MATHEMATICAL ENGINEERING
TECHNICAL REPORTS

Bisubmodular Function Maximization and Extensions

Satoru IWATA, Shin-ichi TANIGAWA
and Yuichi YOSHIDA

METR 2013–16 September 2013

DEPARTMENT OF MATHEMATICAL INFORMATICS
GRADUATE SCHOOL OF INFORMATION SCIENCE AND TECHNOLOGY
THE UNIVERSITY OF TOKYO
BUNKYO-KU, TOKYO 113-8656, JAPAN

WWW page: http://www.keisu.t.u-tokyo.ac.jp/research/techrep/index.html

Bisubmodular Function Maximization and Extensions

Satoru Iwata [*] Shin-ichi Tanigawa [†] Yuichi Yoshida [‡]

September 2013

Abstract

This paper presents simple greedy approximation algorithms for maximizing bisubmodular functions, extending the double-greedy algorithms for submodular function maximization due to Buchbinder, Feldman, Naor, and Schwartz (2012). Our deterministic algorithm provides an approximate solution that achieves at least one third of the optimal value, whereas the output of our randomized algorithm achieves at least a half of the optimal value at expectation.

We also extend the approach to provide constant factor approximation algorithms for maximizing k-submodular functions and skew-bisubmodular functions, which are recently introduced as generalizations of bisubmodular functions.

1 Introduction

Let V be a finite nonempty set of cardinality n and 3^V denote the set of ordered pairs of disjoint subsets of V. Two binary operations $\sqcup$ and $\sqcap$ on 3^V are defined by

$$\begin{aligned}(X_1, Y_1) \sqcup (X_2, Y_2) &= ((X_1 \cup X_2) \setminus (Y_1 \cup Y_2), (Y_1 \cup Y_2) \setminus (X_1 \cup X_2)),\\ (X_1, Y_1) \sqcap (X_2, Y_2) &= (X_1 \cap X_2, Y_1 \cap Y_2).\end{aligned}$$

A function $f : 3^V \to \mathbb{R}$ is called *bisubmodular* if it satisfies

$$f(X_1, Y_1) + f(X_2, Y_2) \geq f((X_1, Y_1) \sqcup (X_2, Y_2)) + f((X_1, Y_1) \sqcap (X_2, Y_2))$$

for every pair of (X_1, Y_1) and (X_2, Y_2) in 3^V. Examples of bisubmodular functions include the rank functions of delta-matroids and the cut capacity functions of bidirected networks. Combinatorial algorithms for minimizing bisubmodular functions have been developed in [4, 9].

A recent paper of Singh, Guillory, and Bilme [10] addressed the problem of maximizing general bisubmodular functions in the contexts of sensor placement and feature selection. They showed that a certain class of bisubmodular functions admit a constant-factor approximation algorithm for maximization. In the present paper, we provide simple greedy approximation algorithms for bisubmodular function maximization, extending the double-greedy algorithm for submodular function maximization due to Buchbinder, Feldman, Naor, and Schwartz [1].

[*]Department of Mathematical Informatics, Graduate School of Information Science and Technology, University of Tokyo, Tokyo 113-8656, Japan (iwata@mist.i.u-tokyo.ac.jp). Supported by JSPS Grant-in-Aid for Scientific Research (B) No. 23300002.

[†]Research Institute for Mathematical Sciences, Kyoto University, Kyoto 606-8502 Japan (tanigawa@kurims.kyoto-u.ac.jp). Supported by JSPS Grant-in-Aid for Scientific Research (B).

[‡]National Institute of Informatics, and Preferred Infrastructure, Inc. (yyoshida@nii.ac.jp). Supported by JSPS Grant-in-Aid for Research Activity Start-up (24800082), MEXT Grant-in-Aid for Scientific Research on Innovative Areas (24106001), and JST, ERATO, Kawarabayashi Large Graph Project.

A bisubmodular function generalizes a submodular function as follows. Let 2^V denote the family of all the subsets of V. A function $g : 2^V \to \mathbb{R}$ is called *submodular* if it satisfies

$$g(Z_1) + g(Z_2) \geq g(Z_1 \cup Z_2) + g(Z_1 \cap Z_2)$$

for every pair of Z_1 and Z_2 in 2^V. For a submodular function g, a function $f : 3^V \to \mathbb{R}$ defined by

$$f(X, Y) = g(X) + g(V \setminus Y) - g(V),$$

is bisubmodular.

Submodular function maximization contains NP-hard optimization problems such as max cut and set cover. It is known to be intractable in the standard oracle model. However, approximation algorithms have been studied extensively. In particular, Feige, Mirrokni, and Vondrák [3] have developed constant factor approximation algorithms for the unconstrained maximization of nonnegative submodular functions and shown that no approximation algorithm can achieve the ratio better than 1/2. Buchbinder, Feldman, Naor, and Schwartz [1] provide much simpler algorithms that substantially improve the approximation factor. Their deterministic and randomized versions achieve the factors of 1/3 and 1/2, respectively.

One particularly novel idea in the algorithms of Buchbinder, Feldman, Naor, and Schwartz [1] is to keep a pair of nested subsets instead of a single subset. In view of bisubmodular function maximization, however, one can claim that their algorithm simply keeps a single member of 3^V. Extending their algorithm to the framework of bisubmodular functions clarifies the background for their algorithm to work so effectively.

As the pair of nested subsets coincide at the termination of the double-greedy algorithms, our algorithms always return a partition, i.e., a pair of disjoint subsets whose union is V. The following lemma justifies this restriction by showing that a bisubmodular function is maximized by a partition.

