appearing in Cohen et al. (2006). Let $(k+1)^V$ be the family of all subpartitions of $V$ to $k$ subsets, that is, $(k+1)^V := \{(V_1,\dots,V_k) \mid V_i \subseteq V \ (\forall i \in [k]), V_i \cap V_j = \emptyset \ (\forall i,j \in [k], i \neq j)\}$. For $\boldsymbol{x} = (X_1,\dots,X_k), \boldsymbol{y} = (Y_1,\dots,Y_k) \in (k+1)^V$, $\boldsymbol{x} \sqcup \boldsymbol{y}, \boldsymbol{x} \sqcap \boldsymbol{y} \in (k+1)^V$ are defined by

$$\boldsymbol{x} \sqcap \boldsymbol{y} := (X_1 \cap Y_1, \dots, X_k \cap Y_k) \text{ and}$$
$$\boldsymbol{x} \sqcup \boldsymbol{y} := \left( (X_1 \cup Y_1) \setminus \left( \bigcup_{i \neq 1} (X_i \cup Y_i) \right), \dots, (X_k \cup Y_k) \setminus \left( \bigcup_{i \neq k} (X_i \cup Y_i) \right) \right),$$

respectively. A function $f\colon (k+1)^V \to \mathbb{R}$ is called *$k$-submodular* if for any $\boldsymbol{x}, \boldsymbol{y} \in (k+1)^V$, $f(\boldsymbol{x}) + f(\boldsymbol{y}) \geq f(\boldsymbol{x} \sqcup \boldsymbol{y}) + f(\boldsymbol{x} \sqcap \boldsymbol{y})$ hold[1]. If $k = 1$, then it matches the submodularity. In this paper, we identify $\{0,1,\dots,k\}^n$ with $(k+1)^V$; we correspond a element in $(k+1)^V$, which is a subpartition $(V_1, V_2, \dots, V_k)$ of $V$, with $n$-dimensional vector $\boldsymbol{x} \in \{0,1,\dots,k\}^n$ by defining $e$-th element of $\boldsymbol{x}$ with $i$ if $e \in V_i$. By indicating this $\boldsymbol{x}$, we write as $\boldsymbol{x} \in (k+1)^V$ in this paper. As applications of maximization of monotone $k$-submodular functions, Ohsaka and Yoshida (2015) list influence maximization with $k$ kinds of items and sensor placement with $k$ kinds of sensors, and Qian et al. (2017) state information coverage maximization with $k$ topics.

For $k$-submodular maximization, there have been studies on approximation algorithms with constraints. In this paper, we deal with the matroid constraint and the individual size constraint, whose definitions are given in Section 1.1. Ohsaka and Yoshida (2015) gave an approximation ratio $1/2$ for monotone case with the total size constraint. Later, Sakaue (2017) gave the same approximation ratio $1/2$ for monotone case with the matroid constraint, which includes the total size-constrained case. For individual-size-constrained problem, Ohsaka and Yoshida (2015) gave an approximation ratio $1/3$ of the greedy algorithm.

There are situations such that combination is needed for modeling the problem. For example, if one deals with $k$ kinds of things, the objective function contains some error, and also a budget constraint should be considered, then a combined model is necessary and we want to know performance bounds of (possibly simple) algorithms for those problems. The details of applications setings are described in Section 1.3. Analyses in this paper are for the framework of greedy algorithms or residual random greedy algorithms (Buchbinder et al., 2014).

In this paper, we deal with two types of constraints and three concepts on functions. Two constraints are the matroid constraint and the individual size constraint, which are

1. It seems that $k$-submodular problems can be reduced to a submodular problem over an extended ground set with a partition matroid constraint, but in general it is not.

explained in Section 1.1. Here, we shortly explain the three concepts on functions. The first one is *curvature* for monotone $k$-submodular functions, which represents, roughly speaking, how much the $k$-submodular function curves on the increment on the addition of a new element to some $V_i$ ($i \in [k]$) of a subpartition $(V_1, \dots, V_k)$. The second one is *weak $k$-submodularity*, which is a $k$-submodularity-like property with the relaxation on the ratio of increment between addition of a set and addition of each elements of the set. The last one is *approximate $k$-submodularity* (Nguyen and Thai, 2020), which is also a $k$-submodularity-like property, but with the relaxation on the lower and upper bounds of function values. The details of these three concepts are explained in Section 3. Note that for $k$-submodular maximization, greedy(-type) algorithms are well-studied ones and thus our analyses in this paper indicate that the greedy algorithm satisfies the better bound when the curvature is restricted or greedy-type algorithms satisfy certain approximation ratios for weakly or approximately $k$-submodular maximization with some type of constraints.

### 1.1. Problem Settings

In this paper, we deal with two types of problems for nonnegative monotone function maximization, with the matroid constraint and with the individual size constraint. For $\boldsymbol{x} \in (k+1)^V$, let $\mathrm{supp}(\boldsymbol{x}) := \{e \in [n] \mid \boldsymbol{x}(e) \neq 0\}$.

**Matroid Constraint** A *matroid* is a pair $(V, \mathcal{I})$ of a set $V$ (called a *ground set*) and a set family $\mathcal{I} \subseteq 2^V$ (called the *family of independent sets*) satisfying following three conditions: (i) $\emptyset \in \mathcal{I}$, (ii) for $A, B \subseteq V$, if $A \subseteq B$ and $B \in \mathcal{I}$, then $A \in \mathcal{I}$, and (iii) for $A, B \in \mathcal{I}$, if $|A| < |B|$, then there exists $e \in B \setminus A$ such that $A \cup \{e\} \in \mathcal{I}$. The set family of maximal elements of $\mathcal{I}$ is denoted by $\mathcal{B}$ (called the *base family*). It is known that every element in $\mathcal{B}$ is of the same size and this size is called the *rank* of the matroid. A matroid constraint for monotone $k$-submodular maximization was dealt with by Sakaue (2017). Under the matroid constraint with a matroid $(V, \mathcal{I})$, a solution $\boldsymbol{x} \in (k+1)^V$ is feasible if $\mathrm{supp}(\boldsymbol{x}) \in \mathcal{I}$. Note that the matroid constraint is a generalization of the total size constraint (a matroid $(V, \mathcal{I})$ with $\mathcal{I} = \{S \subseteq V \mid |S| \leq B\}$ (called a *uniform matroid*) is corresponding to the total size constraint with $B$ (a solution $\boldsymbol{x} \in (k+1)^V$ is feasible if $|\mathrm{supp}(\boldsymbol{x})| \leq B$)).

**Individual Size Constraint** A size constraint for each $i \in [k]$ for $k$-submodular maximization is considered in Ohsaka and Yoshida (2015). More precisely, for $\boldsymbol{x} \in (k+1)^V$ and $i \in [k]$, let $\mathrm{supp}_i(\boldsymbol{x}) = \{e \in [n] \mid \boldsymbol{x}(e) = i\}$. Then, given $B_1, B_2, \dots, B_k \in \mathbb{Z}_{\geq 0}$ such that $\sum_{i=1}^{k} B_i \leq n$, a solution $\boldsymbol{x} \in (k+1)^V$ is feasible if $|\mathrm{supp}_i(\boldsymbol{x})| \leq B_i$ for each $i \in [k]$.

**Problem Frameworks** Problems in this paper belong to either of the followings:

**Problem 1** *Given a function $f\colon (k+1)^V \to \mathbb{R}_{\geq 0}$, find $\boldsymbol{x} \in (k+1)^V$ maximizing $f(\boldsymbol{x})$ with the matroid constraint.*

**Problem 2** *Given a function $f\colon (k+1)^V \to \mathbb{R}_{\geq 0}$, find $\boldsymbol{x} \in (k+1)^V$ maximizing $f(\boldsymbol{x})$ with the individual size constraint.*

### 1.2. Contribution

Our aim of this paper is to analyze approximation ratios of well-studied greedy(-type) algorithms for several problems of $k$-submodular maximization and related problems.

