Analysis of An Approach for the Set Maxima Problem

Vincenzo Liberatore*

Abstract

The *set maxima problem* is defined as follows: given a totally ordered universe X of size n and $\mathcal{S} \subseteq 2^X$, $|\mathcal{S}| = n$, find $\max S$ for all $S \in \mathcal{S}$. It has been conjectured that this problem can be solved deterministically with $O(n)$ comparisons. In this paper, we discuss the proposal of Komlós that the minimum spanning tree verification algorithm should provide a solution paradigm for the set maxima problem. We generalize Komlós' algorithm to a larger set of problems arising from the verification of the optimum basis of a very large class of matroids. We will also provide an algorithm for the complementary case of maxima over fundamental matroid hyperplanes.

1 Introduction

Let X be a set of n distinct elements on which a total order is defined, and $\mathcal{S} \subseteq 2^X$ a collection of n nonempty subsets of X. The *set maxima problem* consists of computing $\max S$ for all $S \in \mathcal{S}$; all computation is assumed to occur in the comparison tree model. Any instance of the set maxima problem can be trivially solved with $O(n \log n)$ comparisons by sorting the elements in X and extracting the maxima from the sorted universe with no additional comparisons. No better deterministic algorithm is presently known for this general version of the problem.

Fredman formulated the set maxima problem and proved that at most $\binom{2n-1}{n-1}$ arrangements of maxima are possible [11], showing that the information-theoretic bound is only $\Omega(n)$ comparisons. An $O(n)$ randomized algorithm is given in [9], but no linear deterministic algorithm is known except for some particular cases. However, Fredman's proof and the existence of a linear randomized algorithm made easy to conjecture that also a deterministic linear algorithm should exist for this problem.

Recently, a "rank-sequence algorithm" has been proposed [2] with the following properties. The algorithm is parametric with respect to a subset $\rho \subseteq \{1, 2, \ldots, n\}$ which is named the rank sequence. The rank-sequence algorithm runs with a linear number of comparisons when $\mathcal{S}$ is the collection of hyperplanes of the projective geometry $PG(d, q)$ once d is a fixed constant and ρ has been chosen in a suitable way. The rank-sequence algorithm has been conjectured to have a version that runs with $O(n)$ comparisons for any instance of the set maxima problem because projective geometries are intrinsically hard instances of the set maxima problem [2]. Unfortunately, we will show that for any choice of the rank-sequence ρ there is an instance of the set maxima problem that forces the rank-sequence algorithm to execute a superlinear number of comparisons, even if ρ is chosen on the basis of an examination of $\mathcal{S}$.

A particularly interesting case of the set maxima problem arises in graph theory as the verification of whether a given spanning tree of a weighed undirected graph has minimum cost. It is

*Department of Computer Science, Hill Center, Rutgers University, New Brunswick, NJ 08903. E-mail: vliberatore@acm.org. Partly supported by an NSF grant through DIMACS.

well known that this problem is essentially the set maxima problem for which the universe X is the set of tree edges and the family of sets $\mathcal{S}$ is the collection of paths linking the endpoints of every nontree edge [1]. For this problem, Komlós has found an optimal algorithm [13], and in passing, we observe that Komlós' solution yielded very recently the first randomized algorithm for finding minimum spanning trees with a linear number of comparisons [12].

Both Komlós' algorithm and the rank-sequence algorithm have been tentatively conjectured as prototypes of an efficient algorithm for the set maxima problem [2, 13]. Because of the strength of these guesses and because several authorities pointed out the difficulty of the problem[1], we lost any interest in seeking a novel solution for the set maxima problem, and we focused on the analysis of the two strategies above. In the next section, we will analyze the rank-sequence algorithm, and then we will state our results and outline the structure of this paper.

2 The Rank-Sequence Algorithm

We now analyze the rank-sequence algorithm of Bar-Noy, Motwani, and Naor [2] in the critical case $m = n$. The rank-sequence algorithm works as follows. Define the rank of an element $x \in X$ as the number of elements in X that are greater than or equal to x. Let $\rho = \langle r_1, r_2, \ldots, r_k \rangle$ be a sequence of integers with $n \geq r_1 \geq r_2 \geq \ldots \geq r_k \geq 1$; ρ will be the rank sequence and the rank-sequence algorithm is parametric with respect to ρ.

Algorithm 1 *Rank-Sequence Algorithm [2].*

1. compute the sequence $\langle z_1, z_2, \ldots, z_k \rangle \in X^k$ such that the rank of z_i is r_i $(1 \leq i \leq k)$;
2. for $i = 0, 1, \ldots, k$, let

$$Z_i = \begin{cases} \{x \in X | x \leq z_1\} & \text{if } i = 0 \\ \{x \in X | z_i < x \leq z_{i+1}\} & \text{if } 1 \leq i < k \\ \{x \in X | z_k < x\} & \text{otherwise} \end{cases}$$

3. for $j = 1, \ldots, m$, let $S'_j = S_j \cap Z_{l(j)}$ where $l(j)$ is the largest index i such that $S_j \cap Z_i \neq \emptyset$,
4. compute $\max S'_j$ for all j's by brute force search.

The set S'_j is called the reduced set of S_j. We can verify if two reduced sets are equal without executing any comparison, and, if indeed two reduced sets are equal, we will eliminate one of the two and reduce the size of the problem for free.

Let R be a random variable that takes a value $i \in \{0, 1, \ldots, k\}$ with the probability p_i that an element taken uniformly at random from X is in Z_i. We recall [2] that

Lemma 2.1 *The generic algorithm takes* $\Omega(n(1 + H(R))) + \sum_{j=1}^{n}(|S'_j| - 1)$ *comparisons.*

Lemma 2.2 *If* $x \geq 1$*, then* $\lfloor x \rfloor \geq \frac{x}{2}$.

Proof. If $1 \leq x < 2$, then $1 = \lfloor x \rfloor > \frac{x}{2}$. If $x \geq 2$, then $\lfloor x \rfloor \geq x - 1 \geq \frac{x}{2}$. □

We begin by presenting a proof of the following well-known fact [8]:

[1] Jeff Kahn offered the author a Paul Erdös T-shirt if we can solve the set maxima problem.

Theorem 2.3 *For infinitely many values of n and for any choice of the rank sequence there is an instance of the set maxima problem that forces the generic algorithm to execute $\Omega(n \log n)$ comparisons.*

Intuitively, if the rank sequence is coarse, then we can insert many sets in its largest subdivision and increase the brute force search cost; if the rank sequence is fine, then the determination of $\langle z_1, z_2, \dots, z_k$ will require too many comparisons.

Proof. Let Z_h be the largest of the Z_i's $(0 \le i \le k)$ and let $\nu = |Z_h| \ge \frac{n}{k+1}$ and $p_h = \frac{\nu}{n}$. The concavity of logarithms implies that

$$H(R) \ge \left\lfloor \frac{1}{p_h} \right\rfloor p_h \log \frac{1}{p_h} = \Omega\left(\log \frac{1}{p_h}\right) .$$

If $\nu \ne \omega(1)$, there are infinitely many values of n where $p_h \le \frac{c}{n}$ for some positive constant c, and for those values of n $H(R) = \Omega(\log n)$, and the generic algorithm executes $\Omega(n \log n)$ comparisons.

Suppose now that $\nu = \omega(1)$ and $p_h = \omega\left(\frac{1}{n}\right)$. The subsets of Z_h of size greater than $\frac{\nu}{3}$ that do not include z_{h+1} are at least $2^{\nu-1} - 2^{(\nu-1)H\left(\frac{1}{3}\right)}$ for ν big enough, and among these we can choose $n-1$ distinct reduced sets as long as n is big enough. We then choose also that $X \in \mathcal{S}$ in order to satisfy the condition that all elements in X appear in $\mathcal{S}$. Then, the generic algorithm executes

$$\Omega\left(n(1 + H(R)) + (n-1)\left(\frac{p_h n}{3} - 1\right)\right) = \Omega\left(n\left(\log \frac{1}{p_h} + \frac{1}{3}(n-1)p_h\right)\right)$$

comparisons. The derivative of $\log \frac{1}{x} + \frac{1}{3}(n-1)x$ with respect to x is positive for $x > \frac{3 \ln 2}{n-1}$, and so, for $\nu = \omega(1)$, the number of comparisons is increasing with p_h. Therefore, the number of comparisons is at least $\Omega(n \log n)$. $\square$

The best value of p_h for the generic algorithm is $p_h \simeq \frac{2}{n}$, in which case the algorithm executes $\Omega(n \log n)$ comparisons. In other words, the generic algorithm is not substantially different than sorting.

Corollary 2.4 *The number of comparisons executed by the generic algorithm is not in $O(n)$.*

Corollary 2.5 *There is an instance of the set maxima problem that can be solved in a linear number of comparisons, but such that for any choice of the rank sequence and for infinitely many values of n, the generic algorithm executes $\Omega(n \log n)$ comparisons.*

In the proof, we will need to invoke the existence of the symmetric heap algorithm [13] that finds the maxima over intervals with $O(n)$ comparisons.

