Minimum Degree Orderings

HIROSHI NAGAMOCHI*

Department of Applied Mathematics and
Physics
Graduate School of Informatics
Kyoto University
nag@amp.i.kyoto-u.ac.jp

Abstract: It is known that, given an edge-weighted graph, a maximum adjacency ordering (MA ordering) of vertices can find a special pair of vertices, called a *pendent pair*, and that a minimum cut in a graph can be found by repeatedly contracting a pendent pair, yielding one of the fastest and simplest minimum cut algorithms. In this paper, we provide another ordering of vertices, called a minimum degree ordering (MD ordering) as a new fundamental tool to analyze the structure of graphs. We prove that an MD ordering finds a different type of special pair of vertices, called a *flat pair*, which actually can be obtained as the last two vertices after removing a vertex with the minimum degree repeatedly. By contracting flat pairs, we can find not only a minimum cut but also all extreme subsets of a given graph. These results can be extended to the problem of finding extreme subsets in symmetric submodular set functions.

1 Introduction

Let $\Re$ and $\Re_+$ denote the sets of reals and nonnegative reals, respectively. Let V be a finite set, where we denote $n = |V|$. A singleton set $\{v\}$ is called *trivial* and may be written as v. Let $(G = (V, E), w)$ be a graph with vertex set V, edge set E and weight function $w : E \mapsto \Re_+$. The *cut function* of (G, w) is defined by set function $d_{(G,w)} : 2^V \mapsto \Re^+$ such that $d_{(G,w)}(X) = \sum\{w(e) \mid e \in E,\ e \cap X \neq \emptyset \neq e - X\}$. For two specified vertices u and v, let $\lambda_{(G,w)}(u, v)$ denote the local-edge-connectivity $\min\{d_{(G,w)}(X) \mid u \in X \subseteq V - v\}$.

Analyzing the connectivity structure of a given graph is an important research issue, and several compact representations of connectivity structure of graphs such as Gomory-Hu trees [6], cactus representations [3] and extreme subsets [20] have been discovered. These representations have numerous applications to design of efficient graphs algorithms (see [9]).

Computing a minimum cut X, i.e., a nonempty subset $X \subset V$ that minimizes $d_{(G,w)}$, is one of the basic problems in the issue, and has been studied extensively (see [2]). In particular, for undirected graphs, several algorithms that compute a minimum cut without relying on a maximum flow algorithm are known so far. One of such algorithms is based on a structural feature of graphs. A pair of vertices u and v is called a *pendent pair* if it satisfies

$$d_{(G,w)}(X) \geq \min\{d_{(G,w)}(u), d_{(G,w)}(v)\} \text{ for all } X \subseteq V \text{ with } |X \cap \{u, v\}| = 1. \tag{1}$$

i.e., $\lambda_{(G,w)}(u, v) = \min\{d_{(G,w)}(u), d_{(G,w)}(v)\}$. The existence of pendent pairs is implied by Gomory-Hu trees that represent the structure of all local-edge-connectivities [6]. An edge-weighted spanning tree $(T = (V, F), w')$ is called a *Gomory-Hu tree* of (G, w) if (i) $\lambda_{(T,w')}(x, y) = \lambda_{(G,w)}(x, y)$ holds for all $x, y \in V$ and (ii) for each edge $e = \{u, v\} \in F$, the two components $T_1 = (V_1, F_1)$ and $T_2 = (V_2, F_2)$ in $(V, F - e)$ satisfy $d_{(G,w)}(V_1) = d_{(G,w)}(V_2) = \lambda_{(G,w)}(u, v)$. For example, Figure 1(b) shows a Gomory-Hu tree for the graph $G = (V, E)$ in Fig. 1(a). By

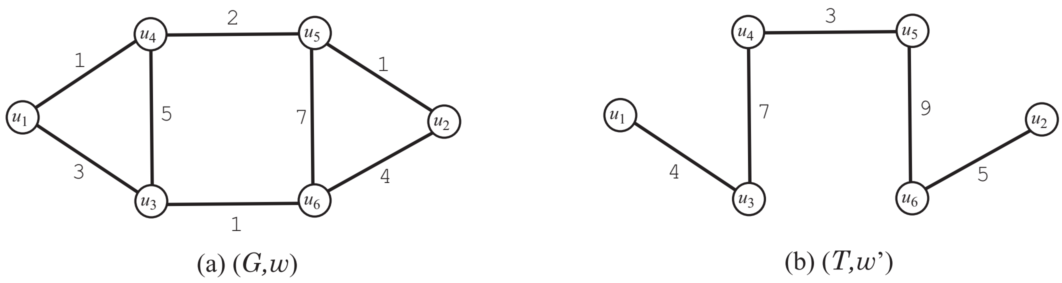


Figure 1: (a) An edge-weighted graph $(G = (V, E), w)$ and (b) A Gomory-Hu tree $(T = (V, F), w')$ of the graph (G, w), where the numbers beside edges indicate their weights.

definition of Gomory-Hu trees, every leaf vertex u and its unique neighbour v in (T, w') give a pendent pair. In the graph (G, w) in Figure 1(a), $\{u_3, v_1\}$ and $\{u_6, u_2\}$ are pendent pairs.

Interestingly a pendent pair can be found the following simple procedure. An ordering $\sigma = (v_1, v_2, \ldots, v_n)$ of the vertices in an edge-weighted graph (G, w) is called a *maximum adjacency ordering* (MA ordering, for short) if it satisfies

$$d_{(G,w)}(\{v_1, v_2, \ldots, v_{i-1}\}, v_i) \geq d_{(G,w)}(\{v_1, v_2, \ldots, v_{i-1}\}, v_j), \quad 2 \leq i \leq j \leq n, \tag{2}$$

where $d_{(G,w)}(X, Y)$ denotes $\sum\{w(e) \mid e = \{x, y\} \in E,\ x \in X,\ y \in Y\}$. It is shown that an MA ordering identifies a pendent pair of (G, w) as its last two vertices; i.e., it holds

$$d_{(G,w)}(v_n) = \lambda_{(G,w)}(v_{n-1}, v_n). \tag{3}$$

For example, $\sigma = (u_1, u_3, u_4, u_5, u_6, u_2)$ is an MA ordering of the graph (G, w) in Figure 1(a), indicating that $\{u_6, u_2\}$ is a pendent pair.

Based on property (3), one can design an algorithm to compute a minimum cut in a graph (G, w) by repeating to identify a pendent pair and contract the pair into a single vertex (see [12, 19]). This yields one of the fastest and simplest minimum cut algorithms, which runs in $O(nm + n^2 \log n)$ time, where $m = \sum_{e \in E} |e|$. However, no algorithm that constructs a Gomory-Hu tree of a given graph (G, w) by using MA orderings is known. Importantly Queyranne [16] extended the minimum cut algorithm for graphs to a combinatorial strongly polynomial algorithm for minimizing a *symmetric* submodular set function. Recently combinatorial strongly polynomial algorithms for minimizing general submodular set functions have been obtained by Iwata et al. [5] and Schrijver [18]. However, for minimizing symmetric submodular set functions, Queyranne's algorithm remains significantly simpler than these algorithms.

