

Maximizing k -Submodular Functions and Beyond*

Justin Ward[†]

Department of Computer Science, University of Warwick, UK

J.D.Ward@warwick.ac.uk

Stanislav Živný[‡]

Department of Computer Science, University of Oxford, UK

standa.zivny@cs.ox.ac.uk

Abstract

We consider the maximization problem in the value oracle model of functions defined on k -tuples of sets that are submodular in every orthant and r -wise monotone, where $k \geq 2$ and $1 \leq r \leq k$. We give an analysis of a deterministic greedy algorithm that shows that any such function can be approximated to a factor of $1/(1+r)$. For $r = k$, we give an analysis of a randomised greedy algorithm that shows that any such function can be approximated to a factor of $1/(1+\sqrt{k/2})$, thus improving on the factor of $1/(1+k)$ obtained by our deterministic greedy algorithm.

In the case of $k = r = 2$, the considered functions correspond precisely to bisubmodular functions, in which case we obtain an approximation guarantee of $1/2$. We show that, as in the case of submodular functions, this result is the best possible in both the value query model, and under the assumption that $NP \neq RP$.

Extending a result of Ando et al., we show that for any $k \geq 3$ submodularity in every orthant and pairwise monotonicity (i.e. $r = 2$) precisely characterize k -submodular functions. Consequently, we obtain an approximation guarantee of $1/3$ (and thus independent of k) for the maximization problem of k -submodular functions.

1 Introduction

Given a finite nonempty set U , a set function $f : 2^U \rightarrow \mathbb{R}_+$ defined on subsets of U is called *submodular* if for all $S, T \subseteq U$,

$$f(S) + f(T) \geq f(S \cap T) + f(S \cup T).$$

Submodular functions are a key concept in operations research and combinatorial optimization [29, 28, 38, 34, 10, 24, 19]. Examples of submodular functions include cut capacity functions, matroid rank functions, and entropy functions. Submodular functions are often considered to be a discrete analogue of convex functions [26].

*An extended abstract of this work appeared in the *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2014 [41].

[†]J.W. was supported by EPSRC grants EP/J021814/1 and EP/D063191/1.

[‡]S.Ž. was supported by a Royal Society University Research Fellowship.

Both minimizing and maximizing submodular functions have been considered extensively in the literature, in both constrained and unconstrained settings. Submodular function maximization is easily shown to be NP-hard [34] since it generalizes many standard NP-hard problems such as the maximum cut problem [12, 9]. In contrast, the problem of minimizing a submodular function can be solved efficiently with only polynomially many evaluations of the function [19] either by using the ellipsoid algorithm [13, 14], or by using one of several combinatorial algorithms that have been obtained in the last decade [33, 20, 17, 18, 30, 22].

Following a question by Lovász [26], a generalization of submodularity to biset functions has been introduced. Given a finite nonempty set U , a function $f : 3^U \rightarrow \mathbb{R}_+$ defined on pairs of disjoint subsets of U is called *bisubmodular* if for all pairs (S_1, S_2) and (T_1, T_2) of disjoint subsets of U ,

$$f(S_1, S_2) + f(T_1, T_2) \geq f((S_1, S_2) \sqcap (T_1, T_2)) + f((S_1, S_2) \sqcup (T_1, T_2)),$$

where we define

$$(S_1, S_2) \sqcap (T_1, T_2) = (S_1 \cap T_1, S_2 \cap T_2),$$

and

$$(S_1, S_2) \sqcup (T_1, T_2) = ((S_1 \cup T_1) \setminus (S_2 \cup T_2), (S_2 \cup T_2) \setminus (S_1 \cup T_1)).$$

Bisubmodular functions were originally studied in the context of rank functions of delta-matroids [4, 6]. Bisubmodularity also arises in bicooperative games [3] as well as variants of sensor placement problems and coupled feature selection problems [35]. The minimization problem for bisubmodular functions was solved by using ellipsoid method in [32]. More recently, combinatorial [11] and strongly polynomial [27] algorithms for minimizing bisubmodular functions have been developed.

In this paper, we study the natural generalization of submodular and bisubmodular functions: given a natural number $k \geq 1$ and a finite nonempty set U , a function $f : (k+1)^U \rightarrow \mathbb{R}_+$ defined on k -tuples of pairwise disjoint subsets of U is called *k-submodular* if for all k -tuples $S = (S_1, \dots, S_k)$ and $T = (T_1, \dots, T_k)$ of pairwise disjoint subsets of U ,

$$f(S) + f(T) \geq f(S \sqcap T) + f(S \sqcup T),$$

where we define

$$S \sqcap T = (S_1 \cap T_1, \dots, S_k \cap T_k),$$

and

$$S \sqcup T = ((S_1 \cup T_1) \setminus \bigcup_{i \in \{2, \dots, k\}} (S_i \cup T_i), \dots, (S_k \cup T_k) \setminus \bigcup_{i \in \{1, \dots, k-1\}} (S_i \cup T_i)).$$

Under this definition, 1-submodularity corresponds exactly to the standard notion of submodularity for set functions, and similarly 2-submodularity corresponds to bisubmodularity. (We note that Ando has used the term *k-submodular* to study a different class of functions [1].)

1.1 Related work

The terminology for k -submodular functions was first introduced in [15] but the concept has been studied previously in [7]. The concept of k -submodularity is a special case of strong tree submodularity [23] with the tree being a star on $k + 1$ vertices.

To the best of our knowledge, it is not known whether the ellipsoid method can be employed for minimizing k -submodular functions for $k \geq 3$ (some partial results can be found in [15]), let alone whether there is a (fully) combinatorial algorithm for minimizing k -submodular functions for $k \geq 3$. However, it has recently been shown that explicitly given k -submodular functions can be minimized in polynomial time [36]¹, and these results have proved useful in the design of fixed-parameter algorithms [40].

Some results on maximizing special cases of bisubmodular functions have appeared in Singh, Guillory, and Bilmes [35], who showed that simple bisubmodular function can be represented as a matroid constraint and a single submodular function, thus enabling the use of existing algorithms in some special cases. Unfortunately they show that this approach may require that the submodular function take negative values and so the approach does not work in general. (We note that our definition of bisubmodularity corresponds to directed bisubmodularity in [35].)

A different generalization of bisubmodularity, called skew bisubmodularity, has proved important in classifying finite-valued CSPs on domains with three elements [16]; this result was then generalized by a complexity classification of finite-valued CSPs on domains of arbitrary size [37]. Explicitly given skew bisubmodular functions can be minimized efficiently by results of Thapper and Živný [36]. The general question of whether all bisubmodular, and, more generally, k -submodular functions can be approximately maximized was left open.

1.2 Contributions

Following the question by Lovász [26] of whether there are generalizations of submodularity that preserve some nice properties such as efficient optimization algorithms, we consider the class of functions that are submodular in every orthant and r -wise monotone (the precise definition is given in Section 2), which includes as special cases bisubmodular and k -submodular functions.

Specifically, we consider the problem of *maximizing* bisubmodular and, more generally, k -submodular functions in the *value oracle model*. We provide the first approximation guarantees for maximizing a general bisubmodular or k -submodular function.

In Section 3, we prove that for any $k \geq 2$, k -submodular functions are precisely the k -set functions that are submodular in every orthant and pairwise monotone, thus extending the result from [2] that showed this result for $k = 2$.

In Section 4, we show that the naive random algorithm that simply returns a random partition of the ground set U is $1/4$ -approximation for maximizing any bisubmodular function and a $1/k$ -approximation for maximizing a k -submodular function with $k \geq 3$. We also show that our analysis is tight.

In Section 5, we show that a simple greedy algorithm for maximizing k -set functions that are submodular in every orthant and r -wise monotone for some $1 \leq r \leq k$ achieves a factor of $1/(1+r)$. We also show that our analysis is tight. Consequently, this algorithm achieves a factor of $1/3$ for maximizing k -submodular functions.

¹In fact, results in [36] imply that much larger classes of functions can be minimized in polynomial time, including as one special case functions that are (strong) tree submodular, which in turn includes k -submodular functions.

In Section 6, we develop a randomized greedy algorithm for maximizing k -set functions that are submodular in every orthant and k -wise monotone. The algorithm is inspired by the algorithm of Buchbinder et al. [5] for unconstrained submodular maximization. We show that this algorithm approximates any such k -set function to a factor of $1/(1 + \sqrt{k/2})$.

Finally, in Section 7, we relate our results on bisubmodular functions and existing results on submodular functions via a known embedding of submodular functions into bisubmodular functions. Using this embedding we can translate inapproximability results for submodular function into analogous results for bisubmodular functions. Moreover, we show that the algorithm of Buchbinder et al. [5] may be viewed as a special case of our algorithm applied to this embedding.

Recently, Iwata, Tanigawa, and Yoshida [21] have independently obtained a $1/k$ -approximation algorithm for maximizing k -submodular functions. Here we improve this factor to $1/3$, while also considering several other algorithms and generalizations of k -submodular functions.

2 Preliminaries

We denote by \mathbb{R}_+ the set of all non-negative real numbers. Let U be a ground set containing n elements and $k \geq 1$ be a fixed integer. We consider functions that assign a value in \mathbb{R}_+ to each partial assignment of the values $\{1, \dots, k\}$ to the elements of U . We can represent each such partial assignments as vectors \mathbf{x} in $\{0, \dots, k\}^U$, where we have $x_e = 0$ if element $e \in U$ is not assigned any value in $\{1, \dots, k\}$, and otherwise have x_e equal to the value assigned to e . It will be useful to consider the partial assignment obtained from another (possibly partial) assignment \mathbf{x} by “forgetting” the values assigned to all elements except for some specified set $S \subseteq U$. We represent this as the vector $\mathbf{x}|_S$ whose coordinates are given by $(\mathbf{x}|_S)_e = x_e$, for all $e \in S$ and $(\mathbf{x}|_S)_e = 0$ for all $e \in U \setminus S$. Note that $\mathbf{x}|_S$ is similar to the projection of \mathbf{x} onto S , but we here require that all coordinates $e \notin S$ be set to 0, while the standard notion of projection would remove these coordinates from the resulting vector. In particular, this means that $\mathbf{x}|_S$ and \mathbf{x} both have n coordinates.

In order to relate our results to existing work on submodular functions, we shall also use terminology from set functions. In this setting, we consider k -set functions, which assign a value to each tuple of k disjoint sets $S = (S_1, \dots, S_k)$, where $S_i \subseteq U$ and $S_i \cap S_j = \emptyset$ for all $i \neq j$. It is straightforward to check that the two notions are equivalent by having $e \in S_i$ if and only if $x_e = i$. Then, we have $x_e = 0$ if and only if e does not appear in any of the sets S_1, \dots, S_k .

The solution space over which we optimize our functions is thus the set of partitions of some subset $U' \subseteq U$ into k disjoint sets, where in our vector notation U' is equivalent to the set of coordinates in \mathbf{x} that are non-zero. We shall refer to a partition of the entire ground set U as an *orthant* of U , and use the word *partial solution* to refer to a partition of some subset of U , to emphasize that they may not necessarily assign every element in U to a set. Given a partial solution \mathbf{s} and an orthant \mathbf{t} , we say that \mathbf{s} is in orthant \mathbf{t} if $\mathbf{s} = \mathbf{t}|_A$ for some set $A \subseteq U$. That is, \mathbf{s} is in orthant \mathbf{t} if and only if \mathbf{s} agrees with \mathbf{t} on all non-zero values.