Lemma 1.1. *For any bisubmodular function $f : 3^V \to \mathbb{R}_+$, there exists a partition that attains the maximum value of f.*

Proof. Suppose that a disjoint pair $(S, T) \in 3^V$ attains the maximum value of f. Then it follows from the bisubmodularity of f that

$$f(S, V \setminus S) + f(V \setminus T, T) \geq 2f(S, T), \tag{1}$$

which implies that $f(S, V \setminus S) = f(V \setminus T, T) = f(S, T)$. Thus the maximum value of f is attained by a partition of V. □

Maximizing submodular functions is extensively investigated for its application to machine learning such as viral marketing, information cascading, and sensor placement. For example, given a set V of candidate positions, we define a function $f : 2^V \to \mathbb{R}$ so that $f(X)$ for $X \subseteq V$ represents the utility obtained by placing information sources or sensors at the positions in X. Then, $f(X)$ often becomes a submodular function. When we have two kinds of resources in these problems, by defining $f : 3^V \to \mathbb{R}$ so that $f(X, Y)$ for $(X, Y) \in 3^V$ represents the utility obtained by placing the first kind of resources at X and the second kind of resources at Y, we often obtain a bisubmodular function [10]. By extending this idea to the setting with k types of sensors, it is natural to think of a generalized submodular functions with k arguments. In fact, Huber and Kolmogorov [6] discussed k-submodularity as a generalization of bisubmodularity.

Let $(k+1)^V := \{(X_1, \ldots, X_k) \mid X_i \subseteq V \, (i = 1, \ldots, k), X_i \cap X_j = \emptyset \, (i \neq j)\}$. A function $f : (k+1)^V \to \mathbb{R}$ is called k-submodular if

$$\begin{aligned} f(X_1, \ldots, X_k) + f(Y_1, \ldots, Y_k) \quad \geq \quad & f(X_1 \cap Y_1, \ldots, X_k \cap Y_k) \\ & + f(X_1 \cup Y_1 \setminus (\bigcup_{i \neq 1} X_i \cup Y_i), \ldots, X_k \cup Y_k \setminus (\bigcup_{i \neq k} X_i \cup Y_i)). \end{aligned}$$

For the k-submodular function maximization problem with $k \geq 2$, we show that the randomized greedy algorithm finds a solution whose value is at least $\frac{1}{k}$ of the optimal value.

As another extension of the bisubmodularity, Huber, Krokhin, and Powell [7] have introduced the concept of skew-bisubmodularity. For $\alpha \in [0,1]$, a function $f : 3^V \to \mathbb{R}$ is called *α-bisubmodular* if, for any (X_1, Y_1) and (X_2, Y_2) in 3^V,

$$\begin{aligned} f(X_1, Y_2) + f(X_2, Y_2) \geq & f((X_1, Y_1) \sqcap (X_2, Y_2)) \\ & + \alpha f((X_1, Y_1) \sqcup (X_2, Y_2)) + (1-\alpha) f((X_1, Y_1) \dot{\sqcup} (X_2, Y_2)), \end{aligned} \tag{2}$$

where $(X_1, Y_1) \dot{\sqcup} (X_2, Y_2) = (X_1 \cup X_2, (Y_1 \cup Y_2) \setminus (X_1 \cup X_2))$. A function $f : 3^V \to \mathbb{R}$ is called *skew-bisubmodular* if it is α-bisubmodular for some $\alpha \in [0,1]$.

While it was left open in [7] to decide whether α-bisubmodular functions can be minimized in polynomial time in the value oracle model, Huber and Krokhin [8] have announced that the minimization problem is indeed tractable by means of the ellipsoid method (see also [5]). Fujishige, Tanigawa, and Yoshida [5] have also provided a natural interpretation of the skew-bisubmodularity in the context of discrete convex analysis.

We show that a randomized greedy algorithm provides an approximate solution within the factor of $\frac{2\sqrt{\alpha}}{(1+\sqrt{\alpha})^2}$ for maximizing an α-bisubmodular function. Combining this with another simple algorithm, we obtain an approximate algorithm whose approximate ratio is at least $\frac{8}{25}$ for any $\alpha \in [0,1]$.

The rest of this paper is organized as follows. In Section 2, we present a greedy algorithm for bisubmodular function maximization and show that it is a 1/3-approximation algorithm. In Section 3, we show that a randomized version of this greedy algorithm achieves a performance guarantee within a factor of 1/2, which is optimal. Section 4 is then devoted to k-submodular function maximization. In Section 5, we analyze a randomized greedy algorithm for maximizing α-bisubmodular functions, and then we present an improvement that leads to a constant-factor approximation algorithm. The inapproximability of this problem is discussed in Section 6.

2 A greedy algorithm

In this section, we present a simple greedy algorithm for finding a partition that maximizes a nonnegative bisubmodular function approximately and show that the output achieves at least one third of the optimal value.

The algorithm is described in Algorithm 1. The algorithm keeps $(A, B) \in 3^V$. Initially, we set $A := \emptyset$ and $B := \emptyset$. The algorithm repeats the following iteration until $A \cup B = V$. Each iteration starts with selecting an element $u \in V \setminus (A \cup B)$. If $f(A \cup \{u\}, B) \geq f(A, B \cup \{u\})$, then add u to A. Otherwise, add u to B. Since the bisubmodularity of f implies

$$f(A \cup \{u\}, B) + f(A, B \cup \{u\}) \geq 2f(A, B), \tag{3}$$

this iteration never reduces the value of $f(A, B)$.