- (Section 4) For the maximization of monotone $k$-submodular functions with curvature $c$, we show that the greedy algorithm yields an approximation ratio of $1/(1+c)$ for the problem with the matroid constraint and $1/(1+2c)$ for the problem with the individual size constraint. Technically, we utilize a term which is bounded by 0 in the analysis by Ohsaka and Yoshida (2015) and Sakaue (2017) by using curvature. Since approximation ratio better than $(k+1)/2k$ cannot be achieved in polynomial time (Iwata et al., 2016), our result shows hopeful viewpoint for maximization of some type of $k$-submodular functions.
- (Section 5) For the maximization of monotone $\gamma$-weakly $k$-submodular functions, we show that the residual random greedy algorithm yields an expected approximation ratio of $(1+1/\gamma)^{-2}$ for the problems with the matroid constraint or the individual size constraint. These results and arguments toward the results imply that we can expect that some type of results on submodular function maximization may be able to be brought to the $k$-submodular statement by the similar discussion to ours.
- (Section 6) For the maximization of monotone $\varepsilon$-approximately $k$-submodular functions, we show that the greedy algorithm yields an approximation ratio of $\frac{1}{2}\left(\frac{1-\varepsilon}{1+\varepsilon}\right)\frac{1}{1+\frac{\varepsilon B}{1-\varepsilon}}$ for the problem with the matroid constraint, where $B$ is a rank of the matroid of constraint. This result includes the total size constrained case by Nguyen and Thai (2020). Note that maximization of approximately $k$-submodular functions with the individual size constraint was dealt with by Zheng et al. (2021) for the greedy algorithm.

### 1.3. On Applications

In this section, we state the applications of constraints or function types appearing in this paper. *Matroid constraint* includes partition matroid constraint, which corresponds with constraints for groups of candidate places (see, e.g., Friedrich et al. (2019) (not for $k$-submodular functions)). *Individual size constraint* corresponds with, for instance, the budget for each type of items (see, e.g., Ohsaka and Yoshida (2015)).

For a ground set $V$, the curvature of a monotone submodular function can be calculated in linear times of function calls, and thus in a situation such that function value can be estimated, then we can estimate the curvature. For $k$-submodular functions, if $k$ is not so large, then we can estimate the value of curvature in some situations. *Curvature for $k$-submodular functions* is useful to know the performance bound of the algorithm.

Weak submodular functions have applications on machine learning (Elenberg et al., 2017; Chen et al., 2018). The *weakly $k$-submodular function* is a certain relaxation of the $k$-submodular function and thus it can be utilized in a situation such that the objective function is close to $k$-submodular but precisely not.

If the objective function $f(S)$ ($S \subseteq V$) is a submodular function and for observation $F(S)$ there is an inconsistent noise ($f(S) = \mathbb{E}[F(S)]$), then if noise is bounded and one observes enough times, then the estimated value obeys an approximately submodular function (Horel and Singer, 2016). By the same type of argument, *approximately $k$-submodular functions* are useful when the objective function is a $k$-submodular function and observation has some bounded inconsistent noise. Note that Zheng et al. (2021) stated on the cases that a function is not exactly $k$-submodular because of noise or uncertainty and they dealt with approximately $k$-submodular maximization with the total size constraint and that with the individual size constraint.

Combination of the above constraints or concepts is necessary for modeling of complex real-world problems. Influence maximization for $k$ types of items and sensor placement for $k$ types of sensors are listed as applications of monotone $k$-submodular maximization (with the total size constraint or the individual size constraint) by Ohsaka and Yoshida (2015), and information coverage maximization with $k$ topics is stated by Qian et al. (2017). For example, in influence maximization application, the following requirements may arise simultaneously; (i) the sender wants to spread $k$ kinds of information and the objective is to maximize the number of people who influenced by at least one type of information, (ii) fair constraint on the candidate places, and (iii) the observed function contains inconsistent noise. For the above situation, monotone approximately $k$-submodular maximization with the matroid constraint can be utilized. Other combinations also can be considered similarly.

## 2. Related Works

Tables 1, 2, and 3 are the results of this paper and those of corresponding previous works that our algorithm utilizes.

Table 1: Approximation ratios for monotone $k$-submodular maximization with general curvature (without bound) and bounded curvature $c$.

| | matroid constraint | individual size constraint |
|---|---|---|
| general | $\frac{1}{2}$ (Sakaue, 2017) | $\frac{1}{3}$ (Ohsaka and Yoshida, 2015) |
| bounded curvature | $\frac{1}{1+c}$ (Theorem 1) | $\frac{1}{1+2c}$ (Theorem 2) |

Table 2: Approximation ratios for monotone weakly ($k$-)submodular functions. The term $\gamma$ appearing in the right two approximation ratios is one used for the definition of $\gamma$-weakly $k$-submodular functions (Definition 2). (If $k = 1$, then the property of this $\gamma$ matches those of the $\gamma$ appearing in the leftmost approximation ratio.)

| $\gamma$-weakly submodular (matroid constraint) | $\gamma$-weakly $k$-submodular (matroid constraint) | $\gamma$-weakly $k$-submodular (individual size constraint) |
|---|---|---|
| $\left(1+\frac{1}{\gamma}\right)^{-2}$ (Chen et al., 2018) | $\left(1+\frac{1}{\gamma}\right)^{-2}$ (Theorem 5) | $\left(1+\frac{1}{\gamma}\right)^{-2}$ (Theorem 6) |

For bisubmodular functions, which are $k$-submodular functions with $k = 2$, approximate maximization algorithm was studied in Singh et al. (2012). For unconstrained problems, Ward and Živný (2014) gave the first approximation-guaranteed algorithm for $k$-submodular maximization, the approximation ratio of which is $1/(1 + \sqrt{k/2})$. Iwata et al. (2016)

Table 3: Approximation ratios for monotone approximately ($k$-)submodular maximization. ("$\varepsilon$-approx." means "$\varepsilon$-approximately" in each definition.)

| $\varepsilon$-approx. submodular (matroid constraint) | $\varepsilon$-approx. $k$-submodular (total size constraint) | $\varepsilon$-approx. $k$-submodular (matroid constraint) |
|---|---|---|
| $\frac{1}{2}\left(\frac{1-\varepsilon}{1+\varepsilon}\right)\frac{1}{1+\frac{\varepsilon B}{1-\varepsilon}}$ (Horel and Singer, 2016) | $\frac{1}{2}\left(\frac{1-\varepsilon}{1+\varepsilon}\right)\frac{1}{1+\frac{\varepsilon B}{1-\varepsilon}}$ (Zheng et al., 2021) | $\frac{1}{2}\left(\frac{1-\varepsilon}{1+\varepsilon}\right)\frac{1}{1+\frac{\varepsilon B}{1-\varepsilon}}$ (Theorem 7) |

gave a randomized 1/2-approximation algorithm for nonmonotone case and a randomized $k/(2k-1)$-approximation algorithm for monotone case. Later, Oshima (2021) gave a randomized $(k^2+1)/(2k^2+1)$-approximation algorithm for nonmonotone case. Iwata et al. (2016) also showed that an approximation ratio better than $(k+1)/2k$ for monotone $k$-submodular maximization would need exponential queries. For online setting, Soma (2019) gave algorithms corresponding to Iwata et al. (2016)'s results above, that is, no 1/2-regret algorithm for nonmonotone case and no $k/(2k-1)$-regret algorithm for monotone case.