In this paper, we prove that a different structural feature of graphs can be used to design a simple and efficient connectivity algorithm. We define a new type of special pair of vertices, called a "flat pair," based on "extreme subsets" of graphs. We then introduce another ordering of vertices, called "a minimum degree ordering" (MD ordering) to identify a flat pair. A nonempty proper subset X of V is called an *extreme subset* of a graph $(G = (V, E), w)$ if

$$d_{(G,w)}(Y) > d_{(G,w)}(X) \text{ for all nonempty proper subsets } Y \text{ of } X. \tag{4}$$

We denote by $\mathcal{X}(G, w)$ the family of all extreme subsets of (G, w). Any singleton $\{v\}$, $v \in V$ is an extreme subset. Note that at least one of extreme subsets corresponds to a minimum cut, and no two extreme subsets intersect each other. Figure 2(a) shows the extreme subsets of the graph (G, w) in Fig. 1(a), and Figure 2(b) shows its tree representation that indicates the inclusionwise relation among the extreme subsets.

Extreme subsets are originally introduced by Watanabe and Nakamura [20] to solve the edge-connectivity augmentation problem, and extreme subsets of graphs are currently an important

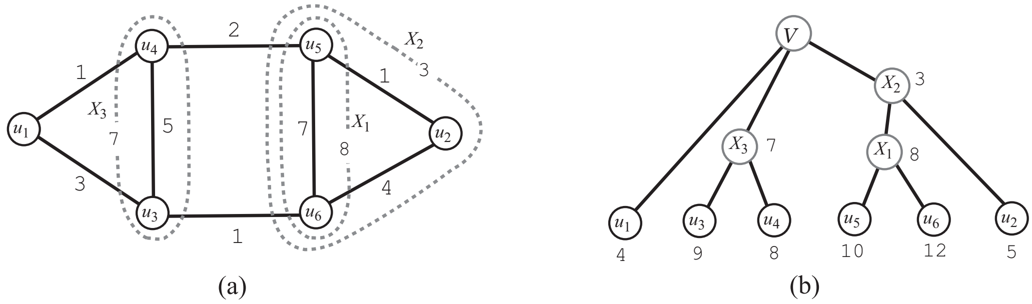


Figure 2: (a) The extreme subsets in $\mathcal{X}(G,w)$ of the graph (G,w) in Fig. 1(a), where the numbers beside edges indicate their weights and each of the nontrivial extreme subsets $X_1, X_2, X_3 \in \mathcal{X}(G,w)$ is depicted by a dotted closed curve; (a) The tree representation for $\mathcal{X}(G,w)$.

tool to design efficient algorithms for solving graph connectivity problems such as the source location problem [9, 17], the minimum k-way cut problem [15], and the dynamic minimum cut problem [10].

In this paper, we call a pair $\{u,v\}$ in a graph (G,w), of vertices u and v a *flat pair* if it satisfies

$$d_{(G,w)}(X) \geq \min\{d_{(G,w)}(x) \mid x \in X\} \text{ for all } X \subseteq V \text{ with } |X \cap \{u,v\}| = 1. \tag{5}$$

We will prove that such a pair always exists. Observe that no nontrivial extreme subset X separates a flat pair $\{u,v\}$; i.e., $\{u,v\} \subseteq X$ or $\{u,v\} \cap X = \emptyset$ holds for any nontrivial extreme set X. A flat pair must correspond to two leaves u and v with the same parent in the tree representation of extreme subsets. In Figure 2(a), $\{u_3,u_4\}$ and $\{u_5,u_6\}$ are flat pairs.

In this paper, we call an ordering $\pi = (v_1, v_2, \ldots, v_n)$ of the vertices in a graph (G,w) a *minimum degree ordering* (MD ordering, for short) if it satisfies

$$d_{(G_{i-1},w)}(v_i) = \min\{d_{(G_{i-1},w)}(v) \mid v \in V - \{v_1, v_2, \ldots, v_{i-1}\}\} \quad 1 \leq i \leq j \leq n-1, \tag{6}$$

where $G_{i-1} = G - \{v_1, v_2, \ldots, v_{i-1}\}$ denotes the graph obtained from G by removing vertices $v_1, v_2, \ldots, v_{i-1}$ together with all edges incident to them. Thus, an MD ordering can be easily obtained just by removing a vertex with the minimum degree in the remaining graph. For the graph in Figure 2(a), $\pi = (u_1, u_2, u_3, u_4, u_5, u_6)$ is an MD ordering.

Interestingly the following fact holds: the last two vertices in an MD ordering gives a flat pair. We prove this, and then show that all extreme subsets of a graph (G,w) can be computed by using flat pairs in $O(mn + n^2 \log n)$ time. It is already known [9] that all extreme subsets in a graph can be computed in $O(mn + n^2 \log n)$ time by applying an MA ordering after augmenting the given graph with a new vertex and edges. However, the augmenting process is rather artificial, and no direct extension of this algorithm to the case of submodular set functions has been successful.

Orderings of (6) in a special case where G is an unweighted simple graph are known as δ-slicings [7] or smallest-last orderings [8] which are introduced to study the structure of induced subgraphs. However, the fact that MD orderings identify flat pairs was not known, and extensions to hypergraphs or set functions are not trivial (see Appendix for a computing process in the case of hypergraphs).

In this paper, we define flat pairs and MD orderings in terms of set functions on V, and prove that extreme subsets of symmetric submodular set functions can be computed with the same time complexity of Queyranne's algorithm. Since one of the extreme subsets minimizes the set function, our new algorithm also solves the minimization problem for symmetric submodular set functions.

The rest of the paper is organized as follows. Section 2 introduces basic notions on submodular or posi-modular set functions, and states our main result in terms of set functions. Section 3 then proves that the last two elements in an MD ordering in a symmetric and crossing submodular or intersecting posi-modular set function is a flat pair. Section 4 gives an algorithm that computes all extreme subsets in such a set function. Section 5 concludes.

2 Preliminaries

Let V be a finite set, where we denote $n = |V|$. A singleton set $\{v\}$ is called *trivial* and may be written as v. The union of a set X and a singleton $\{v\}$ may be written as $X + v$. A subset $X \subseteq V$ *separates* two elements $u, v \in V$ if $|X \cap \{u, v\}| = 1$. For two subsets $X, Y \subseteq V$, we say that X and Y *intersect* each other if $X \cap Y \neq \emptyset$, $X - Y \neq \emptyset$ and $Y - X \neq \emptyset$ hold, and say that X and Y *cross* each other if, in addition, $V - (X \cup Y) \neq \emptyset$ holds. A family $\mathcal{X} \subseteq 2^V$ of subsets of V is called *laminar* if no two subsets in $\mathcal{X}$ intersect each other.