We now analyze this greedy algorithm to show that its output achieves at least one third of the maximum value of f. Let (S, T) be an optimal solution. By Lemma 1.1 we may assume $S \cup T = V$. We put $X := A \cup S \setminus B$ and $Y := B \cup T \setminus A$. As the algorithm updates (A, B), we suppose (X, Y) changes accordingly. At the termination of the algorithm, it follows from $A \cup B = V$ that $(X, Y) = (A, B)$.

We analyze how a potential Ψ, defined by $\Psi := f(X, Y) + 2f(A, B)$, changes in the process of the algorithm.

Lemma 2.1. *The value of Ψ never decreases in the process of the algorithm.*

Algorithm 1

Input: A non-negative bisubmodular function $f : 3^V \to \mathbb{R}_+$ on a finite set V.

```
1: Initialize (A, B) = (∅, ∅).
2: while ∃u ∈ V \ (A ∪ B) do
3:    if f(A ∪ {u}, B) ≥ f(A, B ∪ {u}) then
4:       A := A ∪ {u};
5:    else
6:       B := B ∪ {u};
7: return (A, B).
```

Proof. We first suppose that $u \in S$. If $f(A \cup \{u\}, B) \geq f(A, B \cup \{u\})$, then (A, B) will be replaced by $(A \cup \{u\}, B)$ and (X, Y) does not change, which imply that Ψ does not decrease in this iteration. Otherwise, (A, B) and (X, Y) will be replaced by $(A, B \cup \{u\})$ and $(X \setminus \{u\}, Y \cup \{u\})$, respectively. By the bisubmodularity of f, we have

$$\begin{aligned} f(X \setminus \{u\}, Y \cup \{u\}) + f(A \cup \{u\}, B) &\geq f(A, B) + f(X \setminus \{u\}, Y), \\ f(X \setminus \{u\}, Y) + f(A \cup \{u\}, B) &\geq f(A, B) + f(X, Y). \end{aligned}$$

Since $f(A, B \cup \{u\}) > f(A \cup \{u\}, B)$, these inequalities imply

$$f(X \setminus \{u\}, Y \cup \{u\}) + 2f(A, B \cup \{u\}) > f(X, Y) + 2f(A, B),$$

which means that Ψ does not decrease in this iteration.

Since $S \cup T = V$, the remaining case is when $u \in T$. Notice however that the roles of S and T, A and B, and X and Y are symmetric in the algorithm. Therefore, the same argument implies that Ψ does not decrease in this case as well. □

Theorem 2.2. *The value of $f(A, B)$ at the termination is at least $\frac{1}{3} f(S, T)$.*

Proof. We track the value of Ψ. Initially, we have $(A, B) = (\emptyset, \emptyset)$ and $(X, Y) = (S, T)$, which imply $\Psi \geq f(S, T)$. By Lemma 2.1, we have $\Psi \geq f(S, T)$ throughout the algorithm. At the termination, since $A \cup B = V$, we have $(X, Y) = (A, B)$, and hence $3f(A, B) = \Psi$. Thus we obtain $3f(A, B) \geq f(S, T)$. □

3 A randomized greedy algorithm

In this section, we present a randomized version of the greedy algorithm. The algorithm is shown in Algorithm 2, where we start with $(A, B) = (\emptyset, \emptyset)$ and iteratively add a new element u to A or B until we get a partition (A, B) of V.

We now analyze this randomized algorithm to show that its output achieves at least a half of the maximum value of f at expectation. Let (S, T) be an optimal solution. By Lemma 1.1, we may assume $S \cup T = V$. We put $X := A \cup S \setminus B$ and $Y := B \cup T \setminus A$. As the algorithm updates (A, B), we suppose (X, Y) changes accordingly.

The following lemma is a crucial observation for the analysis, which evaluates the expected change $\Delta\Phi$ of a potential Φ, defined by $\Phi = f(A, B) + f(X, Y)$. Note that Φ is a random variable.

Lemma 3.1. *The expectation $\mathbb{E}[\Phi]$ never decreases in the process of the algorithm.*

Algorithm 2

Input: A non-negative bisubmodular function $f : 3^V \to \mathbb{R}_+$ on a finite set V.

Initialize $(A, B) = (\emptyset, \emptyset)$.
while $\exists u \in V \setminus (A \cup B)$ **do**
  Let $a := f(A \cup \{u\}, B) - f(A, B)$ and $b := f(A, B \cup \{u\}) - f(A, B)$.
  if $a \leq 0$ **then** insert u to B.
  else if $b \leq 0$ **then** insert u to A.
  else add u to A with probability $\frac{a}{a+b}$ and to B with probability $\frac{b}{a+b}$.
return (A, B).

Proof. Let $u \in V \setminus (A \cup B)$ be the element chosen in an iteration. We now analyze the change $\Delta\Phi$ of Φ in this iteration.

We first suppose that $u \in T$. In this case, $u \notin X$ and $u \in Y$. By the bisubmodularity, we have

$$f(X \cup \{u\}, Y \setminus \{u\}) + f(A, B \cup \{u\}) \geq f(A, B) + f(X, Y \setminus \{u\}),$$
$$f(X, Y \setminus \{u\}) + f(A, B \cup \{u\}) \geq f(A, B) + f(X, Y),$$

which imply

$$f(X \cup \{u\}, Y \setminus \{u\}) - f(X, Y) \geq 2(f(A, B) - f(A, B \cup \{u\})) = -2b. \tag{4}$$

Since $a + b \geq 0$ by (3), we have the following three cases.