For constrained problems, as written in Section 1, Ohsaka and Yoshida (2015) gave approximation ratios 1/2 and 1/3 for the total size constrained problem and for the individual size constrained problem, respectively, and Sakaue (2017) gave an approximation ratio 1/2 for the matroid-constrained problem. Recently, Tang et al. (2021) proposed a $(\frac{1}{2}-\frac{1}{2\mathrm{e}})$-approximation algorithm for monotone $k$-submodular maximization under a knapsack constraint. There has been another approach from the view of multi-objective evolutionary algorithms, and Qian et al. (2017) gave an algorithm whose expected number of iterations to obtain a 1/2-approximate solution for monotone $k$-submodular maximization with the total size constraint is polynomial.

Curvature (Conforti and Cornuéjols, 1984) is a parameter which indicates, roughly speaking, how much the function curves. When curvature is $c$, Conforti and Cornuéjols (1984) showed that the greedy algorithm gives $1/(1+c)$-approximation for monotone submodular maximization with the matroid constraint. Especially, for the problem with the uniform matroid constraint, Conforti and Cornuéjols (1984) gave an approximation ratio $(1-\mathrm{e}^{-c})/c$ for the greedy algorithm. Later, Vondrák (2010) gave the same approximation ratio $(1-\mathrm{e}^{-c})/c$ for the general matroid constraint. Many studies on submodular maximization utilize the curvature of a function to give an approximation ratio (Sviridenko et al., 2017; Friedrich et al., 2019).

Submodularity ratio was introduced by Das and Kempe (2011) and later there have been extensive works on weakly submodular maximization; matroid-constrained problem (Sun et al., 2020), introducing another definition for nonmonotone functions (Santiago and Yoshida, 2020), and streaming setting with the total size constraint (Elenberg et al., 2017).

Related to approximate submodularity (Horel and Singer, 2016), Hassidim and Singer (2017) gave an approximation algorithm for maximization of submodular functions with noisy oracle with size constraint. Approximately $k$-submodular functions are defined and dealt with by Nguyen and Thai (2020) (they called functions as "under noise"). In fact, Nguyen and Thai (2020) dealt with a generalized problem, that is, streaming setting with the total size constraint. Recently, Zheng et al. (2021) gave approximation algorithms for maximization of approximately $k$-submodular functions with the total size constraint and the individual size constraint.

Setting with the combination of concepts was also considered. For example, Bian et al. (2017) gave an analysis for non-submodular functions utilizing curvature and submodularity ratio.

Another generalization of submodular functions on domain is DR-submodular functions on the integer lattice, dealt with by Soma and Yoshida (2015).

## 3. Preliminaries

In this paper, the cardinality of a set $V$ is denoted by $n$. We often identify $[n]$ with $V$ for the notational convenience. For $\boldsymbol{x}, \boldsymbol{y} \in (k+1)^V$, we say $\boldsymbol{x}$ and $\boldsymbol{y}$ are *support-disjoint* if $\mathrm{supp}(\boldsymbol{x}) \cap \mathrm{supp}(\boldsymbol{y}) = \emptyset$. For $\boldsymbol{x}, \boldsymbol{y} \in (k+1)^V$, we write as $\boldsymbol{x} \preceq \boldsymbol{y}$ if $\boldsymbol{x} \sqcup \boldsymbol{y} = \boldsymbol{y}$ holds. For $e \in [n] \setminus \mathrm{supp}(\boldsymbol{x})$ and $i \in [k]$, we denote by $\boldsymbol{x} + (e, i)$ the vector in $(k+1)^V$ obtained from $\boldsymbol{x}$ by changing element corresponding to $e$ from 0 to $i$. Similarly, we denote by $\boldsymbol{x} - (e', i')$ the vector in $(k+1)^V$ obtained from $\boldsymbol{x}$ by setting $e'$-th element (that is, $i'$) to 0. For $e \in [n] \setminus \mathrm{supp}(\boldsymbol{x})$ and $i \in [k]$, $\Delta_{e,i} f(\boldsymbol{x}) := f(\boldsymbol{x} + (e,i)) - f(\boldsymbol{x})$. In this paper, we often abuse the notation for $\boldsymbol{x} \in (k+1)^V$ by identifying the vector and the set of its elements; i.e., $\boldsymbol{x} = \{(1, \boldsymbol{x}(1)), (2, \boldsymbol{x}(2)), \ldots, (n, \boldsymbol{x}(n))\}$. Note that for the above notation, it is necessary to be well-defined; e.g., for $\boldsymbol{x} \in (k+1)^V$ with $\boldsymbol{x}(e) \neq 0$, $\boldsymbol{x} + (e, i)$ is not well-defined. For support-disjoint $\boldsymbol{x}, \boldsymbol{y} \in (k+1)^V$, let $f(\boldsymbol{y} \mid \boldsymbol{x}) := f(\boldsymbol{x} \sqcup \boldsymbol{y}) - f(\boldsymbol{x})$. We sometimes write a singleton $\{e\}$ just as $e$. A characterization of $k$-submodularity was given by Ward and Živný (2016); $f$ is $k$-submodular if and only if $f$ is *orthant submodular* and *pairwise monotone*. A function $f$ is called orthant submodular if $\Delta_{(e,i)} f(\boldsymbol{x}) \geq \Delta_{(e,i)} f(\boldsymbol{y})$ holds for any $\boldsymbol{x}, \boldsymbol{y} \in (k+1)^V$ with $\boldsymbol{x} \preceq \boldsymbol{y}$ when $e \notin \mathrm{supp}(\boldsymbol{y})$ and $i \in [k]$, and $f$ is called pairwise monotone if $\Delta_{(e,i_1)} f(\boldsymbol{x}) + \Delta_{(e,i_2)} f(\boldsymbol{x}) \geq 0$ holds for any $\boldsymbol{x} \in (k+1)^V$ with $e \notin \mathrm{supp}(\boldsymbol{x})$ and $i_1 \neq i_2$. A $k$-submodular function $f$ is called *monotone* if for any $\boldsymbol{x}, \boldsymbol{y} \in (k+1)^V$ with $\boldsymbol{x} \preceq \boldsymbol{y}$, $f(\boldsymbol{x}) \leq f(\boldsymbol{y})$ holds.

**Curvature for $k$-Submodular Functions** *Curvature* for a monotone submodular function was introduced by Conforti and Cornuéjols (1984). Intuitively, curvature represents how far the function is from a modular function. For a monotone submodular function $f\colon 2^V \to \mathbb{R}_{\geq 0}$, the curvature $c$ is defined by $c = 1 - \min_{v \in V} \frac{f(V) - f(V \setminus \{v\})}{f(v)}$. If $f$ is monotone, then $0 \leq c \leq 1$ holds. There have been studies considering curvature for nonmonotone functions in, e.g., Friedrich et al. (2019). By using curvature for submodular functions, Conforti and Cornuéjols (1984) showed that the greedy algorithm gives $1/(1+c)$-approximation for monotone submodular maximization with the matroid constraint. By the extension of the curvature of submodular functions, a curvature of $k$-submodular functions can be considered. Note that if $k = 1$, then Definition 1 coincides with the curvature of submodular functions. Also note that quite similar curvature for normalized monotone $k$-multi-submodular functions was introduced in Santiago and Shepherd (2019).

**Definition 1** *For a $k$-submodular function $f\colon (k+1)^V \to \mathbb{R}_{\geq 0}$, the curvature $c$ is defined by*

$$c = 1 - \min_{i \in [k], e \in [n], \boldsymbol{a} \in (k+1)^{V \setminus \{e\}}} \frac{\Delta_{e,i} f(\boldsymbol{a})}{f((e,i))}.$$

**Weakly $k$-Submodular Functions** For set functions, as a measure for the closeness to submodular functions, *submodularity ratio* was introduced by Das and Kempe (2011). Deeply related to the submodularity ratio, weak submodularity was defined as follows. (Note that the below definition of weak submodularity corresponds with the definition of submodularity ratio which can be seen, for example, in Chen et al. (2018).) A nonnegative monotone function $f\colon 2^V \to \mathbb{R}_{\geq 0}$ is called $\gamma$-weakly submodular if for all disjoint $S, T \subseteq V$, $\sum_{e \in T} f(e \mid S) \geq \gamma f(T \mid S)$ holds. A generalized concept of weak submodularity to weak $k$-submodularity can be considered as follows.

