Divide-and-Conquer Algorithms for Partitioning Hypergraphs and Submodular Systems*

Kazumasa Okumoto†   Takuro Fukunaga‡   Hiroshi Nagamochi‡

Abstract

The submodular system k-partition problem is a problem of partitioning a given finite set V into k non-empty subsets $V_1, V_2, \ldots, V_k$ so that $\sum_{i=1}^{k} f(V_i)$ is minimized where f is a non-negative submodular function on V, and k is a fixed integer. This problem contains the hypergraph k-cut problem. In this paper, we design the first exact algorithm for $k = 3$ and approximation algorithms for $k \geq 4$. We also analyze the approximation factor for the hypergraph k-cut problem.

1 Introduction

A set function $f : 2^V \to \Re$ on a finite set V is called *submodular* if it satisfies $f(X) + f(Y) \geq f(X \cap Y) + f(X \cup Y)$ for every pair of sets $X, Y \subseteq V$ [1, 2]. Moreover, f is called *symmetric* if $f(X) = f(V \setminus X)$ for every $X \subseteq V$, and *non-negative* if $f(X) \geq 0$ for every $X \subseteq V$. A *submodular system* is defined as a pair (V, f) of a finite set V and a submodular function f on V.

A k-partition P_k of (V, f) is defined as a partition of V into k non-empty sets $V_1, V_2, \ldots, V_k \subseteq V$, i.e., $V_i \neq \emptyset$ for $i \in \{1, 2, \ldots, k\}$, $V_i \cap V_j = \emptyset$ for $1 \leq i < j \leq k$, and $\cup_{i=1}^{k} V_i = V$. We denote the partition by $[V_1, V_2, \ldots, V_k]$. The cost of P_k, denoted by $f(P_k)$ or $f(V_1, V_2, \ldots, V_k)$, is defined as $\sum_{i=1}^{k} f(V_i)$. In this paper, we consider the *submodular system k-partition problem*, that asks to find a minimum cost k-partition of given submodular system (V, f). Throughout this paper, it is supposed that f is non-negative and given as an oracle which returns $f(X)$ for $X \subseteq V$.

Submodularity often plays an essential role in studies on the connectivity of graphs. In fact, the submodular system k-partition problem generalizes the *graph k-cut problem* and *hypergraph k-cut problem*. For a graph or hypergraph $G = (V, E)$ with weight $w : E \to \Re$, a k-cut is defined as a set of edges whose removal divides G into at least k connected components. The k-cut problem asks to find a minimum weight k-cut of the given graph or hypergraph.

For a graph G and $X \subseteq V$, define $f(X)$ as $\sum_{e \in \delta(X)} w(e)$ where $\delta(X)$ denotes the set of edges joining vertices in X and $V \setminus X$. It is known that this set function is symmetric and submodular. For a k-partition $[V_1, V_2, \ldots, V_k]$ of V, $\sum_{i=1}^{k} f(V_i)$ is exactly twice the weight of a k-cut disconnecting $V_1, V_2, \ldots, V_k$ each other. Hence the k-cut problem of undirected graphs can be formulated as the k-partition problem with symmetric submodular systems.

For hypergraphs, the cut function defined above does not formulate the k-cut problem as the submodular k-partition problem. However, by defining submodular functions appropriately, it is possible to formulate the hypergraph k-cut problem as the submodular system k-partition problem (see Section 2). Note that the submodular functions used for this is not symmetric.

The graph k-cut problem is one of the fundamental problems in combinatorial optimization. It is closely related to the reliability of networks, and has many applications, for example, to the traveling salesperson problem, VLSI design, and evolutionary tree construction [4, 11]. Goldschmidt and Hochbaum [6] showed that the problem is NP-hard when k is not fixed. For fixed k, they presented a polynomial-time algorithm. Its running time is $O(n^{k^2}T(n,m))$ where $T(n,m)$ is time for computing max-flow in a graph consisting of n vertices and m edges. Note that $T(n,m)$ is known to be $O(mn\log(n^2/m))$ [5]. After their work, many polynomial-time algorithms for fixed k are obtained. An algorithm due to Kamidoi, Yoshida and Nagamochi [8] runs in $O(n^{4k/(1-1.71/\sqrt{k})-34}T(n,m))$. An algorithm due to Xiao [17] runs in $O(n^{4k-\log k})$. An algorithm due to Thorup [14] runs in $\tilde{O}(n^{2k})$. In addition, Karger and Stein [9] gave a random algorithm running in $O(n^{2(k-1)}\log^3 n)$.

For the hypergraph k-cut problem, Xiao [16] designed a polynomial-time algorithm to the case of $k = 3$. However, no polynomial-time algorithm is known when k is a fixed integer larger than 3. With regards to approximation algorithms, Zhao, Nagamochi and Ibaraki [18] gave an algorithm achieving the approximation factor $(1 - \frac{2}{k})\min\{k, d_{\max}\}$ with the result due to Xiao [16], where $d_{\max}$ denotes the maximum size of hyperedges. Moreover, with the reduction to the terminal k-vertex cut problem in bipartite graphs (see Section 2), we can apply the LP-rounding algorithm due to Garg, Vazirani and Yannakakis [3]. It achieves the approximation factor $(2 - \frac{2}{k})$.

Little is known about the submodular k-partition problem. Queyranne [13] gave a $(2 - \frac{2}{k})$-approximation algorithm for the problem with symmetric submodular functions. Zhao, Nagamochi and Ibaraki [18] presented a $(k - 1)$-approximation algorithm for the problem with non-negative submodular functions.

Besides the reduction from the hypergraph k-cut problem to the submodular system k-partition problem and the terminal k-vertex cut problem, the contribution of this paper is to design algorithms for the submodular system k-partition problem. For $k = 3$, we present a polynomial-time algorithm for $k = 3$. For $k \geq 4$, we give approximation algorithms. We also discuss the approximation factor of the algorithms for the hypergraph k-cut problem. Table 1 summarizes these approximation factors. Our algorithms perform well especially for small k. For hypergraph k-cut problem with $k \geq 5$, our algorithms present worse approximation factor than the algorithm due to Garg, Vazirani and Yannakakis [3] for the terminal k-vertex cut problem. However, our algorithms have an advantage since they do not need to solve the linear programming problems.

The key of our algorithms is *uncrossing operation*, which has been applied to many problems related to submodular functions. In this paper, we prove a theorem on uncrossing k-partitions and 2-partitions satisfying some conditions (Theorem 4). This theorem is originally proven by Xiao [16] for hypergraphs. We reveal in this paper that his result essentially relies only on submodularity of the cut function in hypergraphs.

The rest of this paper is organized as follows. Section 2 introduces notations. It also describes the reduction of the hypergraph k-cut problem to the terminal k-vertex cut problem in bipartite graphs, and shows that the submodular system k-partition problem contains the hypergraph k-cut problem. Section 3 proves several properties of k-partitions. Section 4 presents an exact algorithm for the submodular system 3-partition problem. Section 5 gives an approximation algorithm for

Table 1: Comparison of approximation factors by the previous best and our algorithms

	hypergraph k-cut problem		submodular system k-partition problem	
$k = 3$	1	(Xiao [16])	2	(Zhao et al. [18])
			1	(This paper)
$k = 4$	1.5	(Garg et al. [3])	3	(Zhao et al. [18])
	4/3	(This paper)	1.5	(This paper)
$k = 5$	1.6	(Garg et al. [3])	4	(Zhao et al. [18])
	5/3	(This paper)	2	(This paper)
$k \geq 6$	$2 - 2/k$	(Garg et al. [3])	$k - 1$	(Zhao et al. [18])
	$k/2 - 1$	(This paper)	$k + 1 - 2\sqrt{k-1}$	(This paper)

the submodular system 4-partition problem. It also analyses the approximation factor of the algorithm for the hypergraph 4-cut problem. Section 6 designs an approximation algorithm for the submodular system k-partition problem for $k \geq 5$ together with discussing the approximation factor for the hypergraph k-cut problem.

2 Preliminaries

2.1 Notations

In this paper, $\Re$ and $\Re_+$ stand for the sets of reals and non-negative reals, respectively. For a submodular system (V, f), we denote $|V|$ by n. Each $v \in V$ is called a *vertex*. The complement of a subset X of V is denoted by $\overline{X}$ (i.e., $\overline{X} = V \setminus X$).

Let $[V_1, V_2, \ldots, V_k]$ be a k-partition of a submodular system (V, f). Each V_i, $i \in \{1, 2, \ldots, k\}$ is called a *component* of the partition. A k-partition is called *h-size* if its all components contain at least h vertices. We note that a minimum h-size k-partition stands for a partition of minimum cost among all the h-size k-partitions, and a h-size minimum k-partition stands for a h-size partition of minimum cost among all the k-partitions.

For $U \subset V$, define a set function f^U on U such that $f^U(X) = f(X)$ for all $X \subseteq U$. Notice that if f is submodular, then f^U is submodular. We call a submodular system (U, f^U) *subsystem of (V, f) induced by U*.

For $U \subset V$, define V_U as a set obtained from V by replacing U with a new single vertex u (i.e., $V_U = (V \setminus U) \cup \{u\}$). Moreover, define f_U as a set function on V_U such that for every $X \subseteq V_U$, $f_U(X) = f(X)$ if $u \notin X$ and $f_U(X) = f((X \setminus \{u\}) \cup U)$ otherwise. It is easy to check that if f is submodular, then f_U is submodular. We call this operation *shrinking U into u*.

We say that a 2-partition $[X, \overline{X}]$ *crosses* a k-partition $[Y_1, Y_2, \ldots, Y_k]$ if both $X \setminus Y_i$ and $\overline{X} \setminus Y_i$ are non-empty for all $i \in \{1, 2, \ldots, k\}$. This means that, if $[X, \overline{X}]$ does not cross $[Y_1, Y_2, \ldots, Y_k]$, then there exists some $i \in \{1, 2, \ldots, k\}$ such that $X \subseteq Y_i$ or $\overline{X} \subseteq Y_i$. If $k = 2$, then $[X, \overline{X}]$ crosses $[Y_1, Y_2]$ whenever none of $X \cap Y_1$, $X \cap Y_2$, $\overline{X} \cap Y_1$ and $\overline{X} \cap Y_2$ is non-empty.

Suppose that a k-partition $[V_1, V_2, \ldots, V_k]$ and a 2-partition $[X, \overline{X}]$ of (V, f) satisfies $X \subseteq V_i$ with some $i \in \{1, 2, \ldots, k\}$. Then (V_X, f_X) has a k-partition $[V_1', V_2', \ldots, V_k']$ that satisfies $f_X(V_1', V_2', \ldots, V_k') = f(V_1, V_2, \ldots, V_k)$; I.e, each k-partition of (V, f) whose some component contains X corresponds to a k-partition of (V_X, f_X). The contrary also holds. Throughout this paper,

we do not distinguish such a k-partition of (V, f) with the corresponding one of (V_X, f_X).

For a finite set V, a hyperedge is defined as a subset of V. A hypergraph is defined as a pair (V, E) of V and a set E of hyperedges on V. Let $U_1, U_2, \ldots, U_\ell$ be disjoint subsets of V. Then $\delta(U_1, U_2, \ldots, U_\ell)$ denotes the set of edges in E that intersect at least two of $U_1, U_2, \ldots, U_\ell$. Moreover, $\delta_e(U_1, U_2, \ldots, U_\ell)$ denotes the set of edges in $\delta(U_1, U_2, \ldots, U_\ell)$ that do not intersect $V \setminus (U_1 \cup U_2 \cup \cdots \cup U_\ell)$. When a weight $w : E \to \Re_+$ is given, we define $w(U_1, U_2, \ldots, U_i) = \sum_{e \in \delta(U_1, U_2, \ldots, U_\ell)} w(e)$, and $w_e(U_1, U_2, \ldots, U_\ell) = \sum_{e \in \delta_e(U_1, U_2, \ldots, U_\ell)} w(e)$.