Proof. Let $\mathcal{S} = \{\{x \in X : \max\{0, i - \sqrt{n}\} \le \operatorname{rank}(x) \le i\} : 1 \le i \le n\}$. $\mathcal{S}$ is a collection of intervals and can be solved in $O(n)$ comparisons. If $p_i < \frac{2}{\sqrt{n}}$, then we can attribute to each element in Z_i an entropy cost of $\log \frac{\sqrt{n}}{2} = \Omega(\log n)$. If $p_i \ge \frac{2}{\sqrt{n}}$, then the total brute force cost on Z_i is $\left\lfloor \frac{p_i n}{2} \right\rfloor (\sqrt{n} - 1)$, and we can attribute to each element in Z_i a cost of $\Omega(\sqrt{n})$. On the whole, the cost is $\Omega(n \log n)$. $\square$

The previous corollary entails that no adaptive version of the generic algorithm that fixes the rank sequence after having inspected $\mathcal{S}$ could produce the set maxima with a linear number of comparisons. The rank-sequence algorithm seems to work only when the set $\mathcal{S}$ in some sense possesses random-like properties [2].

Since the generic algorithm fails to find a solution in $O(n)$ comparisons, we are led to believe that a solution for this problem cannot make decisions based only on selected elements of givens ranks, but must examine the structure of the family $\mathcal{S}$.

3 Matroids

3.1 Representation with Matroids

We assume familiarity with the notion of matroids [6, 14, 19, 21], and we give the following definitions for the sole purpose of fixing our notation. Let $\mathcal{M}$ be a matroid [14, 21] with element set $E(\mathcal{M})$, basis set $\mathcal{B}(\mathcal{M})$, circuit set $\mathcal{C}(\mathcal{M})$, and hyperplane set $\mathcal{H}(\mathcal{M})$. $\mathcal{M}^*$ is the dual matroid of $\mathcal{M}$.

Let $B \in \mathcal{B}(\mathcal{M})$. For $e \in E - B$, let $C_{\mathcal{M}}(e, B)$ be the unique fundamental circuit of e with respect to B in $\mathcal{M}$. We will omit the subscript $\mathcal{M}$ when no confusion can arise and we will let $B(e) = C(e, B) - \{e\}$. Let $\lambda_{\mathcal{M}}$ be the exchange closure function of $\mathcal{M}$; we will omit the subscript $\mathcal{M}$ when no confusion can arise. We define $H_e = \lambda(B - \{e\}) \in \mathcal{H}(\mathcal{M})$ for $e \in B$. Notice that $H_e - B = \{g \notin B : e \notin B(g)\}$. If A is a matrix, $\mathcal{M}[A]$ will be the matroid represented by A, and $E(\mathcal{M}[A])$ is the set of column labels of A.

In the rest of this section we will give two reformulations of the set maxima problem in terms of binary matroids. First, we will show that the set maxima problem is equivalent to the problem of computing the maxima of sets of the form $B(e)$ of a simple weighed binary matroid. This reduction is suitable to generalize Komlós' algorithm and will be investigated in section 5. We will also show that the set maxima problem is equivalent to the problem of computing the maxima of sets of the form $H_e - B$ of a weighed binary matroid; this reformulation will be investigated in section 6. In the rest of this section, we will outline the nature of these two reformulations. We will assume that $|\mathcal{S}| = m$ could be different from n as this simplifies the statement and analysis of the intermediate phases of our algorithms. We will let $[n] = \{1, 2, \ldots, n\}$, $\mathcal{S} = \{S_1, S_2, \ldots, S_m\}$, and $X = \{x_1, x_2, \ldots, x_n\}$.

Define $\mathcal{M}_{\mathcal{B}} = \mathcal{M}[(I|D)]$ where $D = (d_{ij})$ is an $n \times m$ matrix given by

$$d_{ij} = \begin{cases} 1 & \text{if } x_i \in S_j \\ 0 & \text{otherwise} \end{cases} .$$

Let $w : [n] \to X$ be a weight function defined on the basis $[n] \in \mathcal{B}(\mathcal{M}_{\mathcal{B}})$ by $w(i) = x_i$. Then, $\max S_j = \max B(n + j)$ for $j = 1, 2, \ldots, m$, and the set maxima problem is equivalent to the problem of computing the maxima of sets of the form $B(e)$ of a simple weighed binary matroids.

The second reformulation is as follows. Define $\mathcal{M}_{\mathcal{H}} = \mathcal{M}[(I|D')]$ where $D' = \left(d'_{ij}\right)$ is an $m \times n$ matrix given by

$$d'_{ij} = \begin{cases} 1 & \text{if } x_j \notin S_i \\ 0 & \text{otherwise} \end{cases} .$$

Let $w : [m + n] - [n] \to X$ be a weight function defined on the cobasis $[m + n] - [n]$ of $\mathcal{M}_{\mathcal{H}}$ by $w(n + j) = x_j$. Then, $\max S_i = \max(H_i - [n])$ for $i = 1, 2, \ldots, m$, and the set maxima problem is equivalent to the problem of computing the maxima of sets of the form $H_e - B$ of a weighed binary matroids.

3.2 Fundamental Circuit Maxima

Given a simple weighed binary matroid $\mathcal{M}$ and a basis $B \in \mathcal{B}(\mathcal{M})$, the *fundamental circuit maxima problem* is the problem of finding $\max B(e)$ for all $e \notin B$. If $\mathcal{M}$ is graphic, then Komlós's algorithm [13] solves the fundamental circuit maxima problem. In this paper, we generalize Komlós' algorithm as follows:

Theorem 3.1 *There is an algorithm that runs in a linear number of comparisons for all those set maxima problems that can be expressed as a fundamental circuit maxima problem on a matroid that is obtained by means of direct sums and 2-sums starting from matroids each of which is either regular or isomorphic to F_7 or to F_7^*.*

In particular, fundamental circuit maxima over regular matroids can be found with a linear number of comparisons. But more is true: piecing together regular matroids and F_7, we obtain an algorithm for the class of matroids that enjoy the strong max-flow min-cut property [15, 19, 20]; to the best of our knowledge, this is the larger proper subclass of binary matroids that arises in a natural setting.

Theorem 3.1 constitutes a proper generalization of Komlós' algorithm, as established by observing that, for $B = [3] \in \mathcal{B}(F_7)$, $\mathcal{S} = \{B(e) : e \in B\}$ is a valid set family for the set maxima problem, that any other binary matroid that defines the same fundamental circuit family $\mathcal{S}$ contains a minor isomorphic to F_7, and that F_7 is an excluded minor for the class of graphic matroids.

3.3 The Hyperplane Maxima Problem

Given a simple binary matroid $\mathcal{M}$ and a $B \in \mathcal{B}(\mathcal{M})$, the *fundamental hyperplane maxima problem* is the problem of finding $\max(H_e - B)$ for all $e \in B$. The fundamental hyperplane maxima problem arises as complementary to the fundamental circuit maxima problem because of the following easy

Proposition 3.2 *Let $\mathcal{M}^* = (E, \lambda^*)$ be a matroid, $e \in B^* \in \mathcal{B}(\mathcal{M}^*)$, $B = E - B^* \in \mathcal{B}(\mathcal{M})$, and $H_e^* = \lambda^*(B^* - \{e\})$. Then, $H_e^* - B^* = B - B(e)$.*

Proof. Let r^* be the rank function on $\mathcal{M}^*$, r the rank function on $\mathcal{M}$, and λ the closure function of $\mathcal{M}$. $g \in H_e^* - B^* = H_e^* \cap B$ if and only if $g \in B$ and $r^*((B^* - \{e\}) \cup \{g\}) = r^*(B^* - \{e\})$. Rewriting this equality in terms of r and noticing that $g \in B$, we obtain $r((B \cup \{e\}) - \{g\}) + 1 = r(B \cup \{e\}) = r(\mathcal{M})$, and $g \in H_e^* \cap B$ if and only if $g \in B$ and $(B \cup \{e\}) - \{g\} \notin \mathcal{B}(\mathcal{M})$, which happens if and only if $g \in B - B(e)$. $\square$

Our main result on hyperplane maxima is as follows:

Theorem 3.3 *There is an algorithm that runs in a linear number of comparisons for all those set maxima problem that can be expressed as a fundamental hyperplane maxima problem on a matroid that is obtained by means of direct sums, 2-sums, and 3-sums starting from matroids each of which is either regular or isomorphic to F_7 or to F_7^*.*

It is not hard to verify that the fundamental hyperplane maxima of $\mathcal{M}[(I|A)]$ for

$$A = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

can be found by the algorithm promised by theorem 3.3, but not by that promised by theorem 3.1.

The rest of the paper is organized as follows. In section 4, we will establish some properties of the matroid sum operator.

4 Sums

In this section, we will establish a few properties of the matroid sum operations. These properties are purely combinatorial and non-algorithmic, but will be used in the next sections to prove the correctness of our algorithms. Although most of the following results establish natural properties of the sum operators, we were not able to locate any of them in the literature.