Consider the operations \min_0 and \max_0 given by

$$\min_0(s, t) \stackrel{\text{def}}{=} \begin{cases} 0, & s \neq 0, t \neq 0, s \neq t \\ \min(s, t), & \text{otherwise} \end{cases}$$

and

$$\max_0(s, t) \stackrel{\text{def}}{=} \begin{cases} 0, & s \neq 0, t \neq 0, s \neq t \\ \max(s, t), & \text{otherwise,} \end{cases}$$

where $\min(s, t)$ (respectively, $\max(s, t)$) returns the smaller (respectively, the larger) of s and t with respect to the usual order on the integers. Then, for vectors \mathbf{s} and \mathbf{t} in $\{0, \dots, k\}^U$ we let $\min_0(\mathbf{s}, \mathbf{t})$ (respectively, $\max_0(\mathbf{s}, \mathbf{t})$) denote the vector obtained from applying \min_0 (respectively, \max_0) to \mathbf{s} and \mathbf{t} coordinate-wise. Using these operations we can define the general class of k -submodular functions:

Definition 1. Given a natural number $k \geq 1$ and a finite nonempty set U , a function $f : \{0, \dots, k\}^U \rightarrow \mathbb{R}_+$ is called k -submodular if for all \mathbf{s} and \mathbf{t} in $\{0, \dots, k\}^U$,

$$f(\mathbf{s}) + f(\mathbf{t}) \geq f(\min_0(\mathbf{s}, \mathbf{t})) + f(\max_0(\mathbf{s}, \mathbf{t})). \quad (1)$$

Note that if \mathbf{s} and \mathbf{t} are both orthants, then we have $\min_0(\mathbf{s}, \mathbf{t}) = \max_0(\mathbf{s}, \mathbf{t}) = \text{id}_0(\mathbf{s}, \mathbf{t})$, where the operation id_0 on each coordinate of \mathbf{s} and \mathbf{t} is given by $\text{id}_0(s, t) = s = t$ if $s = t$, and $\text{id}_0(s, t) = 0$ otherwise. Thus, if f is a k -submodular function, we have

$$f(\mathbf{s}) + f(\mathbf{t}) \geq 2f(\text{id}_0(\mathbf{s}, \mathbf{t})) \quad (2)$$

for any two orthants \mathbf{s} and \mathbf{t} of U .

Example 2. The well-known Max-Cut problem demonstrates that maximizing (1-)submodular functions is NP-hard, even if the objective function is given explicitly [12]. We show that the same hardness result holds for any $k \geq 1$. Consider the function $f^{(u,v)} : \{0, \dots, k\}^{\{u,v\}} \rightarrow \mathbb{R}_+$ given by² $f^{(u,v)}(x_u, x_v) = \llbracket x_u \neq x_v \rrbracket$. It is easy to check that $f^{(u,v)}$ is k -submodular. Given a graph (V, E) with $V = \{1, \dots, n\}$, we consider the function $f(\mathbf{x}) = \sum_{\{i,j\} \in E} f^{(i,j)}(x_i, x_j)$. Because f is a positive combination of k -submodular functions, it is also k -submodular. Moreover, maximizing f amounts to solving the Max- k -Cut problem, which is NP-hard [31].

While concise, Definition 1 gives little intuition in the traditional setting of set functions. We now consider this setting in order to provide some intuition. Consider two partial solutions $S = (S_1, \dots, S_k)$ and $T = (T_1, \dots, T_k)$ and let \mathbf{s} and \mathbf{t} be the vectors in $\{0, \dots, k\}^U$ representing S and T , respectively. Consider some element $e \in U$. We have $\min_0(s_e, t_e) = i \neq 0$ precisely when $s_e = t_e = i \neq 0$. Thus, the vector $\min_0(\mathbf{s}, \mathbf{t})$ in Definition 1 corresponds exactly to the coordinate-wise intersection $(S_1 \cap T_1, \dots, S_k \cap T_k)$ of S and T . Similarly, $\max_0(s_e, t_e) = i \neq 0$ precisely when either $s_e = t_e \neq 0$ or when one of s_e, t_e is $i \neq 0$ and the other is 0. Thus, the vector $\max_0(\mathbf{s}, \mathbf{t})$ corresponds exactly to the coordinate-wise union of S and T after we have removed any element e occurring in two different sets in S and T . That is, if we set $X_{-i} = \bigcup_{j \neq i} (S_j \cup T_j)$, then $\max_0(\mathbf{s}, \mathbf{t})$ corresponds to $((S_1 \cup T_1) \setminus X_{-1}, \dots, (S_k \cup T_k) \setminus X_{-k})$. The removal of X_{-i} from the i th union effectively enforces the condition that no element occurs in two different sets in the resulting partial solution.

²Here and throughout, we employ the Iverson bracket notation $\llbracket p \rrbracket$ to denote a value that is 1 when statement p is true and 0 when p is false.

The following equivalences, first observed by Cohen et al. [7], allow us to relate k -submodular functions to existing families of set functions. When $k = 2$, Definition 1 requires that

$$f(S_1, S_2) + f(T_1, T_2) \geq f(S_1 \cap T_1, S_2 \cap T_2) + f((S_1 \cup T_1) \setminus (S_2 \cup T_2), (S_2 \cup T_2) \setminus (S_1 \cup T_1)),$$

which agrees exactly with the definition of bisubmodular functions given in [10]. When $k = 1$, there is only a single set in each partial solution, and hence a single non-zero value in each corresponding vector, and so $X_{-1} = \emptyset$. Thus, Definition 1 requires that

$$f(S_1) + f(T_1) \geq f(S_1 \cap T_1) + f(S_1 \cup T_1),$$

which agrees exactly with the standard definition of submodular functions [29].

It is well-known that for standard set functions submodularity is equivalent to the property of *diminishing marginal returns*. Let $f : 2^U \rightarrow \mathbb{R}_+$ be a set function on U and define the marginal value of e with respect to S as $f_e(S) \stackrel{\text{def}}{=} f(S \cup \{e\}) - f(S)$ for all $S \subseteq U$ and $e \notin S$. Then, f is submodular if and only if

$$f_e(A) \geq f_e(B)$$

for all $A \subseteq B$ and $e \notin B$.

We shall see that marginal returns also play an important role in characterizing k -submodular functions. In this setting, however, we must specify not only which element we are adding to the solution, but which set in the partition it is being added to. For a k -set function $f : \{0, \dots, k\}^U \rightarrow \mathbb{R}_+$, an element $e \in U$, and a value $i \in \{1, \dots, k\}$, we define the marginal value $f_{i,e}(S)$ by

$$f_{i,e}(S) \stackrel{\text{def}}{=} f(S_1, \dots, S_{i-1}, S_i \cup \{e\}, S_{i+1}, \dots, S_k) - f(S_1, \dots, S_k)$$

for any partial solution $S = (S_1, \dots, S_k)$ such that $e \notin S_i$ for any i . Equivalently, in vector notation, we have

$$f_{i,e}(\mathbf{s}) \stackrel{\text{def}}{=} f(\mathbf{s} + i \cdot \mathbf{1}_e) - f(\mathbf{s}),$$

where \mathbf{s} is any partial solution satisfying $s_e = 0$, and $\mathbf{1}_e$ denotes the unit vector that is 1 in coordinate e and 0 in all other coordinates.

Definition 3. Let $k \geq 1$, and $1 \leq r \leq k$. We say that a function $f : \{0, \dots, k\}^U \rightarrow \mathbb{R}_+$ is:

- *submodular in every orthant*, if for any two partial solutions \mathbf{a} and \mathbf{b} in the same orthant of U , $f(\mathbf{a}) + f(\mathbf{b}) \geq f(\min_0(\mathbf{a}, \mathbf{b})) + f(\max_0(\mathbf{a}, \mathbf{b}))$.
- *r -wise monotone*, if for any element e , any partial solution \mathbf{s} with $s_e = 0$, and any set of r distinct values $I \in \binom{\{1, \dots, k\}}{r}$:

$$\sum_{i \in I} f_{i,e}(\mathbf{s}) \geq 0.$$

We remark that the case of $k = r = 1$ corresponds to monotone submodular functions. In the case of $k = r = 2$, Ando, Fujishige, and Naito [2] have shown that these two properties give an exact characterization of the class of bisubmodular functions. In Section 3, we extend their result by showing that submodularity in every orthant and pairwise monotonicity in fact precisely characterize k -submodular functions for all $k \geq 2$.

Let us now give some justification for the terminology “submodular in every orthant.” Let \mathbf{x} be an orthant of U . Given a k -submodular function f , we call set function $h : 2^U \rightarrow \mathbb{R}_+$ defined for any $S \subseteq U$ by

$$h(S) \stackrel{\text{def}}{=} f(\mathbf{x}|_S)$$

the function *induced by* \mathbf{x} and f . In the language of set functions, the function h is obtained by first assigning each element e in U to a single set X_i (where $i = x_e$). Then, $h(S)$ is simply the value of $f(S \cap X_1, \dots, S \cap X_k)$. We now show f is k -submodular in an orthant (in the sense of Definition 3) if and only if the function h induced by this orthant and f is submodular.

Lemma 4. *Let (X_1, \dots, X_k) be an orthant of U , with vector representation \mathbf{x} . Then, f is k -submodular in the orthant \mathbf{x} if and only if the function h induced by \mathbf{x} and f is submodular.*

Proof. Let A and B be two subsets of U , with associated partial solutions $\mathbf{a} = \mathbf{x}|_A$ and $\mathbf{b} = \mathbf{x}|_B$ in orthant \mathbf{x} . Then, note that $e \in A \cap B$ if and only if $\min(a_e, b_e)$ is non-zero, and $e \in A \cup B$ if and only if $\max(a_e, b_e)$ is non-zero. Moreover, since \mathbf{a} and \mathbf{b} agree on all non-zero coordinates, we have $\min_0(\mathbf{a}, \mathbf{b}) = \min(\mathbf{a}, \mathbf{b})$ and $\max_0(\mathbf{a}, \mathbf{b}) = \max(\mathbf{a}, \mathbf{b})$. Hence,

$$\begin{aligned} h(A \cup B) &= f(\mathbf{x}|_{A \cup B}) = f(\max(\mathbf{x}|_A, \mathbf{x}|_B)) = f(\max_0(\mathbf{x}|_A, \mathbf{x}|_B)) = f(\max_0(\mathbf{a}, \mathbf{b})), \\ h(A \cap B) &= f(\mathbf{x}|_{A \cap B}) = f(\min(\mathbf{x}|_A, \mathbf{x}|_B)) = f(\min_0(\mathbf{x}|_A, \mathbf{x}|_B)) = f(\min_0(\mathbf{a}, \mathbf{b})). \end{aligned}$$

Thus, we have

$$h(A) + h(B) \geq h(A \cap B) + h(A \cup B)$$

for any $A, B \subseteq U$ if and only if

$$f(\mathbf{a}) + f(\mathbf{b}) \geq f(\min_0(\mathbf{a}, \mathbf{b})) + f(\max_0(\mathbf{a}, \mathbf{b}))$$

for the associated partial solutions \mathbf{a}, \mathbf{b} in orthant \mathbf{x} . □

Many of our proofs will use this connection between the standard notion of submodularity and the k -set functions in Definition 1. Specifically, we shall make use of the following result from Lee, Sviridenko, and Vondrák [25], which we restate here.

Lemma 5 ([25, Lemma 1.1]). *Let f be a non-negative submodular function on U . Let $S, C \subseteq U$ and let $\{T_\ell\}_{\ell=1}^t$ be a collection of subsets of $C \setminus S$ such that each element of $C \setminus S$ appears in exactly p of these subsets. Then*

$$\sum_{\ell=1}^t [f(S \cup T_\ell) - f(S)] \geq p[f(S \cup C) - f(S)].$$

In fact, the following weaker statement will be sufficient for our purposes:

Corollary 6 (of Lemma 5). *Let f be a non-negative submodular function on U . Let $S, C \subseteq U$ and let $\{T_\ell\}_{\ell=1}^t$ be a collection of subsets of $C \setminus S$ such that each element of $C \setminus S$ appears in exactly p of these subsets. Then*

$$\sum_{\ell=1}^t f(S \cup T_\ell) \geq pf(S \cup C).$$

Proof. Add $\sum_{\ell=1}^t f(S)$ to each side of the inequality in Lemma 5. This gives

$$\begin{aligned} \sum_{\ell=1}^t f(S \cup T_\ell) &\geq p \cdot f(S \cup C) - p \cdot f(S) + \sum_{\ell=1}^t f(S) \\ &= p \cdot f(S \cup C) + (t - p) \cdot f(S) \\ &\geq p \cdot f(S \cup C), \end{aligned}$$

since $p \leq t$. □

3 Characterization of k -Submodularity

Theorem 7. *Let $f : \{0, \dots, k\}^U \rightarrow \mathbb{R}_+$ be a k -set function, where $k \geq 2$. Then, f is k -submodular if and only if f is submodular in every orthant and pairwise monotone.*

In order to prove Theorem 7, we shall make use of the following lemma, which allows us to generalize pairwise monotonicity to solutions that disagree on the placement of *multiple* elements e .