A *set function* f on V is a function $f : 2^V \to \Re$. A set function f is called *fully* (resp., *intersecting, crossing*) *submodular* if

$$f(X) + f(Y) \geq f(X \cap Y) + f(X \cup Y) \tag{7}$$

holds for every (resp., intersecting, crossing) pair of sets $X, Y \subseteq V$.A set function f is called *fully* (resp., *intersecting, crossing*) *posi-modular* if

$$f(X) + f(Y) \geq f(X - Y) + f(Y - X) \tag{8}$$

holds for every (resp., intersecting, crossing) pair of sets $X, Y \subseteq V$ [13]. Notice that the class of crossing submodular set functions is wider than that of fully submodular set functions. A set function f is called *symmetric* if

$$f(X) = f(V - X) \text{ for all } X \subseteq V. \tag{9}$$

Every symmetric and fully (resp., intersecting, crossing) submodular set function is fully (resp., intersecting, crossing) posi-modular.

Given a set function f on V the set function f' obtained by *contracting* two elements $x, y \in V$ into a new element z is defined by $V' = (V - \{x, y\}) \cup \{z\}$ and

$$f'(X) = \begin{cases} f(X) & \text{if } z \notin X \subseteq V' \\ f((X - z) \cup \{x, y\}) & \text{if } z \in X \subseteq V'. \end{cases}$$

A nonempty proper subset X of V is called an *extreme subset* of f if

$$f(Y) > f(X) \text{ for all nonempty proper subsets } Y \text{ of } X.$$

We denote by $\mathcal{X}(f)$ the family of all extreme subsets of a set function f. Any trivial set $\{v\}$, $v \in V$ is an extreme subset. By definition, any nonempty subset X contains an extreme subset X' with $f(X') \leq f(X)$. In particular, $\mathcal{X}(f)$ contains a *minimizer* of f, i.e., a subset X with $f(X) = \min_{Y \in 2^V - \{\emptyset, V\}} f(Y)$. The family of extreme subsets of a set function becomes laminar only when it is intersecting posi-modular or symmetric and crossing submodular in the following sense.

Lemma 1 *Let f be a set function on a finite set V, and $\mathcal{X}(f)$ be the family of extreme subsets of f.*

(i) *If f is intersecting posi-modular or symmetric and crossing submodular, then $\mathcal{X}(f)$ is laminar.*

(ii) *There is a crossing posi-modular set function f such that $\mathcal{X}(f)$ is not laminar.*

(iii) *There is an asymmetric and fully submodular set function f such that $\mathcal{X}(f)$ is not laminar.*

PROOF: (i) Recall that a symmetric and crossing submodular set function is crossing posi-modular. Let $X, Y \in \mathcal{X}(f)$ be two intersecting extreme subsets. By definition we would have $f(X) + f(Y) < f(X-Y) + f(Y-X)$, which is a contradiction if f is intersecting posi-modular or X and Y cross each other. Hence we only have to consider the case where f is symmetric and X and Y do not cross each other. In this case, we have $f(X-Y) = f(V-Y) = f(Y)$ and $f(Y-X) = f(V-X) = f(X)$, which again contradicts the above inequality.

(ii) Let $V = \{v_1, v_2, v_3\}$, $f(X) = 1$ if $|X| = 2$ and $f(X) = 2$ otherwise. Since $|V| = 3$, there is no crossing pair, and f is considered as an intersecting posi-modular. However, $\{X \mid |X| = 2,\ X \subseteq V\} \subseteq \mathcal{X}(f)$ holds, and $\mathcal{X}(f)$ is not laminar.

(iii) Let $(D = (V, A), w)$ be a digraph such that vertex set $V = \{v_1, v_2, v_3, v_4\}$ and arc set $A = \{a_1 = (v_1, v_3), a_2 = (v_1, v_4), a_3 = (v_2, v_3), a_4 = (v_2, v_4), a_5 = (v_3, v_4), a_6 = (v_4, v_3)\}$, and $d^+ : 2^V \mapsto \Re_+$ be the cut function such that $d^+(X)$ denotes the number of arcs outgoing from X to $V - X$. The set function d^+ is known to be fully submodular. However, $\{v_1, v_3, v_4\}, \{v_2, v_3, v_4\} \in \mathcal{X}(f)$ holds, and $\mathcal{X}(f)$ is not laminar. ∎

For a set function f on a set V, let T_f denote the time to evaluate the function value $f(X)$ of a given subset $X \subseteq V$. In this paper, we prove the next result, which also solves the minimization of f since $\mathcal{X}(f)$ contains a minimizer of f.

Theorem 2 *Let f be a set function on V with $n = |V| \geq 2$. If f is symmetric and crossing submodular or intersecting submodular and posi-modular, then the family $\mathcal{X}(f)$ of extreme subsets of f can be found in $O(n^3 T_f)$ time.* ∎

An important example of symmetric and fully submodular functions is the cut functions of hypergraphs. Let $(G = (V, E), w)$ be a hypergraph with vertex set V, hyperedge set E ($\subseteq 2^V - (\{\emptyset\} \cup \{\{v\} \mid v \in V\})$) and weight function $w : E \mapsto \Re_+$. The *cut function* of (G, w) is defined by $d_{(G,w)} : 2^V \mapsto \Re^+$ such that

$$d_{(G,w)}(X) = \sum \{w(e) \mid e \in E,\ e \cap X \neq \emptyset \neq e - X\}, \tag{10}$$

where we let $d_{(G,w)}(\emptyset) = d_{(G,w)}(V) = 0$. We see that $d_{(G,w)}$ is symmetric and fully submodular. Figure 2(a) illustrates all extreme subsets in $\mathcal{X}(d_{(G,w)})$ for the cut function $d_{(G,w)}$ of an edge-weighted graph $(G = (V, E), w)$.

3 Minimum Degree Orderings

To show Theorem 2, this section will introduce a new ordering of V for set functions f. Before showing this, we first review a related ordering, called a maximum adjacency ordering. A pair of elements $u, v \in V$ is called a *pendent pair* of f if

$$f(X) \geq \min\{f(u), f(v)\} \text{ for all subsets } X \text{ that separate } u \text{ and } v.$$

Given a set function f on V with $n = |V| \geq 2$, an ordering $\sigma = (v_1, v_2, \ldots, v_n)$ is a *maximum adjacency ordering* (MA ordering, for short) of V if it satisfies

$$f(v_i) - f(V_{i-1} + v_i) \geq f(v_j) - f(V_{i-1} + v_j), \quad 1 \leq i \leq j \leq n, \tag{11}$$

where $V_0 = \emptyset$ and $V_i = \{v_1, v_2, \ldots, v_i\}$ $(1 \leq i \leq n-1)$.