We will assume familiarity with the symmetric difference operation $\triangle$ and the parallel connection operation P_N. Since slightly different versions of the sum operations have appeared in the literature, we make the following non-restrictive assumptions following Seymour [16]:

- no basepoint in a 2-sum is a loop or a coloop of either part;
- the size of the parts of a 2-sum exceeds 2;
- in any 3-sum $\mathcal{M}_1 \oplus_3 \mathcal{M}_2$, $E(\mathcal{M}_1) \cap E(\mathcal{M}_2)$ does not contain a cocircuit of either part;
- the size of the parts of a 3-sum exceeds 6.

We also observe that the sum of matroids is commutative by the symmetry of the definition. Moreover, the symmetric difference operators enjoys the following associativity property.

Proposition 4.1 *Let $\mathcal{M}_1$, $\mathcal{M}_2$, and $\mathcal{M}_3$ be binary matroids with $E(\mathcal{M}_1) \cap E(\mathcal{M}_2) \cap E(\mathcal{M}_3) = \emptyset$. Then, $\mathcal{M}_1 \triangle (\mathcal{M}_2 \triangle \mathcal{M}_3) = (\mathcal{M}_1 \triangle \mathcal{M}_2) \triangle \mathcal{M}_3$.*

Proof. Let $\mathcal{M}_L = \mathcal{M}_1 \triangle (\mathcal{M}_2 \triangle \mathcal{M}_3)$ and $\mathcal{M}_R = (\mathcal{M}_1 \triangle \mathcal{M}_2) \triangle \mathcal{M}_3$. Clearly, $E(\mathcal{M}_L) = E(\mathcal{M}_R)$. Let $C_i \in \mathcal{C}(\mathcal{M}_i)$ ($i \in [3]$), and suppose by contradiction that $C = C_1 \triangle C_2 \triangle C_3$ is a circuit of $\mathcal{M}_L$, but not of $\mathcal{M}_R$. Then, $C_2 \triangle C_3$ is a subset of $E(\mathcal{M}_2) \triangle E(\mathcal{M}_3)$, but that there is an $e \in (C_1 \triangle C_2) \cap (E(\mathcal{M}_1) \cap E(\mathcal{M}_2))$. So, $e \notin E(\mathcal{M}_i) \cap E(\mathcal{M}_3)$ ($i = 1, 2$). Also, if $e \in C_1$, then $e \in C$ because $e \notin E(\mathcal{M}_3)$; if $e \in C_2$, then $e \in C_2 \triangle C_3$ because $e \notin E(\mathcal{M}_3)$. In either case, $e \in C \cap E(\mathcal{M}_1) \cap E(\mathcal{M}_2)$, which is a contradiction. Conversely, if we suppose that C is a circuit of $\mathcal{M}_R$, but not of $\mathcal{M}_L$, we reach an analogous contradiction. □

We recall [4, 6] the following

Lemma 4.2 *Let $\mathcal{M}_1$ and $\mathcal{M}_2$ be matroids with $E(\mathcal{M}_1) \cap E(M_2) = \{p\}$. Then,*

$$\begin{aligned} \mathcal{B}(P(\mathcal{M}_1, \mathcal{M}_2)) \quad = \quad & \{B_1 \cup B_2 \cup \{p\} : B_i \cup \{p\} \in \mathcal{B}(\mathcal{M}_i), i = 1,2\} \cup \\ & \{B_1 \cup B_2 : p \notin B_1 \cup B_2, B_i \in \mathcal{B}(\mathcal{M}_i), B_j \cup \{p\} \in \mathcal{B}(\mathcal{M}_j), i, j \in \{1,2\}, i \neq j\}\ . \end{aligned}$$

It follows that

Proposition 4.3 *If $\mathcal{M} = \mathcal{M}_1 \oplus_2 \mathcal{M}_2$, $E(\mathcal{M}_1) \cap E(\mathcal{M}_2) = \{p\}$, then $\mathcal{B}(\mathcal{M}) \subseteq \mathcal{B}(P(\mathcal{M}_1, \mathcal{M}_2))$.*

Proof. Since $\mathcal{M} = P(\mathcal{M}_1, \mathcal{M}_2) \backslash \{p\}$, for any $B \in \mathcal{B}(\mathcal{M})$ there is a $B' \in \mathcal{B}(P(\mathcal{M}_1, \mathcal{M}_2))$ such that $B = B' - \{p\}$. Suppose there is a $B \in \mathcal{B}(\mathcal{M}) - \mathcal{B}(P(\mathcal{M}_1, \mathcal{M}_2))$; then $p \in B'$. Moreover, B is a maximal set of the form $B' - \{p\}$, implying that p belongs to every basis of $P(\mathcal{M}_1, \mathcal{M}_2)$. Then, by lemma 4.2, for both $i = 1, 2$, there are $B_i \cup \{p\} \in \mathcal{B}(\mathcal{M}_i)$. Suppose that for $i = 1$ or $i = 2$ there is also a $B_i' \in \mathcal{B}(\mathcal{M}_i)$ such that $p \notin B_i'$. Then, by lemma 4.2, for $j \in \{1,2\} - \{i\}$, $p \notin B_i' \cup B_j - \{p\} \in \mathcal{B}(P(\mathcal{M}_1, \mathcal{M}_2))$, which is a contradiction. Therefore, p is a coloop of $\mathcal{M}_1$ and $\mathcal{M}_2$, which contradicts our assumptions. □

By lemma 4.2 and proposition 4.3, if $B \in \mathcal{B}(\mathcal{M}_1 \oplus_2 \mathcal{M}_2)$, for $i, j \in \{1,2\}$, $i \neq j$, there are B_i and B_j such that

$$\begin{array}{rll} B_i = & (B \cap E(\mathcal{M}_i)) \cup \{p\} & \in \mathcal{B}(\mathcal{M}_i) \\ B_j = & B \cap E(\mathcal{M}_j) & \in \mathcal{B}(\mathcal{M}_j) \,. \end{array}$$

The basis B_i will be said to be the basis of $\mathcal{M}_i$ induced by B; analogously, the basis B_j will be said to be the basis of $\mathcal{M}_j$ induced by B, and we will say that the 2-sum is unbalanced from $\mathcal{M}_i$ to $\mathcal{M}_j$.

Recall [4, 6] that

Lemma 4.4 *Let $\mathcal{M}_1$ and $\mathcal{M}_2$ be matroids with $E(\mathcal{M}_1) \cap E(M_2) = \{p\}$. Let λ_P be the closure function of $\mathcal{M}_P = P(\mathcal{M}_1, \mathcal{M}_2)$, λ_i be the closure function of $\mathcal{M}_i$ $(i = 1,2)$ Then, for all $X_i \subseteq E(\mathcal{M}_i)$ $(i = 1,2)$,*

$$\lambda_P(X_1 \cup X_2) = \begin{cases} \bigcup_{i=1}^{2} \lambda_i(X_i \cup \{p\}) & \textit{if } p \in \lambda_i(X_i) \textit{ for } i = 1 \textit{ or } 2 \\ \bigcup_{i=1}^{2} \lambda_i(X_i) & \textit{otherwise} \end{cases}$$

Proposition 4.5 *If $\mathcal{M} = \mathcal{M}_1 \oplus_2 \mathcal{M}_2$ and the sum is unbalanced from $\mathcal{M}_2$ to $\mathcal{M}_1$, $E(\mathcal{M}_1) \cap E(\mathcal{M}_2) = \{p\}$, $B \in \mathcal{B}(\mathcal{M})$, B_i is the basis of $\mathcal{M}_i$ induced by B $(i = 1,2)$, λ is the closure function of $\mathcal{M}$, λ_i the closure function of $\mathcal{M}_i$ $(i = 1,2)$ and, for any $e \in B \cap E(\mathcal{M}_i)$, $H_e = \lambda(B - \{e\})$, $H_e^{(i)} = \lambda_i(B_i - \{e\})$, then,*

$$H_e = \begin{cases} H_e^{(1)} \cup E(\mathcal{M}_2) - \{p\} & \textit{if } e \in B_1, p \in H_e^{(1)} \\ H_e^{(1)} \cup H_p^{(2)} & \textit{if } e \in B_1, p \notin H_e^{(1)} \\ E(\mathcal{M}_1) \cup H_e^{(2)} - \{p\} & \textit{if } e \in B_2 \end{cases}$$

Proof. Write $H_e = \lambda_P(B - \{e\}) - \{p\}$, and substitute the expression of λ_P given by lemma 4.4 for $X_i = B \cap E(\mathcal{M}_i)$ $(i = 1,2)$. The proof is then completed by a case analysis. □

Let us turn now to 3-sums. We recall [5] the following

Lemma 4.6 *Let $\mathcal{M}_1$ and $\mathcal{M}_2$ be two matroids, $N = \mathcal{M}_1|T = \mathcal{M}_2|T$ a triangle, and $\mathcal{M}_P = P_N(\mathcal{M}_1, \mathcal{M}_2)$. Let $X_i \subseteq E(\mathcal{M}_i)$ for $i = 1,2$. Let λ_i the closure function of $\mathcal{M}_i$ $(i = 1,2)$ and λ_P the closure function of $\mathcal{M}_P$. Let r_i the rank function of $\mathcal{M}_i$ $(i = 1,2)$ and r_P the rank function of $\mathcal{M}_P$. Then,*