Lemma 8. *Let $k \geq 2$ and suppose that $f : \{0, \dots, k\}^U \rightarrow \mathbb{R}_+$ is submodular in every orthant and pairwise monotone. Let \mathbf{a} and \mathbf{b} in $\{0, \dots, k\}^U$ satisfy $0 \neq a_e \neq b_e \neq 0$ for all $e \in I$ and $a_e = b_e$ for all $e \in U \setminus I$, and define $\mathbf{c} = \mathbf{a}|_{U \setminus I} = \mathbf{b}|_{U \setminus I}$. Then, $f(\mathbf{a}) + f(\mathbf{b}) \geq 2f(\mathbf{c})$.*

Proof. The proof is by induction on the size of I . In the case that $|I| = 0$, the claim is trivial. Suppose, then, that $|I| = p > 0$ and so I contains at least one element e . We can represent \mathbf{a} and \mathbf{b} as $\mathbf{a} = \mathbf{c} + \mathbf{x}$, and $\mathbf{b} = \mathbf{c} + \mathbf{y}$ where \mathbf{x} and \mathbf{y} are vectors in $\{0, \dots, 1\}^U$ satisfying $0 \neq x_e \neq y_e \neq 0$ for all $e \in I$, and $x_e = y_e = 0$ for all $e \in U \setminus I$.

Let $e \in I$ be some element on which \mathbf{a} and \mathbf{b} disagree. We define $\bar{\mathbf{x}} = \mathbf{x}|_{I \setminus \{e\}}$, $\bar{\mathbf{y}} = \mathbf{y}|_{\{e\}}$, and $\mathbf{z} = \bar{\mathbf{x}} + \bar{\mathbf{y}}$. Then, we have

$$f(\mathbf{a}) + f(\mathbf{b}) = f(\mathbf{c} + \mathbf{x}) + f(\mathbf{c} + \mathbf{y}) = [f(\mathbf{c} + \mathbf{x}) + f(\mathbf{c} + \mathbf{z})] + [f(\mathbf{c} + \mathbf{y}) + f(\mathbf{c} + \mathbf{z})] - 2f(\mathbf{c} + \mathbf{z}). \quad (3)$$

The solutions $\mathbf{c} + \mathbf{x}$ and $\mathbf{c} + \mathbf{z}$ disagree on precisely the single element e in I and are non-zero for this element. Thus, pairwise monotonicity of f implies that

$$f(\mathbf{c} + \mathbf{x}) + f(\mathbf{c} + \mathbf{z}) \geq 2f(\mathbf{c} + \bar{\mathbf{x}}). \quad (4)$$

Similarly, $\mathbf{c} + \mathbf{y}$ and $\mathbf{c} + \mathbf{z}$ disagree on precisely those $p - 1$ elements in $I \setminus \{e\}$ and are non-zero for these elements. Thus, by the induction hypothesis

$$f(\mathbf{c} + \mathbf{y}) + f(\mathbf{c} + \mathbf{z}) \geq 2f(\mathbf{c} + \bar{\mathbf{y}}). \quad (5)$$

Combining (3), (4), and (5) we obtain

$$f(\mathbf{a}) + f(\mathbf{b}) \geq 2f(\mathbf{c} + \bar{\mathbf{x}}) + 2f(\mathbf{c} + \bar{\mathbf{y}}) - 2f(\mathbf{c} + \mathbf{z}). \quad (6)$$

Now, we note that $\mathbf{c} + \bar{\mathbf{x}}$ and $\mathbf{c} + \bar{\mathbf{y}}$ are both in the orthant $\mathbf{c} + \mathbf{z}$. Thus, from submodularity in every orthant,

$$f(\mathbf{c} + \bar{\mathbf{x}}) + f(\mathbf{c} + \bar{\mathbf{y}}) \geq f(\min_0(\mathbf{c} + \bar{\mathbf{x}}, \mathbf{c} + \bar{\mathbf{y}})) + f(\max_0(\mathbf{c} + \bar{\mathbf{x}}, \mathbf{c} + \bar{\mathbf{y}})) = f(\mathbf{c}) + f(\mathbf{c} + \mathbf{z}). \quad (7)$$

Combining (6) and (7) we obtain

$$f(\mathbf{a}) + f(\mathbf{b}) \geq 2f(\mathbf{c}) + 2f(\mathbf{c} + \mathbf{z}) - 2f(\mathbf{c} + \mathbf{z}) = 2f(\mathbf{c}). \quad \square$$

We now return to the proof of Theorem 7.

Proof of Theorem 7. We begin by showing the necessity of the two properties. Suppose that f is k -submodular. Then, submodularity in every orthant follows directly from (1). For pairwise monotonicity, let \mathbf{s} satisfy $s_e = 0$. Consider any pair of distinct values i, j from $\{1, \dots, k\}$, and let $\mathbf{s}^i = \mathbf{s} + i \cdot \mathbf{1}_e$ and $\mathbf{s}^j = \mathbf{s} + j \cdot \mathbf{1}_e$. Then,

$$\begin{aligned} f_{i,e}(\mathbf{s}) + f_{j,e}(\mathbf{s}) &= f(\mathbf{s}^i) - f(\mathbf{s}) + f(\mathbf{s}^j) - f(\mathbf{s}) \\ &\geq f(\min_0(\mathbf{s}^i, \mathbf{s}^j)) + f(\max_0(\mathbf{s}^i, \mathbf{s}^j)) - 2f(\mathbf{s}) \\ &= f(\mathbf{s}) + f(\mathbf{s}) - 2f(\mathbf{s}). \end{aligned}$$

We now show that submodularity in every orthant and pairwise monotonicity imply k -submodularity. Let f be a function that is submodular in every orthant and pairwise monotone, and consider two arbitrary vectors \mathbf{x} and \mathbf{y} in $\{0, \dots, k\}^U$. Let I be the set of all elements $e \in U$ for which $x_e \neq 0$, $y_e \neq 0$ and $x_e \neq y_e$. We can write

$$f(\mathbf{x}) + f(\mathbf{y}) = f(\mathbf{x}) + f(\mathbf{y}|_{U \setminus I}) + f(\mathbf{y}) + f(\max_0(\mathbf{x}, \mathbf{y})) - f(\mathbf{y}|_{U \setminus I}) - f(\max_0(\mathbf{x}, \mathbf{y})). \quad (8)$$

We note that \mathbf{x} and $\mathbf{y}|_{U \setminus I}$ are in the same orthant, since they agree on all non-zero coordinates. Thus,

$$\begin{aligned} f(\mathbf{x}) + f(\mathbf{y}|_{U \setminus I}) &\geq f(\min_0(\mathbf{x}, \mathbf{y}|_{U \setminus I})) + f(\max_0(\mathbf{x}, \mathbf{y}|_{U \setminus I})) \\ &= f(\min_0(\mathbf{x}, \mathbf{y})) + f(\max_0(\mathbf{x}, \mathbf{y}) + \mathbf{x}|_I), \end{aligned} \quad (9)$$

where in the final equation we have used the fact that for all $e \in I$, $x_e \neq 0$, $y_e \neq 0$ and $x_e \neq y_e$ and so $\min_0(x_i, y_i) = \max_0(x_i, y_i) = 0$. Similarly, we have \mathbf{y} and $\max_0(\mathbf{x}, \mathbf{y})$ in the same orthant, and so

$$\begin{aligned} f(\mathbf{y}) + f(\max_0(\mathbf{x}, \mathbf{y})) &\geq f(\min_0(\mathbf{y}, \max_0(\mathbf{x}, \mathbf{y}))) + f(\max_0(\mathbf{y}, \max_0(\mathbf{x}, \mathbf{y}))) \\ &= f(\mathbf{y}|_{U \setminus I}) + f(\max_0(\mathbf{x}, \mathbf{y}) + \mathbf{y}|_I). \end{aligned} \quad (10)$$

Combining (8), (9), and (10), we obtain

$$f(\mathbf{x}) + f(\mathbf{y}) \geq f(\min_0(\mathbf{x}, \mathbf{y})) + f(\max_0(\mathbf{x}, \mathbf{y}) + \mathbf{x}|_I) + f(\max_0(\mathbf{x}, \mathbf{y}) + \mathbf{y}|_I) - f(\max_0(\mathbf{x}, \mathbf{y})). \quad (11)$$

Finally, from Lemma 8 we have:

$$f(\max_0(\mathbf{x}, \mathbf{y}) + \mathbf{x}|_I) + f(\max_0(\mathbf{x}, \mathbf{y}) + \mathbf{y}|_I) \geq 2f(\max_0(\mathbf{x}, \mathbf{y})). \quad (12)$$

Combining (11) and (12) then gives

$$f(\mathbf{x}) + f(\mathbf{y}) \geq f(\min_0(\mathbf{x}, \mathbf{y})) + f(\max_0(\mathbf{x}, \mathbf{y})). \quad \square$$

We now provide an example of a natural class of k -set functions which are submodular in every orthant and k -wise monotone but not k -submodular.

Example 9. Let $f^{(u,v)} : \{0, \dots, k\}^{\{u,v\}} \rightarrow \mathbb{R}_+$ be given by:

$$f^{(u,v)}(x_u, x_v) = \begin{cases} 0, & x_u = x_v = 0 \\ \frac{1}{k} \sum_{i=1}^k \mathbb{I}[x_u < i] = \frac{k-x_u}{k}, & x_u \neq 0, x_v = 0 \\ \frac{1}{k} \sum_{i=1}^k \mathbb{I}[i < x_v] = \frac{x_v-1}{k}, & x_u = 0, x_v \neq 0 \\ \mathbb{I}[x_u < x_v], & \text{otherwise.} \end{cases}$$

The function $f^{(u,v)}$ has the following intuitive interpretation: we begin with the valued constraint $\mathbb{I}[x_u < x_v]$, where x_u and x_v range over $\{1, \dots, k\}$. This gives a function that is defined on all orthants. We extend the function to partial assignments by setting $f^{(u,v)}(0, 0) = 0$, and otherwise assigning $f^{(u,v)}(x_u, 0)$ and $f^{(u,v)}(0, x_v)$ the probability that $x_u > i$ and $i > x_v$, respectively, when i is chosen uniformly at random from $\{1, \dots, k\}$.

The function $f^{(u,v)}$ arises in the following graph layout problem: we are given a directed graph $G = (V, E)$ and a number k , and we wish to partition V into k layers so that as many directed edges as possible travel from a lower- to a higher-numbered layer. This problem is equivalent to maximizing the function $f(\mathbf{x}) : \{0, \dots, k\}^V \rightarrow \mathbb{R}_+$ given by $f(\mathbf{x}) = \sum_{(u,v) \in E} f^{(u,v)}(x_u, x_v)$. Although this function allows some vertices to remain unassigned, k -wise monotonicity implies that there is always a maximizer of f that is an orthant.

We now show that $f^{(u,v)}$ is submodular in every orthant and k -wise monotone. Fix an orthant $(x_u = i, x_v = j)$, where $i, j \in \{1, \dots, k\}$, and let h be the submodular function induced by $f^{(u,v)}$ and this orthant. If $i \geq j$, we have

$$\begin{aligned} h_u(\emptyset) &= h(\{u\}) - h(\emptyset) = \frac{k-i}{k} & h_v(\emptyset) &= h(\{v\}) - h(\emptyset) = \frac{j-1}{k} \\ h_u(\{v\}) &= h(\{u, v\}) - h(\{v\}) = -\frac{j-1}{k} & h_v(\{u\}) &= h(\{u, v\}) - h(\emptyset) = -\frac{k-i}{k}, \end{aligned}$$

while if $i < j$ (and hence $i \leq j-1$), we have:

$$\begin{aligned} h_u(\emptyset) &= h(\{u\}) - h(\emptyset) = \frac{k-i}{k} = 1 - \frac{i}{k} & h_v(\emptyset) &= h(\{v\}) - h(\emptyset) = \frac{j-1}{k} \geq \frac{i}{k} \\ h_u(\{v\}) &= h(\{u, v\}) - h(\{v\}) = 1 - \frac{j-1}{k} \leq 1 - \frac{i}{k} & h_v(\{u\}) &= h(\{u, v\}) - h(\emptyset) = 1 - \frac{k-i}{k} = \frac{i}{k}. \end{aligned}$$

In all cases, we observe that the marginals of h are decreasing, and so h is a submodular function.