Queyranne [16] obtained the following result.

Theorem 3 [16] *For a given symmetric and crossing submodular function f on V with $n = |V| \geq 2$, let $\sigma = (v_1, v_2, \ldots, v_n)$, be an MA ordering of V. Then the last two elements v_{n-1} and v_n give a pendent pair.* ∎

Based on this, the following results are known.

Theorem 4 [16] *For a given symmetric and crossing submodular set function f on V with $n = |V| \geq 2$, a set $X \in 2^V - \{\emptyset, V\}$ that minimizes f can be found in $O(n^3 T_f)$ time.* ∎

Theorem 5 [13] *For a given intersecting submodular and posi-modular set function f on V with $n = |V| \geq 2$, a set $X \in 2^V - \{\emptyset, V\}$ that minimizes f can be found in $O(n^3 T_f)$ time.* ∎

In this paper, we introduce a new pair and a new ordering of V for set functions f. We call a pair of elements $u, v \in V$ a *flat pair* of f if

$$f(X) \geq \min_{x \in X} f(x) \text{ for all subsets } X \text{ that separate } u \text{ and } v. \tag{12}$$

Given a set function f on V with $n = |V| \geq 2$, we call an ordering $\pi = (v_1, v_2, \ldots, v_n)$ a *minimum degree ordering* (MD ordering, for short) of V if it satisfies

$$f(v_i) + f(V_{i-1} + v_i) \leq f(v_j) + f(V_{i-1} + v_j), \quad 1 \leq i \leq j \leq n, \tag{13}$$

where $V_0 = \emptyset$ and $V_i = \{v_1, v_2, \ldots, v_i\}$ $(1 \leq i \leq n-1)$. It is not difficult to see that, after choosing V_{i-1}, the next element v_i can be chosen from $V - V_i$ by evaluating $f(v) + f(V_{i-1} + v)$ for all $v \in V - V_{i-1}$ and that an MD ordering can be found in $O(n^2 T_f)$ time.

We here consider the time complexity for computing an MD ordering of the cut function $d_{(G,w)}$ of (10) in an edge-weighted hypergraph $(G = (V, E), w)$. For this, we define induced hypergraphs as follows. For a subset $X \subseteq V$, the hypergraph $G\langle X \rangle$ induced by X is defined to be an edge-weighted hypergraph $(X, A_X \cup B_X)$ with an edge weight function $w_X : E \to \Re_+$ such that

$$\begin{aligned} A_X &= \{e \in E \mid e \subseteq X\}, \\ B_X &= \{e - X \mid e \in E, e - X \neq \emptyset, |e \cap X| \geq 2\}, \\ w_X(e) &= \begin{cases} w(e) & \text{if } e \in A_X \\ w(e)/2 & \text{if } e \in B_X. \end{cases} \end{aligned}$$

Note that if G contains only graph edges (i.e., $|e| = 2$, $e \in E$) then $B_X = \emptyset$.

Lemma 6 *For an edge-weighted hypergraph $(G = (V, E), w)$, an ordering $\pi = (v_1, v_2, \ldots, v_n)$ such that*

$$d_{(G\langle V - V_{i-1} \rangle, w_{V - V_{i-1}})}(v_i) = \min\{d_{(G\langle V - V_{i-1} \rangle, w_{V - V_{i-1}})}(v) \mid v \in V - V_{i-1}\}, \; i = 1, 2, \ldots, n-1 \tag{14}$$

is an MD ordering of the cut function d of G. An MD ordering π of d can be found in $O(m + n \log n)$ time and $O(m + n)$ space, where $n = |V|$ and $m = \sum_{e \in E} |e|$.

Proof: See Appendix. ∎

An example of computing process of MD orderings in an edge-weighted hypergraph is presented in Appendix.

As an analogous result to Theorem 3, we show that there exists in a flat pair in a symmetric and crossing submodular function f and that it can be found by an MD ordering of f.

Theorem 7 *For a symmetric and crossing submodular set function f on V with $n = |V| \geq 2$, let $\pi = (v_1, v_2, \ldots, v_n)$ be an MD ordering of V. Then the last two vertices v_{n-1} and v_n give a flat pair.* ∎

We prove Theorem 7 after showing a lemma.

Lemma 8 *Let f be a symmetric and crossing submodular set function on V. For a subset $Z \subset V$, let g be a set function on $V - Z$ such that $g(X) = f(X) + f(Z \cup X)$, $X \subseteq V - Z$. Then g is symmetric and crossing submodular.*

Proof: Let X be an arbitrary subset of $V - Z$. We show $g(X) = g((V-Z)-X))$. By definition of g, we have $g((V-Z)-X)) = f((V-Z)-X)) + f(Z \cup ((V-Z)-X)) = f(V-(Z \cup X)) + f(V-X)$, which is $f(Z \cup X) + f(X) = g(X)$ by symmetry of f. Hence g is symmetric. For two crossing subsets $X, Y \subseteq V - Z$, we have by the submodularity of f

$$\begin{aligned} g(X) + g(Y) &= f(X) + f(Y) + f(Z \cup X) + f(Z \cup Y) \\ &\geq f(X \cap Y) + f(X \cup Y) + f(Z \cup (X \cap Y)) + f(Z \cup X \cup Y) \\ &\geq g(X \cap Y) + g(X \cup Y). \end{aligned}$$

Therefore g is crossing submodular. ∎

Proof of Theorem 7 For each $i = 0, 1, \ldots, n-2$, we define set function f_i of $V - X_i$ by

$$f_i(X) = f(X) + f(V_i \cup X),\ X \subseteq V - V_i,$$

which is symmetric and crossing submodular on $V - V_i$ by Lemma 8. By induction on $i = n-2, n-3, \ldots, 1, 0$, we prove that

$$f_i(X) \geq \min_{x \in X} f_i(x) \text{ for all } X \subseteq V - V_i \text{ that separate } v_{n-1} \text{ and } v_n. \tag{15}$$

Since $f_0(X) = 2f(X)$, it suffices to show that (15) holds for $i = 0$. We easily see that (15) holds for $i = n - 2$. Now we assume that (15) holds for $i = j$. We prove that (15) holds for $i = j - 1$. Let X be an arbitrary subset of $V - V_{j-1}$ that separates v_{n-1} and v_n.