The cost of a k-cut for a hypergraph G can be represented by $w(V_1, V_2, \ldots, V_k)$ where $V_1, V_2, \ldots, V_k$ represent vertex sets of connected components after removing the k-cut. Hence the hypergraph k-cut problem can be regarded as a problem of computing a k-partition $[V_1, V_2, \ldots, V_k]$ of V minimizing $w(V_1, V_2, \ldots, V_k)$. Hence we denote a k-cut by a k-partition of V in this paper.

For $U \subseteq V$, G^U denotes the hypergraph obtained by removing all vertices in $V \setminus U$ and hyperedges intersecting $V \setminus U$ from G, and G_U denotes the hypergraph obtained from G by shrinking U into a single vertex.

Given two vertices s and t, a (s, t)-*partition* is a 2-partition such that s and t are in different components. In this paper, time complexities of algorithms are evaluated by the number of computing minimum (s, t)-partitions. A minimum (s, t)-partition of a submodular system (V, f) can be computed by minimizing (asymmetric) submodular function f' on $V \setminus \{t\}$ which is defined with an enough large constant M by $f'(X) = f^{V \setminus \{t\}}(X)$ if $s \in X$ and $f'(X) = f^{V \setminus \{t\}}(X) + M$ otherwise. See [7] for recent algorithmic development of the submodular function minimization. Gueyranne [12] showed that computing minimum (s, t)-partitions of symmetric submodular systems is as hard as minimizing (asymmetric) submodular functions. In hypergraphs (V, E), minimum (s, t)-partitions can be computed by finding a maximum flow in digraphs with $|V| + 2|E|$ vertices and $|E| + 2 \sum_{e \in E} |e|$ edges [10].

2.2   Reduction of the hypergraph k-cut problem

In the *terminal k-vertex cut problem*, an undirected graph (V, E) with vertex weight $w : V \to \Re$ and k terminals $t_1, t_2, \ldots, t_k \in V$ are given. A *vertex cut* is defined as a subset U of V whose removal disconnects given terminals each other. The objective of the problem is to find a vertex cut U minimizing $\sum_{v \in U} w(v)$. First, let us observe that the hypergraph k-cut problem can be reduced to the terminal k-vertex cut problem with bipartite graphs.

Let $G = (V, E)$ be a hypergraph with weight $w : E \to \Re_+$. Let V_E be the set of vertices corresponding to E while $v_e \in V_E$ denotes the vertex corresponding to $e \in E$. Define E' as a set of edges on $V \cup V_E$ such that E' contains an edge joining $u \in V$ and $v_e \in V_E$ if and only if the hyperedge e contains u in G. Then the bipartite graph $B = (V \cup V_E, E')$ can be defined from G. Define a vertex-weight $w' : V \cup V_E$ such that $w'(v) = +\infty$ for $v \in V$ and $w'(v_e) = w(e)$ for $v_e \in V_E$.

Choosing k terminals from V, solve the terminal k-vertex cut problem with B and w'. Obtained minimum vertex cut U contains no vertex in V by the definition of w'. Let $F \subseteq E$ be the set of hyperedges corresponding to vertices in U. Then F is a k-cut of hypergraph G because removing F from G disconnects the k vertices in V chosen as terminals in the terminal k-vertex cut problem each other. Moreover, $\sum_{e \in F} w(e) = \sum_{v \in U} w'(v)$ holds. Therefore, by solving the terminal k-vertex cut problem with choosing every set of k vertices in V as terminals, we can solve the hypergraph k-cut problem.

Notice that this reduction preserves the approximation factor. Hence with a $\left(2 - \frac{2}{k}\right)$-approximation algorithm [3] to the terminal k-vertex cut problem, we obtain the following.

Theorem 1. *The hypergraph k-cut problem can be approximated within a factor of $2 - \frac{2}{k}$ if k is fixed.* $\qquad\square$

We can reduce the hypergraph k-cut problem to the submodular k-partition problem as follows. Obtain a directed hypergraph by orienting $G = (V, E)$ arbitrarily. In other words, for each hyperedge $e \in E$, define the *head* denoted by $h(e)$ as an arbitrary vertex v contained by e. Moreover, define a set function $f_G : 2^V \rightarrow \Re$ as the *in-degree function* of the directed hypergraph with hyperedge weight $w : E \rightarrow \Re_+$, i.e., $f_G(X) = \sum \{w(e) \mid e \in E, h(e) \in X, e \setminus X \neq \emptyset\}$ for $X \subseteq V$. It is easy to prove that function f_G is submodular. A k-cut $[V_1, V_2, \ldots, V_k]$ of the hypergraph $G = (V, E)$ satisfies $w(V_1, V_2, \ldots, V_k) = f_G(V_1, V_2, \ldots, V_k)$. Therefore, we obtain the following.

Theorem 2. *The hypergraph k-cut problem is contained by the submodular system k-partition problem.* $\qquad\square$

We note that the submodular function defined from the hypergraph is not symmetric. This is the difference between the graph k-cut problem and the hypergraph k-cut problem.

3 Basic property of partitions

In this section, we prove two important properties of k-partitions.

Theorem 3. *Let $[X_1, X_2, \ldots, X_k]$ be a minimum k-partition of a submodular system (V, f). Moreover, let $X = \cup_{j=1}^{h} X_{i_j}$ for $\{i_1, i_2, \ldots, i_h\} \subseteq \{1, 2, \ldots, k\}$. Then $[X_{i_1}, X_{i_2}, \ldots, X_{i_h}]$ is a minimum h-partition of (X, f^X).*

Proof. Assuming that $\{i_1, i_2, \ldots, i_h\} = \{1, 2, \ldots, h\}$, we show that $[X_1, X_2, \ldots, X_h]$ is a minimum h-partition of (X, f^X). Suppose the contrary, i.e., there exists a h-partition $[Y_1, Y_2, \ldots, Y_h]$ of (X, f^X) such that $f^X(Y_1, Y_2, \ldots, Y_h) < f^X(X_1, X_2, \ldots, X_h)$. In this case, we obtain

$$\sum_{i=1}^{h} f(Y_i) + \sum_{i=h+1}^{k} f(X_i) = \sum_{i=1}^{h} f^X(Y_i) + \sum_{i=h+1}^{k} f(X_i) < \sum_{i=1}^{h} f^X(X_i) + \sum_{i=h+1}^{k} f(X_i) = \sum_{i=1}^{k} f(X_i).$$

Since $[Y_1, \ldots, Y_h, X_{h+1}, \ldots, X_k]$ is a k-partition of (V, f), this inequality contradicts the definition of $[X_1, X_2, \ldots, X_k]$. $\qquad\square$

Theorem 4. *Let $P = [X_1, X_2]$ and $P_k = [Y_1, Y_2, \ldots, Y_k]$ be 2- and k-partitions of a submodular system (V, f), respectively. Let Z_{ij} denote $X_i \cap Y_j$ for each $i \in \{1, 2\}$ and $j \in \{1, 2, \ldots, k\}$. If some pair of $i \in \{1, 2\}$ and $j \in \{1, 2, \ldots, k\}$ satisfies $Z_{i'j'} \neq \emptyset$ for all $i' \neq i$ and $j' \neq j$, and $f(Z_{ij}, \overline{Z_{ij}}) \geq f(P)$, then*

$$P_k' = [Y_1 \setminus X_i, \ldots, Y_{j-1} \setminus X_i, X_i \cup Y_j, Y_{j+1} \setminus X_i, \ldots, Y_k \setminus X_i]$$

is a k-partition of (V, f) that satisfies $f(P_k') \leq f(P_k)$.

Proof. Assume that $i = j = 1$. It is obvious from the assumption that P'_k is a k-partition of (V, f). Hence in the following, let us prove $f(P'_k) \le f(P_k)$.

By the submodularity of f, it holds that

$$f(Y_1) + f(X_1) \ge f(Y_1 \cap X_1) + f(Y_1 \cup X_1) = f(Z_{11}) + f(Y_1 \cup X_1). \tag{1}$$

Similarly it follows that

$$f(X_2) + \sum_{h=2}^{k} f(Y_h) \ge f(Y_2 \cap X_2) + f(Y_2 \cup X_2) + \sum_{h=3}^{k} f(Y_h)$$

$$\ge f(Y_2 \cap X_2) + f((Y_2 \cup X_2) \cap Y_3) + f(Y_2 \cup Y_3 \cup X_2) + \sum_{h=4}^{k} f(Y_h)$$

$$= f(Y_2 \cap X_2) + f(Y_3 \cap X_2) + f(Y_2 \cup Y_3 \cup X_2) + \sum_{h=4}^{k} f(Y_h)$$

$$\ge \cdots$$

$$\ge \sum_{h=2}^{k} f(Y_h \cap X_2) + f(Y_2 \cup Y_3 \cup \cdots \cup Y_k \cup X_2)$$

$$= \sum_{h=2}^{k} f(Z_{2h}) + f(\overline{Z_{11}}). \tag{2}$$

Summing (1) and (2) gives

$$\sum_{h=1}^{k} f(Y_h) + f(X_1) + f(X_2) \ge f(Y_1 \cup X_1) + \sum_{h=2}^{k} f(Z_{2h}) + f(Z_{11}) + f(\overline{Z_{11}}). \tag{3}$$

Moreover, it follows from the assumption that

$$f(X_1) + f(X_2) \le f(Z_{11}) + f(\overline{Z_{11}}). \tag{4}$$

Therefore by (3) and (4),

$$f(P_k) = \sum_{h=1}^{k} f(Y_h) \ge f(Y_1 \cup X_1) + \sum_{h=2}^{k} f(Z_{2h}) = f(P'_k).$$

$\square$

Theorems 3 and 4 have been already proven in [16] for hypergraphs. In particular, Theorem 4 is important because it tells that when a minimum k-partition crosses a 2-partition and they satisfies the conditions, we can uncross them by finding another minimum k-partition.

4 Exact Algorithm for the submodular system 3-partition problem

In this section, we show that the algorithm in [16] for the hypergraph 3-cut problem can be extended to submodular systems straightforwardly. The following theorem on structure of minimum 3-partitions is a consequence of Theorem 4.

Theorem 5. *Let $P = [X_1, X_2]$ be a minimum 2-size 2-partition of a submodular system (V, f) with $|V| \geq 7$. If (V, f) has a 2-size minimum 3-partition, then it has a minimum 3-partition not crossed by P.*

Proof. Assume that (V, f) has a minimum 3-partition $P_3 = [Y_1, Y_2, Y_3]$ which is 2-size (i.e., $|Y_i| \geq 2$, $i \in \{1, 2, 3\}$). We prove the theorem by constructing a minimum 3-partition of (V, f) not crossed by P.

Let $Z_{i,j} = X_i \cap Y_j$ for $i \in \{1, 2\}$ and $j \in \{1, 2, 3\}$. For $i \in \{1, 2\}$, let n_i denote the number of components of P_3 intersecting X_i. Without loss of generality, we suppose that $n_1 \leq n_2$. If $n_1 = 1$, then P do not cross P_3 because $X_1 \subseteq Y_j$ holds for some $j \in \{1, 2, 3\}$. Hence in the following, we consider the case where $n_2 \geq n_1 \geq 2$.

First, let us examine the case of $n_1 = 2$. Without loss of generality assume that $Z_{1,1} \neq \emptyset$, $Z_{1,2} \neq \emptyset$, and $Z_{1,3} = \emptyset$. It follows that $|Z_{2,3}| = |Y_3| \geq 2$ because P_3 is 2-size. Since $[Z_{2,3}, \overline{Z_{2,3}}]$ is a 2-size 2-partition of (V, f), $f(Z_{2,3}, \overline{Z_{2,3}}) \geq f(P)$ holds by the definition of P. Theorem 4 then implies that $[Z_{1,1}, Z_{1,2}, X_2 \cup Y_3]$ is also a minimum 3-partition of (V, f). Observe that this 3-partition is not crossed by P since $X_2 \subseteq X_2 \cup Y_3$.