In order to show that $f^{(u,v)}$ is k -wise monotone, we note that $f_{i,e}^{(u,v)}(0, 0)$ is non-negative for all values of i and e , and so $\sum_{i=1}^k f_{i,e}^{(u,v)}(0, 0) \geq 0$ for all $e \in \{u, v\}$. For the remaining marginals, suppose that $j \neq 0$. Then, for we have

$$\begin{aligned} \sum_{i=1}^k f_{i,u}^{(u,v)}(0, j) &= \sum_{i=1}^k \left[\mathbb{I}[i < j] - \frac{1}{k} \sum_{p=1}^j \mathbb{I}[p < j] \right] = \sum_{i=1}^k \mathbb{I}[i < j] - \sum_{p=1}^k \mathbb{I}[p < j] = 0, \\ \sum_{i=1}^k f_{i,v}^{(u,v)}(j, 0) &= \sum_{i=1}^k \left[\mathbb{I}[j < i] - \frac{1}{k} \sum_{p=1}^j \mathbb{I}[j < p] \right] = \sum_{i=1}^k \mathbb{I}[j < i] - \sum_{p=1}^k \mathbb{I}[j < p] = 0. \end{aligned}$$

Finally, in order to see that $f^{(u,v)}$ is not pairwise monotone (and hence not k -submodular), we suppose that $k \geq 3$, and note that $f^{(u,v)}(2, 0) = \frac{k-2}{k}$, but both $f^{(u,v)}(2, 1) = 0$ and $f^{(u,v)}(2, 2) = 0$. Thus, $f_{1,v}^{(u,v)}(2, 0) + f_{2,v}^{(u,v)}(2, 0) = \frac{2(2-k)}{k} < 0$.

4 The Naive Random Algorithm

We now consider the performance of the naive random algorithm for maximizing a k -submodular function $f : \{0, \dots, k\}^U \rightarrow \mathbb{R}_+$. Note that pairwise monotonicity of f , guaranteed by Theorem 7, implies that any partial solution $S \in \{0, \dots, k\}^U$ can be extended greedily to an orthant of U without any loss in the value of f , since for every element $e \notin S$, we must have $f_{i,e}(S) \geq 0$ for some $i \in \{1, \dots, k\}$. Thus, we may assume without loss of generality that f takes its maximum value on some orthant \mathbf{o} . We now consider the expected performance of a random algorithm that simply selects an orthant of U uniformly at random.

Theorem 10. *Let $f : \{0, \dots, k\}^U \rightarrow \mathbb{R}_+$ be a k -submodular function attaining its maximum value on orthant \mathbf{o} , and let \mathbf{x} be an orthant of U selected uniformly at random. Then, $\mathbb{E}[f(\mathbf{x})] \geq \frac{1}{4}f(\mathbf{o})$ if $k = 2$, and $\mathbb{E}[f(\mathbf{x})] \geq \frac{1}{k}f(\mathbf{o})$ if $k \geq 3$.*

We present the analysis for the case in which $k \geq 3$ first, as it is simpler and will aid in motivating some of the constructions used for the case $k = 2$.

4.1 Analysis for $k \geq 3$

Let $h : 2^U \rightarrow \mathbb{R}_+$ be the submodular function induced by \mathbf{o} and f . For each $e \in U$ we consider a fixed permutation π_e on the set $\{1, \dots, k\}$ with the property that $\pi_e(o_e) = o_e$ and $\pi_e(z) \neq z$ for all $z \in \{1, \dots, k\} \setminus \{o_e\}$.³ Then, we denote by $\pi(\mathbf{x})$ the vector $(\pi_e(x_e))_{e \in U}$.

Let $P(A)$ be the set of orthants of U that agree with \mathbf{o} on exactly those coordinates $e \in A$. The following lemma allows us to relate the sum of the values of all partitions in $P(A)$ to the value of \mathbf{o} .

Lemma 11. *For each set $A \subseteq U$,*

$$\sum_{\mathbf{x} \in P(A)} f(\mathbf{x}) \geq |P(A)| \cdot h(A) = (k-1)^{n-|A|} h(A).$$

Proof. Consider the sum $\sum_{\mathbf{x} \in P(A)} f(\pi(\mathbf{x}))$. Because $\pi_e(x_e) = o_e$ if and only if $x_e = o_e$ already, we have $\pi(\mathbf{x}) \in P(A)$ if and only if $\mathbf{x} \in P(A)$. Then, because each π_e is a bijection, we have

$$\sum_{\mathbf{x} \in P(A)} f(\mathbf{x}) = \sum_{\mathbf{x} \in P(A)} f(\pi(\mathbf{x})),$$

and so,

$$\sum_{\mathbf{x} \in P(A)} f(\mathbf{x}) = \frac{1}{2} \left[\sum_{\mathbf{x} \in P(A)} f(\mathbf{x}) + \sum_{\mathbf{x} \in P(A)} f(\pi(\mathbf{x})) \right] = \frac{1}{2} \sum_{\mathbf{x} \in P(A)} [f(\mathbf{x}) + f(\pi(\mathbf{x}))]. \quad (13)$$

Now, we note that \mathbf{x} and $\pi(\mathbf{x})$ are both orthants. Thus, from (2) we have

$$f(\mathbf{x}) + f(\pi(\mathbf{x})) \geq 2\text{id}_0(\mathbf{x}, \pi(\mathbf{x})).$$

³Such a permutation can be obtained by taking, for example, $\pi_e(o_e) = o_e$, $\pi_e(o_e - 1) = o_e + 1$, and $\pi(z) = z + 1 \pmod k$ for all other $z \in \{1, \dots, k\}$.

Consider an arbitrary coordinate $e \in U$. If $e \in A$ we have $x_e = o_e$ and so $\pi_e(x_e) = x_e$ and hence $\text{id}_0(x_e, \pi_e(x_e)) = x_e$. If $e \notin A$, then we have $x_e \neq o_e$ and so $\pi_e(x_e) \neq x_e$ and hence $\text{id}_0(x_e, \pi_e(x_e)) = 0$. Thus,

$$2\text{id}_0(\mathbf{x}, \pi(\mathbf{x})) = 2f(\mathbf{o}|_A) = 2h(A).$$

Combining this with (13) we have,

$$\sum_{\mathbf{x} \in P(A)} f(\mathbf{x}) = \frac{1}{2} \sum_{\mathbf{x} \in P(A)} [f(\mathbf{x}) + f(\pi(\mathbf{x}))] \geq \sum_{\mathbf{x} \in P(A)} h(A) = (k-1)^{n-|A|} h(A),$$

since there are precisely $k-1$ choices $i \neq o_e$ for x_e for each of the $n-|A|$ coordinates $e \notin A$. \square

We now complete the proof of Theorem 10 in the case $k \geq 3$. We formulate the expectation as

$$\mathbb{E}[f(\mathbf{x})] = \frac{1}{k^n} \sum_{i=0}^n \sum_{A \in \binom{U}{i}} \sum_{\mathbf{x} \in P(A)} f(\mathbf{x}).$$

Using Lemma 11 we obtain

$$\sum_{i=0}^n \sum_{A \in \binom{U}{i}} \sum_{\mathbf{x} \in P(A)} f(\mathbf{x}) \geq \sum_{i=0}^n \sum_{A \in \binom{U}{i}} (k-1)^{n-i} h(A). \quad (14)$$

Consider a fixed value $i \in \{0, \dots, n\}$. Each element $e \in U$ appears in exactly $\binom{n-1}{i-1}$ of the $\binom{n}{i}$ sets $A \in \binom{U}{i}$. Because h is submodular, Corollary 6 then implies that

$$\sum_{A \in \binom{U}{i}} h(A) \geq \binom{n-1}{i-1} h(U) = \binom{n-1}{i-1} f(\mathbf{o}). \quad (15)$$

Combining (14) and (15) with our formulation of $\mathbb{E}[f(\mathbf{x})]$ we obtain:

$$\begin{aligned} \mathbb{E}[f(\mathbf{x})] &\geq \frac{1}{k^n} \sum_{i=0}^n \binom{n-1}{i-1} (k-1)^{n-i} f(\mathbf{o}) \\ &= \frac{(k-1)^{n-1}}{k^n} \sum_{i=0}^n \binom{n-1}{i-1} (k-1)^{-(i-1)} f(\mathbf{o}) \\ &= \frac{(k-1)^{n-1}}{k^n} \sum_{i=0}^{n-1} \binom{n-1}{i} (k-1)^{-i} f(\mathbf{o}) \\ &= \frac{(k-1)^{n-1}}{k^n} \cdot \left(1 + \frac{1}{k-1}\right)^{n-1} \cdot f(\mathbf{o}) \\ &= \frac{(k-1)^{n-1}}{k^n} \cdot \frac{k^{n-1}}{(k-1)^{n-1}} \cdot f(\mathbf{o}) \\ &= \frac{1}{k} \cdot f(\mathbf{o}). \end{aligned}$$

4.2 Analysis for $k = 2$

Now we consider the case in which f is a bisubmodular function, i.e. the case of $k = 2$. In the previous analysis of k -submodular functions for $k \geq 3$ we used a bijection π_e on $\{1, \dots, k\}$ with the property that $\pi_e(o_e) = o_e$ and $\pi_e(z) \neq z$ for all $z \neq o_e$. However, when $k = 2$, no such bijection exists and we must adopt a different approach.

Suppose again that f attains its maximum on orthant $\mathbf{o} \in \{1, 2\}^U$. For a value $v \in \{1, 2\}$ we let $\bar{v} \stackrel{\text{def}}{=} (v \bmod 2) + 1$ (i.e. the other value in $\{1, 2\}$). Then, for any disjoint subsets A and B of U we define the (partial) solution $T(A, B)$ by

$$T(A, B)_i = \begin{cases} o_i, & i \in A \\ \bar{o}_i, & i \in B \\ 0, & \text{otherwise} \end{cases}.$$

It will simplify our analysis to work with symmetrized values, which depend only on the sizes of the sets A and B chosen. We define

$$F_{i,j} = \binom{n}{i}^{-1} \binom{n-i}{j}^{-1} \sum_{A \in \binom{U}{i}} \sum_{B \in \binom{U \setminus A}{j}} [f(T(A, B))].$$

Then, $F_{i,j}$ gives the average value of f over all partial solutions on $i + j$ elements that agree with \mathbf{o} on exactly i non-zero elements, disagree with it on exactly j non-zero elements, and are zero for the remaining $n - i - j$ elements. In particular, we have $F_{n,0} = f(\mathbf{o})$, and $F_{i,n-i} = \binom{n}{i}^{-1} \sum_{A \in \binom{U}{i}} f(T(A, U \setminus A))$. Our next lemma relates these two values.

Lemma 12. *For all i such that $0 \leq i \leq n$,*

$$F_{i,n-i} \geq \frac{i(i-1)}{n(n-1)} F_{n,0}. \quad (16)$$

Proof. We prove two separate inequalities which together imply the lemma. First, we shall show that for all $1 \leq i \leq n-1$,

$$F_{i,n-i} \geq F_{i-1,n-i-1}. \quad (17)$$

We do this by showing that a related inequality holds for arbitrary sets of the appropriate size, and then average over all possible sets to obtain (17). Fix $1 \leq i \leq n-1$ and let A be any subset of U of size $i+1$. Set $B = U \setminus A$ and let x and y any two distinct elements in A . Consider the solutions $T(A-x, B+x)$ and $T(A-y, B+y)$ ⁴. They are both orthants and agree on all elements except x and y . Thus, from (2), the inequality

$$\begin{aligned} f(T(A-x, B+x)) + f(T(A-y, B+y)) &\geq 2\text{id}_0(T(A-x, B+x), T(A-y, B+y)) \\ &= 2f(T(A-x-y, B)) \end{aligned}$$

holds for any such choice of A , x , and y , where $|A| = i+1$ and $|B| = |U \setminus A| = n-i-1$. Averaging the resulting inequalities over all possible choices for A , $B = U \setminus A$, x , and y and dividing both sides by 2 then gives (17).

⁴Here, we employ the shorthand $A+x$ for $A \cup \{x\}$ and $A-x$ for $A \setminus \{x\}$.