- If $a \geq 0$ and $b < 0$, the algorithm changes (A, B) to $(A \cup \{u\}, B)$. Hence, by (4), $\mathbb{E}[\Delta\Phi] = f(A \cup \{u\}, B) - f(A, B) + f(X \cup \{u\}, Y \setminus \{u\}) - f(X, Y) \geq a - 2b \geq 0$.
- If $a < 0$ and $b \geq 0$, then the algorithm changes (A, B) to $(A, B \cup \{u\})$. Hence, $\mathbb{E}[\Delta\Phi] = f(A, B \cup \{u\}) - f(A, B) = b \geq 0$.
- If $a > 0$ and $b > 0$, the algorithm changes (A, B) to $(A \cup \{u\}, B)$ with probability $\frac{a}{a+b}$ and to $(A, B \cup \{u\})$ with probability $\frac{b}{a+b}$. Therefore, by (4), $\mathbb{E}[\Delta\Phi] = \frac{a}{a+b}(f(A \cup \{u\}, B) - f(A, B) + f(X \cup \{u\}, Y \setminus \{u\}) - f(X, Y)) + \frac{b}{a+b}(f(A, B \cup \{u\}) - f(A, B)) \geq \frac{a}{a+b}(a - 2b) + \frac{b^2}{a+b} = \frac{(a-b)^2}{a+b} \geq 0$.

Thus in each of these cases, $\mathbb{E}[\Phi]$ does not decrease.

We next consider the case that $u \in S$. A symmetric argument, however, works to prove that $\mathbb{E}[\Delta\Phi] \geq 0$. □

Using Lemma 2.1, we obtain the following theorem, which establishes that the greedy algorithm has a guaranteed approximation ratio of 1/2.

Theorem 3.2. *The expectation $\mathbb{E}[f(A, B)]$ at the termination is at least $\frac{1}{2} f(S, T)$.*

Proof. We track the value of $\Phi = f(A, B) + f(X, Y)$. Initially, we have $(A, B) = (\emptyset, \emptyset)$ and $(X, Y) = (S, T)$, which imply $\Phi \geq f(S, T)$. By Lemma 2.1, we have $\mathbb{E}[\Phi] \geq f(S, T)$ throughout the execution of the algorithm. At the termination, since $A \cup B = V$, we have $(X, Y) = (A, B)$, and hence $2\mathbb{E}[f(A, B)] = \mathbb{E}[\Phi] \geq f(S, T)$. □

Algorithm 3

Input: A non-negative k-submodular function $f : (k+1)^V \to \mathbb{R}_+$ on a finite set V.

1: Initialize $(A_1, \dots, A_k) = (\emptyset, \dots, \emptyset)$.
2: **while** $\exists u \in V \setminus \bigcup_{i=1}^k A_i$ **do**
3: Let $a_i := f(A_1, \dots, A_i \cup \{u\}, \dots, A_k) - f(A_1, \dots, A_i, \dots, A_k)$ for $1 \le i \le k$.
4: Let $I_+ := \{i \in [k] \mid a_i \ge 0\}$.
5: Let $a := \sum_{i \in I_+} a_i$.
6: **if** $a = 0$ **then** insert u to A_i with probability $\frac{1}{k}$.
7: **else** add u to A_i with probability $\frac{a_i}{a}$.
8: **return** $(A_1, \dots, A_k)$.

4 Maximizing k-submodular functions

The interpretation of the algorithm by Buchbinder et al. [1] as a greedy algorithm in the framework of bisubmodular functions leads to further extensions of the technique. In this section we show that the randomized greedy algorithm can be extended to k-submodular functions without any difficulty.

The algorithm is given in Algorithm 3, which is a straightforward adaptation of the randomized greedy algorithm for bisubmodular maximization. Namely, instead of making a partition into two subsets, the algorithm constructs a partition of V into k subsets. The following is the counterpart of Lemma 1.1.

Lemma 4.1. *For any k-submodular function $f : (k+1)^V \to \mathbb{R}_+$, there exists a partition of V into k subsets that attains the maximum value of f.*

Proof. Suppose that $(S_1, \dots, S_k) \in (k+1)^V$ attains the maximum value of f. By the k-submodularity of f, we have

$$f(V \setminus \bigcup_{i \ne 1} S_i, S_2, \dots, S_k) + f(S_1, V \setminus \bigcup_{i \ne 2} S_i, S_3, \dots, S_k) \ge 2f(S_1, S_2, \dots, S_k).$$

Thus the maximum is attained by a partition of V. □

We also note that, by the k-submodularity of f, we have

$$f(\dots, A_i \cup \{u\}, \dots, A_j, \dots) + f(\dots, A_i, \dots, A_j \cup \{u\}, \dots) \ge f(\dots, A_i, \dots, A_j, \dots),$$

and hence $a_i + a_j \ge 0$ holds at line 3. Therefore, if $a = 0$ at line 6, we have $a_i = 0$ for all i.

The following theorem provides a performance guarantee of this algorithm.

Theorem 4.2. *The randomized greedy algorithm for maximizing k-submodular functions provides an approximate solution within a factor of $\frac{1}{k}$.*

For the proof, let $(S_1, \dots, S_k)$ be an optimal solution with $V = \bigcup_{i=1}^k S_i$ and put $X_i = (A_i \cup S_i) \setminus \bigcup_{j \ne i} A_j$ for $1 \le i \le k$. As the algorithm updates $(A_1, \dots, A_k)$, we suppose $(X_1, \dots, X_k)$ changes accordingly. We now analyze a potential Φ_k, defined by $\Phi_k = f(A_1, \dots, A_k) + \frac{1}{k-1} f(X_1, \dots, X_k)$.