1. *$r(\mathcal{M}_P) = r(\mathcal{M}_1) + r(\mathcal{M}_2) - 2$;*
2. *$r_P(X_1 \cup X_2) = r_2(X_2 \cup (\lambda_1(X_1) \cap T)) + r_1(X_1 \cup T) - 2$;*
3. *$\lambda_P(X_1 \cup X_2) = \lambda_2((\lambda_1(X_1) \cap T) \cup X_2) \cup \lambda_1((\lambda_2((\lambda_1(X_1) \cap T) \cup X_2) \cap T) \cup X_1)$;*

Corollary 4.7 *Let $\mathcal{M}_1$ and $\mathcal{M}_2$ be matroids, where $N = \mathcal{M}_1|T = \mathcal{M}_2|T$, and $\mathcal{M} = \mathcal{M}_1 \oplus_3 \mathcal{M}_2$. Then, $r(\mathcal{M}) = r(\mathcal{M}_1) + r(\mathcal{M}_2) - 2$.*

Proof. Let $\mathcal{M}_P = P_N(\mathcal{M}_1, \mathcal{M}_2)$, $B \in \mathcal{B}(\mathcal{M})$, and $B_P \in \mathcal{B}(\mathcal{M}_P)$ such that $B = B_P - T$. Then, $r(\mathcal{M}) = |B| \leq |B_P| = r(\mathcal{M}_P)$. Suppose $r(\mathcal{M}) < r(\mathcal{M}_P)$. Then, by maximality of B, any basis of $\mathcal{M}_P$ would intersect T and T would contain a cocircuit of $\mathcal{M}_P$, and so a cocircuit of $\mathcal{M}_P|E(\mathcal{M}_1) = \mathcal{M}_1$. □

Now, we can prove the following theorem.

Theorem 4.8 *If $\mathcal{M} = \mathcal{M}_1 \oplus_3 \mathcal{M}_2$, $N = \mathcal{M}_1|T = \mathcal{M}_2|T$, and $B \in \mathcal{B}(\mathcal{M})$, then there are $B_N \in \mathcal{B}(N)$ and $B_i \in \mathcal{B}(\mathcal{M}_i)$ for $i = 1, 2$, such that $B_1 \cup B_2 = B \cup B_N$, and $B_1 \cap B_2 = \emptyset$.*

Proof. Let $\mathcal{M}_P = P_N(\mathcal{M}_1, \mathcal{M}_2)$ and $B \in \mathcal{B}(\mathcal{M})$. Then, $B = B_P - T$ for some $B_P \in \mathcal{B}(\mathcal{M}_P)$; by lemma 4.6.(1) and corollary 4.7, $B = B_P$. Define $B^{(i)} = B \cap E(\mathcal{M}_i)$ for $i = 1, 2$ and observe that $B^{(i)}$ is an independent set of $\mathcal{M}_i$.

We will show that there are $Q_1, Q_2 \subseteq T$ such that $B_i = B^{(i)} \cup Q_i \in \mathcal{B}(\mathcal{M}_i)$. Indeed, let r_P be the rank function of $\mathcal{M}_P$ and r_i be the rank function of $\mathcal{M}_i$ $(i = 1, 2)$, and

$$r(\mathcal{M}_1) + r(\mathcal{M}_2) - 2 = r(\mathcal{M}_P) = r_P(B) = r_2(B^{(2)} \cup (\lambda_1(B^{(1)}) \cap T)) + r_1(B^{(1)} \cup T) - 2\ ,$$

and since $r_1(B^{(1)} \cup T) \leq r(\mathcal{M}_1)$, $r_2(B^{(2)} \cup (\lambda_1(B^{(1)}) \cap T)) \geq r(\mathcal{M}_2)$ and, by the independence augmentation axiom, $Q_2 \subseteq \lambda_1(B^{(1)}) \cap T \subseteq T$. Analogously, $Q_1 \subseteq T$, and $B_1 \cup B_2 \subseteq B \cup T$.

Observe that $|B_i| - 2 \leq |B^{(i)}| \leq |B_i|$ and $|B| = |B^{(1)}| + |B^{(2)}| = |B_1| + |B_2| - 2$. $|B^{(i)}| = |B_i|$ $(i \in \{1, 2\})$ implies $|B_j| - 2 = |B^{(j)}|$ and $|B_j \cap T| = 2$ $(i \neq j \in \{1, 2\})$, which gives the result.

Suppose now that $|B^{(i)}| = |B_i| - 1$ for $i = 1, 2$. If $B_1 \cap B_2 = \{q\} \subset T$, then $q \notin \lambda_1(B^{(1)})$ and there is another element in T, say p, such that $p \notin \lambda_1(B^{(1)})$. Then, $q \in B_1(p)$ and define $B_1' = (B_1 \cup \{p\}) - \{q\} \in \mathcal{B}(\mathcal{M}_1)$, where $B^{(1)} \subset B_1'$ and $B_1' \cap B_2 = \emptyset$. □

The bases B_1 and B_2 given by the previous theorem will be said to be the bases of $\mathcal{M}_1$ and $\mathcal{M}_2$ induced by B. If $|B_i| > |B_j|$ we will say that the sum is unbalanced from $\mathcal{M}_i$ to $\mathcal{M}_j$ $(i, j \in \{1, 2\}, i \neq j)$; we will say that the sum is balanced otherwise.

We will use the following lemma due to Seymour [16]:

Lemma 4.9 *Let $\mathcal{M}_1$ and $\mathcal{M}_2$ be binary matroids and $T = E(\mathcal{M}_1) \cap E(\mathcal{M}_2)$. Suppose that $\mathcal{M} = \mathcal{M}_1 \oplus_3 \mathcal{M}_2$. If $C \in \mathcal{C}(\mathcal{M})$ and $Y_i = C \cap E(\mathcal{M}_i) \neq \emptyset$ $(i = 1, 2)$, then there exists a $z \in T$ such that for $i = 1, 2$, $Y_i \cup \{z\} \in \mathcal{C}(\mathcal{M}_i)$.*

Corollary 4.10 *Let $\mathcal{M}_1$ and $\mathcal{M}_2$ be binary matroids and $T = E(\mathcal{M}_1) \cap E(\mathcal{M}_2)$. Suppose that $\mathcal{M} = \mathcal{M}_1 \oplus_3 \mathcal{M}_2$. If $C \in \mathcal{C}(\mathcal{M})$ and $Y_i = C \cap E(\mathcal{M}_i) \neq \emptyset$ $(i = 1, 2)$, then there exists a unique $z \in T$ such that for $i = 1, 2$, $Y_i \cup \{z\} \in \mathcal{C}(\mathcal{M}_i)$.*

Proof. If there were two distinct z_1 and z_2, then $(Y_1 \cup \{z_1\}) \triangle (Y_1 \cup \{z_2\}) = \{z_1, z_2\} \subset T \in \mathcal{C}(\mathcal{M}_1)$ would contain a circuit of $\mathcal{M}_1$. □

We now prove the analogous of proposition 4.5.

Proposition 4.11 *Let $\mathcal{M} = \mathcal{M}_1 \oplus_3 \mathcal{M}_2$, $N = \mathcal{M}_1|T = \mathcal{M}_2|T$, $B \in \mathcal{B}(\mathcal{M})$, $B_i \in \mathcal{B}(\mathcal{M}_i)$ the basis of $\mathcal{M}_i$ induced by B $(i = 1, 2)$ λ is the closure function of $\mathcal{M}$, λ_i the closure function of $\mathcal{M}_i$ $(i = 1, 2)$, and for any $e \in B$, $H_e = \lambda(B - \{e\})$, $H_e^{(i)} = \lambda_i(B_i - \{e\})$. If $|B_1 \cap T| = |B_2 \cap T| = 1$, then,*

$$H_e = \begin{cases} H_e^{(i)} \cup E(\mathcal{M}_j) & \text{if } e \in B_i, p \in B_j \cap T \cap H_e^{(i)}, i, j \in \{1, 2\}, i \neq j \\ H_e^{(i)} \cup H_p^{(j)} & \text{if } e \in B_i, p \in (B_j \cap T) - H_e^{(i)}, i, j \in \{1, 2\}, i \neq j \end{cases}$$

Proof. Write $H_e = \lambda_P(B - \{e\}) - \{p\}$, and substitute the expression of λ_P given by lemma 4.6 for $X_i = B \cap E(\mathcal{M}_i)$ $(i = 1, 2)$. The proof is then completed by a case analysis. □

If $B \cap E(\mathcal{M}_i) \in \mathcal{B}(\mathcal{M}_i)$ for $i = 1$ or 2, the hyperplanes are not given by an expression as simple as that in the previous theorem. However, we will now prove a weaker version of the previous result on a minor of the matroid $\mathcal{M}$; such a result will be sufficient to justify our algorithms. For $\eta \in E(\mathcal{M})$ we will let $A(\eta) = B - B(\eta)$ and $\mathcal{M}_\eta = \mathcal{M}/A(\eta)$. We notice that $\mathcal{B}(\mathcal{M}_\eta) = \mathcal{B}(\mathcal{M}/A(\eta)) = \{B' \subseteq (E - B) \cup B(\eta) : B' \cup (B - B(\eta)) \in \mathcal{B}(\mathcal{M})\} \ni B(\eta)$.