Next, we show that for any $1 \leq i \leq n-1$,

$$F_{i-1,n-i-1} \geq \frac{i-1}{i+1} F_{i+1,n-i-1}. \quad (18)$$

Again fix $i \geq 1$, let A be any subset of U of size $i+1$ and set $B = U \setminus A$. Let h be the submodular function induced by the orthant $T(A, B)$ and f . Note then, that we can express h as $h(X) = T(A \cap X, B \cap X)$. We consider the sum:

$$\sum_{C \in \binom{A}{2}} [f(T(A \setminus C, B)) - T(\emptyset, B)] = \sum_{C \in \binom{A}{2}} [h(U \setminus C) - h(B)]$$

Each element of A appears in exactly $\binom{|A|-1}{2} = \binom{i}{2}$ of the sets $U \setminus C$ above (one for each way to choose a two element set C from the remaining $|A| - 1$ elements). Applying Corollary 6 we then obtain

$$\sum_{C \in \binom{A}{2}} h(U \setminus C) \geq \binom{i}{2} h(U) = \binom{i}{2} T(A, B).$$

Altogether, we obtain the inequality

$$\sum_{C \in \binom{A}{2}} f(T \setminus C, B) \geq \binom{i}{2} T(A, B),$$

valid for any choice of A , with $|A| = i+1$, and $|B| = |U \setminus A| = n-i-1$. Averaging the resulting inequalities over all possible choices for A , we obtain

$$\binom{i+1}{2} F_{i-1,n-i-1} \geq \binom{i}{2} F_{i+1,n-i-1},$$

which is equivalent to (18).

Combining (17) and (18) then gives the symmetrized inequality

$$F_{i,n-i} \geq \frac{i-1}{i+1} F_{i+1,n-i-1}. \quad (19)$$

The desired inequality (16) then follows from reverse induction on i . If $i = n$, then (16) is trivial. For the inductive step, we suppose that $1 \leq i \leq n-1$. Then, applying (19) followed by the induction hypothesis gives

$$F_{i,n-i} \geq \frac{i-1}{i+1} F_{i+1,n-i-1} \geq \frac{i-1}{i+1} \cdot \frac{(i+1)i}{n(n-1)} F_{n,0} = \frac{i(i-1)}{n(n-1)} F_{n,0}.$$

If $i = 0$, we cannot apply (19). In this case, however, (16) follows directly from non-negativity of f . \square

We now complete the proof of Theorem 10 in the case that $k = 2$. We can formulate the expectation in terms of our symmetric notation as

$$\mathbb{E}[f(\mathbf{x})] = 2^{-n} \sum_{i=0}^n \sum_{A \in \binom{U}{i}} T(A, U \setminus A) = 2^{-n} \sum_{i=0}^n \binom{n}{i} F_{i,n-i}.$$

Then, we have

$$\begin{aligned}
2^{-n} \sum_{i=0}^n \binom{n}{i} F_{i,n-i} &\geq 2^{-n} \sum_{i=2}^n \binom{n}{i} F_{i,n-i} \\
&\geq 2^{-n} \sum_{i=2}^n \binom{n}{i} \frac{i(i-1)}{n(n-1)} F_{n,0} \\
&= 2^{-n} \sum_{i=2}^n \binom{n-2}{i-2} F_{n,0} \\
&= 2^{-n} \sum_{i=0}^{n-2} \binom{n-2}{i} F_{n,0} \\
&= 2^{-n} \cdot 2^{n-2} F_{n,0} \\
&= \frac{1}{4} f(\mathbf{o}),
\end{aligned}$$

where the first inequality follows from non-negativity of f (and hence of F) and the second inequality follows from Lemma 12.

Example 13. As a tight example for $k = 2$, we consider the function $f^{(u,v)}$ defined as in Example 9 for the special case in which $k = 2$. Then, the resulting function is submodular in every orthant and 2-wise monotone and hence must be bisubmodular. Moreover, the probability that a random orthant will set $x_u = 1$, and $x_v = 2$ is $\frac{1}{4}$, and the function has value 0 for all other orthants. Thus, $\mathbb{E}[f^{(u,v)}(\mathbf{x})] = \frac{1}{4}$, whereas the maximum value is 1.

This example is easily extended to ground sets $U = \{u\} \cup V$ of arbitrary size, by setting $f(\mathbf{x}) = \sum_{v \in V} f^{(u,v)}(x_u, x_v)$. This function is also bisubmodular as it is a positive combination of bisubmodular functions. Moreover, the assignment setting $x_u = 1$ and $x_v = 2$ for all $v \in V$ has value $|V|$, but by linearity of expectation a uniform random assignment has expected value only $\frac{1}{4}|V|$.

Example 14. As a tight example for $k \geq 3$, we consider the single-argument k -submodular function $f^{(e)} : \{0, \dots, k\}^{\{e\}}$ given by $f(x_e) = \mathbb{I}[x_e = 1]$. It is easy to verify that this function is indeed k -submodular. Moreover, a uniform random assignment sets $x_e = 1$ with probability only $\frac{1}{k}$, and so $\mathbb{E}[f^{(e)}(x_e)] = \frac{1}{k}$. Similar to the previous example, we can generalize to an arbitrary ground set U by setting $f(\mathbf{x}) = \sum_{e \in U} f^{(e)}(x_e)$. We note also that the value 1 in the definition of each $f^{(e)}$ can be replaced by any value $p \in \{1, \dots, k\}$.

5 A Deterministic Greedy Algorithm

In this section we consider a deterministic greedy algorithm for maximizing a k -set function $f : \{0, \dots, k\}^U \rightarrow \mathbb{R}_+$, that is submodular in every orthant and r -wise monotone for some $1 \leq r \leq k$, where $k \geq 2$. As a special case, we obtain an approximation algorithm for k -submodular functions.

The algorithm begins with the initial solution $\mathbf{s} = \mathbf{0}$ and considers elements of the ground set U in some arbitrary order, permanently setting $s_e = i$ for each element e , based on the increase that this gives in f . Specifically, the algorithm sets s_e to the value i that yields the largest marginal

Algorithm 1 Deterministic Greedy

```
s  $\leftarrow$  0
for each  $e \in U$  do
  for  $i = 1$  to  $k$  do
     $y_i \leftarrow f_{i,e}(\mathbf{s})$ 
   $y = \max(y_1, \dots, y_k)$ 
  Let  $q$  be the smallest value from  $\{1, \dots, k\}$  so that  $y_i = y$ .
   $s_e \leftarrow q$ 
return  $\mathbf{s}$ 
```

increase $f_{i,e}(S)$ in f with respect to the current solution \mathbf{s} . If there is more than one option we set s_e the smallest such i giving the maximal increase.

Theorem 15. *Let \mathbf{s} be the solution produced by the deterministic greedy algorithm on some instance $f : \{0, \dots, k\}^U \rightarrow \mathbb{R}_+$ that is submodular in every orthant and r -wise monotone for some $1 \leq r \leq k$, and let \mathbf{o} be the optimal solution for this instance. Then,*

$$(1 + r)f(\mathbf{s}) \geq f(\mathbf{o}).$$

Proof. Our analysis considers two sequences of n solutions. First let, $\mathbf{s}^{(j)}$ be the algorithm's solution after j elements of U have been considered, and let $U^{(j)}$ be the set of elements that have been considered. Let $\mathbf{o}^{(j)} = \mathbf{o}|_{U \setminus U^{(j)}} + \mathbf{s}^{(j)}$ be a partial solution that agrees with $\mathbf{s}^{(j)}$ on the placement of the elements considered by the greedy algorithm in its first j phases and with \mathbf{o} on the placement of all other elements. Note that in particular we have $\mathbf{o}^{(0)} = \mathbf{o}$ and $\mathbf{o}^{(n)} = \mathbf{s}$. Our analysis of the greedy algorithm will bound the loss in $f(\mathbf{o}^{(j)})$ incurred at the each stage by the improvement in $\mathbf{s}^{(j)}$ made by the algorithm. In Lemma 16, we show that for every $0 \leq j \leq n$, $f(\mathbf{o}^{(j)}) - f(\mathbf{o}^{(j+1)}) \leq r[f(\mathbf{s}^{(j+1)}) - f(\mathbf{s}^{(j)})]$.

Summing this inequality from $j = 0$ to $n - 1$, we obtain

$$\sum_{j=0}^{n-1} [f(\mathbf{o}^{(j)}) - f(\mathbf{o}^{(j+1)})] \leq r \sum_{j=0}^{n-1} [f(\mathbf{s}^{(j+1)}) - f(\mathbf{s}^{(j)})].$$

Telescoping the summations on each side, we then have

$$f(\mathbf{o}^{(0)}) - f(\mathbf{o}^{(n)}) \leq r [f(\mathbf{s}^{(n)}) - f(\mathbf{s}^{(0)})].$$

The theorem then follows immediately from the facts $\mathbf{o}^{(0)} = \mathbf{o}$, $\mathbf{o}^{(n)} = \mathbf{s}^{(n)} = \mathbf{s}$, and $\mathbf{s}^{(0)} \geq 0$. \square

It remains to show the following inequality.

Lemma 16. *For $0 \leq j \leq n - 1$,*

$$f(\mathbf{o}^{(j)}) - f(\mathbf{o}^{(j+1)}) \leq r [f(\mathbf{s}^{(j+1)}) - f(\mathbf{s}^{(j)})].$$

Proof. Let e be the element considered in the $(j + 1)$ th phase of the algorithm.

We define the solution $\mathbf{t} = \mathbf{o}|_{U \setminus U^{(j+1)}} + \mathbf{s}^{(j)}$, and let $a_i = f_{i,e}(\mathbf{t})$ for $1 \leq i \leq k$. Then, we note that for any value i , $\mathbf{t} + i \cdot \mathbf{1}_e$ and $\mathbf{s}^{(j)} + i \cdot \mathbf{1}_e$ are in the same orthant. For some value i , let h be the submodular function induced by this orthant and f . Then h must be submodular, and so

$$y_i = f_{i,e}(\mathbf{s}^{(j)}) = h_e(U^{(j)}) \geq h_e(U \setminus \{e\}) = f_{i,e}(\mathbf{t}) = a_i.$$

Suppose that in the optimal solution we have $o_e = p$ but the greedy algorithm sets $s_e \leftarrow q$. Then, we observe that $f(\mathbf{o}^{(j)}) = f(\mathbf{t}) + f_{p,e}(\mathbf{t})$ and $f(\mathbf{o}^{(j+1)}) = f(\mathbf{t}) + f_{q,e}(\mathbf{t})$, and so

$$f(\mathbf{o}^{(j)}) - f(\mathbf{o}^{(j+1)}) = f_{p,e}(\mathbf{t}) - f_{q,e}(\mathbf{t}) = a_p - a_q.$$

Similarly,

$$f(\mathbf{s}^{(j+1)}) - f(\mathbf{s}^{(j)}) = f_{p,e}(\mathbf{s}^{(j)}) = y_j.$$

By r -wise monotonicity, for any $I \subseteq \{1, \dots, k\}$ with $|I| = r$ we have $\sum_{\ell \in I} a_\ell \geq 0$ and thus $-a_q \leq \sum_{\ell \in I \setminus \{q\}} a_\ell$. Therefore, $a_p - a_q \leq a_p + \sum_{\ell \in I \setminus \{q\}} a_\ell \leq r \cdot y_q$ as $a_i \leq y_i$ for every $1 \leq i \leq k$ and $y_q = \max(y_1, \dots, y_k)$. \square

Combining Theorems 7 and 15 gives us the following.

Corollary 17. *Let \mathbf{s} be the solution produced by the deterministic greedy algorithm for some k -submodular function $f : \{0, \dots, k\}^U \rightarrow \mathbb{R}_+$, and let \mathbf{o} be an optimal solution for this instance. Then,*

$$\frac{1}{3}f(\mathbf{s}) \geq f(\mathbf{o}).$$

The following is a tight example for Theorem 15.

Example 18. Let $0 \leq r \leq k$ and consider the function $f^{(u,v)} : \{0, \dots, k\}^{\{u,v\}} \rightarrow \mathbb{R}_+$ given by $f^{(u,v)}(x_u, x_v) = \frac{1}{r+1} \llbracket x_u \neq 0 \rrbracket + \frac{r}{r+1} \llbracket x_u \neq 1 \wedge x_v = 2 \rrbracket$. We shall first show that $f^{(u,v)}$ is submodular in every orthant and r -wise monotone.