Case-1. $v_j \notin X$: Let $x^* = \operatorname{argmin}\{f_{j-1}(x) \mid x \in X\}$. For two crossing sets $V_{j-1} \cup X$ and $V_j + x^*$, we have by submodularity of f

$$f(V_{j-1} \cup X) + f(V_j + x^*) \geq f(V_j \cup X) + f(V_{j-1} + x^*).$$

From this and induction hypothesis $f_j(X) \geq f_j(x^*)$, we have

$$\begin{aligned} f_{j-1}(X) - f_{j-1}(x^*) &= f(X) + f(V_{j-1} \cup X) - f(V_{j-1} + x^*) - f(x^*) \\ &\geq f(X) + f(V_j \cup X) - f(V_j + x^*) - f(x^*) \\ &= f_j(X) - f_j(x^*) \geq 0. \end{aligned}$$

Hence $f_{j-1}(X) \geq f_{j-1}(x^*) \geq \min_{x \in X} f_{j-1}(x)$.

Case-2. $v_j \in X$: By the choice of v_j, $f_{j-1}(v_j) = \min_{x \in V - V_j} f_{j-1}(x)$. Consider subset $Y = (V - V_j) - X$, which separates v_{n-1} and v_n, and hence $f_{j-1}(Y) \geq \min_{y \in Y} f_{j-1}(y)$ holds by the argument in Case-1. Since $f_{j-1}(Y) = f_{j-1}(X)$ holds by symmetry, we have $f_{j-1}(X) = f_{j-1}(Y) \geq \min_{y \in Y} f_{j-1}(y) \geq f_{j-1}(v_j) = \min_{x \in X} f_{j-1}(x)$, as required.

Therefore, (15) holds for $i = j$. This proves that (15) holds for $i = 0$, i.e., Theorem 7. ∎

Corollary 9 *Let V be a finite set $n = |V| \geq 2$. Every symmetric and crossing submodular function f on V such that $f(v) = k$, $v \in V$ for some $k \in \Re$ admits a pair $\{u, v\} \subseteq V$ that is pendent and flat at the same time.*

PROOF: We easily see that any MD ordering π of f is also an MA ordering if $f(v) = k$, $v \in V$. Hence the last two vertices in π is pendent and flat by Theorems 3 and 7. ∎

There is a symmetric and crossing submodular function f which has no pair that is pendent and flat at the same time. For example, the cut function $d_{(G,w)}$ in Fig. 2(a) has no such pairs, since $\{u_3, u_4\}$ and $\{u_5, u_6\}$ are the flat pairs of $d_{(G,w)}$, but neither of them is pendent.

Corollary 10 *Let f be a set function f on V with $n = |V| \geq 2$. If f is symmetric and crossing submodular or intersecting submodular and posi-modular, then a flat pair of f can be found in $O(n^2 T_f)$ time.*

PROOF: If f is symmetric and crossing submodular, then we compute an MD ordering π of f in $O(n^2 T_f)$ time and choose the pair of the last two elements in π, which is flat by Theorem 7. Consider the case where f is intersecting submodular and posi-modular, where we assume $f(\emptyset) = f(V) = -\infty$ as it does not lose the intersecting submodularity and posi-modularity of f. In this case, we work with the following set function $g : 2^{V+s} \mapsto \Re \cup \{-\infty\}$ (where s is a new element): For each $X \subseteq V + s$, let

$$g(X) = \begin{cases} f(X) & \text{if } s \notin X \\ f(V-(X-s)) & \text{if } s \in X. \end{cases} \tag{16}$$

It is known [13] that, for an intersecting submodular and posi-modular set function f on V, the above set function g is symmetric and crossing submodular on $V + s$. Let π_g be an MD ordering of g, where the first element in π_g must be s since $g(s) = f(V) = +\infty$. Then the last two elements $u, v \in V$ in π_g give a flat pair of g. We see that $\{u, v\}$ is also flat in f, since any subset $X \subseteq V$ with $f(X) < \min_{x \in X} f(x)$ would imply $g(X) < \min_{x \in X} g(x)$, contradicting that $\{u, v\}$ is flat in g. Therefore, we can find a flat pair in $O(n^2 T_f)$ time even if f is intersecting submodular and posi-modular. ∎

4 Computing Extreme Subsets

This section presents an algorithm for computing all extreme subsets of a set function f by using flat pairs. For any nonempty subset $Y \subseteq V$, there is an extreme subset $Y^* \in \mathcal{X}(f)$ with $Y^* \subseteq Y$ and $f(Y^*) \leq f(Y)$. Hence we see that $f(Y) > f(X)$ for all nonempty $Y \subset X$ if and only if $f(Z) > f(X)$ for all $Z \subset X$ with $Z \in \mathcal{X}(f)$. From this observation and the fact that no nontrivial extreme subset $X \in \mathcal{X}(f)$ separates any flat pair, we obtain the following algorithm for computing all extreme subsets of a set function f that admits flat pairs.

After initializing by $\mathcal{X} := \{\{v\} \mid v \in V\}$ $(\subseteq \mathcal{X}(f))$, we repeat a procedure of contracting a flat pair $n-2$ times. Let V^i, $i = n, n-1, \ldots, 2$, be the set of elements obtained after contracting the first $n - i$ flat pairs, where $|V^i| = i$ holds. For each element $x \in V^i$, let $V[x] \subseteq V$ denote the set of all elements that have been contracted into x. We maintain the property that

$$\mathcal{X} \text{ consists of all extreme subsets } X \in \mathcal{X}(f) \text{ with } X \subseteq V[x] \text{ and } x \in V^i. \tag{17}$$

After contracting a flat pair $u^i, v^i \in V^i$ into a single element z^i, we add $V[z^i]$ to $\mathcal{X}$ if $V[z^i] \in \mathcal{X}(f)$, so that (17) holds in the resulting set $V^{i-1} = (V^i - \{u^i, v^i\}) \cup \{z^i\}$ of elements. We can test whether $V[z^i] \in \mathcal{X}(f)$ or not by checking if $f(V[z^i]) < \min_{Y \in \mathcal{X}: Y \subset V[z^i]} f(Y)$. To facilitate this test, we also maintain $\mu(x) = \min_{Y \in \mathcal{X}: Y \subset V[x]} f(Y)$, $x \in V^i$ for each $i = n, n-1, \ldots, 2$. The entire algorithm is described as follows.