Let λ_η be the closure function on $\mathcal{M}_\eta$, and for $e \in B(\eta)$, $H_e^\eta = \lambda_\eta(B(\eta) - \{e\})$ the hyperplane of $\mathcal{M}_\eta$ obtained by the removal of e from the basis $B(\eta)$. Then,

$$\begin{aligned} H_e^\eta - B(\eta) &= \lambda_\eta(B(\eta) - \{e\}) - B(\eta) = \\ &= \lambda((B(\eta) - \{e\}) \cup (B - B(\eta))) - (B - B(\eta)) - B(\eta) \\ &= H_e - B\ . \end{aligned}$$

We need the following preliminary lemmata.

Lemma 4.12 *If $\mathcal{M}$ is a binary matroid, $N = \mathcal{M}|T$ is a triangle, $T = \{p, q, r\}$, $B \in \mathcal{B}(\mathcal{M})$ with $B \cap T = \{p, q\}$, $\eta \in C \in \mathcal{C}(\mathcal{M})$, and $C \cap T = \{r\}$, $C \subseteq B \cup \{r, \eta\}$, then $(C - \{r\}) \cup \{p, q\} = C(\eta, B)$.*

Proof. $(C - \{r\}) \cup \{p, q\}$ is the disjoint union of circuits of $\mathcal{M}$ and contains only one non-basis element, namely η. Therefore, $(C - \{r\}) \cup \{p, q\}$ consists only of $C(\eta, B)$. □

Lemma 4.13 *If $\mathcal{M}$ is a binary matroid, $N = \mathcal{M}|T$ is a triangle, and $B \in \mathcal{B}(\mathcal{M})$ with $B \cap T = \emptyset$, then there is a $z \in T$ such that $B(z) = \triangle_{x \in T - \{z\}} B(x)$.*

Proof. If $z \in T$, then $B(z) \neq \emptyset$, otherwise z would be a loop of $\mathcal{M}$ and could not belong to a triangle. First, we prove that there are two elements $p, q \in T$ with the property that both $B(p) - B(q) \neq \emptyset$ and $B(q) - B(p) \neq \emptyset$. Suppose not. Then, let $T = \{p, q, r\}$ and assume $B(p) \subseteq B(r)$ save renaming of the elements in T. Indeed, $B(p) \subset B(r)$ otherwise $C(p, B) \triangle C(r, B) = \{p, q\}$ contains a circuit. So, $C(r, B) \triangle C(p, B) \triangle T$ is a disjoint union of circuits and it is equal to $((B(r) - B(p)) \cup \{p, r\}) \triangle T = (B(r) - B(p)) \cup \{q\} \subseteq B \cup \{q\}$. Since $B \cup \{q\}$ contains only the circuit $C(q, B)$, $B(q) = B(r) - B(p)$. So, $B(q) - B(p) \neq \emptyset$ and $B(p) - B(q) = B(p) \neq \emptyset$.

Then, $C(p, B) \triangle C(q, B) \triangle T$ is a disjoint union of circuits and it is equal to $(B(p) \triangle B(q)) \cup \{r\} \subseteq B \cup \{r\}$. Since $B \cup \{r\}$ contains only the circuit $C(r, B)$, $B(r) = B(p) \triangle B(q)$. □

Corollary 4.14 *If $\mathcal{M}$ is a binary matroid, $N = \mathcal{M}|T$ is a triangle, and $B \in \mathcal{B}(\mathcal{M})$ with $B \cap T = \emptyset$, then for all distinct $x, y, z \in T$, $B(x) - B(y)$ and $B(y) - B(x)$ partition $B(z)$.*

Lemma 4.15 *Let $\mathcal{M}$ be a matroid. If $X \subseteq B \in \mathcal{B}(\mathcal{M})$, then $B - X \in \mathcal{B}(\mathcal{M}/X)$.*

Proof. Clearly, $X \in \mathcal{I}(\mathcal{M}|X)$. Also, $\lambda_{\mathcal{M}|X}(X) = \lambda(X) - (E(\mathcal{M}) - X) \supseteq X$, so, $X \in \mathcal{B}(\mathcal{M}|X)$. Since $B = (B - X) \cup X$ and $B - X \subseteq E(\mathcal{M}) - X$, then $B - X \in \mathcal{B}(\mathcal{M}/X)$. □

Lemma 4.16 *Let $\mathcal{M}$ be a matroid, $X \subseteq B \in \mathcal{B}(\mathcal{M})$, and $e \in E(\mathcal{M}) - B$. Then, $C_{\mathcal{M}}(e, B) - X = C_{\mathcal{M}/X}(e, B - X)$.*

Proof. Suppose there is a $D \in \mathcal{C}(\mathcal{M}/X)$ with $D \subseteq C_{\mathcal{M}}(e, B) - X$. Then, $D = D' - X$ for some $D' \in \mathcal{C}(\mathcal{M})$ and $D' - X \subseteq B \cup \{e\}$. Since $X \subseteq B$, $D' \subseteq B \cup \{e\}$, and, by uniqueness of the fundamental circuit, $D' = C_{\mathcal{M}}(e, B)$. □

Theorem 4.17 *Let $\mathcal{M} = \mathcal{M}_1 \oplus_3 \mathcal{M}_2$, $N = \mathcal{M}_1|T = \mathcal{M}_2|T$, $T = \{p,q,r\}$, $B \in \mathcal{B}(\mathcal{M})$, λ the closure function of $\mathcal{M}$, λ_i the closure function of $\mathcal{M}_i$ ($i = 1,2$), $\eta \in E(\mathcal{M}_2) - B - \lambda_2(T)$, $e \in B(\eta) \cap E(\mathcal{M}_1)$, $H_e = \lambda(B - \{e\})$, $B_i \in \mathcal{B}(\mathcal{M}_i)$ a basis of $\mathcal{M}_i$ induced by B ($i = 1,2$). If $B_1 \cap T = \emptyset$, then, there is a a basis $B_2' \in \mathcal{B}(\mathcal{M}_2)$ induced by B such that $|B_2'(\eta) \cap T| = 2$, and there is a unique $x \in (B_2'(\eta) \cap T) - H_e^{(1)}$ and $H_e - B = (H_e^{(1)} \cup H_x^{(2)}) - T - B$, where $H_e^{(1)} = \lambda_1(B_1 - \{e\})$ and $H_x^{(2)} = \lambda_2(B_2' - \{x\})$.*

Proof. By definition, $H_e - B = \{g \in E(\mathcal{M}) - B : e \notin B(g)\}$, $(H_e - B) \cap E(\mathcal{M}_1) = \{g \in E(\mathcal{M}_1) - T - B_1 : e \notin B_1(g)\} = H_e^{(1)} - T - B_1$. It remains to analyze $(H_e - B) \cap E(\mathcal{M}_2)$. Let λ_η be the closure function of $\mathcal{M}_\eta = \mathcal{M}/A(\eta)$. Recall that $B(\eta) \in \mathcal{B}(\mathcal{M}_\eta)$ and let $H_e^\eta = \lambda_\eta(B(\eta) - \{e\})$; since $e \in B(\eta)$, $H_e - B = H_e^\eta - B(\eta)$. We will now consider $(H_e^\eta - B(\eta)) \cap E(\mathcal{M}_2)$.

Let $Y_i = B(\eta) \cap E(\mathcal{M}_i)$ and $r \in T$ such that $Y_i \cup \{r\} \in \mathcal{C}(\mathcal{M}_i)$ ($i = 1,2$). Since $B_1 \cap T = \emptyset$, then $|B_2 \cap T| = 2$. If $r \in B_2$, then there is a $y \in T - \{r\}$ with $y \notin B_2$ and so $r \in B_2(y)$, so that we can take $B_2' = (B_2 - \{r\}) \cup \{y\} \in \mathcal{B}(\mathcal{M}_2)$ and B_2' is a basis of $\mathcal{M}_2$ induced by B. Henceforth we will assume without loss of generality that $r \notin B_2$. Let $T = \{p,q,r\}$. Then, by lemma 4.12, $Y_2 \cup \{p,q,\eta\} = C(\eta, B_2)$ and $B_2(\eta) = Y_2 \cup \{p,q\}$.

We remark that $e \in Y_1 = B_1(r)$. By corollary 4.14, $B_1(p) - B_1(q)$ and $B_1(q) - B_1(p)$ partition $B_1(r) = Y_1$, which proves that there is a unique $x \in B_2(\eta) \cap T$ such that $e \in B_1(x)$, and so a unique $x \in (B_2(\eta) \cap T) - H_e^{(1)}$; for convenience, let us say that $x = p$. We notice that $B_1(r) = Y_1 \in \mathcal{B}(\mathcal{M}/A_1(r))$ and, by lemma 4.16, $e \in C_{\mathcal{M}_1}(p, B_1) \cap Y_1 = C_{\mathcal{M}_1/A_1(r)}(p, Y_1)$.