At the beginning of the algorithm, $\Phi_k = f(\emptyset, \dots, \emptyset) + \frac{1}{k-1} f(S_1, \dots, S_k) \ge \frac{1}{k-1} f(S_1, \dots, S_k)$. At the end of the algorithm, since $(A_1, \dots, A_k)$ is a partition of V, we have $(X_1, \dots, X_k) = (A_1, \dots, A_k)$, which implies $\mathbb{E}[\Phi_k] = \frac{k}{k-1} \mathbb{E}[f(A_1, \dots, A_k)]$. Therefore, if the expected amount of change $\Delta\Phi_k$ of Φ_k is non-negative at each iteration, we obtain $\mathbb{E}[f(A_1, \dots, A_k)] \ge \frac{1}{k} f(S_1, \dots, S_k)$ at the end of the algorithm. We now show that this is the case.

Lemma 4.3. *The expectation $\mathbb{E}[\Phi_k]$ never decreases in the process of the algorithm.*

Proof. Let $u \in V \setminus \bigcup_{i=1}^k A_i$ be the element chosen in an iteration. We now analyze the change $\Delta\Phi_k$ of Φ_k in this iteration.

Since $\{S_1, \ldots, S_k\}$ is a partition of V, without loss of generality we assume $u \in S_1$. As in the case of bisubmodular functions, it follows from the k-submodularity of f that

$$\begin{aligned} & f(X_1 \setminus \{u\}, X_2, \ldots, X_{i-1}, X_i \cup \{u\}, X_{i+1}, \ldots, X_k) - f(X_1, \ldots, X_k) \\ & \geq 2(f(A_1, \ldots, A_k) - f(A_1 \cup \{u\}, A_2, \ldots, A_k)) \\ & = -2a_1 \end{aligned} \tag{5}$$

for $2 \leq i \leq k$.

If $a > 0$, then $(A_1, \ldots, A_k)$ and $(X_1, \ldots, X_k)$ are changed to $(A_1, \ldots, A_i \cup \{u\}, \ldots, A_k)$ and $(X_1 \setminus \{u\}, X_2, \ldots, X_{i-1}, X_i \cup \{u\}, X_{i+1}, \ldots, X_k)$, respectively, with probability $\frac{a_i}{a}$ for each $i \in I_+$. Therefore, we have

$$\begin{aligned} \mathbb{E}[\Delta\Phi_k] &\geq \sum_{i \in I_+ \setminus \{1\}} \frac{a_i}{a}\left(a_i - \frac{2}{k-1}a_1\right) + \frac{(\max\{a_1, 0\})^2}{a}. \\ &= \frac{1}{a}\left(\sum_{i \in I_+} a_i^2 - \frac{2}{k-1}\sum_{i \in I_+ \setminus \{1\}} a_i a_1\right). \end{aligned} \tag{6}$$

If $\{1\} \notin I_+$, $a_i a_1 \leq 0$ for all $i \in I_+$, and hence we have $\mathbb{E}[\Delta\Phi_k] \geq 0$ from (6). Otherwise, we have

$$\mathbb{E}[\Delta\Phi_k] \geq \frac{1}{a}\left(\frac{k-2}{k-1}\sum_{i \in I_+ \setminus \{1\}} a_i^2 + \frac{1}{k-1}\sum_{i \in I_+ + \setminus \{1\}} (a_i - a_1)^2\right) \geq 0.$$

from (6).

If $a = 0$, then $(A_1, \ldots, A_k)$ and $(X_1, \ldots, X_k)$ are changed to $(A_1, \ldots, A_i \cup \{u\}, \ldots, A_k)$ and $(X_1 \setminus \{u\}, X_2, \ldots, X_{i-1}, X_i \cup \{u\}, X_{i+1}, \ldots, X_k)$, respectively, with probability $\frac{1}{k}$ for each $1 \leq i \leq k$. As we mentioned in the remark after the algorithm, $a_i = 0$ holds for all $1 \leq i \leq k$ when $a = 0$. Therefore by (5), $f(X_1 \setminus \{u\}, X_2, \ldots, X_{i-1}, X_i \cup \{u\}, X_{i+1}, \ldots, X_k) - f(X_1, \ldots, X_k) \geq 0$ for each $2 \leq i \leq k$. This in turn implies $\mathbb{E}[\Delta\Phi_k] \geq 0$. □

5 Approximability of maximizing α-bisubmodular functions

In this section, we discuss the problem of maximizing an α-bisubmodular function. An adaptation of the greedy algorithm is shown to achieve the approximation ratio of $\frac{2\sqrt{\alpha}}{(1+\sqrt{\alpha})^2}$ for $\alpha \in [0, 1]$. This ratio converges to zero as α goes to zero. In order to improve the performance for small α, we give another simple approximation algorithm that achieves the approximation ratio of $\frac{1}{3+2\alpha}$ in Section 5.2. By taking the maximum of the outputs of these two algorithms, we obtain the approximation ratio of $\frac{8}{25}$ for any $\alpha \in [0, 1]$ (the minimum of the two ratio is achieved when $\alpha = \frac{1}{16}$).

Concerning the maximum of an α-bisubmodular function, we have the following counterpart of Lemma 1.1.

Lemma 5.1. *For any α-bisubmodular function $f : 3^V \to \mathbb{R}_+$ with $\alpha \in [0, 1]$, there exists a partition of V that attains the maximum value of f.*

Proof. Suppose that $(S,T) \in 3^V$ attains the maximum value of f. By the α-bisubmodularity of f, we have

$$\alpha f(S, V \setminus S) + f(V \setminus T, T) \geq (1+\alpha) f(S,T),$$

which implies that $f(S, V\setminus S) = f(V \setminus T, T) = f(S,T)$. Thus the maximum value of f is attained by a partition of V. □

5.1 A randomized greedy algorithm

We now extend the randomized greedy algorithm for the bisubmodular function. Intuitively, α-bisubmodularity is a variant of bisubmodularity directed toward the first argument by parameter α. Following this intuition, we shall adjust the choice probability. The algorithm we discuss here is the same as Algorithm 2 except that it adds u to A with probability $\frac{\alpha a}{\alpha a+b}$ and to B with probability $\frac{b}{\alpha a+b}$ (where a and b are as defined in line 3). Note that, by the α-bisubmodularity of f, we have

$$\alpha f(A \cup \{u\}, B) + f(A, B \cup \{u\}) \geq (1+\alpha) f(A,B) \tag{7}$$

for any $(A,B) \in 3^V$ and $u \in V \setminus (A \cup B)$, which implies $\alpha a + b \geq 0$.