Next, let us examine the remaining case, i.e., $n_1 = n_2 = 3$. For all $i \in \{1, 2\}$ and $j \in \{1, 2, 3\}$, $Z_{i,j} \neq \emptyset$ holds in this case. Since $|V| \geq 7$, some $i \in \{1, 2\}$ and $j \in \{1, 2, 3\}$ satisfy $|Z_{i,j}| \geq 2$. Let $|Z_{1,1}| \geq 2$ for example. Then $[Z_{1,1}, \overline{Z_{1,1}}]$ is a 2-size 2-partition. Hence by the definition of P, $f(Z_{1,1}, \overline{Z_{1,1}}) \geq f(P)$ holds. Theorem 4 implies that $[X_1 \cup Y_1, Z_{2,2}, Z_{2,3}]$ is also a minimum 3-partition of (V, f). Observe that this 3-partition is not crossed by P since $X_1 \subseteq X_1 \cup Y_1$. $\square$

Theorem 5 tells us that at least one of the following cases holds:

(i) $|V| \leq 6$;

(ii) There exists a minimum 3-partition which is not 2-size, i.e., $[\{v\}, V_1, V_2]$ with some $v \in V$, and $V_1, V_2 \subseteq V \setminus \{v\}$;

(iii) For any minimum 2-size 2-partition, there exists a minimum 3-partition not crossed by it.

Let us see how to compute a solution in each case. In the case of (i), it is possible to compute an optimal solution in constant time. In the case of (ii), by Theorem 3, $[V_1, V_2]$ is a minimum 2-partition of the subsystem $(V \setminus \{v\}, f^{V \setminus \{v\}})$. Hence it is possible to compute an optimal solution by computing a minimum 2-partition for the subsystem induced by $V \setminus \{v\}$ with every $v \in V$,

Let us discuss the case of (iii). For computing a minimum 2-size 2-partition, the algorithm presented by Vazirani and Yannakakis [15] can be used. Their algorithm enumerates all the 2-partitions in the order of non-decreasing costs and with the delay between two successive outputs at most $O(n)$ minimum (s, t)-partition computations. Since the number of 2-partitions not being 2-size is n, we need to compute at most $n + 1$ outputs to obtain a minimum 2-size 2-partition. Hence we can find a minimum 2-size 2-partition by computing minimum (s, t)-partition $O(n^2)$ times.

Let $P = [X_1, X_2]$ be a minimum 2-size 2-partition computed as above. Since a 3-partition not crossed by P has a corresponding 3-partition in either (V_{X_1}, f_{X_1}) or (V_{X_2}, f_{X_2}). Hence by solving the problem recursively for these two systems, we can find an optimal solution for the original system.

The entire of the algorithm is described as follows. For a set $\mathcal{P}$ of partitions, we let $\min \mathcal{P}$ return a partition of minimum cost in $\mathcal{P}$.

Algorithm MIN3PT(V,f)

Input: A submodular system (V, f)

Output: A 3-partition of (V, f)

Step 1: Initialize the solution S with an arbitrary 3-partition of (V, f).

Step 2: If $|V| \leq 6$, then enumerate all 3-partitions of V, and terminate with outputting a minimum 3-partition among them.

Step 3: For every $v \in V$, compute a minimum 2-partition $[V_1, V_2]$ of $(V \setminus \{v\}, f^{V \setminus \{v\}})$ and $S \leftarrow \min\{S, [\{v\}, V_1, V_2]\}$.

Step 4: Compute a minimum 2-size 2-partition $[X_1, X_2]$ of (V, f). Then terminate with outputting $\min\{S, \text{MIN3PT}(V_{X_1}, f_{X_1}), \text{MIN3PT}(V_{X_2}, f_{X_2})\}$.

Theorem 6. *Algorithm* MIN3PT *computes a minimum 3-partition of a submodular system by computing minimum (s, t)-partitions $O(n^3)$ times.*

Proof. Let $x = |X_1|$. Since $[X_1, X_2]$ is 2-size, $2 \leq x \leq n - 2$. Moreover, $|V_{X_1}| = n - x + 1$ and $|V_{X_2}| = x + 1$. Define $D(n)$ as the number of (s, t)-partition computations used in MIN3PT(V, f) when $|V| = n$. We have already seen that all operations except executing MIN3PT(V_{X_1}, f_{X_1}) and MIN3PT(V_{X_2}, f_{X_2}) can be done in $O(n^2)$ (s, t)-partition computations. Hence we have

$$D(n) \leq D(n - x + 1) + D(x + 1) + O(n^2).$$

It is easy to see that $D(n) = O(n^3)$ satisfies this inequality. $\square$

5 Approximation algorithm for 4-partition problem

5.1 Submodular system 4-partition problem

In this section, we consider the submodular 4-partition problem with non-negative submodular functions. As a consequence of Theorem 4, we obtain the following theorem.

Theorem 7. *Let $P = [X_1, X_2]$ be a minimum 3-size 2-partition of a submodular system (V, f) with $|V| \geq 17$. Then (V, f) has a minimum 4-partition $P_4 = [Y_1, Y_2, Y_3, Y_4]$ satisfying one of the following conditions up to changing indices of components in P and P_4:*

(i) *P_4 is not 3-size;*

(ii) *$Y_1 \cup Y_2 = X_1$ and $Y_3 \cup Y_4 = X_2$;*

(iii) *$1 \leq |X_1 \cap Y_1| \leq |X_1 \cap Y_2| \leq 2$, $X_1 \cap Y_3 = X_1 \cap Y_4 = \emptyset$, and $X_2 \cap Y_j \neq \emptyset$ for $j \in \{1, 2, 3, 4\}$;*

(iv) *P does not cross P_4.*

Proof. Assuming no minimum 4-partition satisfies (i), (ii) and (iii), we construct a minimum 4-partition satisfying (iv). Let $P_4 = [Y_1, Y_2, Y_3, Y_4]$ be an arbitrary minimum 4-partition of (V, f). Since it does not satisfy (i), P_4 is 3-size (i.e., $|Y_i| \geq 3$ for $i \in \{1, 2, 3, 4\}$). Let n_i denote the number of components of P_4 intersecting with X_i for $i \in \{1, 2\}$, and we assume without loss of generality

8

that $n_1 \leq n_2$. If $n_1 = 1$, then P does not cross P_4 because $X_1 \subseteq Y_j$ holds for some $j \in \{1, 2, 3, 4\}$. Moreover if $n_1 = n_2 = 2$, then the second condition holds. Hence in the following, we consider the case where $n_1 \geq 2$ and $n_2 \geq 3$. We denote $X_i \cap Y_j$ by $Z_{i,j}$ for $i \in \{1, 2\}$ and $j \in \{1, 2, 3, 4\}$.

First, let us consider the case of $n_1 = 2$ and $n_2 \geq 3$. Assume without loss of generality that $X_1 \subseteq Y_1 \cup Y_2$ (i.e., $Z_{1,3} = Z_{1,4} = \emptyset$). Since $Y_3 \neq \emptyset$ and $Y_4 \neq \emptyset$, it holds that $Z_{2,3} \neq \emptyset$ and $Z_{2,4} \neq \emptyset$. If $Z_{2,1} = \emptyset$, then $|Z_{1,1}| = |Y_1| \geq 3$ holds because P_4 is 3-size, and $Z_{2,2} \neq \emptyset$ follows because $n_2 \geq 3$. Since $[Z_{1,1}, \overline{Z_{1,1}}]$ is a 3-size 2-partition of (V, f) in this case, $f(Z_{1,1}, \overline{Z_{1,1}}) \geq f(P)$ by the definition of P. Hence Theorem 4 implies that $[X_1 \cup Z_{2,1}, Z_{2,2}, Z_{2,3}, Z_{2,4}]$ is also a minimum 4-partition of (V, f) if $Z_{2,1} = \emptyset$. Notice that P does not cross this 4-partition since $X_1 \subseteq X_1 \cup Z_{2,1}$. Similarly if $Z_{2,2} = \emptyset$, then $[Z_{2,1}, X_1 \cup Z_{2,2}, Z_{2,3}, Z_{2,4}]$ is a minimum 4-partition not crossed by P.

Let $Z_{2,1} \neq \emptyset$ and $Z_{2,2} \neq \emptyset$. Since P_4 does not satisfy (iii), $|Z_{1,1}| \geq 3$ or $|Z_{1,2}| \geq 3$ holds. Suppose that $|Z_{1,1}| \geq 3$. Then $[Z_{1,1}, \overline{Z_{1,1}}]$ is a 3-size 2-partition of (V, f), and hence $f(Z_{1,1}, \overline{Z_{1,1}}) \geq f(P)$ holds by the definition of P. Theorem 4 implies that $[X_1 \cup Z_{2,1}, Z_{2,2}, Z_{2,3}, Z_{2,4}]$ is also a minimum 4-partition of (V, f). Notice that P does not cross this 4-partition since $X_1 \subseteq X_1 \cup Z_{2,1}$. Similarly if $|Z_{1,2}| \geq 3$, then $[Z_{2,1}, X_1 \cup Z_{2,2}, Z_{2,3}, Z_{2,4}]$ is a minimum 4-partition not crossed by P.

Next, let us consider the case of $n_2 \geq n_1 = 3$. We assume without loss of generality that $Z_{1,1} = \emptyset$ and $Z_{1,j} \neq \emptyset$ for $j \in \{2, 3, 4\}$. Since P_4 is 3-size, $|Z_{2,1}| = |Y_1| \geq 3$. Hence $[Z_{2,1}, \overline{Z_{2,1}}]$ is a 3-size 2-partition of (V, f), and $f(Z_{2,1}, \overline{Z_{2,1}}) \geq f(P)$ follows by the definition of P. Theorem 4 then implies that $[X_2 \cup Z_{1,1}, Z_{1,2}, Z_{1,3}, Z_{1,4}]$ is also a minimum 4-partition of (V, f). Notice that P does not cross this 4-partition since $X_2 \subseteq X_2 \cup Z_{1,1}$.

In the last, let us consider the case of $n_1 = n_2 = 4$, i.e., $Z_{i,j} \neq \emptyset$ holds for all $i \in \{1, 2\}$ and $j \in \{1, 2, 3, 4\}$. Since $|V| \geq 17$, $|Z_{i,j}| \geq 3$ holds for some $i \in \{1, 2\}$ and $j \in \{1, 2, 3, 4\}$. Assume without loss of generality that $|Z_{1,1}| \geq 3$. Then $[Z_{1,1}, \overline{Z_{1,1}}]$ is a 3-size 2-partition, and $f(Z_{1,1}, \overline{Z_{1,1}}) \geq f(P)$ holds by the definition of P. Theorem 4 then implies that $[X_1 \cup Z_{2,1}, Z_{2,2}, Z_{2,3}, Z_{2,4}]$ is also a minimum 4-partition of (V, f). Notice that P does not cross this 4-partition since $X_1 \subseteq X_1 \cup Z_{2,1}$. $\square$

Now we give our algorithm. When $|V| \leq 16$, it is possible to compute an optimal solution in constant time. When $|V| \geq 17$, Theorem 7 shows that there exists an optimal solution satisfying one of the four conditions with a minimum 3-size 2-partition P. If (i) or (ii) is satisfied, then it is possible to compute an optimal solution in polynomial time. If (iii) is satisfied, then it is possible to compute a 1.5-approximate solution in polynomial time, or to apply our algorithm recursively. If (iv) is satisfied, then it is possible to apply our algorithm recursively. We prove these facts from now on.

Before that, let us mention that a minimum 3-size 2-partition is computable as computing a minimum 2-size 2-partition in Section 4. Since the number of 2-partitions not being 3-size is $O(n^2)$, it suffices to compute a minimum (s, t)-partition $O(n^3)$ times.

First, let us examine the case where (i) is satisfied, i.e., there exists a minimum 4-partition $[U, V_1, V_2, V_3]$ with $U \subseteq V$, $1 \leq |U| \leq 2$, and $V_1, V_2, V_3 \subseteq V \setminus U$. By Theorem 3, $[V_1, V_2, V_3]$ is a minimum 3-partition of the subsystem induced by $V \setminus U$. Such a 3-partition can be found in $O(n^3)$ (s, t)-partition computations by using Algorithm MIN3PT in Section 4. By executing this for every $U \subseteq V$ with $1 \leq |U| \leq 2$, it is possible to compute an optimal solution in this case. The number of (s, t)-partition computations is $O(n^5)$.