Algorithm ExtremeSubsets
Input: A set function f on a finite set V, where $n = |V| \geq 2$.
Output: A laminar family $\mathcal{X} \subseteq 2^V - \{\emptyset, V\}$ of extreme subsets of f.
1 $\mathcal{X} := \{\{v\} \mid v \in V\}$;
2 Let $\mu(v) := f(v)$ for all $v \in V$;
3 $V^n := V$; $f^n := f$;
4 **for** $i := n$ to 3 **do**
5 Find a flat pair $\{u^i, v^i\}$ of f^i;
6 $V^{i-1} := (V^i - \{u^i, v^i\}) \cup \{z^i\}$;
7 Let f^{i-1} be the set function on V^{i-1} obtained from f^i by contracting elements u^i and v^i into a single element z^i;
8 Let $V[z^i] \subseteq V$ be the set of all elements that have been contracted into z^i; /* $f(V[z^i]) = f^{i-1}(z^i)$ */
9 **if** $f^{i-1}(z^i) < \min\{\mu(u^i), \mu(v^i)\}$ **then**
10 $\mathcal{X} := \mathcal{X} \cup \{V[z^i]\}$; $\mu(z^i) := f^{i-1}(z^i)$
11 **else**
12 $\mu(z^i) := \min\{\mu(u^i), \mu(v^i)\}$
13 **end if**
14 **end for**

From the above argument, we see that algorithm ExtremeSubsets computes the set $\mathcal{X}(f)$ of all extreme sets of f correctly as long as we can always find a flat pair in line 5. If a given set function f is symmetric and crossing submodular or intersecting submodular and posi-modular, then it is not difficult to see that each set function f^i obtained from f by contracting elements remains symmetric and crossing submodular or intersecting submodular and posi-modular. Therefore, by Corollary 10, we can find a flat pair in $O(n^2 T_f)$ time. Then the running time of ExtremeSubsets is $O(n^3 T_f)$. This establishes Theorem 2.

By Lemma 6, we easily see that, for hypergraphs, algorithm ExtremeSubsets can be implemented to run in $O(n(m + n \log n))$ time and $O(m + n)$ space.

Corollary 11 *For an edge-weighted hypergraph $(G = (V, E), w)$, the set $\mathcal{X}(d)$ of all extreme subsets can be found in $O(n(m + n \log n))$ time and $O(m + n)$ space, where $n = |V|$ and $m = \sum_{e \in E} |e|$.* ∎

A computing process of ExtremeSubsets applied to an edge-weighted hypergraph is presented in Appendix.

5 Conclusion

MA orderings were originally introduced to find a forest decomposition of multigraphs in linear time by Nagamochi and Ibaraki [11]. They realized that the an MA ordering identifies a pendent pair, and based on this, they proposed an $O(nm + n^2 \log n)$ time algorithm for computing a minimum cut in an edge-weighted graph without relying on a maximum flow algorithm. The algorithm is then extended to an $O(n^3 T_f)$ time algorithm for minimizing a symmetric and crossing submodular set function by Queyranne [16] and for an intersecting posi-modular set function by Nagamochi and Ibaraki [13]. For graphs, MA orderings can be used to sparsify multigraphs [11] and to find a maximum flow between a pendent pair in an edge-weighed graphs [1]. However, for symmetric submodular or posi-modular set functions, Queyranne's and Nagamochi and Ibaraki's algorithms based on pendent pairs can find a minimizer only.

Our new algorithm beased on flat pairs can find not only a minimizer but also all extreme subsets. Interestingly, the algorithm works for the class of intersecting posi-modular or symmetric and crossing submodular set functions, which is shown by Lemma 1 to be a maximal class of set functions whose extreme subsets always form laminar.

References

[1] S. R. Arikati and K. Mehlhorn: A correctness certificate for the Stoer-Wagner min-cut algorithm, Information Processing Letters, vol. 70, 1999, pp. 251–254.

[2] C. S. Chekuri, A. V. Goldberg, D. R. Karger, M. S. Levine and C. Stein, Experimental study of minimum cut algorithms, Proc. 8th Annual ACM-SIAM Symposium on Discrete Algorithms, (1997) 324–333.

[3] E.A. Dinits, A.V. Karzanov and M.V. Lomonosov, On the structure of a family of minimal weighted cuts in a graph, Studies in Discrete Optimization (in Russian) A.A. Fridman (Ed.) Nauka, Moscow, (1976), 290–306.

[4] M. L. Fredman and R. E. Tarjan, Fibonacci heaps and their uses in improved network optimization algorithms, J. Assoc. Compt. Mach., 34 (1987) 596–615.

[5] S. Iwata, L. Fleischer, and S. Fujishige: A combinatorial strongly polynomial algorithm for minimizing submodular functions, Journal of ACM, 48 (2001) 761–777.

[6] R.E. Gomory and T.C. Hu, Multi-terminal network flows, SIAM J. Appl. Math., 9 (1961) 551–570.

[7] D. W. Matula, k-Components, clusters, and slicings in graphs, SIAM J. Applied Mathematics, vol. 22, (1972) 459–480.

[8] D. W. Matula, and L. L. Beck, Smallest-last ordering and clustering and graph coloring algorithms, J. ACM, vol. 30, (1983) 417–427.

[9] H. Nagamochi, Graph algorithms for network connectivity problems, J. Operations Research Society of Japan, 47 (2004) 199–223.

[10] H. Nagamochi, Computing a minimum cut in a graph with dynamic edges incident to a designated vertex, Inst. Electron. Inform. Comm. Eng. Trans. Fundamentals (to appear).

[11] H. Nagamochi and T. Ibaraki, A linear-time algorithm for finding a sparse k-connected spanning subgraph of a k-connected graph, Algorithmica, 7 (1992) 583–596.

[12] H. Nagamochi and T. Ibaraki, Computing edge-connectivity of multigraphs and capacitated graphs, SIAM J. Discrete Mathematics, 5 (1992) 54–66.

[13] H. Nagamochi and T. Ibaraki, A note on minimizing submodular functions, Information Processing Letters, 67 (1998) 239–244.

[14] H. Nagamochi and T. Ibaraki, Augmenting edge-connectivity over the entire range in $\tilde{O}(nm)$ time, J. Algorithms, 30 (1999) 253–301.

[15] H. Nagamochi and Y. Kamidoi, Minimum cost subpartitions in graphs, Information Processing Letters, 102 (2007) 79–84.

[16] M. Queyranne, Minimizing symmetric submodular functions, Mathematical Programming, 82 (1998) 3–12.

[17] M. Sakashita, K. Makino, S. Fujishige, Minimum cost source location problems with flow requirements, Lecture Notes in Computer Science vol. 3887 (2006) 769–780.

[18] A. Schrijver, A combinatorial algorithm minimizing submodular function in strongly polynomial time, J. Combinatorial Theory (B), 80 (2000) 346–355.

[19] M. Stoer and F. Wagner, A simple min cut algorithm, J. ACM, 44 (1997) 585–591.

[20] T. Watanabe and A. Nakamura, Edge-connectivity augmentation problems, J. Comp. System Sci., 35 (1987) 96–144.