**Definition 2** *A nonnegative monotone function $f\colon (k+1)^V \to \mathbb{R}_{\geq 0}$ is called $\gamma$-weakly $k$-submodular if for all support-disjoint $\boldsymbol{x}, \boldsymbol{y} \in (k+1)^V$, it holds that*

$$\sum_{(e,i) \in \boldsymbol{y}} f((e,i) \mid \boldsymbol{x}) \geq \gamma f(\boldsymbol{y} \mid \boldsymbol{x}).$$

**Approximately $k$-Submodular Functions** In Horel and Singer (2016), apporoximately submodular functions were defined as follows. For $\varepsilon > 0$, $f\colon 2^V \to \mathbb{R}$ is $\varepsilon$-approximately submodular if there exists a submodular function $\tilde{f}\colon 2^V \to \mathbb{R}$ satisfying $(1-\varepsilon)\tilde{f}(S) \leq f(S) \leq (1+\varepsilon)\tilde{f}(S)$ for all $S \subseteq V$. As a generalization of approximate submodularity, approximate $k$-submodularity was introduced by Nguyen and Thai (2020) in which the notion is considered as $k$-submodularity "with noise".

**Definition 3 (Nguyen and Thai (2020))** *For $\varepsilon > 0$, $f\colon (k+1)^V \to \mathbb{R}$ is $\varepsilon$-approximately $k$-submodular if there exists a $k$-submodular function $\tilde{f}$ satisfying*

$$(1-\varepsilon)\tilde{f}(\boldsymbol{x}) \leq f(\boldsymbol{x}) \leq (1+\varepsilon)\tilde{f}(\boldsymbol{x})$$

*for all $\boldsymbol{x} \in (k+1)^V$.*

**Greedy Algorithm** In this paper, we utilize the frameworks of greedy algorithm and residual random greedy algorithm (Buchbinder et al., 2014). For residual random greedy algorithm, we explain details in Section 5. Here, we describe the framework of greedy algorithm for $f\colon (k+1)^V \to \mathbb{R}$ in Algorithm 1. For the matroid constraint, $B$ is equal to the rank of an input matroid by Lemma 1 in Sakaue (2017), and for the individual size constraint, $B$ is equal to the sum of the input constants $B_i$ for $i \in [k]$. In Algorithm 1, for each step $j = 1, \ldots, B$, we select $(e^{(j)}, i^{(j)})$ to add to $\boldsymbol{s}^{(j-1)}$.

**Algorithm 1** Greedy Algorithm for the (Approximately) $k$-Submodular Maximization Problem

$\boldsymbol{s}^{(0)} \leftarrow \mathbf{0}$.
**for** $j = 1, \ldots, B$ **do**
  Let $(e^{(j)}, i^{(j)}) \in ([n] \setminus \mathrm{supp}(\boldsymbol{s}^{(j-1)})) \times [k]$ be a pair such that $\boldsymbol{s}^{(j-1)} + (e^{(j)}, i^{(j)})$ is feasible maximizing $f(\boldsymbol{s}^{(j-1)} + (e^{(j)}, i^{(j)}))$.
  $\boldsymbol{s}^{(j)} \leftarrow \boldsymbol{s}^{(j-1)} + (e^{(j)}, i^{(j)})$.
**end for**
Return $\boldsymbol{s}^{(B)}$.

## 4. Bounded Curvature

In this section, we give approximation ratios for greedy algorithms on monotone nonnegative $k$-submodular maximization with the matroid constraint or the individual size constraint by utilizing the curvature of a $k$-submodular function (Definition 1).

### 4.1. Matroid Constraint

We give the following approximation ratio of greedy algorithm using curvature for the problem of monotone $k$-submodular maximization with the matroid constraint. Note that the approximation ratio in Theorem 1 matches that for the matroid constrained case given by Sakaue (2017) if $c = 1$. In the case of $c < 1$, our approximation ratio is better than $1/2$. Since the matroid constraint is the generalization of the total size constraint, we have the same approximation ratio result for the problem with the total size constraint.

**Theorem 1** *The greedy algorithm yields an approximation ratio $1/(1+c)$ for monotone $k$-submodular maximization with the matroid constraint, where $c$ is the curvature of an input function.*

**Proof** For the proof of Theorem 1, we utilize the following lemma by Sakaue (2017). ($\mathcal{B}$ is a base family of a matroid $(V, \mathcal{I})$.)

**Lemma 1 (Sakaue (2017))** *Let $S \in \mathcal{I}$, $T \in \mathcal{B}$ with $S \subsetneq T$, and $e \notin S$ with $S \cup \{e\} \in \mathcal{I}$. Then, there exists $e' \in T \setminus S$ with $(T \setminus \{e'\}) \cup \{e\} \in \mathcal{B}$.*

Proof follows Sakaue (2017); the analogical discussions can be seen in, e.g., Ohsaka and Yoshida (2015); Iwata et al. (2016); Ward and Živný (2016). Let $\boldsymbol{o}$ be an optimal solution. We define $\boldsymbol{s}^{(0)}, \boldsymbol{s}^{(1)}, \ldots, \boldsymbol{s}^{(B)}, \boldsymbol{o}^{(0)}, \boldsymbol{o}^{(1)}, \ldots, \boldsymbol{o}^{(B)}$ as $\boldsymbol{s}^{(0)} = \mathbf{0}$, $\boldsymbol{o}^{(0)} = \boldsymbol{o}$, and as for each step $j \in [B]$, we add $(e^{(j)}, i^{(j)})$ which is selected in step $j$ of the greedy algorithm to obtain $\boldsymbol{s}^{(j)}$ from $\boldsymbol{s}^{(j-1)}$. Let $o^{(j)} := e^{(j)}$ if $e^{(j)} \in \mathrm{supp}(\boldsymbol{o}^{(j-1)})$, otherwise, set $o^{(j)}$ such that $(\mathrm{supp}(\boldsymbol{o}^{(j-1)}) \setminus \{o^{(j)}\}) \cup \{e^{(j)}\} \in \mathcal{B}$ holds. Note that $o^{(j)}$ satisfying the above property exists by virtue of Lemma 1. We make $\boldsymbol{o}^{(j)}$ from $\boldsymbol{o}^{(j-1)}$ by setting $o^{(j)}$-th element to 0 and setting $e^{(j)}$-th element to $i^{(j)}$. Then, $\boldsymbol{s}^{(B)} = \boldsymbol{o}^{(B)}$ holds and we denote this by $\boldsymbol{s}$.

We show that for each $j \in [B]$,

$$c(f(\boldsymbol{s}^{(j)}) - f(\boldsymbol{s}^{(j-1)})) \geq f(\boldsymbol{o}^{(j-1)}) - f(\boldsymbol{o}^{(j)})$$

holds. For each $j \in [B]$, let $\boldsymbol{o}^{(j-1/2)}$ be a vector made from $\boldsymbol{o}^{(j-1)}$ by setting $o^{(j)}$-th element to 0, and let $y^{(j)} := \Delta_{e^{(j)}, i^{(j)}} f(\boldsymbol{s}^{(j-1)})$, $a^{(j-1/2)} := \Delta_{o^{(j)}, \boldsymbol{o}^{(j-1)}(o^{(j)})} f(\boldsymbol{o}^{(j-1/2)})$, and $a^{(j)} := \Delta_{e^{(j)}, i^{(j)}} f(\boldsymbol{o}^{(j-1/2)})$. Note that $f(\boldsymbol{s}^{(j)}) - f(\boldsymbol{s}^{(j-1)}) = y^{(j)}$ and $f(\boldsymbol{o}^{(j-1)}) - f(\boldsymbol{o}^{(j)}) = a^{(j-1/2)} - a^{(j)}$.