Notice that since $p, q \notin B$, $A(\eta) = A_1(r) \cup (A_2(\eta) - \{p,q\})$. By lemma 4.16 and corollary 4.14, $C_{\mathcal{M}_1/A_1(r)}(p, Y_1) = (C_{\mathcal{M}_1}(p, B_1) \cap Y_1) = Y_1 - C_{\mathcal{M}_1}(q, B_1) = Y_1 - ((C_{\mathcal{M}_1}(q, B_1) \cap Y_1) = Y_1 - C_{\mathcal{M}_1/A_1(r)}(q, Y_1)$, giving that

$$C_{\mathcal{M}_1/A_1(r)}(p, Y_1) \cap C_{\mathcal{M}_1/A_1(r)}(q, Y_1) = \emptyset \ . \tag{1}$$

For $g \in E(\mathcal{M}_2) - B - T$ define $Y_i(g) = C(g, B) \cap E(\mathcal{M}_i)$ and z to be the element in T so that $Y_i(g) \cup \{z\} \in \mathcal{C}(\mathcal{M}_i)$ ($i = 1,2$). Since $C_{\mathcal{M}_1}(z, B_1) = Y_1(g) \cup \{z\}$, by lemma 4.16, $C_{\mathcal{M}_1/A_1(r)}(z, Y_1) = (Y_1(g) \cap Y_1) \cup \{z\}$. Then, $B(g) \cap Y_1 = Y_1(g) \cap Y_1 = C_{\mathcal{M}_1/A_1(r)}(z, Y_1) - \{z\}$. So, $e \in B(g) \cap Y_1$ if and only if $e \in C_{\mathcal{M}_1/A_1(r)}(z, Y_1) - \{z\}$, and, by (1), $z \neq q$. We now prove that $z \neq q$ if and only if $p \in B_2(g)$, or, which is the same, $g \notin H_p^{(2)}$. If $z = p$, then $p \in Y_2(g) \cup \{p\} = B_2(g)$; if $z = r$, by lemma 4.12, $p \in Y_2(g) \cup \{p,q\} \in B_2(g)$. Conversely, if $p \in B_2(g)$, but we assume by contradiction that $Y_2(g) \cup \{q\} \in \mathcal{C}(\mathcal{M}_2)$, then we would obtain that $B_2(g) = Y_2(g) \cup \{q\} \not\ni p$. It follows that $(H_e^\eta - B) \cap E(\mathcal{M}_2) = \{g \in E(\mathcal{M}_2) - T - B : p \notin B_2(g)\} = H_p^{(2)} - T - B$. $\square$

5 Fundamental Circuits

In this section, we will prove theorem 3.1 on the maxima of fundamental circuits.

If $\mathcal{M}$ is a graphic matroid with $r(\mathcal{M}) = n$ and $|E(\mathcal{M})| = m$, Komlós' algorithm finds the fundamental circuit maxima with $O(m+n)$ comparisons [13]. Any algorithm for verifying the optimum basis of a graph can in fact be represented as a transmuter, for whose definition we refer to [18]. Let m' be the number of edges in the transmuter and n' the number of vertices of the transmuter that do not correspond to elements of $\mathcal{M}$; the number of comparisons is $m' - n' - (m - n + 1)$. Then, Komlós result can be restated by saying that there is a transmuter associated with $\mathcal{M}$ with $m' - n' - (m - n) - 1 = O(n + m)$.

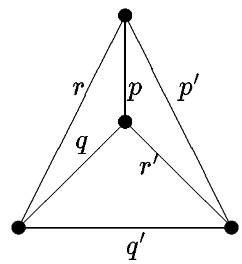

Figure 1: A ΔY Triangle and Star

Let $G = (V, E)$ be an undirected graph, T a spanning tree of G, and consider the problem of finding the fundamental circuit maxima of the cographic matroid $\mathcal{M} = \mathcal{M}(G)$. It is not hard to see that this problem can be restated as that of finding $\max\{w(f) : f \notin T, e \in T(f)\}$ for all $e \in T$. This problem can be solved by visiting in reverse order a transmuter [17]; in particular, if we visit Komlós' transmuter in reverse order, we obtain the maxima at the cost of $m' - n' - n + 1 = O(m+n)$ comparisons.

If the matroid $\mathcal{M}$ specified by theorem 3.1 is not connected, then the fundamental circuit maxima problem reduces to the same problem on each connected component of $\mathcal{M}$; from now on, we will assume that $\mathcal{M}$ is connected. The following algorithm establishes theorem 3.1 by finding the fundamental circuit maxima of a connected matroid obtained by means of 2-sums starting from matroids each of which is either regular or isomorphic to F_7 or F_7^*. In the algorithm, we will execute ΔY exchanges as defined in [19], and the elements involved to the exchange are named in agreement with figure 1. The reason why the algorithm executes ΔY exchanges will be explained when we argue its correctness. In the appendix, we will give a sample run of this algorithm.

Algorithm 2 *Fundamental circuit maxima.*

1. If $\mathcal{M}$ is graphic, solve the fundamental circuit maxima problem with Komlós' algorithms; if $\mathcal{M}$ is cographic, solve the fundamental circuit maxima problem with a reverse visit of Komlós' transmuter; if $\mathcal{M}$ is isomorphic to R_{10}, F_7, or F_7^*, solve the fundamental circuit maxima problem on $\mathcal{M}$ with a constant number of comparisons;

2. If $\mathcal{M}$ is not graphic, cographic, or isomorphic to R_{10}, F_7, or F_7^*, find $\mathcal{M}_1$ and $\mathcal{M}_2$ such that

   - $\mathcal{M} = \mathcal{M}_1 \oplus_k \mathcal{M}_2$ ($k \in \{2,3\}$), and
   - if $\mathcal{M}_2$ is graphic, there is no other decomposition $\mathcal{M} = \mathcal{M}_1' \oplus_j \mathcal{M}_2'$ with $j \in \{2,3\}$, $E(\mathcal{M}_2) \subset E(\mathcal{M}_2')$, and $\mathcal{M}_2'$ graphic,
   - if $\mathcal{M}_2$ is cographic, there is no other decomposition $\mathcal{M} = \mathcal{M}_1' \oplus_j \mathcal{M}_2'$ with $j \in \{2,3\}$, $E(\mathcal{M}_2) \subset E(\mathcal{M}_2')$, and $\mathcal{M}_2'$ cographic, and
   - $|E(\mathcal{M}_2)|$ is minimal

3. if $k = 2$ and the sum is unbalanced from $\mathcal{M}_i$ to $\mathcal{M}_j$ ($i, j \in \{1,2\}$), then solve recursively the fundamental circuit maxima problem on $\mathcal{M}_j$, assign the maximum along $B_j(p)$ to the basepoint p, and solve recursively the set maxima problem on $\mathcal{M}_i$;

4. if $k = 3$ and the sum is unbalanced from $\mathcal{M}_i$ to $\mathcal{M}_j$ $(i, j \in \{1,2\})$, let $T = \{p,q,r\} = E(\mathcal{M}_1) \cap E(\mathcal{M}_2)$ and $p, q \in B_i$, and
   (a) solve recursively the fundamental circuit maxima problem on $\mathcal{M}_j$,
   (b) if the maximum on $B_j(p)$ is equal to the maximum on $B_j(q)$ and greater than the maximum on $B_j(r)$, then execute a ΔY exchange on $\mathcal{M}_i$ along T, and, by defining elements as in figure 1, take $p', q', r' \in B_i$ and assign the weight of p' and q' as the maximum weight on $B_j(r)$ and the weight of r' as the maximum on $B_j(p)$; otherwise assign to p the maximum weight along $B_j(p)$ and to q the maximum weight along $B_j(q)$, finally take the basis of $\mathcal{M}_i$ after the ΔY exchange to be $(B_i - \{p,q\}) \cup \{p', q', r'\}$;
   (c) solve recursively the fundamental circuit maxima problem on $\mathcal{M}_i$;
5. if the sum is balanced, let $T = \{p,q,r\} = E(\mathcal{M}_1) \cap E(\mathcal{M}_2)$ and $p \in B_1$, $q \in B_2$, and
   (a) solve recursively the fundamental circuit problem on $\mathcal{M}_2$,
   (b) assign to $p \in B_1 \cap E(\mathcal{M}_2)$ the maximum weight of $B_2(p)$, solve recursively the fundamental circuit maxima problem on $\mathcal{M}_1$, and
   (c) update the solution in $\mathcal{M}_2$ by comparing the value of the solution found in step 5a with the maximum weight in $B_1(q)$ for all $e \in E(\mathcal{M}_2)$ such that $q \in B_2(e)$.

We now turn to prove the correctness of the algorithm.