Next, let us consider the case where (ii) is satisfied. Now we can obtain $P = [X_1, X_2]$ as stated above. By Theorem 3, $[Y_1, Y_2]$ is a minimum 2-partition of the subsystem (X_1, f^{X_1}), and $[Y_3, Y_4]$

is a minimum 2-partition of the subsystem (X_2, f^{X_2}). Hence an optimal solution can be found by computing minimum 2-partitions twice, which needs $O(n)$ minimum (s, t)-partition computations.

When (iv) is satisfied, we can recursively apply our algorithm to (V_{X_1}, f_{X_1}) and (V_{X_2}, f_{X_2}). Hence the remaining case is when (iii) is satisfied. We here suppose that the other three conditions are not satisfied. In contrast with the cases above, we do not know how to compute an optimal solution. However, we can compute a 1.5-approximate solution in this case, or find a 2-size 2-partition not crossing an optimal solution. For proving this fact, we present several lemmas. We use the notations in the proof of Theorem 7.

Lemma 1. *Suppose that P and P_4 in Theorem 7 satisfy (iii). Then,*

$$f(Z_{1,1}) + f(Z_{1,2}) + f(X_2) \le f(P_4).$$

Proof. It follows from the submodularity and non-negativity of f that

$$f(P_4) \ge f(Y_1) + f(Y_2) + f(Y_3 \cup Y_4). \tag{5}$$

Since P_4 is 3-size, a 2-partition $[Y_1 \cup Y_2, Y_3 \cup Y_4]$ is 3-size. Thus $f(P) \le f(Y_1 \cup Y_2, Y_3 \cup Y_4)$ holds by the definition of P. Recall that $Z_{1,1} \ne \emptyset$, $Z_{1,2} \ne \emptyset$, and $Y_3 \cup Y_4 = Z_{2,3} \cup Z_{2,4}$ now. Hence by Theorem 4,

$$f(Y_1) + f(Y_2) + f(Y_3 \cup Y_4) \ge f(Z_{1,1}) + f(Z_{1,2}) + f(X_2) \tag{6}$$

holds. (5) and (6) give the required inequality. □

Lemma 2. *Suppose that P and P_4 of Theorem 7 satisfy (iii). Moreover, suppose that P crosses any minimum 4-partition of (V, f). Then,*

$$f(Z_{1,1}, Z_{1,2}) < 2f(X_1) + f(X_2).$$

Proof. If $f(Z_{1,1}, \overline{Z_{1,1}}) \ge f(X_1, X_2)$, then $[X_1 \cup Y_1, Z_{2,2}, Z_{2,3}, Z_{2,4}]$ is a minimum 4-partition of (V, f) by Theorem 4. Notice that P does not cross this 4-partition, contradicting to the assumption. Hence $f(X_1, X_2) > f(Z_{1,1}, \overline{Z_{1,1}})$ holds. Similarly, $f(X_1, X_2) > f(Z_{1,2}, \overline{Z_{1,2}})$ also holds. Summing these gives

$$2f(X_1, X_2) > f(Z_{1,1}, \overline{Z_{1,1}}) + f(Z_{1,2}, \overline{Z_{1,2}}). \tag{7}$$

By the submodularity and non-negativity of f,

$$f(\overline{Z_{1,1}}) + f(\overline{Z_{1,2}}) \ge f(\overline{Z_{1,1}} \cup \overline{Z_{1,2}}) + f(\overline{Z_{1,1}} \cap \overline{Z_{1,2}}) = f(V) + f(X_2) \ge f(X_2).$$

Thus the required inequality is proven. □

From Lemmas 1 and 2, we can derive the following.

Lemma 3. *Suppose that P and P_4 in Theorem 7 satisfy (iii), and that P crosses any minimum 4-partition of (V, f). Moreover, assume that there exists a minimum 2-partition of (X_1, f^{X_1}) different from $[Z_{1,1}, Z_{1,2}]$. Then a minimum 3-partition $[R_1, R_2, R_3]$ of (X_1, f^{X_1}) satisfies*

$$f(X_2, R_1, R_2, R_3) \le 1.5f(P_4).$$

Proof. Let $[A, B]$ be a minimum 2-partition of (X_1, f^{X_1}) different from $[Z_{1,1}, Z_{1,2}]$. By the definition,

$$f(Z_{1,1}, Z_{1,2}) \geq f(A, B)$$

holds. By Lemma 1,

$$f(P_4) \geq f(Z_{1,1}, Z_{1,2}, X_2)$$

also holds. Hence

$$f(P_4) + f(Z_{1,1}) + f(Z_{1,2}) \geq f(Z_{1,1}, Z_{1,2}, X_2) + f(A, B). \tag{8}$$

Notice that $Z_{1,1} \cup Z_{1,2} \cup A = X_1$ and $Z_{1,1} \cap (Z_{1,2} \cup A) = Z_{1,1} \cap A$. By the submodularity of f,

$$f(Z_{1,1}) + f(Z_{1,2}) + f(A) \geq f(Z_{1,1}) + f(Z_{1,2} \cup A) + f(Z_{1,2} \cap A)$$
$$\geq f(X_1) + f(Z_{1,1} \cap A) + f(Z_{1,2} \cap A).$$

Hence we obtain

$$f(Z_{1,1}, Z_{1,2}, X_2) + f(A, B) \geq f(X_1) + f(Z_{1,1} \cap A) + f(Z_{1,2} \cap A) + f(X_2) + f(B). \tag{9}$$

Since $[A, B]$ and $[Z_{1,1}, Z_{1,2}]$ are different, we can assume without loss of generality that $A \cap Z_{1,1} \neq \emptyset$ and $A \cap Z_{1,2} \neq \emptyset$. Then $[Z_{1,1} \cap A, Z_{1,2} \cap A, B]$ is a 3-partition of (X_1, f^{X_1}). By the definition of $[R_1, R_2, R_3]$,

$$f(Z_{1,1} \cap A) + f(Z_{1,2} \cap A) + f(B) \geq f(R_1, R_2, R_3) \tag{10}$$

holds. From (8), (9) and (10), we obtain

$$f(P_4) + f(Z_{1,1}) + f(Z_{1,2}) - f(X_1) \geq f(X_2) + f(R_1, R_2, R_3). \tag{11}$$

On the other hand, Lemmas 1 and 2 show that

$$f(Z_{1,1}) + f(Z_{1,2}) - f(X_1) \leq \min\{f(P_4) - f(X_1) - f(X_2), f(X_1) + f(X_2)\}.$$

The maximum of the right-hand side of this inequality is $0.5f(P_4)$, which is achieved by $f(X_1) + f(X_2) = 0.5f(P_4)$. This and (11) give the required inequality. $\qquad\square$

Let $[A, B]$ and $[R_1, R_2, R_3]$ be minimum 2- and 3-partitions of (X_1, f^{X_1}) respectively. We suppose that $|A| \geq |B|$. Notice that $|A| \geq 2$ because $|X_1| \geq 3$. Lemma 3 tells that, when only the third condition of Theorem 7 is satisfied, $A = Z_{1,1}$, $A = Z_{1,2}$, or $[R_1, R_2, R_3, X_2]$ is a 1.5-approximate solution. If $A = Z_{1,1}$ or $A = Z_{1,2}$, then shrinking A preserves P_4 because $[A, \overline{A}]$ does not cross P_4. Hence in this case, we can either find a 1.5-approximate solution or apply the algorithm recursively to (V_A, f_A).

We describe the entire of our algorithm below.

Algorithm MIN4PT(V, f)

Input: A submodular system (V, f)

Output: A 4-partition of (V, f)

Step 1: Initialize the solution S with an arbitrary 4-partition of (V, f).

Step 2: If $|V| \leq 16$, then enumerate all 4-partitions of V, and terminate with outputting a minimum 4-partition among them.

Step 3: For every $U \subseteq V$ with $1 \leq |U| \leq 2$, compute a minimum 3-partition $\{V_1, V_2, V_3\}$ of $(V \setminus U, f^{V \setminus U})$ by using Algorithm MIN3PT, and $S \leftarrow \min\{S, [U, V_1, V_2, V_3]\}$. (This step is for (i) and (iii).)

Step 4: Compute a minimum 3-size 2-partition $[X_1, X_2]$ of (V, f) such that $|X_1| \leq |X_2|$.

Step 5: Compute a minimum 2-partition $[Q_1, Q_2]$ of (X_1, f^{X_1}) and a minimum 2-partition $[Q_3, Q_4]$ of (X_2, f^{X_2}). Then $S \leftarrow \min\{S, [Q_1, Q_2, Q_3, Q_4]\}$. (This step is for (ii).)

Step 6: If $|X_1| \geq 5$, then set $S \leftarrow \min\{S, \text{MIN4PT}(V_{X_1}, f_{X_1}), \text{MIN4PT}(V_{X_2}, f_{X_2})\}$. (This step is for (iv).)

Step 7: If $|X_1| \leq 4$, then compute a minimum 2-partition $[A, B]$ with $|A| \geq |B|$, and set $S \leftarrow \min\{S, \text{MIN4PT}(V_A, f_A), \text{MIN4PT}(V_{X_2}, f_{X_2})\}$. (This step is for (iii) and (iv).)

Step 8: Terminate with outputting S.

Theorem 8. *Algorithm* MIN4PT *computes a 1.5-approximate solution of the submodular system 4-partition problem by computing minimum (s, t)-partitions $O(n^6)$ times.*

Proof. First, let us prove the 1.5-approximability of the algorithm by the induction on n. When $|V| \leq 16$, Step 2 of the algorithm computes an optimal solution. Below, we examine the cases where the conditions discussed in Theorem 7 are satisfied by the minimum 3-size 2-partition $[X_1, X_2]$ computed in the algorithm and some minimum 4-partition. When (i) of Theorem 7 is satisfied, Steps 3 of the algorithm finds an optimal solution as we have already seen. When (ii) of Theorem 7 is satisfied, Step 5 of the algorithm finds an optimal solution. Then the remaining case is when (iii) or (iv) of Theorem 8 holds.

Consider the case where (iv) of Theorem 7 is satisfied. In this case, shrinking X_1 or X_2 preserves an optimal solution. Hence when $|X_5| \geq 5$, $\text{MIN4PT}(V_{X_1}, f_{X_1})$ or $\text{MIN4PT}(V_{X_2}, f_{X_2})$ in Step 6 of the algorithm returns a 1.5-approximate solution by the inductive hypothesis. When $|X_5| \leq 4$, $\text{MIN4PT}(V_A, f_A)$ or $\text{MIN4PT}(V_{X_2}, f_{X_2})$ in Step 7 of the algorithm returns a 1.5-approximate solution by the inductive hypothesis (Notice that $A \subseteq X_1$).

Consider the case where (iii) of Theorem 7 is satisfied. In this case, $|X_1| \leq 4$ holds. Lemma 3 implies that $A = Z_{1,1}$, $A = Z_{1,2}$, or $[R_1, R_2, R_3, X_2]$ is a 1.5-approximate solution where $[R_1, R_2, R_3]$ is a minimum 3-partition of (X_1, f^{X_1}). If $A = Z_{1,1}$ or $A = Z_{1,2}$ holds, then $\text{MIN4PT}(V_A, f_A)$ returns a 1.5-approximate solution by the inductive hypothesis. Otherwise, Step 3 computes a 1.5-approximate solution since each of R_i, $i = 1, 2, 3$ contains at most two elements.