Appendix

Proof of Lemma 6

Note that, for each $x \in V - V_{i-1}$, it holds

$$\begin{aligned}
&d_{(G,w)}(x) + d_{(G,w)}(V_{i-1}+x) - d_{(G,w)}(V_{i-1})\\
&= 2\sum\{w(e) \mid x \in e \in E, e \subseteq V - V_{i-1}\}\\
&\quad + \sum\{w(e) \mid x \in e \in E, e \cap V_{i-1} \neq \emptyset \neq e \cap (V - V_{i-1} - x)\}\\
&= 2\sum\{w(e) \mid x \in e \in E, e \cap V_{i-1} = \emptyset\}\\
&\quad + \sum\{w(e) \mid x \in e \in E, e \cap V_{i-1} \neq \emptyset, |e \cap (V - V_{i-1})| \geq 2\}\\
&= 2\sum\{w(e) \mid e \in A_{V-V_{i-1}}, e \cap x \neq \emptyset \neq e \cap (V - V_{i-1} - x)\}\\
&\quad + \sum\{w(e) \mid e \in B_{V-V_{i-1}}, e \cap x \neq \emptyset \neq e \cap (V - V_{i-1} - x)\}\\
&= \sum\{w_{V-V_{i-1}}(e) \mid e \in A_{V-V_{i-1}} \cup B_{V-V_{i-1}}, e \cap x \neq \emptyset \neq e \cap (V - V_{i-1} - x)\}\\
&= 2d_{(G\langle V-V_{i-1}\rangle, w_{V-V_{i-1}})}(x).
\end{aligned}$$

Hence any ordering $\pi = (v_1, v_2, \ldots, v_n)$ in the statement of the lemma is an MD ordering of d. We consider the time complexity. A hypergraph $G = (V, E)$ can be stored as a set of adjacency lists $Adj(u)$, $u \in V$ in $O(m+n)$ space, by which we can find the set $\Gamma_G(u)$ of all vertices adjacent to u in $O(|\Gamma_G(u)|)$ time. To obtain an MD ordering of d, we repeat choosing a vertex u with the minimum degree $d_{(G\langle V-V_{i-1}\rangle, w_{V-V_{i-1}})}(v)$ in the current graph $G\langle V-V_{i-1}\rangle$ and removing the vertex $v_i = u$ from the graph to obtain $G\langle V-V_i\rangle$. We maintain a data structure Q that contains all vertices $u \in V - V_i$ along with the current degree $d_{(G\langle V-V_{i-1}\rangle, w_{V-V_{i-1}})}(u)$, $u \in V - V_i$. After choosing v_i, we update the degree $d_{(G\langle V-V_{i-1}\rangle, w_{V-V_{i-1}})}(u)$ of each vertex $u \in \Gamma_G(v_i) - V_i$ by decreasing by

$$\sum\{\frac{w(e)}{2} \mid u \in e \in E, e \cap V_i = \{v_i\}\} + \sum\{\frac{w(e)}{2} \mid v_i \in e \in E, e - V_i = \{u\}\}.$$

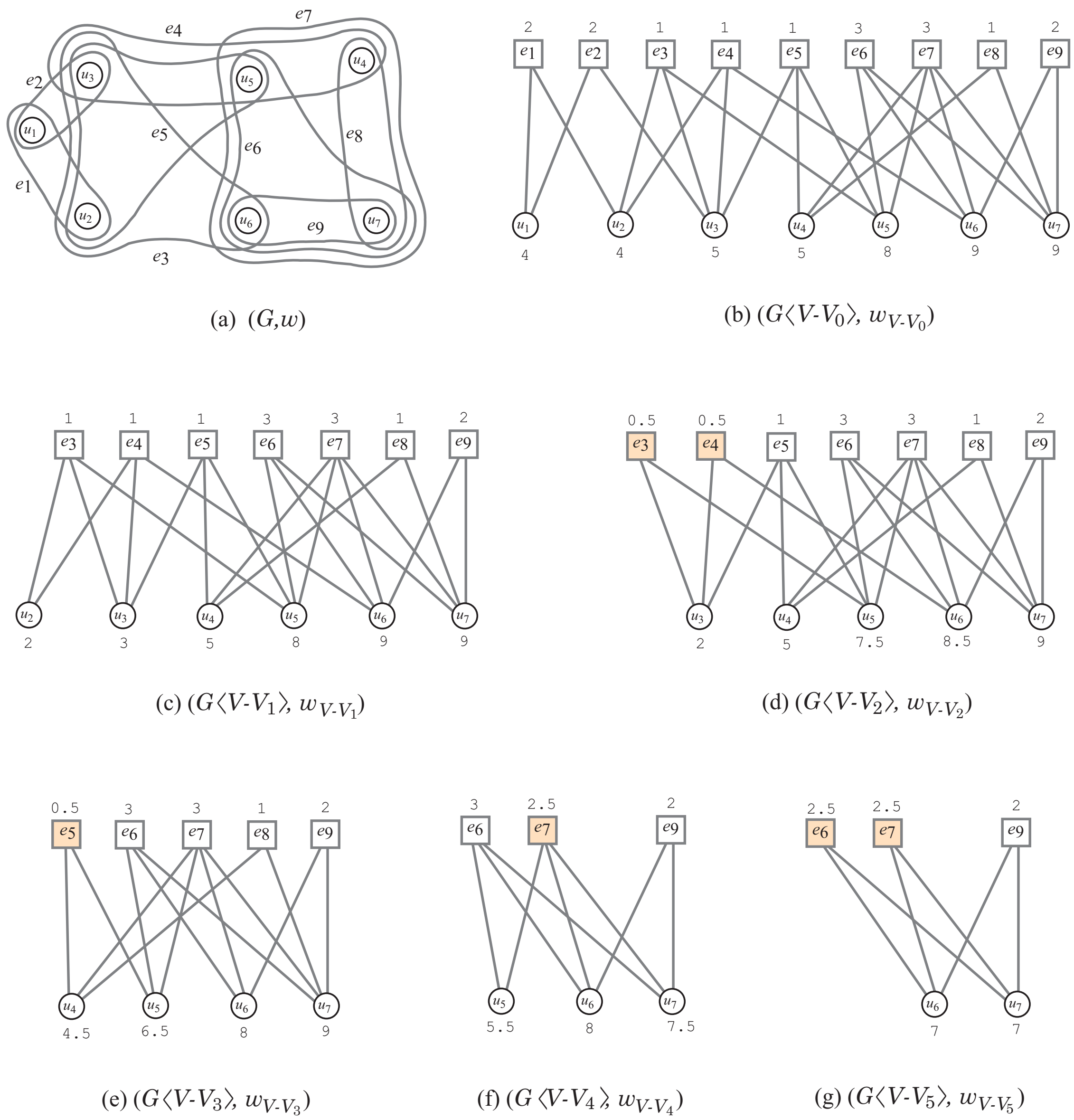
We then choose a vertex u^* with the minimum degree $d_{(G\langle V-V_{i-1}\rangle, w_{V-V_{i-1}})}(u^*)$ from Q, and then delete u^* from Q. It is not difficult to see that the total number of updating degrees is $O(m)$, and the number of extracting a vertex with the minimum degree from Q is n. By using data structure of Fibonacci heap [4] to maintain elements in Q, the above algorithm can be implemented to run in $O(m + n\log n)$ time and $O(m+n)$ space. ∎