Lemma 5.1 *After step 2 of the algorithm above, $\mathcal{M}_2$ is graphic or cographic or isomorphic to R_{10}, F_7, or F_7^*.*

Proof. Suppose to the contrary that $\mathcal{M}_2 = \mathcal{M}_{21} \oplus_j \mathcal{M}_{22}$. If $E(\mathcal{M}_1) \cap E(\mathcal{M}_2) \subseteq E(\mathcal{M}_{21})$, then $|E(\mathcal{M}_1 \triangle \mathcal{M}_{21}) \cap E(\mathcal{M}_{22})| = |E(\mathcal{M}_{21}) \cap E(\mathcal{M}_{22})|$ and, by associativity of symmetric difference, $\mathcal{M} = (\mathcal{M}_1 \oplus_k \mathcal{M}_{21}) \oplus_j \mathcal{M}_{22}$, which contradicts the minimality of $\mathcal{M}_2$.

Suppose now that $k = 3$, $T = E(\mathcal{M}_1) \cap E(\mathcal{M}_2)$, and that $T \cap E(\mathcal{M}_{2i}) \neq \emptyset$ $(i = 1, 2)$. Assume without loss of generality that $\{r\} = T \cap E(\mathcal{M}_{22})$ and that no minor of $\mathcal{M}_2$ is isomorphic to R_{10}, F_7, or F_7^*. We will now show that $\mathcal{M}_{22}$ is a planar matroid.

If $j = 2$, since T is a triangle of $\mathcal{M}_2$, then the basepoint p of the 2-sum is parallel to r. So, if $|E(\mathcal{M}_{22})| > 3$, there is a decomposition where r is deleted from $\mathcal{M}_{22}$ and added in parallel to p in $\mathcal{M}_{21}$, and again $\mathcal{M} = (\mathcal{M}_1 \oplus_3 \mathcal{M}_{21}) \oplus_2 \mathcal{M}_{22}$, contradicting the minimality of $\mathcal{M}_2$. If $|E(\mathcal{M}_{22})| = 3$, then $\mathcal{M}_{22}$ is planar.

If $j = 3$ and $|E(\mathcal{M}_{22})| > 7$, by lemma 4.9, there is a $z \in E(\mathcal{M}_{21}) \cap E(\mathcal{M}_{22})$ parallel to r, and again we can shift r to $\mathcal{M}_{21}$ resulting in $\mathcal{M} = (\mathcal{M}_1 \oplus_3 \mathcal{M}_{21}) \oplus_3 \mathcal{M}_{22}$, which contradicts the minimality of $\mathcal{M}_2$. Suppose now that $|E(\mathcal{M}_{22})| = 7$. Since $\mathcal{M}_{22}$ contains the triangle along which it is conjoined with $\mathcal{M}_{21}$, we obtain that $3 \leq E(\widetilde{\mathcal{M}_{22}}) \leq 6$. Also, $2 \leq r(\widetilde{\mathcal{M}_{22}}) \leq 4$ as $\mathcal{M}_{22}$ is connected and contains a triangle. If $r(\widetilde{\mathcal{M}_{22}}) = 2$, then $\mathcal{M}_{22}$ is the closure of a triangle and hence planar. If $r(\widetilde{\mathcal{M}_{22}}) = 3$, then $\widetilde{\mathcal{M}_{22}}$ is the deletion of at least one element from F_7 and hence $\mathcal{M}_{22}$ is planar. If $r(\widetilde{\mathcal{M}_{22}}) = 4$, since $\mathcal{M}_{22}$ contains a triangle, it is easily seen to be planar.

By repeating step 2 of the algorithm on $\mathcal{M}_{21}$, we can assume that $\mathcal{M}_2$ is the 2-sum or 3-sum of planar matroids plus one other component which is graphic or cographic, so also $\mathcal{M}_2$ is graphic or cographic and cannot be subdivided in step 2. $\square$

We can now prove the correctness of the algorithm. If $\mathcal{M} = \mathcal{M}_1 \oplus_2 \mathcal{M}_2$ with the sum unbalanced from $\mathcal{M}_i$ to $\mathcal{M}_j$ and p the basepoint of the 2-sum, for all $e \in E(\mathcal{M}_i)$, $B_i(e) \triangle B_j(p)$ is a circuit of $\mathcal{M}$ and is contained in $B \cup \{e\}$, which justifies step 3.

In the case of a 3-sum, $\mathcal{M}$ is a regular matroid. If $\mathcal{M} = \mathcal{M}_1 \oplus_3 \mathcal{M}_2$ with the sum balanced, the correctness follows from the observation that $B(e) = B_2(e) \triangle B_1(q)$ for $q \in B_2(e)$ and $B(e) = B_1(e) \triangle B_2(p)$ for $p \in B_1(e)$. If $\mathcal{M} = \mathcal{M}_1 \oplus_3 \mathcal{M}_2$ with the sum unbalanced from $\mathcal{M}_i$ to $\mathcal{M}_j$, then, for $e \in E(\mathcal{M}_i)$, $B(e) = B_i(e) \triangle B_j(z)$ for the z given by lemma 4.10. If $z \in B_i$, the algorithm could continue like before, but if $z \notin B_i$ and the maximum weight element of $B_j(p) \cup B_j(q)$ belongs to $B_j(p) \cap B_j(q)$ and we would attribute to $B_j(z)$ a weight that does not belong to it. In this case, we execute a ΔY exchange. $\mathcal{M}_i$ remains a regular matroid after the ΔY exchange [19], and, with the new definition of B_i, the fundamental circuits have the correct weights.

Now we will assess the number of comparisons executed by the algorithm. The second step of the algorithm does not execute any comparison on the elements in X. The number of comparisons executed in the other steps can be obtained by counting the number of times step 2 is executed. Recall [3] that

Lemma 5.2 *If $\mathcal{M}$ is a regular matroid and $|E(\mathcal{M})| \leq 10$, then $\mathcal{M}$ is either graphic, cographic, or isomorphic to R_{10}.*

If $\mathcal{M}'$ is a matroid obtained during the execution of the algorithm and if $\mathcal{M}'$ is in turn separated in step 2, it can be assumed that $|E(\mathcal{M}')| > |E(F_7)| = 7$. Also, we note that a ΔY exchange does not change the number of elements of a matroid.

Lemma 5.3 *Step 2 of algorithm 2 is executed at most $m - 7$ times.*

Proof. If $\mathcal{M}$ is graphic, cographic, or isomorphic to R_{10}, F_7, or F_7^*, the result is trivially true. Otherwise, notice that $|E(\mathcal{M}_1)| < |E(\mathcal{M})|$, and the result follows by induction hypothesis. □

Notice that each time step 2 is executed we introduce a constant number of elements. Therefore, the total number of elements in the matroids at the bottom of the recursion is $O(m)$. Analogously, the total number of basis elements in the matroids at the bottom of the recursion is $O(m + n)$. On each such a matroid of size m' and rank n', the algorithm runs with $O(m' + n')$ comparisons. The subsequent update of the solution performed when the sum is balanced requires at most $O(m)$ comparisons. By the previous lemma, the total number of comparisons needed to determine whether a ΔY exchange has to be performed is $O(m)$, yielding an $O(m + n)$ bound on the number of comparisons.

6 Fundamental Hyperplanes

In this section we will show an algorithm for the fundamental hyperplane maxima that establishes theorem 3.3. We will present our algorithm by degrees: first, we will exhibit an algorithm for finding the maxima on interval complements. We will then exploit this algorithm to find the fundamental hyperplane maxima of graphic and cographic matroid. Finally, we will utilize the graph algorithms in the procedure promised by the theorem.

6.1 Interval Complements

In this section, we will consider the fundamental hyperplane maxima problem in the case where $\mathcal{M} = \mathcal{M}[(I|D)]$ and D is an $m \times n$ binary matrix with the property that the 1's in each column occur in consecutive positions; matrices of this kind will be called *interval matrices*. We will use the algorithm on interval matrices as a component for the algorithms in the next sections. If D

is an interval matrix, then $\mathcal{M}[(I|D)]$ is graphic, but not necessarily planar: take for example the network matrix corresponding to K_5 with a path as spanning tree. A discussion and a bibliography on interval matrices can be found in [10].

If D is an $m \times n$ interval matrix, the hyperplane maxima problem on $\mathcal{M} = \mathcal{M}[(I|D)]$ can be restated as follows: let X be a set on which a total order is defined, $\mathcal{I} = \{I_1, \ldots, I_n\}$ a family of distinct intervals over $[m]$, $w : \mathcal{I} \to X$ an weight function on $\mathcal{I}$, and consider the problem of computing the values of the function

$$\mu(x) = \max\{w(I) : x \notin I \in \mathcal{I}\}$$

for all $x \in [m]$. We can assume without loss of generality that $[m] \notin \mathcal{I}$ as for no $x \in [m]$ the value of $\mu(x)$ depends on $w([m])$.