Next, let us consider the number of computing minimum (s, t)-partitions. Let $x = |X_1|$. Since $[X_1, X_2]$ is 3-size, $3 \leq x \leq n - 3$. Moreover, $|V_{X_1}| = n - x + 1$ and $|V_{X_2}| = x + 1$ hold. Define $D(n)$ as the number of (s, t)-partition computations by the algorithm. By the discussion above, all operations in Steps 1 to 6 can be done by $O(n^5)$ (s, t)-computations. If $|X_1| \geq 5$, then the algorithm executes the operations in Step 6. In this case, we have

$$D(n) \leq D(x + 1) + D(n - x + 1) + O(n^5). \tag{12}$$

If $|X_1| \leq 4$, then the algorithm executes the operations in Step 7 instead of those in Step 6. In this case, we have

$$D(n) \leq D(x+1) + D(n - |B_1| + 1) + O(n^5) \leq D(5) + D(n-1) + O(n^5). \qquad (13)$$

It is easy to verify that $D(n) = O(n^6)$ satisfies both of (12) and (13). $\qquad \square$

5.2 Hypergraph 4-cut problem

In this section, we consider the hypergraph 4-cut problem. Since this problem is contained by the submodular system 4-partition problem as seen in Theorem 2, theorems in Section 5.1 hold and Algorithm MIN4PT works with the same approximation factor. In what follows, we show that Algorithm MIN4PT achieves better approximation factor for this problem.

Lemma 4. *Assume that the submodular system (V, f) is constructed from a hypergraph $G = (V, E)$ with weight $w : E \to \Re_+$ as described in Section 2.2, and that P and P_4 in Theorem 7 satisfy (iii). If P crosses any minimum 4-partition of (V, f), then*

$$w_e(Z_{1,1}, Z_{1,2}) < \min\{w_e(Z_{1,1}, X_2), w_e(Z_{1,2}, X_2)\}.$$

Proof. We assume without loss of generality that $w_e(Z_{1,2}, X_2) \leq w_e(Z_{1,1}, X_2)$. Notice that all of $Z_{1,1}$, $Z_{2,2}$, $Z_{2,3}$ and $Z_{2,4}$ are non-empty in this case. Hence by Theorem 4, if $w(Z_{1,1}, \overline{Z_{1,1}}) \geq w(X_1, X_2)$, then $[X_1 \cup Y_1, Z_{2,2}, Z_{2,3}, Z_{2,4}]$ is a minimum 4-partition of (V, f). Notice that P does not cross this 4-partition, contradicting to the assumption. Hence,

$$w(Z_{1,1}, \overline{Z_{1,1}}) < w(X_1, X_2)$$

holds. Since $\overline{Z_{1,1}} = X_2 \cup Z_{1,2}$ and $X_2 \cap Z_{1,2} = \emptyset$, we have

$$w(Z_{1,1}, \overline{Z_{1,1}}) = w(Z_{1,1}, X_2) + w_e(Z_{1,1}, Z_{1,2}).$$

Similarly, since $X_1 = Z_{1,1} \cup Z_{1,2}$ and $Z_{1,1} \cap Z_{1,2} = \emptyset$, we have

$$w(X_1, X_2) = w(Z_{1,1}, X_2) + w_e(Z_{1,2}, X_2).$$

From these, $w_e(Z_{1,1}, Z_{1,2}) < w_e(Z_{1,2}, X_2)$ follows. $\qquad \square$

Lemma 5. *Assume that the submodular system (V, f) is constructed from a hypergraph $G = (V, E)$ with weight $w : E \to \Re_+$ as described in Section 2.2, and that P and P_4 in Theorem 7 satisfy (iii). Moreover, suppose that P crosses any minimum 4-partition, and that there exists a minimum 2-partition of G^{X_1} different from $[Z_{1,1}, Z_{1,2}]$. Then a minimum 3-partition $[R_1, R_2, R_3]$ of (X_1, f^{X_1}) satisfies*

$$w(R_1, R_2, R_3, X_2) \leq \frac{4}{3} f(P_4).$$

Proof. By the submodularity of f,

$$f(P_4) \geq w(Y_1, Y_2, Y_3 \cup Y_4) \qquad (14)$$

holds. Let $[A, B]$ be a minimum 2-partition of $G(X_1)$, where we suppose that $|A| \geq |B|$. By the assumption, $A \cap Z_{1,1} \neq \emptyset$ and $A \cap Z_{1,2} \neq \emptyset$. Since P_4 is 3-size, a 2-partition $[Y_1 \cup Y_2, Y_3 \cup Y_4]$ is

13

3-size, and by the definition of P, $w(P) \geq w(Y_1 \cup Y_2, Y_3 \cup Y_4)$ holds. Recall that $Z_{1,3} \cup Z_{1,4} = \emptyset$. Hence by Theorem 4, it holds that

$$w(Y_1, Y_2, Y_3 \cup Y_4) \geq w(Z_{1,1}, Z_{1,2}, X_2). \tag{15}$$

Since $\delta(Z_{1,1}, Z_{1,2}, X_2) \supseteq \delta_e(Z_{1,1}, Z_{1,2}) \cup \delta_e(Z_{1,1}, X_2) \cup \delta_e(Z_{1,2}, X_2)$, and all of $\delta_e(Z_{1,1}, Z_{1,2})$, $\delta_e(Z_{1,1}, X_2)$ and $\delta_e(Z_{1,2}, X_2)$ are disjoint, we have

$$w(Z_{1,1}, Z_{1,2}, X_2) \geq w_e(Z_{1,1}, Z_{12}) + w_e(Z_{1,1}, X_2) + w_e(Z_{1,2}, X_2). \tag{16}$$

By Lemma 4,

$$w_e(Z_{1,1}, Z_{1,2}) + w_e(Z_{1,1}, X_2) + w_e(Z_{1,2}, X_2) \geq 3 w_e(Z_{1,1}, Z_{1,2}). \tag{17}$$

From (14), (15), (16), and (17), it follows that

$$w_e(Z_{1,1}, Z_{1,2}) < \frac{1}{3} f(P_4). \tag{18}$$

On the other hand, $w(R_1, R_2, R_3, X_2) = w_e(R_1, R_2, R_3) + w(X_1, X_2)$ because $\delta(R_1, R_2, R_3, X_2) = \delta_e(R_1, R_2, R_3) \cup \delta(X_1, X_2)$ and $\delta_e(R_1, R_2, R_3) \cap \delta(X_1, X_2) = \emptyset$. Moreover, $w_e(R_1, R_2, R_3) \leq w_e(A \cap Z_{1,1}, A \cap Z_{1,2}, B)$ by the definition of $[R_1, R_2, R_3]$. Hence

$$w(R_1, R_2, R_3, X_2) \leq w_e(A \cap Z_{1,1}, A \cap Z_{1,2}, B) + w(X_1, X_2). \tag{19}$$

Since $\delta_e(A \cap Z_{1,1}, A \cap Z_{1,2}, B) = \delta_e(A \cap Z_{1,1}, A \cap Z_{1,2}) \cup \delta_e(A, B)$ and $\delta_e(A \cap Z_{1,1}, A \cap Z_{1,2}) \cap \delta_e(A, B) = \emptyset$, it holds that

$$w_e(A \cap Z_{1,1}, A \cap Z_{1,2}, B) = w_e(A \cap Z_{1,1}, A \cap Z_{1,2}) + w_e(A, B). \tag{20}$$

Since $\delta_e(A \cap Z_{1,1}, A \cap Z_{1,2}) \subseteq \delta_e(Z_{1,1}, Z_{1,2})$,

$$w_e(A \cap Z_{1,1}, A \cap Z_{1,2}) \leq w_e(Z_{1,1}, Z_{1,2}) \tag{21}$$

holds. By combining (19), (20) and (21), we have

$$w(R_1, R_2, R_3, X_2) \leq w_e(Z_{1,1}, Z_{1,2}) + w_e(A, B) + w(X_1, X_2). \tag{22}$$

Notice that $w_e(Z_{1,1}, Z_{1,2}) + w(X_1, X_2) = w(Z_{1,1}, Z_{1,2}, X_2)$ because $Z_{1,1} \cup Z_{1,2} = X_1$ and $Z_{1,1} \cap Z_{1,2} = \emptyset$. Hence with (14) and (15), it shows that

$$w_e(Z_{1,1}, Z_{1,2}) + w(X_1, X_2) \leq f(P_4). \tag{23}$$

Moreover, $w_e(A, B) \leq w_e(Z_{1,1}, Z_{1,2})$ holds by the definition of $[A, B]$. This and (18) show that

$$w_e(A, B) \leq \frac{1}{3} f(P_4). \tag{24}$$

(22), (23) and (24) give the required inequality. $\qquad \square$

By replacing Lemma 3 with Lemma 5, we have the following.

Theorem 9. *Algorithm* MIN4PT *achieves approximation factor* 4/3 *for the hypergraph* 4-*cut problem.* $\qquad \square$

6 Approximation algorithm for k-partition problem

6.1 Submodular system k-partition problem

In this section, we consider the submodular system k-partition problem with an arbitrary fixed $k \geq 5$. As a consequence of Theorem 4, we obtain the following theorem.

Theorem 10. *Let $P = [X_1, X_2]$ be a minimum h-size 2-partition of a submodular system (V, f) with $|V| > 2k(h-1)$. For $i \in \{1, 2\}$ and some k-partition P_k of (V, f), let n_i denote the number of components of P_k intersecting X_i. For any P, there exists a minimum k-partition $P_k = [Y_1, Y_2, \ldots, Y_k]$ of (V, f) satisfying one of the following conditions up to changing indices of components in P and P_k:*

(i) *P_k is not h-size;*

(ii) *P_k is not crossed by P;*

(iii) *No minimum k-partition of (V, f) satisfies (i) and (ii), $2 \leq n_1 \leq k - 2$, and $2 \leq n_2 \leq k - 2$;*

(iv) *No minimum k-partition of (V, f) satisfies (i) and (ii), $2 \leq n_1 \leq k - 2$, $n_2 = k$, and $|X_1 \cap Y_j| < h$ for $j \in \{1, 2, \ldots, k\}$.*

Proof. Assuming that no minimum k-partition of (V, f) satisfies any of (i), (iii) and (iv), we construct a minimum k-partition of (V, f) satisfying (ii). Let $P_k = [Y_1, Y_2, \ldots, Y_k]$ be an arbitrary minimum k-partition of (V, f). Since it does not satisfy (i), P_k is h-size (i.e., $|Y_j| \geq h$ for $j \in \{1, 2, \ldots, k\}$). We assume without loss of generality that $n_1 \leq n_2$. Note that $n_1 \geq 2$ holds because P_k is not crossed by P. We denote $X_i \cap Y_j$ by $Z_{i,j}$ for $i \in \{1, 2\}$ and $j \in \{1, 2, \ldots, k\}$.

First, let us consider the case where $n_1 = n_2 = k$, i.e., $Z_{i,j} \neq \emptyset$ for every $i \in \{1, 2\}$ and $j \in \{1, 2, \ldots, k\}$. Since $|V| > 2k(h-1)$, some pair of i and j satisfies $|Z_{i,j}| \geq h$. Then $[Z_{i,j}, \overline{Z_{i,j}}]$ is a h-size 2-partition of (V, f). Hence by the definition of P, it holds that $f(Z_{i,j}, \overline{Z_{i,j}}) \geq f(P)$. By these facts and Theorem 4, we can construct another minimum k-partition of (V, f) that is not crossed by P in this case.

Next, consider the case where $n_1 = k - 1$. Without loss of generality, $Z_{1,j} \neq \emptyset$ for $j \in \{1, 2, \ldots, k - 1\}$ and $Z_{1,k} = \emptyset$. Then $Z_{2,k} = Y_k$. Since P_k is h-size, $|Z_{2,k}| = |Y_k| \geq h$. This implies that $[Z_{2,k}, \overline{Z_{2,k}}]$ is a h-size 2-partition of (V, f). Hence by the definition of P, it holds that $f(Z_{2,k}, \overline{Z_{2,k}}) \geq f(P)$. By these facts and Theorem 4, we can construct another minimum k-partition of (V, f) that is not crossed by P in this case.