Fix an orthant $(x_u = i, x_v = j)$ with $j \neq 2$, and let h be the function induced by $f^{(u,v)}$ and this orthant. Then, the marginals of h are given by:

$$\begin{aligned} h_u(\emptyset) &= h(\{u\}) - h(\emptyset) = \frac{1}{r+1} & h_v(\emptyset) &= h(\{v\}) - h(\emptyset) = 0 \\ h_u(\{v\}) &= h(\{u, v\}) - h(\{v\}) = \frac{1}{r+1} & h_v(\{u\}) &= h(\{u, v\}) - h(\{u\}) = 0. \end{aligned}$$

Now, fix an orthant $(x_u = i, x_v = 2)$, and let h be the function induced by $f^{(u,v)}$ and this orthant. We have

$$\begin{aligned} h_u(\emptyset) &= h(\{u\}) - h(\emptyset) = \frac{1}{r+1} & h_v(\emptyset) &= h(\{v\}) - h(\emptyset) = \frac{r}{r+1} \\ h_u(\{v\}) &= h(\{u, v\}) - h(\{v\}) = \frac{1}{r+1} - \frac{r}{r+1} \llbracket i = 1 \rrbracket & h_v(\{u\}) &= h(\{u, v\}) - h(\{u\}) = \frac{r}{r+1} \llbracket i \neq 1 \rrbracket. \end{aligned}$$

In all cases, the marginals of h are decreasing, and so $f^{(u,v)}$ is submodular in every orthant. We now show that $f^{(u,v)}$ is r -wise monotone. We consider an arbitrary set $I \subseteq \{1, \dots, k\}$ with $|I| = r$ and shall show that $\sum_{i \in I} f_{i,e}^{(u,v)}(x_u, x_v) \geq 0$ for all $e \in \{u, v\}$. The marginals of $f^{(u,v)}$ are non-negative,

except the one obtained by setting x_u from 0 to 1 in the case that $x_v = 2$. Thus, the only non-trivial case is that in which $e = u$, $x_u = 0$, $x_v = 2$, and $1 \in I$. In this case,

$$\begin{aligned} \sum_{i \in I} [f_{i,u}^{(u,v)}(0, 2)] &= f^{(u,v)}(1, 2) - f^{(u,v)}(0, 2) + \sum_{i \in I \setminus \{1\}} [f^{(u,v)}(i, 2) - f^{(u,v)}(0, 2)] \\ &= \frac{1}{r+1} - \frac{r}{r+1} + (r-1) \cdot \frac{1}{r+1} = 0. \end{aligned}$$

Now, we analyze the performance of the deterministic greedy algorithm on $f^{(u,v)}$. We suppose, without loss of generality, that the algorithm considers u before v . When u is considered, we have $\mathbf{s} = \mathbf{0}$ and $f_{i,u}^{(u,v)}(0, 0) = \frac{1}{r+1}$ for all $i \in \{1, \dots, k\}$, and so the algorithm sets $s_u = 1$. In the next iteration, we have $f_{i,v}^{(u,v)}(1, 0) = 0$ for all values $i \in \{1, \dots, k\}$, and so the algorithm set $s_v = 1$ and returns $\mathbf{s} = (1, 1)$. We then have $f^{(u,v)}(\mathbf{s}) = \frac{1}{r+1}$, but $f^{(u,v)}(2, 2) = 1$. As in previous examples, we can easily obtain a function over ground sets of arbitrary size by summing the values of several different functions $f^{(u,v)}$.

6 A Randomized Greedy Algorithm

In this section we consider the performance of a simple randomized greedy algorithm for maximizing a k -set function that is submodular in every orthant and k -wise monotone. Our algorithm is inspired by the algorithm of Buchbinder et al. [5] for unconstrained submodular maximization. It begins with the initial solution $\mathbf{s} = \mathbf{0}$ and considers elements of the ground set U in some arbitrary order, permanently setting s_e to some value $i \in \{1, \dots, k\}$, based on the marginal increase in f that this yields. Specifically, the algorithm randomly sets $s_e = i$ with probability proportional to the resulting marginal increase $f_{i,e}(\mathbf{s})$ in f with respect to the current solution \mathbf{s} . If $f_{i,e}(\mathbf{s}) < 0$, we set $s_e = i$ with probability 0. Note that Theorem 7 shows that we cannot have $f_{i,e}(\mathbf{s}) < 0$ for all i , but it may be the case that $f_{i,e}(\mathbf{s}) = 0$ for all i . In this case, we set $s_e = 1$.

Algorithm 2 Randomized Greedy

```

s  $\leftarrow \mathbf{0}$ 
for each  $e \in U$  do
  for  $i = 1$  to  $k$  do
     $y_i \leftarrow \max(0, f_{i,e}(\mathbf{s}))$ 
   $\beta = \sum_{i=1}^k y_i$ 
  if  $\beta \neq 0$  then
    Let  $q \in \{1, \dots, k\}$  be chosen randomly, with  $\Pr[i = \ell] = \frac{y_\ell}{\beta}$  for all  $\ell \in \{1, \dots, k\}$ .
     $s_e \leftarrow q$ 
  else
     $s_e \leftarrow 1$ 
return  $\mathbf{s}$ 

```

Theorem 19. *Let $f : \{0, \dots, k\}^U$ be a k -set function that is submodular in every orthant and k -wise monotone, where $k \geq 2$. Let \mathbf{o} be orthant of U that maximizes f and let \mathbf{s} be the orthant*

produced by the randomized greedy algorithm. Then,

$$\left(1 + \sqrt{\frac{k}{2}}\right) \mathbb{E}[f(\mathbf{s})] \geq f(\mathbf{o}).$$

Proof. As in the analysis of the deterministic greedy algorithm, we consider 2 sequences of n solutions. Let $\mathbf{s}^{(j)}$, and $\mathbf{o}^{(j)}$ be defined as in the proof of Theorem 15, and note that \mathbf{s} (and hence each $\mathbf{s}^{(j)}$) is now a random variable depending on the random choices made by the algorithm. In Lemma 20, we bound the expected decrease $\mathbb{E}[f(\mathbf{o}^{(j)}) - f(\mathbf{o}^{(j+1)})]$ relative to the increase $\mathbb{E}[f(\mathbf{s}^{(j+1)}) - f(\mathbf{s}^{(j)})]$ in each iteration. Specifically, we show that

$$\mathbb{E}[f(\mathbf{o}^{(j)}) - f(\mathbf{o}^{(j+1)})] \leq \sqrt{\frac{k}{2}} \mathbb{E}[f(\mathbf{s}^{(j+1)}) - f(\mathbf{s}^{(j)})] \quad (20)$$

for all j . Summing the resulting inequalities for $j = 0$ to n , we then obtain

$$\sum_{j=0}^n \mathbb{E}[f(\mathbf{o}^{(j)}) - f(\mathbf{o}^{(j+1)})] \leq \sqrt{\frac{k}{2}} \sum_{j=0}^n \mathbb{E}[f(\mathbf{s}^{(j+1)}) - f(\mathbf{s}^{(j)})],$$

which simplifies to

$$\mathbb{E}[f(\mathbf{o}^{(0)})] - \mathbb{E}[f(\mathbf{o}^{(n)})] \leq \sqrt{\frac{k}{2}} (\mathbb{E}[f(\mathbf{s}^{(n)})] - \mathbb{E}[f(\mathbf{s}^{(0)})]) \leq \sqrt{\frac{k}{2}} \mathbb{E}[f(\mathbf{s}^{(n)})].$$

The theorem then follows from the definitions $\mathbf{o}^{(0)} = \mathbf{o}$, and $\mathbf{s}^{(n)} = \mathbf{o}^{(n)} = \mathbf{s}$. \square

We now show that inequality (20) must hold.

Lemma 20. *For any $0 \leq j \leq n$,*

$$\mathbb{E}[f(\mathbf{o}^{(j)}) - f(\mathbf{o}^{(j+1)})] \leq \sqrt{\frac{k}{2}} \mathbb{E}[f(\mathbf{s}^{(j+1)}) - f(\mathbf{s}^{(j)})].$$

Proof. Let e be the element of U considered by the randomized greedy algorithm in the $(j+1)$ th phase, and let $U^{(j)}$ and $\mathbf{o}^{(j)}$ be defined as in the proof of Theorems 19 and 15. We condition on an arbitrary, fixed value for both $\mathbf{s}^{(j)}$, $\mathbf{o}^{(j)}$, and consider the expectation over choices the algorithm makes for e . Because our result will hold for an arbitrary $\mathbf{s}^{(j)}$ or $\mathbf{o}^{(j)}$ it then extends to the expectation over the first j choices made by the algorithm.

As in the proof of Lemma 16, we define the solution $\mathbf{t} = \mathbf{o}|_{U^{(j-1)}} + \mathbf{s}^{(j-1)}$, and set $a_i = f_{i,e}(\mathbf{t})$ for $1 \leq i \leq k$. Let the values y_i be defined as in the algorithm. Then, as in the proof of Lemma 16, submodularity of f in every orthant implies that

$$a_i \leq y_i \text{ for every } i \in \{1, \dots, k\}. \quad (21)$$

Moreover, k -wise monotonicity of f implies that⁵.

$$\sum_i a_i \geq 0. \quad (22)$$

⁵In order to simplify our notation, we assume throughout this section that all summations over i run from $1 \leq i \leq k$, unless other conditions are given.

Finally, by the construction of Algorithm 2, we have $y_i \geq 0$ for each $1 \leq i \leq k$.

Now, let suppose that in the optimal solution $o_e = p$ but the greedy algorithm sets $s_e \leftarrow q$. Then, we have $f(\mathbf{o}^{(j)}) = f(\mathbf{t}) + f_{p,e}(\mathbf{t})$ and $f(\mathbf{o}^{(j+1)}) = f(\mathbf{t}) + f_{q,e}(\mathbf{t})$, and so, as in the proof of Lemma 16,

$$f(\mathbf{o}^{(j)}) - f(\mathbf{o}^{(j+1)}) = f_{p,e}(\mathbf{t}) - f_{q,e}(\mathbf{t}) = a_p - a_q,$$

and

$$f(\mathbf{s}^{(j+1)}) - f(\mathbf{s}^{(j)}) = f_{q,e}(\mathbf{s}^{(j)}) = y_q.$$

For any given value q , the probability that the greedy algorithm makes such a choice is precisely y_q/β , and so

$$\mathbb{E}[f(\mathbf{s}^{(j+1)}) - f(\mathbf{s}^{(j)})] = \frac{1}{\beta} \sum_i y_i^2,$$

and

$$\mathbb{E}[f(\mathbf{o}^{(j)}) - f(\mathbf{o}^{(j+1)})] = \frac{1}{\beta} \sum_i y_i(a_p - a_i) = \frac{1}{\beta} \sum_{i \neq p} y_i(a_p - a_i).$$

In order to prove the lemma it is thus sufficient to show that

$$\sum_{i \neq p} y_i(a_p - a_i) \leq \sqrt{\frac{k}{2}} \sum_i y_i^2. \quad (23)$$

For any value of y_1, \dots, y_k , the left hand side of (23) is upper bounded by the optimal value of the following linear program in a_1, \dots, a_k , whose constraints are given by (21) and (22):

$$\begin{aligned} & \text{maximize} && \sum_{i \neq p} y_i(a_p - a_i) \\ & \text{subject to} && a_i \leq y_i, && \text{for } 1 \leq i \leq k \\ & && \sum_i a_i \geq 0 \end{aligned}$$

We consider an optimal, extreme-point solution a_1^*, \dots, a_k^* for this program. We first note that by increasing a_p we cannot violate the final constraint and can only increase the objective, and so we may assume that $a_p^* = y_p$. Of the remaining k constraints, $k-1$ must be tight, of which $k-2$ must be of the first type. Hence, for all i except at most 1 value $\ell \neq p$, we in fact have $a_i^* = y_i$. This accounts for $k-1$ total tight constraints. The final tight constraint must imply either $a_\ell^* = y_\ell$ or $\sum_i a_i^* = 0$. Because $a_i^* = y_i$ for all $i \neq \ell$, the latter is equivalent to $a_\ell^* = -\sum_{i \neq \ell} y_i$. Moreover, because $y_i \geq 0$ for all i , setting $a_\ell^* = -\sum_{i \neq \ell} y_i$ always gives an objective value at least as large as setting $a_\ell^* = y_\ell$. Thus, we can characterize the optimal solution to this linear program by $a_i^* = y_i$ for all $i \neq \ell$, and $a_\ell^* = -\sum_{i \neq \ell} y_i$, where ℓ is some value distinct from p .