The following theorem provides a performance analysis of this algorithm.

Theorem 5.2. *For any $\alpha \in [0,1]$, the randomized greedy algorithm for maximizing α-bisubmodular functions provides an approximate solution within a factor of $\frac{2\sqrt{\alpha}}{(1+\sqrt{\alpha})^2}$.*

The proof is done in the same manner as before. Let (S,T) be an optimal solution with $S \cup T = V$, put $X := A \cup S \setminus B$ and $Y := B \cup T \setminus A$. As the algorithm updates (A,B), we suppose (X,Y) changes accordingly. To prove this theorem, we consider a deformed potential $\Phi_\alpha = f(A,B) + \frac{2\sqrt{\alpha}}{1+\alpha} f(X,Y)$. At the beginning, since $(A,B) = (\emptyset, \emptyset)$ and $(X,Y) = (S,T)$, we have $\Phi_\alpha = f(\emptyset,\emptyset) + \frac{2\sqrt{\alpha}}{1+\alpha} f(S,T) \geq \frac{2\sqrt{\alpha}}{1+\alpha} f(S,T)$. At the end of the algorithm, since $(A,B) = (X,Y)$, we have $\mathbb{E}[\Phi_\alpha] = \frac{(1+\sqrt{\alpha})^2}{1+\alpha}\mathbb{E}[f(A,B)]$. If the expected amount of change $\Delta\Phi_\alpha$ of Φ_α is non-negative at each iteration, we obtain $\mathbb{E}[f(A,B)] \geq \frac{2\sqrt{\alpha}}{(1+\sqrt{\alpha})^2} f(S,T)$ at the end of the algorithm.

Lemma 5.3. *The expectation $\mathbb{E}[\Phi_\alpha]$ never decreases in the process of the algorithm.*

Proof. Let $u \in V \setminus (A \cup B)$ be the element chosen in an iteration. We now analyze the change $\Delta\Phi_\alpha$ of Φ_α in this iteration.

Suppose that $u \in T$. In this case, $u \notin X$ and $u \in Y$. By α-bisubmodularity, we have

$$\alpha(f(X \cup \{u\}, Y \setminus \{u\}) - f(X,Y)) \geq (1+\alpha)(f(A,B) - f(A, B \cup \{u\})) = -(1+\alpha) b \tag{8}$$

by extending (4). We have the following three cases.

- If $a \geq 0$ and $b < 0$, the algorithm changes (A,B) to $(A \cup \{u\}, B)$. If $\alpha = 0$, we have $b \geq 0$ by $\alpha a + b \geq 0$. Thus, $\alpha > 0$. Then $\mathbb{E}[\Delta\Phi_\alpha] = f(A \cup \{u\}, B) - f(A,B) + \frac{2\sqrt{\alpha}}{1+\alpha}(f(X \cup \{u\}, Y \setminus \{u\}) - f(X,Y)) \geq a - \frac{2\sqrt{\alpha}}{1+\alpha}\frac{(1+\alpha)b}{\alpha} = a - \frac{2b}{\sqrt{\alpha}} \geq 0$.
- If $a < 0$ and $b \geq 0$, then the algorithm changes (A,B) to $(A, B \cup \{u\})$. Hence, $\mathbb{E}[\Delta\Phi_\alpha] = f(A, B \cup \{u\}) - f(A,B) = b \geq 0$.
- If $a > 0$ and $b > 0$, the algorithm changes (A,B) to $(A \cup \{u\}, B)$ with probability $\frac{\alpha a}{\alpha a+b}$ and to $(A, B \cup \{u\})$ with probability $\frac{b}{\alpha a+b}$. Therefore $\mathbb{E}[\Delta\Phi_\alpha] = \frac{\alpha a}{\alpha a+b}(f(A \cup \{u\}, B) - f(A,B) + \frac{2\sqrt{\alpha}}{1+\alpha}(f(X \cup \{u\}, Y \setminus \{u\}) - f(X,Y))) + \frac{b}{\alpha a+b}(f(A, B \cup \{u\}) - f(A,B)) \geq \frac{\alpha a}{\alpha a+b}(a - \frac{2\sqrt{\alpha}}{1+\alpha} \cdot \frac{1+\alpha}{\alpha} b) + \frac{b^2}{\alpha a+b} = \frac{\alpha a^2 - 2\sqrt{\alpha} ab + b^2}{\alpha a+b} = \frac{(\sqrt{\alpha}a - b)^2}{\alpha a+b} \geq 0$.

Now we turn to the case that $x \in S$. In this case, $x \in X$ and $x \notin Y$. From α-bisubmodularity, we have

$$f(X \setminus \{u\}, Y \cup \{u\}) - f(X, Y) \geq (1+\alpha)(f(A, B) - f(A \cup \{u\}, B)) = -(1+\alpha)a. \qquad (9)$$

We consider the following three cases.