Since $(e^{(j)}, i^{(j)})$ is chosen by the greedy algorithm, $y^{(j)} \geq \Delta_{o^{(j)}, \boldsymbol{o}^{(j-1)}(o^{(j)})} f(\boldsymbol{s}^{(j-1)})$ holds. Since $\boldsymbol{s}^{(j-1)} \preceq \boldsymbol{o}^{(j-1/2)}$, we have that $\Delta_{o^{(j)}, \boldsymbol{o}^{(j-1)}(o^{(j)})} f(\boldsymbol{s}^{(j-1)}) \geq a^{(j-1/2)}$. Hence, $a^{(j-1/2)} \leq y^{(j)}$ holds. Now, by the definition of curvature, we have that $a^{(j)} = \Delta_{e^{(j)}, i^{(j)}} f(\boldsymbol{o}^{(j-1/2)}) \geq (1-c)\Delta_{e^{(j)}, i^{(j)}} f(\boldsymbol{s}^{(j-1)}) = (1-c)y^{(j)}$. Therefore, we get that $a^{(j-1/2)} - a^{(j)} \leq cy^{(j)}$, which derives the inequality. Then, we have that $f(\boldsymbol{o}) - f(\boldsymbol{s}) = \sum_{j \in [B]} (f(\boldsymbol{o}^{(j-1)}) - f(\boldsymbol{o}^{(j)})) \leq c \sum_{j \in [B]} (f(\boldsymbol{s}^{(j)}) - f(\boldsymbol{s}^{(j-1)})) = c(f(\boldsymbol{s}) - f(\mathbf{0})) \leq cf(\boldsymbol{s})$, which implies that $f(\boldsymbol{s}) \geq \frac{1}{1+c} f(\boldsymbol{o})$. ■

### 4.2. Individual Size Constraint

For the problem of $k$-submodular maximization with bounded curvature with the individual size constraint, we obtain the following approximation ratio, which is better than (resp. matches) that given by Ohsaka and Yoshida (2015) if $c < 1$ (resp. $c = 1$).

**Theorem 2** *The greedy algorithm yields an approximation ratio $1/(1+2c)$ for monotone $k$-submodular maximization with the individual size constraint, where $c$ is the curvature of an input function.*

**Proof** Proof follows Ohsaka and Yoshida (2015). We show for each $j \in [B]$ that

$$2c(f(\boldsymbol{s}^{(j)}) - f(\boldsymbol{s}^{(j-1)})) \geq f(\boldsymbol{o}^{(j-1)}) - f(\boldsymbol{o}^{(j)})$$

holds. Let $S_i^{(j)} := \mathrm{supp}_i(\boldsymbol{o}^{(j-1)}) \setminus \mathrm{supp}_i(\boldsymbol{s}^{(j-1)})$. For step $j$, if $e^{(j)}$ is in $S_{i'}^{(j)}$ for some $i' \neq i^{(j)}$, then we define as follows: $\boldsymbol{o}^{(j-1/2)}$ be a vector obtained from $\boldsymbol{o}^{(j-1)}$ by setting 0 to $o^{(j)}$-th and $e^{(j)}$-th elements and $\boldsymbol{o}^{(j)}$ be a vector obtained from $\boldsymbol{o}^{(j-1/2)}$ by setting $i^{(j)}$ to $e^{(j)}$-th element and $i'$ to $o^{(j)}$-th element. Otherwise, we define as follows: $\boldsymbol{o}^{(j-1/2)}$ be a vector obtained from $\boldsymbol{o}^{(j-1)}$ by setting 0 to $e^{(j)}$-th element and $\boldsymbol{o}^{(j)}$ be a vector obtained from $\boldsymbol{o}^{(j-1/2)}$ by setting $i^{(j)}$ to $e^{(j)}$-th element.

Let us denote $a^{(j-1/2)} = \Delta_{o^{(j)},i^{(j)}} f(\boldsymbol{o}^{(j-1/2)})$, $a^{(j)} = \Delta_{e^{(j)},i^{(j)}} f(\boldsymbol{o}^{(j-1/2)})$, $b^{(j-1/2)} = \Delta_{e^{(j)},i'} f(\boldsymbol{o}^{(j-1/2)})$, and $b^{(j)} = \Delta_{o^{(j)},i'} f(\boldsymbol{o}^{(j-1/2)})$. Since we use greedy method in each step, $y^{(j)} \geq \Delta_{o^{(j)},i^{(j)}} f(\boldsymbol{s}^{(j-1)})$ and $y^{(j)} \geq \Delta_{e^{(j)},i'} f(\boldsymbol{s}^{(j-1)})$ hold. From $\boldsymbol{s}^{(j-1)} \preceq \boldsymbol{o}^{(j-1/2)}$ and orthant submodularity, $\Delta_{e^{(j)},i'} f(\boldsymbol{s}^{(j-1)}) \geq a^{(j-1/2)}$ and $\Delta_{e^{(j)},i'} f(\boldsymbol{s}^{(j-1)}) \geq b^{(j-1/2)}$ hold. Thus, $2y^{(j)} \geq a^{(j-1/2)} + b^{(j-1/2)}$ holds. By the definition of curvature of $k$-submodular functions, $a^{(j)} \geq (1-c)y^{(j)}$ and $b^{(j)} \geq (1-c)y^{(j)}$ hold. Therefore, we obtain $2cy^{(j)} \geq a^{(j-1/2)} - a^{(j)} + b^{(j-1/2)} - b^{(j)}$. ■

## 5. Weakly $k$-Submodular Functions

In this section, we give an expected approximation ratio of residual random greedy algorithms for problems of maximizing a $\gamma$-weakly $k$-submodular function with the matroid constraint and individual size constraint. We utilize the algorithm framework of residual random greedy algorithm, appearing in Buchbinder et al. (2014). Chen et al. (2018) applied this residual random greedy algorithm to weakly submodular function maximization with matroid constraint. We apply the residual random greedy framework for maximizing weakly $k$-submodular functions. The general framework of the algorithm is in Algorithm 2.

Note that we can set the iteration number $B$ in advance since for both matroid-constrained problem and individual size-constrained problem $B$ can be decided from the input constraint due to the monotonicity of the input function. Algorithm 2 can be used for the problems with the matroid constraint or the individual size constraint. The techniques and lemmas are $k$-submodular counterpart for those in Chen et al. (2018) on the maximization of weakly submodular functions. Chen et al. (2018) showed that the residual random greedy algorithm enjoys the approximation ratio of $(1 + 1/\gamma)^{-2}$ for maximization of monotone $\gamma$-weakly

**Algorithm 2** Common Framework of Residual Random Greedy Algorithm for the (Weakly) $k$-Submodular Maximization Problem

$\boldsymbol{s}^{(0)} \leftarrow \mathbf{0}$.
**for** $j = 1, \ldots, B$ **do**
  Construct $\boldsymbol{m}^{(j)} \in (k+1)^V$ such that $\boldsymbol{s}^{(j-1)} \sqcup \boldsymbol{m}^{(j)}$ is a maximal feasible solution maximizing $\sum_{(e,i)\in \boldsymbol{m}^{(j)}} \Delta_{(e,i)} f(\boldsymbol{s}^{(j-1)})$.
  Pick up $(e^{(j)}, i^{(j)})$ from $\boldsymbol{m}^{(j)}$ uniformly at random.
  $\boldsymbol{s}^{(j)} \leftarrow \boldsymbol{s}^{(j-1)} + (e^{(j)}, i^{(j)})$.
**end for**
Return $\boldsymbol{s}^{(B)}$.

submodular function with the matroid constraint. We list the counterparts of Lemma 3.3, Observation 3.4, Corollary 3.5, Theorem 3.6, and Theorem 1.1 in Chen et al. (2018).