Returning to (23), we have

$$\begin{aligned}
\sum_{i \neq p} y_i(a_p - a_i) &\leq \sum_{i \neq p} y_i(a_p^* - a_i^*) \\
&= \sum_{i \neq p, \ell} y_i(y_p - y_i) + y_\ell \left(y_p + \sum_{i \neq \ell} y_i \right) \\
&= 2y_\ell y_p + \sum_{i \neq p, \ell} [y_\ell y_i + y_p y_i - y_i^2],
\end{aligned}$$

for any $y_1, \dots, y_k \geq 0$. In order to prove (23) it then suffices to show that

$$0 \leq \alpha \sum_i y_i^2 - 2y_\ell y_p - \sum_{i \neq p, \ell} [y_\ell y_i + y_p y_i - y_i^2], \quad (24)$$

where $\alpha = \sqrt{\frac{k}{2}}$. This follows directly from the fact that the right hand side of (24) can be written as the following sum of squares:

$$(y_\ell - y_p)^2 + \sum_{j \neq o, \ell} \left(\sqrt{\frac{\alpha-1}{k-2}} y_\ell - \sqrt{\frac{\alpha+1}{2}} y_j \right)^2 + \sum_{j \neq o, \ell} \left(\sqrt{\frac{\alpha-1}{k-2}} y_p - \sqrt{\frac{\alpha+1}{2}} y_j \right)^2. \quad (25)$$

In order to verify that this is the case, we note that

$$(y_\ell - y_p)^2 = y_\ell^2 - 2y_\ell y_p + y_p^2$$

and

$$\begin{aligned}
\left(\sqrt{\frac{\alpha-1}{k-2}} y_\ell - \sqrt{\frac{\alpha+1}{2}} y_i \right)^2 &= \frac{\alpha-1}{k-2} y_\ell^2 - 2\sqrt{\frac{(\alpha-1)(\alpha+1)}{2(k-2)}} y_\ell y_i + \frac{\alpha+1}{2} y_i^2 \\
&= \frac{\alpha-1}{k-2} y_\ell^2 - 2\sqrt{\frac{\alpha^2-1}{2(k-2)}} y_\ell y_i + \frac{\alpha+1}{2} y_i^2 \\
&= \frac{\alpha-1}{k-2} y_\ell^2 - 2\sqrt{\frac{\frac{k}{2}-1}{2(k-2)}} y_\ell y_i + \frac{\alpha+1}{2} y_i^2 \\
&= \frac{\alpha-1}{k-2} y_\ell^2 - 2\sqrt{\frac{\frac{k-2}{2}}{2(k-2)}} y_\ell y_i + \frac{\alpha+1}{2} y_i^2 \\
&= \frac{\alpha-1}{k-2} y_\ell^2 - 2\sqrt{\frac{1}{4}} y_\ell y_i + \frac{\alpha+1}{2} y_i^2 \\
&= \frac{\alpha-1}{k-2} y_\ell^2 - y_\ell y_i + \frac{\alpha+1}{2} y_i^2,
\end{aligned}$$

and, similarly,

$$\left(\sqrt{\frac{\alpha-1}{k-2}} y_p - \sqrt{\frac{\alpha+1}{2}} y_i \right)^2 = \frac{\alpha-1}{k-2} y_p^2 - y_p y_i + \frac{\alpha+1}{2} y_i^2$$

Thus, (25) is equal to

$$\begin{aligned}
& y_\ell^2 - 2y_\ell y_p + y_p^2 + \sum_{i \neq p, \ell} \left[\frac{\alpha-1}{k-2} y_\ell^2 - y_\ell y_i + \frac{\alpha+1}{2} y_i^2 \right] + \sum_{i \neq p, \ell} \left[\frac{\alpha-1}{k-2} y_p^2 - y_p y_i + \frac{\alpha+1}{2} y_i^2 \right] \\
&= y_\ell^2 - 2y_\ell y_p + y_p^2 + (\alpha-1)y_\ell^2 + (\alpha-1)y_p^2 - \sum_{i \neq p, \ell} \left[y_\ell y_i - \frac{\alpha+1}{2} y_i^2 \right] - \sum_{i \neq p, \ell} \left[y_p y_i - \frac{\alpha+1}{2} y_i^2 \right] \\
&= y_\ell^2 - 2y_\ell y_p + y_p^2 + (\alpha-1)y_\ell^2 + (\alpha-1)y_p^2 - \sum_{i \neq p, \ell} [y_\ell y_i + y_p y_i - (\alpha+1)y_i^2] \\
&= \alpha y_\ell^2 + \alpha y_p^2 - 2y_\ell y_p + \alpha \sum_{i \neq p, \ell} y_i^2 - \sum_{i \neq p, \ell} [y_\ell y_i + y_p y_i - y_i^2] \\
&= \alpha \sum_i y_i^2 - 2y_\ell y_p - \sum_{i \neq p, \ell} [y_\ell y_i + y_p y_i - y_i^2]. \quad \square
\end{aligned}$$

The guarantees we obtain for the randomized greedy algorithm are better than for the deterministic greedy algorithm on k -wise monotone k -set functions. In particular, for bisubmodular functions (i.e. 2-wise monotone 2-set functions) we obtain the same approximation results that Feldman et al. [5] give for submodular functions: namely, Theorem 15 shows that the deterministic greedy algorithm has an approximation ratio of $1/3$ while Theorem 19 shows that the randomized greedy algorithm has an expected approximation ratio of $1/2$. We discuss this connection further in the next section.

Returning to the general setting of r -wise monotone functions, we note that the guarantees of Theorem 19 hold only in the case $r = k$, and also degrade with k . Thus, we might expect that for small values of r the deterministic greedy algorithm may outperform the randomized greedy algorithm. While we do not have a tight example for the randomized greedy algorithm on r -wise monotone k -set functions for every fixed value of r and k , the following example confirms our intuition that the randomized algorithm can indeed perform worse than the deterministic algorithm for k -submodular (i.e. pairwise monotone) functions, once k grows large enough.

Example 21. Consider the weighted set-coverage function $f^{(u,v)} : \{0, \dots, k\}^{\{u,v\}} \rightarrow \mathbb{R}_+$ given as follows. We have a universe $\{a, b\}$ where a has weight 1 and b has weight $\gamma = \frac{1}{\sqrt{k-1}}$. Additionally, we have sets $S_1 = \{a\}$ and $S_i = \{b\}$ for every $2 \leq i \leq k$, and $T_i = \{b\}$, for every $1 \leq i \leq k$. The value of $f^{(u,v)}(x_u, x_v)$ is then simply the total weight of all elements in $S_u \cup T_v$. The function induced by $f^{(u,v)}$ and any orthant is then a weighted set coverage function, and so is submodular. Moreover, all marginals of $f^{(u,v)}$ are non-negative and so $f^{(u,v)}$ is trivially r -wise monotone for any r .

We now consider the performance of the randomized greedy algorithm on $f^{(u,v)}$. We suppose, without loss of generality, that the greedy algorithm considers u before v . Initially we have $\mathbf{s} = \mathbf{0}$, and in the first phase, the algorithm sets $s_u \leftarrow 1$ with probability $\frac{1}{1+(k-1)\gamma}$ and for each $2 \leq i \leq k$, sets $s_u \leftarrow i$ with probability $\frac{\gamma}{1+(k-1)\gamma}$. In the next step, the algorithm considers v . We note that all the sets T_i are identical, and so the algorithm's particular choice in this phase does not affect the final value of the function. The solution \mathbf{s} produced by the algorithm has value $1 + \gamma$ if $s_u = 1$ and γ otherwise. Thus, the expected value of solution produced by the algorithm is:

$$\frac{1 + \gamma + (k-1)\gamma^2}{1 + (k-1)\gamma} = \frac{2 + \gamma}{1 + (k-1)\gamma}.$$

The optimal value of $f^{(u,v)}$ is $1 + \gamma$ and so the expected approximation ratio of the randomized greedy algorithm on $f^{(u,v)}$ is

$$\alpha = \frac{2 + \gamma}{1 + (k - 1)\gamma} \cdot \frac{1}{1 + \gamma} = \frac{2 + \gamma}{1 + (k - 1)\gamma + \gamma + (k - 1)\gamma^2} = \frac{2 + \gamma}{2 + k\gamma}.$$

In particular, for all $k \geq 21$, we have $\alpha < 1/3$. For large k , α is approximately $1 / \left(1 + \sqrt{\frac{k}{4}}\right)$. In the appendix, we show that the randomized greedy algorithm does indeed attain a similar, improved ratio for k -submodular functions.

7 Conclusion

In the preceding sections we have considered the problem of maximizing k -submodular functions by both a random partition and two simple greedy algorithms. In the case of maximizing a bisubmodular function, we obtained the same approximation ratios as those already known in the submodular case: $1/4$ for the naive random solution [9] and $1/2$ via a randomized greedy approach [5]. We can make this correspondence more explicit by considering the following embedding of a submodular function into a bisubmodular function. Given a submodular function $g : 2^U \rightarrow \mathbb{R}_+$, we consider the biset function $f : 3^U \rightarrow \mathbb{R}_+$ defined by

$$f(S, T) \stackrel{\text{def}}{=} g(S) + g(U \setminus T) - g(U). \quad (26)$$

This embedding has been studied by Fujishige and Iwata, who show that the function f is bisubmodular and has the following property: if (S, T) is a minimizer (maximizer) of f then both S and $U \setminus T$ are minimizers (maximizers) of g [11]. Thus, exact 2-submodular function minimization (maximization) is a generalization of 1-submodular function minimization (maximization). We can in fact show a stronger result: that this embedding preserves approximability.

Suppose that some algorithm gives a α -approximation for bisubmodular maximization. Then, consider an arbitrary submodular function g and let f be the embedding of g defined as in (26). Let $O = (O_1, O_2)$ be a maximizer f , and suppose that the algorithm returns a solution $S = (S_1, S_2)$. Then, since f is pairwise monotone, we can greedily extend S to a partition $S' = (S'_1, S'_2)$ of U . Similarly, we can assume without loss of generality that O is a partition of U . Then, we have $f(U \setminus S'_2) = f(S'_1)$ and $f(U \setminus O_2) = f(O_1)$, and so

$$\begin{aligned} g(S'_1) &= \frac{1}{2} (g(S'_1) + g(U \setminus S'_1)) \\ &= \frac{1}{2} (f(S'_1, S'_2) + g(U)) \\ &\geq \frac{1}{2} (\alpha f(O_1, O_2) + g(U)) \\ &= \frac{1}{2} (\alpha g(O_1) + \alpha g(U \setminus O_2) + (1 - \alpha)g(U)) \\ &\geq \frac{1}{2} (\alpha g(O_1) + \alpha g(U \setminus O_2)) \\ &= \alpha g(O_1). \end{aligned}$$

Since O_1 is a maximizer of g , the resulting algorithm is an α -approximation for maximizing g . Hence, the $1/2 + \epsilon$ inapproximability results of [9, 8] hold for bisubmodular maximization as well, in both the value oracle setting and under the assumption that $NP \neq RP$.

The embedding (26) also allows us to provide new intuition for the performance of the randomized greedy algorithm for submodular maximization considered by Buchbinder et al. [5]. This algorithm maintains 2 solutions, S_1 and S_2 which are initially \emptyset and U . At each step, it considers an element e , and either adds e to S_1 or removes e from S_2 , with probability proportional to the resulting increase in the submodular function in either case.

In comparison, we consider the case in which we embed a submodular function g into a bisubmodular function f using (26) and then run the greedy algorithm of Section 6 on f . Suppose at some step we have a current solution $T = (T_1, T_2)$ and we consider element e , and define $S_1 = T_1$ and $S_2 = U \setminus T_2$. The algorithm will add e to either T_1 or T_2 with probability proportional to the resulting increase in f . In the first case, this increase is precisely $g(T_1 + e) - g(T_1) = g(S_1 + e) - g(S_1)$, and adding e to T_1 corresponds to adding e to S_1 . In the second case this increase is precisely $g(U \setminus T_2) - g(U \setminus (T_2 + e)) = g(S_2) - g(S_2 - e)$ and adding e to T_1 corresponds to removing e from S_1 . Thus, the operation of the algorithm of Buchbinder et al. [5] may be viewed as that of the natural, straightforward randomized greedy algorithm presented in Section 6, viewed through the lens of the embedding (26).

An interesting open question is whether the symmetry gap technique from [39, 8] can be generalized to obtain hardness results for k -submodular maximization for $k \geq 3$, and, more generally, for maximizing k -set functions that are submodular in every orthant and r -wise monotone for some $1 \leq r \leq k$.

Acknowledgments

We are grateful to Maxim Sviridenko for many insightful conversations.