- If $a \geq 0$ and $b < 0$, then $\mathbb{E}[\Delta\Phi_\alpha] = f(A + x, B) - f(A, B) = a \geq 0$.
- If $a < 0$ and $b \geq 0$, then $\mathbb{E}[\Delta\Phi_\alpha] = f(A, B \cup \{u\}) - f(A, B) + \frac{2\sqrt{\alpha}}{1+\alpha}(f(X \setminus \{u\}, Y \cup \{u\}) - f(X, Y)) \geq b - \frac{2\sqrt{\alpha}}{1+\alpha}(1+\alpha)a = b - 2\sqrt{\alpha}(1+\alpha)a \geq 0$.
- If $a > 0$ and $b > 0$, then $\mathbb{E}[\Delta\Phi_\alpha] = \frac{b}{\alpha a + b}(f(A, B \cup \{u\}) - f(A, B) + \frac{2\sqrt{\alpha}}{1+\alpha}(f(X \setminus \{u\}, Y \cup \{u\}) - f(X, Y))) + \frac{\alpha a}{\alpha a + b}(f(A \cup \{u\}, B) - f(A, B)) \geq \frac{b}{\alpha a + b}(b - \frac{2\sqrt{\alpha}}{1+\alpha}(1+\alpha)a) + \frac{\alpha a^2}{\alpha a + b} = \frac{b^2 - 2\sqrt{\alpha}ab + \alpha a^2}{\alpha a + b} = \frac{(\sqrt{\alpha}a - b)^2}{\alpha a + b} \geq 0$.

This completes the proof. □

5.2 The second algorithm

In this section, we describe another algorithm for maximizing α-bisubmodular functions, which achieves a better approximation ratio than the randomized greedy algorithm for small α.

For an α-bisubmodular function $f : 3^V \to \mathbb{R}_+$ with $\alpha \in [0, 1]$, we define $f' : 2^V \to \mathbb{R}_+$ by $f'(X) = f(X, \emptyset)$ for $X \in 2^V$. Since $f'(X)$ is a non-negative submodular function, we can apply the randomized double greedy algorithm of [1] to obtain a $\frac{1}{2}$-approximate solution Z to the maximization of $f'(X)$. Our second algorithm for α-bisubmodular function maximization is rather simple: Take the better of $(\emptyset, V)$ and $(Z, \emptyset)$.

Theorem 5.4. *The second algorithm for α-bisubmodular function maximization problem provides a $\frac{1}{3+2\alpha}$-approximate solution for any $\alpha \in [0, 1]$.*

Proof. Let (S, T) be an optimal solution. By the α-bisubmodularity, we have

$$\begin{aligned} f(S, \emptyset) + f(\emptyset, V) &\geq f(\emptyset, \emptyset) + \alpha f(\emptyset, T) + (1-\alpha) f(S, T), \\ f(S, \emptyset) + f(\emptyset, T) &\geq f(\emptyset, \emptyset) + f(S, T), \end{aligned}$$

which imply

$$(1+\alpha) f(S, \emptyset) + f(\emptyset, V) \geq (1+\alpha) f(\emptyset, \emptyset) + f(S, T) \geq f(S, T).$$

Let (A, B) be the output by the algorithm. Then,

$$\begin{aligned} &\mathbb{E}[f(A, B)] \geq \max\left\{\frac{1}{2} f(S, \emptyset), f(\emptyset, V)\right\} \geq \frac{2+2\alpha}{3+2\alpha} \cdot \frac{1}{2} f(S, \emptyset) + \frac{1}{3+2\alpha} f(\emptyset, V) \\ =&\frac{1}{3+2\alpha}\left((1+\alpha) f(S, \emptyset) + f(\emptyset, V)\right) \geq \frac{1}{3+2\alpha} f(S, T). \end{aligned}$$ □

Combining this with the first algorithm, we obtain an approximate solution within a factor of $\max\left\{\frac{2\sqrt{\alpha}}{(1+\sqrt{\alpha})^2}, \frac{1}{3+2\alpha}\right\}$. The minimum of this ratio is $\frac{8}{25}$, which is achieved when $\alpha = \frac{1}{16}$.

6 Inapproximability of maximizing α-bisubmodular functions

In this section, we show that, for every $\alpha \in [0,1]$, obtaining approximation ratio better than 1/2 requires an exponential number of queries. There are existing inapproximability results for maximizing submodular functions, and our strategy is to reduce the submodular function maximization problem to the α-bisubmodular function maximization.

We first show a simple construction of an α-bisubmodular function from a submodular function.

Lemma 6.1. *Let $f : 2^V \to \mathbb{R}$ be a submodular function. For any $\alpha \in [0,1]$, define $g : 3^V \to \mathbb{R}$ by $g(A,B) = f(A) + \alpha f(V \setminus B)$. Then, g is an α-bisubmodular function.*

Proof. Huber et al. [7] showed that $h : 3^V \to \mathbb{R}$ is α-bisubmodular if and only if (i) for any $(A,B) \in 3^V$ with $A \cup B = V$, $h' : 2^V \to \mathbb{R}$ defined by $h'(X) = h(A \cap X, B \cap X)$ for $X \in 2^V$ is submodular, and (ii) for any $(A,B) \in 3^V$ and $u \in V \setminus (A \cup B)$, (7) holds.

To see (i), take any $(A,B) \in 3^V$ with $A \cup B = V$. Since the first term and the second term of g are submodular on 2^A and 2^B, respectively, g satisfies (i).