An example of MD orderings in an edge-weighted hypergraph

Figure 3(a) shows an edge-weighted hypergraph $(G = (V, E), w)$, which is represented by a bipartite graph (V, E, F) in Figure 3(b) in such a way that V and E are the two disjoint vertex sets in the bipartite graph and two vertices $v \in V$ and $e \in E$ are adjacent if and only if e is an edge incident to v in G. Figure 3(b)-(g) show induced hypergraphs $(G\langle V-V_{i-1}\rangle, w_{V-V_{i-1}})$, $i = 1, 2, \ldots, 6$. In this case, we have MD ordering $\pi = (u_1, u_2, u_3, u_4, u_5, u_6)$ and flat pair $\{u_5, u_6\}$.

A computing process of ExtremeSubsets applied to an edge-weighted hypergraph

We apply ExtremeSubsets to the edge-weighted hypergraph (G, w) in Figure 3(a). Figure 4(a)-(f) illustrate its computing process of ExtremeSubsets, where (G^i, w^i) denotes the hypergraph before the iteration for i, i.e., $f^i = d_{(G^i, w^i)}$ holds. Figure 4(g) shows the resulting family of extreme subsets $\mathcal{X}(G, w)$.



(a) (G,w)

(b) $(G\langle V\text{-}V_0\rangle,\ w_{V\text{-}V_0})$

(c) $(G\langle V\text{-}V_1\rangle,\ w_{V\text{-}V_1})$

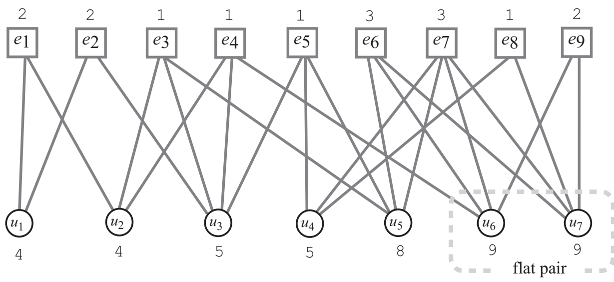
(d) $(G\langle V\text{-}V_2\rangle,\ w_{V\text{-}V_2})$

(e) $(G\langle V\text{-}V_3\rangle,\ w_{V\text{-}V_3})$

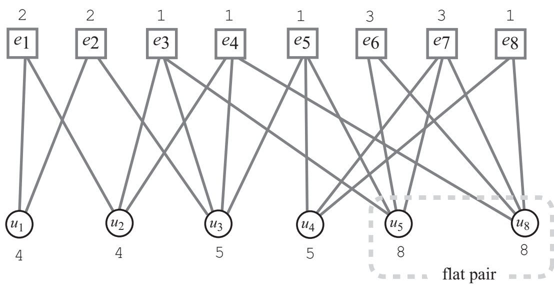
(f) $(G\,\langle V\text{-}V_4\,\rangle,\ w_{V\text{-}V_4})$

(g) $(G\langle V\text{-}V_5\rangle,\ w_{V\text{-}V_5})$

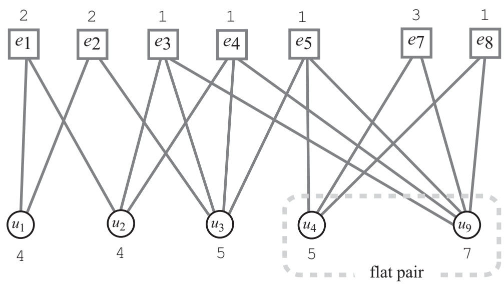
Figure 3: Illustration for computing an MD ordering in a hypergraph; (a) An edge-weighted hypergraph $(G = (V, E), w)$, where the numbers beside edges indicate their weights; (b) The bipartite representation (V, E, F) of G, where the numbers beside vertices $e \in E$ indicate their weights $w(e)$ and the numbers beside vertices $v \in V$ indicate their degree $d_{(G,w)}(v)$; (c)-(g) The bipartite representations of induced hypergraphs $G\langle V - V_{i-1}\rangle$, $i = 2, 3, \ldots, 6$, where the numbers beside vertices $e \in E$ indicate their weights $w_{V-V_{i-1}}(e)$ and the numbers beside vertices $v \in V$ indicate their degree $d_{(G\langle V-V_{i-1}\rangle, w_{V-V_{i-1}})}(v)$.



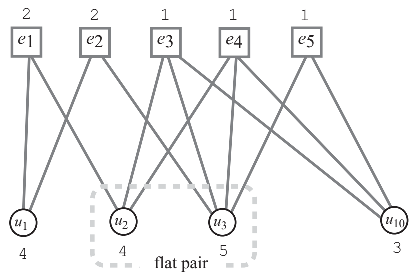
(a) (G^7,w^7)



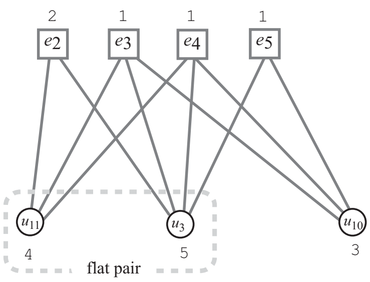
(b) (G^6,w^6)



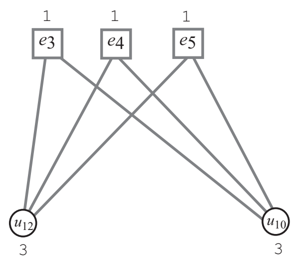
(c) (G^5,w^5)



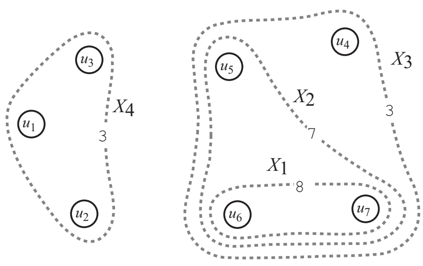
(d) (G^4,w^4)



(e) (G^3,w^3)



(f) (G^2,w^2)



(g) $\mathcal{X}(G,w)$

Figure 4: (a)-(f) A computing process of ExtremeSubsets applied to the edge-weighted hypergraph (G, w) in Figure 3(a); (g) The family of extreme subsets $\mathcal{X}(G, w)$ in the edge-weighted hypergraph (G, w) in Figure 3(a).