References

- [1] Kazutoshi Ando. K -submodular functions and convexity of their Lovász extension. *Discrete Applied Mathematics*, 122(1-3):1–12, 2002.
- [2] Kazutoshi Ando, Satoru Fujishige, and Takeshi Naitoh. A characterization of bisubmodular functions. *Discrete Mathematics*, 148(1-3):299–303, 1996.
- [3] Jesús M. Bilbao, Julio R. Fernández, Nieves Jiménez, and Jorge J. López. Survey of bicooperative games. In Altannar Chinchuluun, Panos M. Pardalos, Athanasios Migdalas, and Leonidas Pitsoulis, editors, *Pareto Optimality, Game Theory and Equilibria*. Springer, 2008.
- [4] André Bouchet. Greedy algorithm and symmetric matroids. *Mathematical Programming*, 38(2):147–159, 1987.
- [5] Niv Buchbinder, Moran Feldman, Joseph Naor, and Roy Schwartz. A tight linear time $(1/2)$ -approximation for unconstrained submodular maximization. In *Proceedings of the 53rd Annual IEEE Symposium on Foundations of Computer Science (FOCS'12)*, pages 649–658. IEEE, 2012.

- [6] Ramaswamy Chandrasekaran and Santosh N. Kabadi. Pseudomatroids. *Discrete Mathematics*, 71(3):205–217, 1988.
- [7] David A. Cohen, Martin C. Cooper, Peter G. Jeavons, and Andrei A. Krokhin. The Complexity of Soft Constraint Satisfaction. *Artificial Intelligence*, 170(11):983–1016, 2006.
- [8] Shahar Dobzinski and Jan Vondrák. From query complexity to computational complexity. In *Proceedings of the 44th Annual ACM Symposium on Theory of Computing (STOC’12)*, pages 1107–1116. ACM, 2012.
- [9] Uriel Feige, Vahab S. Mirrokni, and Jan Vondrák. Maximizing Non-monotone Submodular Functions. *SIAM Journal on Computing*, 40(4):1133–1153, 2011.
- [10] Satoru Fujishige. *Submodular Functions and Optimization*, volume 58 of *Annals of Discrete Mathematics*. North-Holland, Amsterdam, 2nd edition, 2005.
- [11] Satoru Fujishige and Satoru Iwata. Bisubmodular Function Minimization. *SIAM Journal on Discrete Mathematics*, 19(4):1065–1073, 2005.
- [12] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman, 1979.
- [13] M. Grötschel, L. Lovasz, and A. Schrijver. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, 1(2):169–198, 1981.
- [14] M. Grötschel, L. Lovasz, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*, volume 2 of *Algorithms and Combinatorics*. Springer, 1988.
- [15] Anna Huber and Vladimir Kolmogorov. Towards Minimizing k -Submodular Functions. In *Proceedings of the 2nd International Symposium on Combinatorial Optimization (ISCO’12)*, volume 7422 of *Lecture Notes in Computer Science*, pages 451–462. Springer, 2012.
- [16] Anna Huber, Andrei Krokhin, and Robert Powell. Skew bisubmodularity and valued CSPs. *SIAM Journal on Computing*, 43(3):1064–1084, 2014.
- [17] S. Iwata. A fully combinatorial algorithm for submodular function minimization. *Journal of Combinatorial Theory, Series B*, 84(2):203–212, 2002.
- [18] S. Iwata. A faster scaling algorithm for minimizing submodular functions. *SIAM Journal on Computing*, 32(4):833–840, 2003.
- [19] Satoru Iwata. Submodular Function Minimization. *Mathematical Programming*, 112(1):45–64, 2008.
- [20] Satoru Iwata, Lisa Fleischer, and Satoru Fujishige. A combinatorial strongly polynomial algorithm for minimizing submodular functions. *Journal of the ACM*, 48(4):761–777, 2001.
- [21] Satoru Iwata, Shin ichi Tanigawa, and Yuichi Yoshida. Bisubmodular function maximization and extensions. Technical Report METR 2013-16, The University of Tokyo, 2013.

- [22] Satoru Iwata and James B. Orlin. A Simple Combinatorial Algorithm for Submodular Function Minimization. In *Proceedings of the 20th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'09)*, pages 1230–1237, 2009.
- [23] Vladimir Kolmogorov. Submodularity on a tree: Unifying L^\sharp -convex and bisubmodular functions. In *Proceedings of the 36th International Symposium on Mathematical Foundations of Computer Science (MFCS'11)*, volume 6907 of *Lecture Notes in Computer Science*, pages 400–411. Springer, 2011.
- [24] Bernhard Korte and Jens Vygen. *Combinatorial Optimization*, volume 21 of *Algorithms and Combinatorics*. Springer, 4th edition, 2007.
- [25] Jon Lee, Maxim Sviridenko, and Jan Vondrák. Submodular Maximization over Multiple Matroids via Generalized Exchange Properties. *Mathematics of Operations Research*, 35(4):795–806, 2010.
- [26] László Lovász. Submodular Functions and Convexity. In A. Bachem, M. Grötschel, and B. Korte, editors, *Mathematical Programming – The State of the Art*, pages 235–257, Berlin, 1983. Springer.
- [27] S. Thomas McCormick and Satoru Fujishige. Strongly polynomial and fully combinatorial algorithms for bisubmodular function minimization. *Mathematical Programming*, 122(1):87–120, 2010.
- [28] H. Narayanan. *Submodular Functions and Electrical Networks*. North-Holland, Amsterdam, 1997.
- [29] George L. Nemhauser and Laurence A. Wolsey. *Integer and Combinatorial Optimization*. John Wiley & Sons, 1988.
- [30] James B. Orlin. A faster strongly polynomial time algorithm for submodular function minimization. *Mathematical Programming*, 118(2):237–251, 2009.
- [31] Christos H. Papadimitriou and Mihalis Yannakakis. Optimization, Approximation, and Complexity Classes. *Journal of Computer and System Sciences*, 43(3):425–440, 1991.
- [32] Liqun Qi. Directed submodularity, ditroids and directed submodular flows. *Mathematical Programming*, 42(1-3):579–599, 1988.
- [33] Alexander Schrijver. A Combinatorial Algorithm Minimizing Submodular Functions in Strongly Polynomial Time. *Journal of Combinatorial Theory, Series B*, 80(2):346–355, 2000.
- [34] Alexander Schrijver. *Combinatorial Optimization: Polyhedra and Efficiency*, volume 24 of *Algorithms and Combinatorics*. Springer, 2003.
- [35] Ajit P. Singh, Andrew Guillory, and Jeff Bilmes. On bisubmodular maximization. In *Proceedings of the 15th International Conference on Artificial Intelligence and Statistics (AISTATS'12)*, volume 22 of *JLMR Workshop and Conference Proceedings*, pages 1055–1063, 2012.

- [36] Johan Thapper and Stanislav Živný. The power of linear programming for valued CSPs. In *Proceedings of the 53rd Annual IEEE Symposium on Foundations of Computer Science (FOCS'12)*, pages 669–678. IEEE, 2012.
- [37] Johan Thapper and Stanislav Živný. The complexity of finite-valued CSPs. In *Proceedings of the 45th ACM Symposium on the Theory of Computing (STOC'13)*, pages 695–704. ACM, 2013.
- [38] Donald Topkis. *Supermodularity and Complementarity*. Princeton University Press, 1998.
- [39] Jan Vondrák. Symmetry and Approximability of Submodular Maximization Problems. In *Proceedings of the 50th IEEE Symposium on Foundations of Computer Science (FOCS '09)*, pages 651–670. IEEE Computer Society, 2009.
- [40] Magnus Wahlström. Half-integrality, LP-branching and FPT Algorithms. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'14)*, pages 1762–1781. SIAM, 2014.
- [41] Justin Ward and Stanislav Živný. Maximizing bisubmodular and k -submodular functions. In *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'14)*, pages 1468–1481. SIAM, 2014.

A Improved Analysis of Algorithm 2 for k -Submodular Functions

In the case that f is in fact pairwise monotone (and, hence, k -submodular), we can prove the following stronger form of Lemma 20.

Lemma 22. *Suppose that f is k -submodular. Then, for any $0 \leq j \leq n$,*

$$\mathbb{E}[f(\mathbf{o}^{(j)}) - f(\mathbf{o}^{(j+1)})] \leq \alpha \mathbb{E}[f(\mathbf{s}^{(j+1)}) - f(\mathbf{s}^{(j)})].$$

where $\alpha = \max(1, \sqrt{\frac{k-1}{4}})$.

Proof. Using the same notation as in the proof of Lemma 20, we shall now show

$$\sum_{i \neq p} y_i (a_p - a_i) \leq \alpha \sum_i y_i^2, \tag{27}$$

where $\alpha = \max(1, \sqrt{\frac{k-1}{4}})$. As in the proof of Lemma 20, we note that for any value of y_1, \dots, y_k , the left hand side of (27) is upper bounded by the optimal value of a linear program in a_1, \dots, a_k . Now, however, because f is pairwise monotone, we replace the (22) with $\binom{k}{2}$ constraints of the form of $a_i + a_\ell \geq 0$. This gives the program

$$\begin{aligned} & \text{maximize} && \sum_{i \neq p} y_i (a_p - a_i) \\ & \text{subject to} && a_i \leq y_i && 1 \leq i \leq k \\ & && a_i + a_\ell \geq 0 && \forall \{i, \ell\} \in \binom{\{1, \dots, k\}}{2}. \end{aligned}$$

Consider an optimal solution for this program. We note that increasing a_p cannot violate any constraint $a_p + a_\ell \geq 0$, and will increase the objective. Thus, we may assume that $a_p^* = y_p \geq 0$. We now consider 2 cases.

First, suppose that we have $a_\ell^* = -t < 0$ for some $\ell \in \{1, \dots, k\}$ and some value $t > 0$. Because $a_i^* + a_\ell^* \geq 0$ for all $i \neq \ell$, there can be at most one such ℓ . Moreover, we must have $a_i^* \geq t$ for all $i \neq \ell$. For any value $i \notin \{\ell, p\}$, we note that decreasing a_i^* can only increase the objective of our linear program. Thus, in this case, we may assume that $a_i^* = t$ for all $i \notin \{\ell, p\}$, $a_\ell = -t$ and $a_p = y_p$. We can then rewrite our objective as:

$$\sum_{i \neq p} y_i y_p + t \left(y_\ell - \sum_{j \neq \ell, p} y_j \right). \quad (28)$$

Because $t > 0$, we must have $y_\ell \geq \sum_{j \neq \ell, p} y_j$ (otherwise, we could increase (28) by decreasing t). Moreover, we must have $t \leq y_p$, since otherwise we would have $a_p^* + a_\ell^* = y_p - t < 0$. Hence, we have:

$$\sum_{i \neq p} y_i y_p + t \left(y_\ell - \sum_{j \neq \ell, p} y_j \right) \leq \sum_{i \neq p} y_i y_p + y_p y_\ell - y_p \sum_{j \neq \ell, p} y_j = 2y_p y_\ell \leq y_p^2 + y_\ell^2 \leq \sum_i y_i^2,$$

and we have proved (27) with $\alpha = 1$.

Next, suppose that $a_i \geq 0$ for all $i \in \{1, \dots, k\}$. Then, the objective of our program satisfies

$$\begin{aligned} \sum_{i \neq p} y_i (a_p - a_i) &\leq \sum_{i \neq p} y_i a_p \\ &= \sum_{i \neq p} y_i y_p \\ &= \frac{1}{2\sqrt{k-1}} \cdot 2\sqrt{k-1} y_p \sum_{i \neq p} y_i \\ &\leq \frac{1}{2\sqrt{k-1}} \left[(k-1)y_p^2 + \left(\sum_{i \neq p} y_i \right)^2 \right] \\ &\leq \frac{1}{2\sqrt{k-1}} \left[(k-1)y_p^2 + (k-1) \sum_{i \neq p} y_i^2 \right] \\ &= \frac{\sqrt{k-1}}{2} \sum_i y_i^2, \end{aligned}$$

where the second inequality follows from $a^2 + b^2 \geq 2ab$ for any real numbers a and b , and third inequality follows from the Cauchy-Schwarz inequality. Thus, we have proved (27) with $\alpha = \sqrt{\frac{k-1}{4}}$. \square

By replacing Lemma 20 with Lemma 22, in the proof of Theorem 19, we obtain the following result.

Theorem 23. *Let $f : \{0, \dots, k\}^U$ be a k -submodular set function. Let \mathbf{o} be an orthant of U that maximizes f and let \mathbf{s} be the orthant of U produced by the randomized greedy algorithm. Then,*

$$(1 + \alpha) \mathbb{E}[f(\mathbf{s})] \geq f(\mathbf{o}),$$

for $\alpha = \max(1, \sqrt{\frac{k-1}{4}})$.