In order to apply the proof framework for submodular maximization with matroid constraint in Chen et al. (2018) to the $k$-submodular maximization problems, we have to construct $g_j : \boldsymbol{m}^{(j)} \to \boldsymbol{o}^{(j-1)}$ for each $j \in [B]$ (and $\boldsymbol{o}^{(0)}, \boldsymbol{o}^{(1)}, \ldots, \boldsymbol{o}^{(B)} \in (k+1)^V$ are constructed recursively) such that the followings hold for each $j \in [B]$: (i) $\boldsymbol{s}^{(j)}$ and $\boldsymbol{o}^{(j)}$ are support-disjoint, and $\boldsymbol{s}^{(j)} \sqcup \boldsymbol{o}^{(j)}$ is a maximal feasible solution, (ii) $\boldsymbol{o}^{(0)}$ is an optimal solution, and (iii) there exists a bijection $g_j : \boldsymbol{m}^{(j)} \to \boldsymbol{o}^{(j-1)}$ such that for each $(e,i) \in \boldsymbol{m}^{(j)}$, $g_j((e,i)) \sqcup \{(e,i)\} \neq \emptyset$ and that $((\boldsymbol{s}^{(j-1)} \sqcup \boldsymbol{o}^{(j-1)}) \setminus g_j((e,i))) \cup \{(e,i)\}$ is a maximal feasible solution. If $\boldsymbol{s}^{(j)}, \boldsymbol{m}^{(j)}, \boldsymbol{o}^{(j)}$ satisfy the above three conditions, then the same proof framework as Chen et al. (2018) can be utilized. In the following, we show that, for each problem, we can take $\boldsymbol{s}^{(j)}, \boldsymbol{m}^{(j)}, \boldsymbol{o}^{(j)}$ satisfying the above conditions.

### 5.1. On Common Framework

Suppose that we have $\boldsymbol{m}^{(j)}, \boldsymbol{o}^{(j)}, g_j$ with the desired property. Given that $\boldsymbol{m}^{(j)}$ and $\boldsymbol{o}^{(j-1)}$ are fixed for some $j$, we can see that $g_j((e^{(j)}, i^{(j)}))$ is a uniformly random sample from $\boldsymbol{o}^{(j-1)}$, since $(e^{(j)}, i^{(j)})$ is chosen from $\boldsymbol{m}^{(j)}$ uniform randomly and that $g_j$ is a one-to-one mapping. Hence, each $\boldsymbol{o}^{(j)}$ is a random subset of $\boldsymbol{o}^*$ of size $B - j$. The following statements are the counterparts of Lemma 3.3, Observation 3.4, Corollary 3.5, Theorem 3.6, and Theorem 1.1 in Chen et al. (2018), respectively.

**Lemma 2 (Counterpart of Lemma 3.3 in Chen et al. (2018))** *For all* $0 \leq j \leq B$,

$$\mathbb{E}[f(\boldsymbol{o}^{(j)})] \geq \left(1 - \left(\frac{j+1}{B+1}\right)^{\gamma}\right) f(\boldsymbol{o}^*).$$

**Lemma 3 (Counterpart of Observation 3.4 in Chen et al. (2018))** *For all* $j \in [B]$,

$$\mathbb{E}[f(\boldsymbol{s}^{(j)})] - \mathbb{E}[f(\boldsymbol{s}^{(j-1)})] \geq \gamma \cdot \frac{\mathbb{E}[f(\boldsymbol{o}^{(j-1)} \mid \boldsymbol{s}^{(j-1)})]}{B - j + 1}.$$

**Lemma 4 (Counterpart of Corollary 3.5 in Chen et al. (2018))** *For all* $j \in [B]$,

$$\mathbb{E}[f(\boldsymbol{s}^{(j)})] - \mathbb{E}[f(\boldsymbol{s}^{(j-1)})] \geq \gamma \cdot \frac{[1 - (j/(B+1))]^{\gamma} f(\boldsymbol{o}^*) - \mathbb{E}[f(\boldsymbol{s}^{(j-1)})]}{B - j + 1}.$$

By discussion analogous to that in Chen et al. (2018), we obtain an approximation factor $(1+1/\gamma)^{-2} - O(B^{-1})$ (Counterpart of Theorem 3.6 in Chen et al. (2018)). By applying the corresponding discussion in Appendix B in Chen et al. (2018), we obtain the following.

**Theorem 3 (Counterpart of Theorem 1.1 in Chen et al. (2018))** *Residual random greedy algorithm (Algorithm 2) has an expected approximation ratio $(1+1/\gamma)^{-2}$ for the input problem of monotone $\gamma$-weakly $k$-submodular maximization.*

In the following, we claim that for the matroid-constrained and the individual size-constrained problems, Algorithm 2 can be used by change of the procedure to obtain $\boldsymbol{m}^{(j)}$. (On removal of term $\mathrm{O}(B^{-1})$, analogous to Chen et al. (2018), we add enough number of dummy elements $V'$ to $V$ and set a new $k$-submodular function $f' \colon (k+1)^{V \cup V'} \to \mathbb{R}_{\geq 0}$ from the original function $f \colon (k+1)^V \to \mathbb{R}_{\geq 0}$ such that for $\boldsymbol{x}' \in (k+1)^{V' \cup V}$, $f'(\boldsymbol{x}')$ is equal to $f(\boldsymbol{x})$, where $\boldsymbol{x}$ is the element in $(k+1)^{\bar{V}}$ having the same elements for coordinates corresponding to $V$. Analogously, for the individual size-constrained problem, we add enough dummy elements $V'$ to $V$ and increase $B_i$ or keep as it is for $i = 1, \ldots, k$ such that $\sum_{i=1}^{k} B_i$ is increased by $|V'|$. The new function $f'$ is defined the same as that in the matroid-constrained case.)

### 5.2. Matroid Constraint

In order to use the framework of Algorithm 2 for the matroid-constrained problem, we confirm how to construct $\boldsymbol{m}^{(j)}$ in the algorithm and $\boldsymbol{o}^{(j)}$ in the proof. Let $M$ denote the matroid of the given constraint. Then, in the algorithm, for step $j$, let $M^{(j)}$ be a matroid obtained by contraction of $\mathrm{supp}(\boldsymbol{s}^{(j-1)})$ for $M$. (For a matroid $(V, \mathcal{I})$, the rank function $r \colon 2^V \to \mathbb{Z}_{\geq 0}$ is defined by $r(V') := \max_{V'' \subseteq V'} \{|V''| \mid V'' \in \mathcal{I}\}$. For $V' \subseteq V$, contraction of $V'$ for $(V, \mathcal{I})$ is also a matroid with the ground set $V \setminus V'$ and the rank function $\hat{r}(V'') := r(V'' \cup V') - r(V')$ $(V'' \subseteq V \setminus V')$.) For each $e \in [n] \setminus \mathrm{supp}(\boldsymbol{s}^{(j-1)})$, we set $w_e^{\langle i \rangle} := \max_{i \in [k]} w(e, i)$. Then, let $\boldsymbol{m}^{(j)} \in (k+1)^V$ be a vector corresponding to a maximum weight independent set (base) for $M^{(j)}$. Note that this step can be done in polynomial time since we can find a maximum weight independent set for a given matroid in polynomial time by greedy algorithm by virtue of Edmonds (1971). The function $g_j$ must be a one-to-one function that satisfies $g_j((e,i)) \sqcup \{(e,i)\} \neq \emptyset$ and is a base. Existence of such $g_j$ is guaranteed by the following theorem by Brualdi (1969). Note that this logic is same as the argument in Chen et al. (2018).