Next, consider the case where $n_1 \leq k - 2$ and $n_2 = k$. Since P_k does not satisfy (iv), $|Z_{1,j}| \geq h$ for some $j \in \{1, 2, \ldots, k\}$ in this case. That is to say, $[Z_{1,j}, \overline{Z_{1,j}}]$ is a h-size 2-partition of (V, f). Hence by the definition of P, it holds that $f(Z_{1,j}, \overline{Z_{1,j}}) \geq f(P)$. Recall that $Z_{2,j} \neq \emptyset$ for every $j \in \{1, 2, \ldots, k\}$ because $n_2 = k$. By these facts and Theorem 4, we can construct another minimum k-partition of (V, f) that is not crossed by P in this case.

The remaining case is when $n_1 \leq k - 2$ and $n_2 = k - 1$ since (iii) is not satisfied. By $n_2 = k - 1$, we can assume without loss of generality that $Z_{2,1} = \emptyset$ and $Z_{2,j} \neq \emptyset$ for $j \geq 2$. It follows that $Z_{1,1} = Y_1$ from $Z_{2,1} = \emptyset$. Since P_k is h-size, $|Z_{1,1}| = |Y_1| \geq h$. This implies that $[Z_{1,1}, \overline{Z_{1,1}}]$ is a h-size 2-partition of (V, f). Hence by the definition of P, it holds that $f(Z_{1,1}, \overline{Z_{1,1}}) \geq f(P)$. By these facts and Theorem 4, we can construct another minimum k-partition of (V, f) that is not crossed by P in this case. $\square$

Now we present our algorithm. We choose $2k - 3$ as h in Theorem 10. If $|V| \leq 2k(h - 1) = 4k(k-2)$, we compute a minimum k-partition by enumerating all k-partitions. The computational time for this is $O(|V|^k) = O(2^{2k}k^k(k - 2)^k)$.

Note that a minimum h-size 2-partition P can be found by computing minimum (s, t)-partitions $O(n^h)$ times as explained for $h = 2$ in Section 5.1. Below we examine the case where some condition in Theorem 10 holds. We let α_k denote the approximation factor for the submodular system k-partition problem. In the rest of this section, we use the notations defined in the proof of Theorem 10.

Let us consider the case where (i) of Theorem 10 is satisfied. In this case, a minimum k-partition is $[U, V_1, V_2, \ldots, V_{k-1}]$ with some $U \subseteq V$, $1 \leq |U| < h$, and $V_1, V_2, \ldots, V_{k-1} \subseteq V \setminus U$. By Theorem 3, $[V_1, V_2, \ldots, V_{k-1}]$ is a minimum $(k - 1)$-partition of the subsystem induced by $V \setminus U$. Notice that the number of subsets U of V with $1 \leq |U| < h$ is $O(n^{h-1})$. Thus in this case, we can compute an α_{k-1}-approximate solution by applying the algorithm for $(k - 1)$-partition problem $O(n^{h-1}) = O(n^{2k-4})$ times.

Next, let us discuss the case where (ii) of Theorem 10 is satisfied. In this case, shrinking X_1 or X_2 preserves the minimum k-partition P_k of (V, f). Hence we can obtain an approximate solution by applying the algorithm recursively to smaller instances.

Next, let us discuss the case where (iii) of Theorem 10 is satisfied. Recall that at least one of $Z_{1,j} \neq \emptyset$ and $Z_{2,j} \neq \emptyset$ holds for any $j \in \{1, 2, \ldots, k\}$, and hence $n_1 + n_2 \geq k$. For $k - n_2 \leq \ell \leq n_1$, define P_ℓ^1 and $P_{k-\ell}^2$ as a minimum ℓ-partition of (X_1, f^{X_1}) and a minimum $(k - \ell)$-partition of (X_2, f^{X_2}), respectively. We show that both of $f(P_\ell^1)$ and $f(P_{k-\ell}^2)$ are at most $f(P_k)$ for any ℓ. For this, we need the following preparatory lemma.

Lemma 6. *Suppose that P and P_k in Theorem 10 satisfy either (iii) or (iv). Moreover, let $Z_{1,j} \neq \emptyset$ for $j \in \{1, 2, \ldots, n_1\}$. Then,*

$$f(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1}, X_2) \leq f(P_k).$$

Proof. By the submodularity of f,

$$f(P_k) \geq f(Y_1, Y_2, \ldots, Y_{n_1}, \cup_{j=n_1+1}^k Y_j) \tag{25}$$

holds. Since $\cup_{j=n_1+1}^k Z_{2,j} = \cup_{j=n_1+1}^k Y_j \geq (k - n_1)h > h$, $[\cup_{j=n_1+1}^k Z_{2,j}, \overline{\cup_{j=n_1+1}^k Z_{2,j}}]$ is a h-size 2-partition of (V, f). Hence $f(\cup_{j=n_1+1}^k Z_{2,j}, \overline{\cup_{j=n_1+1}^k Z_{2,j}}) \geq f(P)$ holds by the definition of P. Recall that we are assuming $Z_{1,j} \neq \emptyset$ for $j \in \{1, 2, \ldots, n_1\}$. By these facts and Theorem 4,

$$f(Y_1, Y_2, \ldots, Y_{n_1}, \cup_{j=n_1+1}^k Y_j) \geq f(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1}, X_2) \tag{26}$$

holds. Combining (25) and (26) proves the required inequality. $\qquad\square$

Lemma 7. *Suppose that P and P_k in Theorem 10 satisfy (iii). Then a minimum ℓ-partition P_ℓ^1 of (X_1, f^{X_1}) with $\ell \leq n_1$ satisfies $f(P_\ell^1) \leq f(P_k)$, and a minimum $(k - \ell)$-partition $P_{k-\ell}^2$ of (X_2, f^{X_2}) with $k - \ell \leq n_2$ satisfies $f(P_{k-\ell}^2) \leq f(P_k)$.*

Proof. We here show only $f(P_\ell^1) \leq f(P_k)$ because $f(P_{k-\ell}^2) \leq f(P_k)$ can be proven similarly. Suppose without loss of generality that $Z_{1,j} \neq \emptyset$ for $1 \leq j \leq n_1$ and $Z_{1,j} = \emptyset$ for $n_1 + 1 \leq j \leq k$. Lemma 6 implies that $f(X_2, Z_{11}, Z_{12}, \ldots, Z_{1n_1}) \leq f(P_k)$. By the non-negativity of f, Theorem 3, and the definition of P_ℓ^1, it follows that

$$f(P_\ell^1) \leq f(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1}) \leq f(X_2, Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1}).$$

16

This proves the required inequality. □

Lemma 7 tells that the union of an α_ℓ-approximate ℓ-partition of (X_1, f^{X_1}) and an $\alpha_{k-\ell}$-approximate $(k-\ell)$-partition of (X_2, f^{X_2}) achieves approximation factor $\alpha_\ell + \alpha_{k-\ell}$ for any $k-n_2 \leq \ell \leq n_1$. Since we do not know n_1 and n_2 in advance, the algorithm computes such a solution for every $2 \leq \ell \leq k-2$, and outputs the best one among them. This achieves the approximation factor at most $\max_{2 \leq \ell \leq \lceil k/2 \rceil}(\alpha_\ell + \alpha_{k-\ell})$ in this case.

In the last, let us discuss the case where (iv) of Theorem 10. Without loss of generality, suppose that $0 < |Z_{1,j}| < h$ for $j \in \{1, 2, \ldots, n_1\}$ and $Z_{1,j} = \emptyset$ for $j \in \{n_1 + 1, \ldots, k\}$. Our algorithm enumerates 2-partitions, call $[A_i, B_i]$, $i \in \{1, 2, \ldots, p\}$, of (X_1, f^{X_1}) by the algorithm due to Vazirani and Yannakakis [15] until X_1 is divided into at least $k-1$ subsets by all of them. Obviously p is at most $2^{k-2} - 1$. Let $[C_1, C_2, \ldots, C_{k'}]$ be the obtained k'-partition of (X_1, f^{X_1}), where $k-1 \leq k' \leq 2(k-2)$.

Since $|X_1| \geq h = 2(k-2) + 1$, some $i \in \{1, 2, \ldots, k'\}$ satisfies $|C_i| \geq 2$. If all of $[Z_{1,1}, X_1 \setminus Z_{1,1}], [Z_{1,2}, X_1 \setminus Z_{1,2}], \ldots, [Z_{1,n_1}, X_1 \setminus Z_{1,n_1}]$ are enumerated in the above computation, then shrinking each component of $[C_1, C_2, \ldots, C_{k'}]$ preserves the optimal solution. Hence we can reduce the size of the system in this case.

Let us consider the case such that some of $[Z_{1,1}, X_1 \setminus Z_{1,1}], [Z_{1,2}, X_1 \setminus Z_{1,2}], \ldots, [Z_{1,n_1}, X_1 \setminus Z_{1,n_1}]$ is not enumerated. Let $[R_1, R_2, \ldots, R_{k-1}]$ be a minimum $(k-1)$-partition of (X_1, f^{X_1}). In what follows, we prove that $[R_1, R_2, \ldots, R_{k-1}, X_2]$ is a $(k+1 - 2\sqrt{k-1})$-approximate solution in this case. Note that such $[R_1, R_2, \ldots, R_{k-1}]$ can be computed in $O(k^{|X_1|}) = O(k^{hn_1}) = O(k^{(2k-3)(k-2)})$ time.

Without loss of generality, suppose that $[Z_{1,1}, X_1 \setminus Z_{1,1}]$ is not enumerated. Remember that the algorithm due to Vazirani and Yannakakis [15] enumerates all 2-partitions in the order of non-decreasing cost. Hence $f(A_i, B_i) \leq f(Z_{1,1}, X_1 \setminus Z_{1,1})$ for every $i \in \{1, 2, \ldots, p\}$ now. We first show that the cost of $[R_1, R_2, \ldots, R_{k-1}, X_2]$ is bounded in terms of $f(Z_{1,1}, X_1 \setminus Z_{1,1}) - f(X_1)$, and we then give a bound on $f(Z_{1,1}, X_1 \setminus Z_{1,1}) - f(X_1)$.

Lemma 8. *Suppose that P and P_k in Theorem 10 satisfy (iv), and that all of $[A_i, B_i]$, $i \in \{1, 2, \ldots, p\}$ are different from $[Z_{1,1}, X_1 \setminus Z_{1,1}]$. Then,*

$$f(R_1, R_2, \ldots, R_{k-1}, X_2) \leq f(P_k) + (k - 1 - n_1)(f(Z_{1,1}, X_1 \setminus Z_{1,1}) - f(X_1)).$$

Proof. Below we show how to construct a $(k-1)$-partition P' of X_1 such that

$$f(P') \leq f(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1}) + (k - 1 - n_1)(f(Z_{1,1}, X_1 \setminus Z_{1,1}) - f(X_1)).$$

It proves the required inequality because $f(R_1, R_2, \ldots, R_{k-1}) \leq f(P')$ by the definition of $[R_1, R_2, \ldots, R_{k-1}]$, and $f(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1}) \leq f(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1}, X_2) \leq f(P_k)$ by the non-negativity of f and Lemma 6.

Initially set P' to $[Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1}]$. Then, for each $i \in \{1, 2, \ldots, p\}$, we divide each $C \in P'$ intersecting both of A_i and B_i into $C \cap A_i$ and $C \cap B_i$ until P' becomes a $(k-1)$-partition. Let us estimate the increase of $f(P')$ by the division with $[A_i, B_i]$. Assume that $C_1, C_2, \ldots, C_{\ell(i)} \in P'$

17

are divided with $[A_i, B_i]$. The submodularity of f indicates that

$$f(C_1, C_2, \ldots, C_{\ell(i)}) + \ell(i)f(A_i, B_i) \geq \sum_{j=1}^{\ell(i)} \{f(C_j \cap A_i) + f(C_j \cup A_i) + f(B_i)\}$$
$$\geq \sum_{j=1}^{\ell(i)} \{f(C_j \cap A_i) + f(C_j \cap B_i) + f(X_1)\},$$

from which it follows that

$$f(C_1 \cap A_i, C_1 \cap B_i, \ldots, C_{\ell(i)} \cap A_i, C_{\ell(i)} \cap B_i,) - f(C_1, C_2, \ldots, C_{\ell(i)})$$
$$\leq \ell(i)(f(A_i, B_i) - f(X_1)) \leq \ell(i)(f(Z_{1,1}, X_1 \setminus Z_{1,1}) - f(X_1)).$$

Notice that $f(P') - f(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1})$ is at most the sum of this increase over all $i \in \{1, 2, \ldots, p\}$, and $\sum_{i=1}^{p} \ell(i) = k - 1 - n_1$. These complete the proof. $\square$

Now let us estimate $f(Z_{1,1}, X_1 \setminus Z_{1,1}) - f(X_1)$.