Next we check (ii). Take $(A,B) \in 3^V$ with $A \cup B \neq V$ and $u \in V \setminus (A \cup B)$. Then we have

$$
\begin{aligned}
&\alpha g(A \cup \{u\}, B) + g(A, B \cup \{u\}) - (1+\alpha) g(A,B) \\
=&\alpha\big(f(A \cup \{u\}) + \alpha f(V \setminus B)\big) + \big(f(A) + \alpha f(V \setminus (B \setminus \{u\}))\big) - (1+\alpha)\Big(f(A) + \alpha f(V \setminus B)\Big) \\
=&\alpha\Big(\big(f(A \cup \{u\}) - f(A)\big) - \big(f(V \setminus B) - f(V \setminus (B \setminus \{u\}))\big)\Big) \\
\geq& 0,
\end{aligned}
$$

where the inequality follows from the submodularity of f and $A \subseteq V \setminus (B \cup \{u\})$. Thus, g is α-bisubmodular. □

Theorem 6.2. *Let $\alpha \in [0,1]$. There is a polynomial-time reduction from the problem of maximizing non-negative submodular functions to the problem of maximizing non-negative α-bisubmodular functions preserving approximation ratio.*

Proof. Consider $g(A,B) = f(A) + \alpha f(V \setminus B)$. From Lemma 6.1, g is α-bisubmodular. Let A^* be a maximizer of f. Note that $g(A, V \setminus A) = f(A) + \alpha f(V \setminus (V \setminus A)) = (1+\alpha) f(A)$. From Lemma 5.1, there is a maximizer of g of the form $(A, V \setminus A)$. Also, $g(A^*, V \setminus A^*) = (1+\alpha) f(A^*) \geq (1+\alpha) f(A) = g(A, V \setminus A)$ for any A. Therefore, the pair $(A^*, V \setminus A^*)$ is a maximizer of g.

Suppose that there is a polynomial-time δ-approximation algorithm for maximizing non-negative α-bisubmodular functions. Let (X,Y) be the output obtained for g using the algorithm. We can assume that $Y = V \setminus X$. Then,

$$(1+\alpha) f(X) = g(X,Y) \geq \delta g(A^*, V \setminus A^*) = \delta(1+\alpha) f(A^*).$$

Thus, X is a δ-approximation to the maximum of f. □

Feige et al. [3] showed that, for any $\epsilon > 0$, any $(\frac{1}{2} + \epsilon)$-approximation algorithm for the submodular function maximization requires an exponential number of queries. Hence we have the following.

Corollary 6.3. *For every $\alpha \in [0,1]$ and $\epsilon > 0$, any $(\frac{1}{2} + \epsilon)$-approximation algorithm for the α-bisubmodular function maximization requires an exponential number of queries in the value oracle model.*

In the *explicit model*, a function $f : A^n \to \mathbb{Q}$ is expressed as the sum of functions of constant arity. It is shown in [2] that, even in the explicit model, if the submodular function maximization problem admits a $\frac{1}{2} + \epsilon$-approximation algorithm, then NP = RP. Thus, we have the following.

Corollary 6.4. *For every $\alpha \in [0, 1]$ and $\epsilon > 0$, if the α-bisubmodular function maximization problem admits a $\frac{1}{2} + \epsilon$-approximation algorithm, then* NP = RP.

7 Conclusion

We have shown that the double greedy algorithm of Buchbinder et al. [1] for submodular function maximization can be recognized as a greedy algorithm in the bisubmodular setting. This leads to a $\frac{1}{2}$-approximation algorithm for bisubmodular function maximization. The greedy algorithm is further extended to k-submodular functions and α-bisubmodular functions. For α-bisubmodular functions, the approximation ratio converges to zero as α goes to zero. With a simple trick, however, we have devised a constant factor approximation algorithm for α-bisubmodular function maximization. An obvious open problem is to improve this approximation ratio.

References

[1] N. Buchbinder, M. Feldman, J. Naor, and R. Schwartz: A tight linear time (1/2)-approximation for unconstrained submodular maximization. *Proceedings of the 53rd Annual IEEE Symposium on Foundations of Computer Science*, 2012, pp. 649–658.

[2] S. Dobzinski and J. Vondrák: From query complexity to computational complexity, *Proceedings of the 44th ACM Symposium on Theory of Computing*, 2012, pp. 1107–1116.

[3] U. Feige, V. S. Mirrokni, and J. Vondrák: Miximizing nonmonotone submodular functions, *SIAM J. Comput.*, 40 (2011), pp. 1133–1153.

[4] S. Fujishige and S. Iwata: Bisubmodular function minimization, *SIAM J. Discrete Math.*, 19 (2006), pp. 1065–1073.

[5] S. Fujishige, S. Tanigawa, and Y. Yoshida: Generalized skew bisubmodularity: A characterization and a min-max theorem. 2013, RIMS Preprints, RIMS-1781.

[6] A. Huber and V. Kolmogorov: Towards mininizing k-submodular Functions. *Proceedings of 2nd International Symposium on Combinatorial Optimization*, 2012, pp. 451–462.

[7] A. Huber, A. Krokhin, and R. Powell: Skew bisubmodularity and valued CSPs, *Proceedings of the 24th Annual ACM-SIAM Symposium on Discrete Algorithms*, 2013, pp. 1296–1305.

[8] A. Huber and A. Krokhin: A Lovász extension for skew bisubmodular functions. Unpublished manuscript, 2012, October 30. (See also the slides of Anna Huber: VCSPs on Three Elements. The Dagstuhl Seminar 12451 in November 2012 on "The Constraint Satisfaction Problem: Complexity and Approximability").

[9] S. T. McCormick and S. Fujishige: Strongly polynomial and fully combinatorial algorithms for bisubmodular function minimization, *Math. Programming*, 122 (2010), pp. 87–120.

[10] A. P. Singh, A. Guillory, and J. Bilmes: On bisubmodular maximization, *Proceedings of the 15th International Conference on Artificial Intelligence and Statistics*, 2012, pp. 1055–1063.