**Theorem 4 (Brualdi (1969))** *Let $M$ be a matroid and $A_1, A_2$ are bases of $M$. Then, there exists a bijection $g \colon A_1 \to A_2$ such that for each $i \in A_1$, $A_1 \cup \{g(i)\} \setminus \{i\}$ is a base of $M$.*

Analogous to the case for submodular functions with matroid constraints in Chen et al. (2018), by Theorem 4, we can show that we can construct $\boldsymbol{o}^{(j)}$ with the desired property by the following argument. Note that $\boldsymbol{o}^{(0)} = \boldsymbol{o}^*$. Suppose $\boldsymbol{o}^{(j-1)}$ has already been constructed with the desired property. Then, since $\mathrm{supp}(\boldsymbol{s}^{(j-1)} \sqcup \boldsymbol{o}^{(j-1)})$ and $\mathrm{supp}(\boldsymbol{s}^{(j-1)} \sqcup \boldsymbol{m}^{(j)})$ are bases, we can take $g_j \colon \boldsymbol{m}^{(j)} \to \boldsymbol{o}^{(j-1)}$ satisfying the desired property. Therefore, we can use the same-type logic as that in Chen et al. (2018) and obtain the following.

**Theorem 5** *Residual random greedy algorithm (Algorithm 2) has an expected approximation ratio $(1+1/\gamma)^{-2}$ for the monotone $\gamma$-weakly $k$-submodular maximization problem with the matroid constraint.*

### 5.3. Individual Size Constraint

We can determine a vector $\boldsymbol{m}^{(j)}$ maximizing the sum of unit increasing cost and satisfying individual size constraint by using the technique of maximum weight $b$-matching. In maximum weight $b$-matching problem, given a graph $G = (V, E)$, edge-weight function $w \colon E \to \mathbb{R}_{\geq 0}$, and $b \colon V \to \mathbb{Z}_{\geq 0}$, find a maximum weight edge subset such that for each $v \in V$, degree of $v$ is less than or equal to $b(v)$. Precisely, in step $j$, we make an auxiliary weighted bipartite graph such that (i) one vertex set $V_1$ is corresponding to each $e \in [n] \setminus \mathrm{supp}(\boldsymbol{s}^{(j-1)})$, (ii) the other vertex set $V_2$ is corresponding to each $i \in [k]$, (iii) for each edge, the weight is $f((e, i) \mid \boldsymbol{s}^{(j-1)})$, and that (iv) for $i \in [k]$, $b(i)$ is $B_i - |\{e \in [n] \mid (e, i) \in \boldsymbol{s}^{(j-1)}\}|$. Since both $\boldsymbol{s}^{(j-1)} + \boldsymbol{o}^{(j-1)}$ and $\boldsymbol{s}^{(j-1)} + \boldsymbol{m}^{(j)}$ have the desired property, we can set a bijection $g_j \colon \boldsymbol{m}^{(j)} \to \boldsymbol{o}^{(j-1)}$ such that for each $(e, i) \in \boldsymbol{m}^{(j)}$, $g_j((e, i))$ has the same element $i \in [k]$. We set $\boldsymbol{o}^{(j)} = \boldsymbol{o}^{(j-1)} - g_j((e^{(j)}, i^{(j)}))$. Since maximum weight $b$-matching can be solved in polynomial time (see, e.g., Edmonds (1965)), the algorithm runs in polynomial time. By setting as above, we obtain the following.

**Theorem 6** *Residual random greedy algorithm (Algorithm 2) has an expected approximation ratio $(1 + 1/\gamma)^{-2}$ for the monotone $\gamma$-weakly $k$-submodular maximization problem with the individual size constraint.*

## 6. Approximately $k$-Submodular Functions

In this section, we utilize the concept of approximate $k$-submodularity (Definition 3) and give an approximation ratio for the matroid-constrained problem. Corresponding to the approximation ratio $\frac{1}{2}\left(\frac{1-\varepsilon}{1+\varepsilon}\right)\frac{1}{1+\frac{\varepsilon B}{1-\varepsilon}}$ for approximately submodular maximization obtained by Horel and Singer (2016), we show an approximation ratio for approximately $k$-submodular functions. We give the same approximation ratio as Zheng et al. (2021)'s result on total size constrained problem for the matroid-constrained problem, which includes the total size constrained problem.

**Theorem 7** *Let $(V, \mathcal{I})$ be an input matroid, $f \colon 2^V \to \mathbb{R}_{\geq 0}$ be a monotone $\varepsilon$-approximately $k$-submodular function, and $\boldsymbol{s}$ be the output of the greedy algorithm. Then, it holds that*

$$f(\boldsymbol{s}) \geq \frac{1}{2}\left(\frac{1-\varepsilon}{1+\varepsilon}\right)\frac{1}{1+\frac{\varepsilon B}{1-\varepsilon}} \max_{\boldsymbol{x} \colon \mathrm{supp}(\boldsymbol{x}) \in \mathcal{I}} f(\boldsymbol{x}).$$

**Proof** The proof here is analogous to that in Horel and Singer (2016) and Zheng et al. (2021). We define $\boldsymbol{s}^{(0)}, \boldsymbol{s}^{(1)}, \ldots, \boldsymbol{s}^{(B)}, \boldsymbol{o}^{(0)}, \boldsymbol{o}^{(1)}, \ldots, \boldsymbol{o}^{(B)}$ same as those in the proof of Theorem 1, that is, $\boldsymbol{s}^{(0)} = \mathbf{0}$, $\boldsymbol{o}^{(0)}$ is an optimal solution, and as for each step $j \in [B]$, we add $(e^{(j)}, i^{(j)})$, selected by the greedy algorithm with $\boldsymbol{s}^{(j-1)}$ to obtain $\boldsymbol{s}^{(j)}$. Also $o^{(j)}$ is the samely defined as in the proof of Theorem 1; $o^{(j)} = e^{(j)}$ if $e^{(j)} \in \mathrm{supp}(\boldsymbol{o}^{(j-1)})$, otherwise, set $o^{(j)}$ such that $(\mathrm{supp}(\boldsymbol{o}^{(j-1)}) \setminus \{o^{(j)}\}) \cup \{e^{(j)}\} \in \mathcal{B}$ holds. Let $\boldsymbol{t}^{(j)}$ be a vector in $(k+1)^V$ obtained by adding $\boldsymbol{o}^{(j-1)}(o^{(j)})$ at $o^{(j)}$ to $\boldsymbol{s}^{(j-1)}$. Then since $f(\boldsymbol{s}^{(j)}) \geq f(\boldsymbol{t}^{(j)})$, we obtain $\frac{1+\varepsilon}{1-\varepsilon}\tilde{f}(\boldsymbol{s}^{(j)}) \geq \tilde{f}(\boldsymbol{t}^{(j)})$. By the orthant submodulatiry and monotonicity, $\frac{1+\varepsilon}{1-\varepsilon}\tilde{f}(\boldsymbol{s}^{(j)}) - \tilde{f}(\boldsymbol{s}^{(j-1)}) \geq \tilde{f}(\boldsymbol{t}^{(j)}) - \tilde{f}(\boldsymbol{s}^{(j-1)}) = \Delta_{o^{(j)}, \boldsymbol{o}^{(j-1)}(o^{(j)})}\tilde{f}(\boldsymbol{s}^{(j-1)}) \geq \Delta_{o^{(j)}, \boldsymbol{o}^{(j-1)}(o^{(j)})}\tilde{f}(\boldsymbol{o}^{(j-\frac{1}{2})}) \geq$