Lemma 9. *Suppose that P and P_k in Theorem 10 satisfy (iv). Then,*

$$f(Z_{1,1}, X_1 \setminus Z_{1,1}) - f(X_1) \leq \frac{n_1 - 1}{n_1} f(P_k).$$

Proof. Notice that $Z_{2,j} \neq \emptyset$ for all $j \in \{1, 2, \ldots, k\}$ now. Hence by Theorem 4, if $f(Z_{1,j}, \overline{Z_{1,j}}) \geq f(X_1, X_2)$ for some $i \in \{1, 2, \ldots, n_1\}$, then we can obtain a minimum k-partition of (V, f) not crossed by P, a contradiction to the assumption. Thus $f(Z_{1,j}, \overline{Z_{1,j}}) < f(X_1, X_2)$ for every $j \in \{1, 2, \ldots, n_1\}$, implying that

$$\sum_{j=1}^{n_1} f(Z_{1,j}, \overline{Z_{1,j}}) < n_1 f(X_1, X_2).$$

By the submodularity and non-negativity of f,

$$\sum_{j=1}^{n_1} f(Z_{1,j}, \overline{Z_{1,j}}) \geq (n_1 - 1)f(V) + f(X_2) + \sum_{j=1}^{n_1} f(Z_{1,j}) \geq f(X_2) + \sum_{j=1}^{n_1} f(Z_{1,j}).$$

By these, we obtain

$$f(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1}) - f(X_1) \leq (n_1 - 1)f(X_1, X_2). \tag{27}$$

On the other hand,

$$f(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1}) - f(X_1) \leq f(P_k) - f(X_1, X_2) \tag{28}$$

holds by Lemma 6.

By the submodularity,

$$f(Z_{1,1}, X_1 \setminus Z_{1,1}) \leq f(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1})$$

holds. From this, (27), and (28), it follows that

$$f(Z_{1,1}, X_1 \setminus Z_{1,1}) - f(X_1) \leq \min\{(n_1 - 1)f(X_1, X_2), f(P_k) - f(X_1, X_2)\}$$

$f(X_1, X_2) = f(P_k)/n_1$ attains the maximum value of the right-hand side in the above inequality, which presents the required bound of $f(Z_{1,1}, X_1 \setminus Z_{1,1}) - f(X_1)$. $\square$

Lemma 10. *Suppose that P and P_k in Theorem 10 satisfy (iv), and that shrinking the components of the partition $[C_1, C_2, \ldots, C_{k'}]$ does not preserve P_k. Then the k-partition $[R_1, R_2, \ldots, R_{k-1}, X_2]$ of (V, f) achieves the approximation factor $k + 1 - 2\sqrt{k-1}$.*

Proof. By Lemmas 8 and 9,

$$f(R_1, R_2, \ldots, R_{k-1}, X_2) \leq \left\{ 1 + (k - 1 - n_1) \frac{n_1 - 1}{n_1} \right\} f(P_k). \qquad (29)$$

Recall that $2 \leq n_1 \leq k - 2$. Then $n_1 = \sqrt{k-1}$ attains the maximum $(k + 1 - 2\sqrt{k-1}) f(P_k)$ of the right-hand side in (29). $\square$

Below we describe the entire of our algorithm.

Algorithm MINKPT(k,V,f)

Input: A submodular system (V, f) and an integer $k \geq 5$

Output: A k-partition of (V, f)

Step 1: Initialize the solution S with an arbitrary k-partition of (V, f).

Step 2: If $|V| \leq 4k(k-2)$, then enumerate all k-partitions of V, and terminate with outputting minimum one among them.

Step 3: For every $U \subseteq V$ with $|U| \leq 2k - 4$, compute a $(k-1)$-partition $[V_1, V_2, \ldots, V_{k-1}]$ of $(V \setminus U, f^{V \setminus U})$ by using Algorithm MINKPT($k-1,V \setminus U, f^{V \setminus U}$) and set $S \leftarrow \min\{S, [U, V_1, V_2, \ldots, V_{k-1}]\}$. (This step is for (i).)

Step 4: Compute a minimum $(2k - 3)$-size 2-partition $[X_1, X_2]$ of (V, f) such that $|X_1| \leq |X_2|$.

Step 5: For every $2 \leq i \leq k-2$, execute the following operations: Compute a minimum i-partition $[Q_1, Q_2, \ldots, Q_i]$ of (X_1, f^{X_1}) by using Algorithm MINKPT(i, X_1, f^{X_1}); Compute a minimum $(k-i)$-partition $[Q'_1, Q'_2, \ldots, Q'_{k-i}]$ of (X_2, f^{X_2}) by using Algorithm MINKPT($k-i, X_2, f^{X_2}$); Set $S \leftarrow \min\{S, [Q_1, Q_2, \ldots, Q_i, Q'_1, Q'_2, \ldots, Q'_{k-i}]\}$. (This step is for (iii).)

Step 6: If $|X_1| > 2(k-2)^2$, then set $S \leftarrow \min\{S, \text{MINKPT}(k, V_{X_1}, f_{X_1}), \text{MINKPT}(k, V_{X_2}, f_{X_2})\}$. (This step is for (ii).)

Step 7: If $|X_1| \leq 2(k-2)^2$, then execute following operations: Enumerate 2-partitions of (X_1, f^{X_1}) by the algorithm due to Vazirani and Yannakakis [15] until X_1 is divided into at least $k - 1$ subsets, and let (V', f') be the system obtained by shrinking those subsets respectively; Enumerating all $(k - 1)$-partitions of (X_1, f^{X_1}), and call a minimum one among them by $[R_1, R_2, \ldots, R_{k-1}]$; Set

$$S \leftarrow \min\{S, \text{MINKPT}(k, V_{X_2}, f_{X_2}), \text{MINKPT}(k, V', f'), [R_1, R_2, \ldots, R_{k-1}, X_2]\}.$$

(This step is for (ii) and (iv).)

Step 8: Terminate with outputting S.

Theorem 11. *Algorithm* MINKPT *achieves the approximation factor $k + 1 - 2\sqrt{k-1}$ for the submodular system k-partition problem with $k \geq 5$ by computing minimum (s, t)-partitions $O(n^{k^2 - 2k - 2})$ times.*

Proof. First, let us discuss the approximation factor α_k of the algorithm for the minimum k-partition problem. When $n \leq 4k(k-2)$, Step 2 of the algorithm computes an optimal solution. Below, we examine each case where some condition of Theorem 10 are satisfied by the minimum h-size 2-partition $P = [X_1, X_2]$ and some minimum k-partition $P_k = [Y_1, Y_2, \ldots, Y_k]$ of (V, f), where $h = 2k - 3$. We show that α_k satisfies

$$\alpha_k = \max\{k + 1 - 2\sqrt{k-1}, \alpha_{k-1}, \alpha_i + \alpha_{k-i} \mid 2 \leq i \leq \lfloor k/2 \rfloor\} \tag{30}$$

for $k \geq 5$. Remember that $\alpha_2 = \alpha_3 = 1$ and $\alpha_4 = 1.5$ by Theorems 6 and 8. By the induction on k, we can prove that the maximum in (30) is attained by $k + 1 - 2\sqrt{k-1}$.

Let us consider the case where (i) is satisfied. In this case, a component of P_k is some $U \subseteq V$ with $|U| < h$. The other components form a minimum $(k-1)$-partition of $(V \setminus U, f^{V \setminus U})$ by Theorem 3. Algorithm $\text{MINKPT}(k-1, V \setminus U, f^{V \setminus U})$ approximates this $(k-1)$-partition within a factor of α_{k-1}. Hence Step 3 computes an α_{k-1}-approximate solution in this case.

Next, let us consider the case where (ii) is satisfied. In this case, shrinking X_1 or X_2 preserves an optimal solution. Hence, if $|X_1| > 2(k-2)^2$, then $\text{MINKPT}(k, V_{X_1}, f_{X_1})$ or $\text{MINKPT}(k, V_{X_2}, f_{X_2})$ in Step 6 returns an α_k-approximate solution by the induction hypothesis. Notice that (V', f'), computed in Step 7, is the system obtained by shrinking some vertices in X_1. Therefore if $|X_1| \leq 2(k-2)^2$, then $\text{MINKPT}(k, V_{X_2}, f_{X_2})$ or $\text{MINKPT}(k, V', f')$ in Step 7 returns an α_k-approximate solution by the induction hypothesis.

Next, let us consider the case where (iii) is satisfied. In this case, the k-partition $[Q_1, \ldots, Q_i, Q'_1, \ldots, Q'_{k-i}]$ computed in Step 5 achieves the approximation factor $\alpha_i + \alpha_{k-i}$ if $n_1 \leq i \leq k - n_2$ by Lemma 7.

In the last, let us consider the case where (iv) is satisfied. In this case, $|X_1| \leq n_1(h-1) \leq 2(k-2)^2$, i.e., the algorithm executes the operations described in Step 7. By Lemma 10, either an optimal solution remains in (V', f') constructed in Step 7, or $[R_1, R_2, \ldots, R_{k-1}, X_2]$ achieves approximation factor $k + 1 - 2\sqrt{k-1}$. In the former case, $\text{MINKPT}(k, V', f')$ returns an α_k-approximate solution by the induction hypothesis.

By the above discussion, we have proven the approximation factor of Algorithm MINKPT. In the rest of this proof, we analyze the number of computing the minimum (s, t)-partitions. Let $x = |X_1|$. Since $[X_1, X_2]$ is $(2k-3)$-size, $2k - 3 \leq x \leq n - 2k + 3$. Moreover, $|V_{X_1}| = n - x + 1$, $|V_{X_2}| = x + 1$ and $|V'| \leq n - 1$ hold. Define $D_k(n)$ as the number of (s, t)-partition computations used in the algorithm. We have already observed that all operations but executing $\text{MINKPT}(V_{X_1}, f_{X_1})$, $\text{MINKPT}(V_{X_2}, f_{X_2})$ and $\text{MINKPT}(V', f_{V'})$ can be done by $O(n^{2k-4}D_{k-1}(n) + \sum_{i=2}^{k-2} D_i(n))$ minimum (s, t)-partition computations. Hence if $|X_1| > 2(k-2)^2$, we have

$$D_k(n) \leq D_k(n - x + 1) + D_k(x + 1) + O\left(n^{2k-4}D_{k-1}(n) + \sum_{i=2}^{k-2} D_i(n)\right). \tag{31}$$

If $|X_1| \leq 2(k-2)^2$, we have

$$D_k(n) \leq D_k(n - 1) + D_k(2(k-2)^2 + 1) + O\left(n^{2k-4}D_{k-1}(n) + \sum_{i=2}^{k-2} D_i(n)\right). \tag{32}$$

Remember that $D_2(n) = O(n)$, $D_3(n) = O(n^3)$ (Theorem 6), and $D_4(n) = O(n^6)$ (Theorem 8). It is easy to verify that $D(n) = O(n^{k^2 - 2k - 2})$ satisfies both of (31) and (32). $\quad\square$

6.2 Hypergraph k-cut problem

Let α'_k be the approximation factor of Algorithm MINKPT for the hypergraph k-cut problem. In this subsection, we present a better bound on α'_k than presented in Theorem 11 by improving some of lemmas in the previous subsection.

Lemma 11. *Assume that the submodular system (V, f) is constructed from a hypergraph $G = (V, E)$ with weight $w : E \to \Re_+$ as described in Section 2.2, and that P and P_k in Theorem 10 satisfy (iii). Then a minimum ℓ-partition P^1_ℓ of (X_1, f^{X_1}) and a minimum $(k-\ell)$-partition $P^2_{k-\ell}$ of (X_2, f^{X_2}) with $k - n_2 \leq \ell \leq n_1$ satisfy $f(P^1_\ell) + f(P^2_{k-\ell}) \leq 1.5 f(P_k)$.*

Proof. Assume without loss of generality that $Z_{1,i} = \emptyset$, $i \in \{n_1+1, n_1+2, \ldots, k\}$ and that $Z_{2,i} = \emptyset$, $i \in \{1, 2, \ldots, k-n_2\}$. Since $\delta(Y_1, Y_2, \ldots, Y_k) \supseteq \delta_e(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1}) \cup \delta_e(Z_{2,k-n_2+1}, Z_{2,k-n_2+2}, \ldots, Z_{2,k})$ and $\delta_e(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1}) \cap \delta_e(Z_{2,k-n_2+1}, Z_{2,k-n_2+2}, \ldots, Z_{2,k}) = \emptyset$, it holds that

$$w_e(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1}) + w_e(Z_{2,k-n_2+1}, Z_{2,k-n_2+2}, \ldots, Z_{2,k}) \leq f(P_k),$$

and hence

$$\min\{w_e(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1}), w_e(Z_{2,k-n_2+1}, Z_{2,k-n_2+2}, \ldots, Z_{2,k})\} \leq f(P_k)/2.$$

Assume that $w_e(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1}) \leq f(P_k)/2$. By Lemma 6, $f(X_1, Z_{2,k-n_2+1}, \ldots, Z_{2,k}) \leq f(P_k)$ holds. Hence

$$f(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1}) + f(Z_{2,k-n_2+1}, Z_{2,k-n_2+2}, \ldots, Z_{2,k})$$
$$= w_e(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1}) + f(X_1, Z_{2,k-n_2+1}, Z_{2,k-n_2+2}, \ldots, Z_{2,k}) \leq 1.5 f(P_k).$$

Since $f(P^1_\ell) \leq f(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1})$ and $f(P^2_{k-\ell}) \leq f(Z_{2,k-n_2+1}, Z_{2,k-n_2+2}, \ldots, Z_{2,k})$, we have obtained the required inequality.

Even if $w_e(Z_{2,k-n_2+1}, Z_{2,k-n_2+2}, \ldots, Z_{2,k}) \leq f(P_k)/2$, the lemma is proven similarly. $\square$

The following is an improvement of Lemma 9.

Lemma 12. *Assume that the submodular system (V, f) is constructed from a hypergraph $G = (V, E)$ with weight $w : E \to \Re_+$ as described in Section 2.2, and that P and P_k in Theorem 10 satisfy (iv). Then,*

$$f(Z_{1,1}, X_1 \setminus Z_{1,1}) - f(X_1) \leq \begin{cases} f(P_k)/3 & n_1 = 2, \\ f(P_k)/2 & n_1 \geq 3. \end{cases}$$

Proof. Since $f(Z_{1,1}, X_1 \setminus Z_{1,1}) - f(X_1) = w_e(Z_{1,1}, X_1 \setminus Z_{1,1})$ in this case, it suffices to estimate $w_e(Z_{1,1}, X_1 \setminus Z_{1,1})$.

The submodularity and non-negativity of f, and Lemma 6 imply that

$$f(Z_{1,1}, X_1 \setminus Z_{1,1}, X_2) \leq f(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,n_1}, X_2) \leq f(P_k). \tag{33}$$

Moreover,

$$w_e(Z_{1,1}, X_1 \setminus Z_{1,1}) + \sum_{j=1}^{n_1} w_e(Z_{1,j}, Z_{2,j}) \leq f(Z_{1,1}, X_1 \setminus Z_{1,1}, X_2) \tag{34}$$

holds because $\delta_e(Z_{1,1}, X_1 \setminus Z_{1,1})$ and $\delta_e(Z_{1,j}, Z_{2,j})$, $j \in \{1, 2, \ldots, n_1\}$ are all disjoint and contained in $\delta(Z_{1,1}, X_1 \setminus Z_{1,1}, X_2)$.

21

Suppose that $w_e(Z_{1,1}, X_1 \setminus Z_{1,1}) > \sum_{j=2}^{n_1} w_e(Z_{1,j}, Z_{2,j})$. Then $w(Y_1, Y_2, \ldots, Y_k) > w(X_1 \cup Z_{2,1}, Z_{2,2}, Z_{2,3}, \ldots, Z_{2,k})$ holds since $w(Y_1, Y_2, \ldots, Y_k) - w(X_1 \cup Z_{2,1}, Z_{2,2}, Z_{2,3}, \ldots, Z_{2,k}) > w_e(Z_{1,1}, X_1 \setminus Z_{1,1}) - \sum_{j=2}^{n_1} w_e(Z_{1,j}, Z_{2,j})$. This is a contradiction, implying that

$$w_e(Z_{1,1}, X_1 \setminus Z_{1,1}) \leq \sum_{j=2}^{n_1} w_e(Z_{1,j}, Z_{2,j}). \tag{35}$$

Combining (33), (34), and (35) gives the required inequality for $n_1 \geq 3$.

Let us discuss the case where $n_1 = 2$. By Theorem 3, $[Y_1, Y_2]$ is a minimum 2-partition of $(Y_1 \cup Y_2, f^{Y_1 \cup Y_2})$. If $w_e(Z_{1,1}, Z_{1,2}) > w_e(Z_{1,1}, Z_{2,1})$, then $w_e(Z_{1,1}, Y_2 \cup Z_{1,1}) = w_e(Y_1, Y_2) + w_e(Z_{1,1}, Z_{2,1}) - w_e(Z_{1,1}, Y_2) \leq w_e(Y_1, Y_2) + w_e(Z_{1,1}, Z_{2,1}) - w_e(Z_{1,1}, Z_{1,2}) < w_e(Y_1, Y_2)$. This is a contradiction, implying that $w_e(Z_{1,1}, X_1 \setminus Z_{1,1}) = w_e(Z_{1,1}, Z_{1,2}) \leq w_e(Z_{1,1}, Z_{2,1})$. This inequality, (33), (34), and (35) prove the required inequality for $n_1 = 2$. $\square$

Lemma 13. *Assume that the submodular system (V, f) is constructed from a hypergraph $G = (V, E)$ with weight $w : E \to \Re_+$ as described in Section 2.2, that P and P_k in Theorem 10 satisfy (iv), and that shrinking the components of the partition $[C_1, C_2, \ldots, C_{k'}]$ does not preserve P_k. Then the k-partition $[R_1, R_2, \ldots, R_{k-1}, X_2]$ of (V, f) achieves the approximation factor $\max\{k/3, k/2 - 1\}$.*

Proof. By Lemmas 8 and 12,

$$f(R_1, R_2, \ldots, R_{k-1}, X_2) \leq \begin{cases} \frac{k}{3} f(P_k) & n_1 = 2, \\ \frac{k+1-n_1}{2} f(P_k) & n_1 \geq 3. \end{cases}$$

The right-hand side of this inequality is at most $\max(k/3, k/2 - 1) f(P_k)$. $\square$

Lemmas 13 and 11 improve Theorem 11 as follows.

Theorem 12. *For the hypergraph k-cut problem, Algorithm MINKPT achieves the approximation factors $5/3$ for $k = 5$ and $k/2 - 1$ for $k \geq 6$.*

Proof. Let α_k' be the approximation factor achieved by Algorithm MINKPT for the hypergraph k-cut problem. Then by Lemmas 13 and 11, we can improve the equality (30) in the proof of Theorem 11 as follows:

$$\alpha_k' = \max\left\{k/3, k/2 - 1, \alpha_{k-1}', \min(\alpha_i' + \alpha_{k-i}', 1.5\alpha_{k-i}') \mid 2 \leq i \leq \lfloor k/2 \rfloor\right\}.$$

Remember that $\alpha_2' = \alpha_3' = 1$ and $\alpha_4' = 4/3$ by Theorems 6 and 8. By the induction on k, we can prove that the right-hand side of this equality is $5/3$ for $k = 5$ and $k/2 - 1$ for $k \geq 6$. $\square$

7 Concluding remarks

In this paper, we have shown that the hypergraph k-cut problem is contained by the submodular k-partition problem and can be reduced to the terminal k-vertex cut problem in bipartite graphs. In addition, we have presented algorithms for the submodular k-partition problem. For $k = 3$, our algorithm is the first exact algorithm for this problem. For $k \geq 4$, our algorithms achieve the better approximation factors than the previous algorithms. We have also discuss their approximation factors for the hypergraph k-cut problem.

In spite of the progress made by this paper, it remains open whether the hypergraph k-cut problem and the submodular system k-partition problem are polynomial-time solvable or NP-hard for fixed $k \geq 4$. This is a challenging future work.

Acknowledgements

References

[1] A. Frank. "Applications of submodular functions," in *Surveys in Combinatorics, Cambridge London Mathematical Society Lecture Notes Series* 187 (1993) 85-136.

[2] S. Fujishige. Submodular Function and Optimization, *North-Holland, Amsterdam* (1991).

[3] N. Garg, V. V. Vazirani, M. Yannakakis. "Multiway cuts in node weighted graphs," Journal of Algorithms 50 (2004) 49–61.

[4] L. Gasieniec, J. Jansson, A. Lingas, A. Óstlin. "On the complexity of constructing evolutionary trees," Journal of Combinatorial Optimization 3 (1999) 183–197.

[5] A. V. Goldberg, R. E. Tarjan. "A new approach to the maximum flow problem," Journal of the ACM 35 (1988) 921–940.

[6] O. Goldschmidt, D. Hochbaum. "A polynomial algorithm for the k-cut problem for fixed k," Mathematics of Operations Research 19 (1994), 24–37.

[7] S. Iwata. "Submodular function minimization," Mathematical Programming 112 (2008), 45–64.

[8] Y. Kamidoi, N. Yoshida, H. Nagamochi. "A deterministic algorithm for finding all minimum k-way cuts," SIAM Journal on Computing 36 (2006) 1329–1341.

[9] D. R. Karger, C. Stein. "A new approach to the minimum cut problem," Journal of the ACM 43 (1996) 601–640.

[10] E. L. Lawler. "Cutsets and partitions of hypergraphs," Networks 3 (1973) 275–285.

[11] H. Nagamochi. "Algorithms for the minimum partitioning problems in graphs," IEICE Transactions on Information and Systems J86-D-1 (2003) 53–68.

[12] M. Queyranne. "A combinatorial algorithm for minimizing symmetric submodular functions," Mathematical Programming 82 (1998) 3–12.

[13] M. Queyranne. "On optimum size-constrained set partitions," in *Proceedings of AUSSOIS* (1999).

[14] M. Thorup. "Minimum k-way cuts via deterministic greedy tree packing" in *Proceedings of the 40th Annual ACM Symposium on Theory of Computing* (2008) 159–166.

[15] V. V. Vazirani, M. Yannakakis. "Suboptimal cuts: Their enumeration, weight and number," in *Automata, Languages and Programming: Proceedings. of the 19th International Colloquium, Lecture Notes in Computer Science 623* (1992) 366-377.

[16] M. Xiao. "Finding minimum 3-way cuts in hypergraphs," in *Theory and Applications of Models of Computation: Proceedings of the 5th International Conference, Lecture Notes in Computer Science 4978* (2008) 270–281.

[17] M. Xiao. "An Improved divide-and-conquer algorithm for finding all minimum k-way cuts," in *Algorithms and Computation: Proceedings of the 19th International Symposium, Lecture Notes in Computer Science 5369* (2008) 208–219.

[18] L. Zhao, H. Nagamochi, T. Ibaraki. "A unified framework for approximating multiway partition problems," in *Proceedings of the 12th International Symposium on Algorithms and Computation, Lecture Notes in Computer Science 2223* (2001) 682–694.