$\Delta_{o^{(j)},\boldsymbol{o}^{(j-1)}(o^{(j)})}\tilde{f}(\boldsymbol{o}^{(j-\frac{1}{2})}) - \Delta_{e^{(j)},i^{(j)}}\tilde{f}(\boldsymbol{o}^{(j-\frac{1}{2})}) = \tilde{f}(\boldsymbol{o}^{(j-1)}) - \tilde{f}(\boldsymbol{o}^{(j)})$. By taking the summation from $j = 1$ to $B$, $\frac{2\varepsilon}{1-\varepsilon}\sum_{j=1}^{B}\tilde{f}(\boldsymbol{s}^{(j)}) + \tilde{f}(\boldsymbol{s}^{(B)}) - \tilde{f}(\boldsymbol{s}^{(0)}) \geq \tilde{f}(\boldsymbol{o}^{(0)}) - \tilde{f}(\boldsymbol{o}^{(B)})$. Thus, we obtain $2\left(1 + \frac{\varepsilon B}{1-\varepsilon}\right)\tilde{f}(\boldsymbol{s}^{(B)}) \geq \tilde{f}(\boldsymbol{o}^{(0)})$. By using the $\varepsilon$-approximation, we obtain the statement of the theorem. ■

## 7. Concluding Remarks

In this paper, we utilized the three concepts, curvature for $k$-submodular functions, weak $k$-submodularity, and approximate $k$-submodularity, and analyzed approximation ratios of the greedy and residual random greedy algorithms. Concretely, (i) for $k$-submodular functions with bounded curvature, we gave an approximation ratio of the greedy algorithm for the problem with the matroid or the individual size constraints, (ii) for weakly $k$-submodular functions, we gave an approximation ratio of the residual random greedy algorithm with the matroid or the individual size constraints, and (iii) for approximately $k$-submodular functions, we gave an approximation ratio of the greedy algorithm for the matroid-constrained problem.

## References


Andrew An Bian, Joachim M Buhmann, Andreas Krause, and Sebastian Tschiatschek. Guarantees for greedy maximization of non-submodular functions with applications. In *ICML*, pages 498–507, 2017.

Richard A Brualdi. Comments on bases in dependence structures. *Bull. Aust. Math. Soc.*, 1 (2):161–167, 1969.

Niv Buchbinder, Moran Feldman, Joseph Naor, and Roy Schwartz. Submodular maximization with cardinality constraints. In *SODA*, pages 1433–1452, 2014.

Lin Chen, Moran Feldman, and Amin Karbasi. Weakly submodular maximization beyond cardinality constraints: Does randomization help greedy? In *ICML*, pages 804–813, 2018.

David A Cohen, Martin C Cooper, Peter G Jeavons, and Andrei A Krokhin. The complexity of soft constraint satisfaction. *Artif. Intell.*, 170(11):983–1016, 2006.

Michele Conforti and Gérard Cornuéjols. 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.

Abhimanyu Das and David Kempe. Submodular meets spectral: greedy algorithms for subset selection, sparse approximation and dictionary selection. In *ICML*, pages 1057–1064, 2011.

Jack Edmonds. Paths, trees, and flowers. *Can. J. Math.*, 17:449–467, 1965.

Jack Edmonds. Matroids and the greedy algorithm. *Math. Program.*, 1(1):127–136, 1971.

Ethan Elenberg, Alexandros G Dimakis, Moran Feldman, and Amin Karbasi. Streaming weak submodularity: Interpreting neural networks on the fly. In *NIPS*, pages 4044–4054, 2017.

Tobias Friedrich, Andreas Göbel, Frank Neumann, Francesco Quinzan, and Ralf Rothenberger. Greedy maximization of functions with bounded curvature under partition matroid constraints. In *AAAI*, pages 2272–2279, 2019.

Avinatan Hassidim and Yaron Singer. Submodular optimization under noise. In *COLT*, pages 1069–1122, 2017.

Thibaut Horel and Yaron Singer. Maximization of approximately submodular functions. In *NIPS*, pages 3045–3053, 2016.

Anna Huber and Vladimir Kolmogorov. Towards minimizing $k$-submodular functions. In *ISCO*, pages 451–462, 2012.

Satoru Iwata, Shin-ichi Tanigawa, and Yuichi Yoshida. Improved approximation algorithms for $k$-submodular function maximization. In *SODA*, pages 404–413, 2016.

Stefanie Jegelka and Jeff Bilmes. Submodularity beyond submodular energies: coupling edges in graph cuts. In *CVPR*, pages 1897–1904, 2011.

David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *KDD*, pages 137–146, 2003.

Andreas Krause, Ajit Singh, and Carlos Guestrin. Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies. *J. Mach. Learn. Res.*, 9(2), 2008.

Hui Lin and Jeff Bilmes. A class of submodular functions for document summarization. In *ACL-HLT*, pages 510–520, 2011.

George L Nemhauser and Laurence A Wolsey. Best algorithms for approximating the maximum of a submodular set function. *Math. Oper. Res.*, 3(3):177–188, 1978.

George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions—i. *Math. Program.*, 14(1):265–294, 1978.

Lan Nguyen and My T Thai. Streaming $k$-submodular maximization under noise subject to size constraint. In *ICML*, pages 7338–7347, 2020.

Naoto Ohsaka and Yuichi Yoshida. Monotone $k$-submodular function maximization with size constraints. In *NIPS*, pages 694–702, 2015.

Hiroki Oshima. Improved randomized algorithm for $k$-submodular function maximization. *SIAM J. Discrete Math.*, 35(1):1–22, 2021.

Chao Qian, Jing-Cheng Shi, Ke Tang, and Zhi-Hua Zhou. Constrained monotone $k$-submodular function maximization using multiobjective evolutionary algorithms with theoretical guarantee. *IEEE Trans. Evol. Comput.*, 22(4):595–608, 2017.

Shinsaku Sakaue. On maximizing a monotone $k$-submodular function subject to a matroid constraint. *Discrete Optim.*, 23:105–113, 2017.

Richard Santiago and F Bruce Shepherd. Multivariate submodular optimization. In *ICML*, pages 5599–5609, 2019.

Richard Santiago and Yuichi Yoshida. Weakly submodular function maximization using local submodularity ratio. In *ISAAC*, 2020.

Ajit Singh, Andrew Guillory, and Jeff Bilmes. On bisubmodular maximization. In *AISTATS*, pages 1055–1063, 2012.

Tasuku Soma. No-regret algorithms for online $k$-submodular maximization. In *AISTATS*, pages 1205–1214, 2019.

Tasuku Soma and Yuichi Yoshida. A generalization of submodular cover via the diminishing return property on the integer lattice. In *NIPS*, pages 847–855, 2015.

Xin Sun, Dachuan Xu, Dongmei Zhang, and Yang Zhou. An adaptive algorithm for maximization of non-submodular function with a matroid constraint. In *CSoNet*, pages 3–13, 2020.

Maxim Sviridenko. A note on maximizing a submodular set function subject to a knapsack constraint. *Oper. Res. Lett.*, 32(1):41–43, 2004.

Maxim Sviridenko, Jan Vondrák, and Justin Ward. Optimal approximation for submodular and supermodular optimization with bounded curvature. *Math. Oper. Res.*, 42(4):1197–1218, 2017.

Zhongzheng Tang, Chenhao Wang, and Hau Chan. On maximizing a monotone $k$-submodular function under a knapsack constraint. *arXiv preprint arXiv:2105.15159*, 2021.

Jan Vondrák. Submodularity and curvature: The optimal algorithm (combinatorial optimization and discrete algorithms). *RIMS Kokyuroku Bessatsu*, 23:253–266, 2010.

Justin Ward and Stanislav Živný. Maximizing bisubmodular and $k$-submodular functions. In *SODA*, pages 1468–1481, 2014.

Justin Ward and Stanislav Živný. Maximizing $k$-submodular functions and beyond. *ACM Trans. Algorithms*, 12(4):1–26, 2016.

Leqian Zheng, Hau Chan, Grigorios Loukides, and Minming Li. Maximizing approximately $k$-submodular functions. In *SDM*, pages 414–422, 2021.