We will present an algorithm that solves this problem with $O(n + m)$ comparisons. First, we will partition $\mathcal{I}$ in large and small intervals, and find the value of μ relative to these subclasses, then we will combine the solutions at an additional cost of $O(m)$ comparisons. For this purpose, we introduce the following notation. Let

$$\mathcal{G} = \left\{I \in \mathcal{I} : |I| \le \frac{m}{2}\right\} ,$$

$$\mathcal{H} = \left\{I \in \mathcal{I} : |I| > \frac{m}{2}\right\} ,$$

and define

$$\mu_{\mathcal{G}}(x) = \max\{w(I) : x \notin I \in \mathcal{G}\} ,$$

$$\mu_{\mathcal{H}}(x) = \max\{w(I) : x \notin I \in \mathcal{H}\} .$$

Clearly, $\mu(x) = \max\{\mu_{\mathcal{G}}(x), \mu_{\mathcal{H}}(x)\}$, and the values of μ can be computed in $O(m)$ comparisons once $\mu_{\mathcal{G}}$ and $\mu_{\mathcal{H}}$ are known.

Consider first the problem of computing $\mu_{\mathcal{H}}$. Define $\cap\mathcal{H} = \bigcap_{H \in \mathcal{H}} H$. Since $|I| > m/2$ for any $I \in \mathcal{H}$, it follows that $\cap\mathcal{H} \neq \emptyset$. It is possible to compute $\mu_{\mathcal{H}}(x)$ for all $x \notin \cap\mathcal{H}$ with a double scan of $\mathcal{H}$.

Algorithm 3 *Computation of $\mu_{\mathcal{H}}(x)$.*

1. compute $\mu_{\mathcal{H}}(x)$ for all x that are less than any value in $\cap\mathcal{H}$:
   - (a) order $\mathcal{H}$ by decreasing values of starting point, that is, regard $\mathcal{H}$ as the sequence $\langle H_1, H_2, \ldots, H_k\rangle$ where $\min H_j \ge \min H_{j+1}$ for $j = 1, \ldots, k-1$;
   - (b) find the subsequence of indices $i_1, i_2, \ldots, i_h$ such that $i_1 = 1$ and i_{l+1} is the smallest index such that $i_{l+1} > i_l$ and $w(H_{i_{l+1}}) > w(H_{i_l})$;
   - (c) if $x < \min \cap\mathcal{H}$, then assign to $\mu_{\mathcal{H}}(x)$ the weight of the last interval in the sequence $\langle H_{i_1}, H_{i_2}, \ldots, H_{i_h}\rangle$ that does not contain x;
2. compute $\mu_{\mathcal{H}}(x)$ for all x that are greater than any value in $\cap\mathcal{H}$: symmetrical to the previous step.

The computation of the index sequences can be carried out by two linear scans of $\mathcal{H}$, hence the number of comparisons executed by this algorithm is $O(|\mathcal{H}|)$.

We turn now to the computation of $\mu_{\mathcal{G}}$. The algorithm will select the heaviest element I_0 in $\mathcal{G}$ and assign $\mu_{\mathcal{G}}(x) = w(I_0)$ to all $x \notin I_0$. Then, it will proceed recursively inside I_0, which will halve at each step the universe size. The computation of the new heaviest subinterval will be executed with particular care so as to exploit previously executed comparisons, thereby avoiding to add a logarithmic factor to the cost of the algorithm. While stating the algorithm, we will use the following conventions: if D is a sequence of intervals, $D \cap \mathcal{G}$ will denote the subsequence of D whose elements belong to $\mathcal{G}$; Tr will denote the trace function.

Algorithm 4 *Computation of $\mu_{\mathcal{G}}(x)$.*

1. If $\mathcal{G} \neq \emptyset$, find the interval I_0 such that $w(I_0) = \max\{w(I) : I \in \mathcal{G}\}$ as follows:
   (a) for $1 \leq i \leq m$ define $D_i = \langle [i-j, i+j] : 0 \leq j \leq \min\{i-1, m-i\}\rangle$, for $1 \leq i \leq m-1$ define $E_i = \langle [i-j, i+j+1] : 0 \leq j \leq \min\{i-1, m-i-1\}\rangle$, compute the maximum value of w on each initial prefix of $D_i \cap \mathcal{G}$ and $E_i \cap \mathcal{G}$ whose value is not already available; let $d_i = \max\{w(I) : I \in D_i \cap \mathcal{G}\}$, $1 \leq i \leq m$, let $e_i = \max\{w(I) : I \in E_i \cap \mathcal{G}\}$, $1 \leq i \leq m-1$, and $e_m = \min X$;
   (b) determine the interval I_0 whose weight is given by
$$w(I_0) = \begin{cases} \max(\{d_i, e_i : 1 \leq i \leq m\} \cup \{w(\emptyset)\}) & \text{if } \emptyset \in \mathcal{G} \\ \max\{d_i, e_i : 1 \leq i \leq m\} & \text{otherwise} \end{cases}$$
2. Let $\mu_{\mathcal{G}}(x) = w(I_0)$ for all $x \notin I_0$;
3. Compute $\mu_{\mathcal{G}}(x)$ for all $x \in I_0$:
   (a) for all $H \subset I_0$ such that there is an $I \in \mathcal{G}$ with $H = I \cap I_0$ determine $w'(H) = \max\{w(I) : I \in \mathcal{G}, I \cap I_0 = H\}$;
   (b) compute $\mu(x)$ for all $x \in I_0$ with interval set $Tr(\mathcal{G}, I_0) - \{I_0\}$ and weight function w'.

Initially, step 1 takes $O(n)$ comparisons. However, in a recursive call, the maxima of w on any proper initial prefix of $D_i \cap \mathcal{G}$ are already known, and it remains to compute the weight of the last interval of $D_i \cap \mathcal{G}$, which can be done with at most one more comparison. Since the number of points is now $|I_0|$, the total cost of $O(|I_0|)$. The maxima along all the initial prefixes of the E_i's can be analogously obtained in $O(|I_0|)$ comparisons. Also, once all d_i's and e_i's are known, the determination of the new I_0 will take $O(|I_0|)$ comparisons. Since $|I_0|$ is at least halved at each recursive call, the total number of comparisons in step 1 is $O(n) + O(m/2) + O(m/4) + \ldots + O(1) = O(n+m)$. In the computation of w' during step 3a the algorithm executes one comparison for each interval that is discarded for a total of $O(n)$ comparisons. The recursive computations of μ in step 3b breaks down into a call of algorithm 3, a recursive call of algorithm 1, and the comparisons of the values of $\mu_{\mathcal{H}}$ and $\mu_{\mathcal{G}}$. The invocations of algorithm 3 take up at most $O(n)$ comparisons throughout all recursive calls; $\mu_{\mathcal{H}}(x)$ and $\mu_{\mathcal{G}}(x)$ have to be compared only once for any $x \in [m]$ and this operation takes $O(m)$ throughout all the recursive calls of the algorithm. Finally, the recursive call of algorithm 1 have already been accounted for in the terms above.

On the whole, the total number of comparisons is $O(n+m)$.

The two previous algorithms can be implemented in $O(n+m)$ time as follows. In algorithm 3, the sorting steps according to initial and ending point can be carried out with counting sort in time $O(m) + O(n/2) + O(n/4) + \ldots + O(1) = O(m+n)$. In algorithm 4, for each value $i \in [n]$, we have a linked list of the intervals that begin at i; these lists are ordered by ending point. To obtain all these lists we sort first on ending point using counting sorting and then partitioning the result in the final lists. Analogously, for each value $j \in [n]$, we have a linked lists of the intervals that end at j; these lists are ordered by starting point. These lists can again be obtained with counting sorting. We will also construct $2n-1$ linked lists with the maxima on the prefixes of $D_i \cap \mathcal{G}$ and $E_i \cap \mathcal{G}$. These lists can be obtained by sorting by starting point with counting sorting, and then partitioning according to the value of the sum of the initial and ending point. The lists with $D_i \cap \mathcal{G}$ and $E_i \cap \mathcal{G}$ can be updated by visiting the other two sets of lists in $O(m)$ time. All other steps are clearly non-critical.

6.2 Graph Hyperplanes

In this section, we will give an algorithm to find the maxima of fundamental hyperplanes of a graphic matroid. The following observation is crucial for the algorithm: let η be the heaviest non-basis element and suppose that an orientation is fixed on $B(\eta)$. Then, for any non-basis elements f, $B(\eta) \cap B(f)$ is an interval of $B(\eta)$. So, the hyperplane maxima problem can be solved with the algorithm for the interval complement problem that was described in the previous section.

Algorithm 5 *Maxima of hyperplanes of graphic matroids.*

1. determine the heaviest non-basis element η; compute $B(\eta)$;
2. let $I_f = B(\eta) \cap B(f)$ for any $f \in E(\mathcal{M}) - B$, $\mathcal{I} = \{I_f : f \in E(\mathcal{M}) - (B \cup \{\eta\})\}$ and $w(I) = \max\{w(f) : I = I_f\}$; solve the interval complement problem on $\mathcal{I}$ and w and output $\max(H_e - B) = \mu(e)$.

The cost of the algorithm can be estimated as follows. The first step takes $m-1$ comparisons. In the second step, the computation of w requires one comparison for each edge that is discarded. Moreover, the number of intervals is less than n and the number of points is no more than $m+1$, so, the cost of solving the interval complement problem is $O(n+m)$ comparisons. On the whole, the algorithm takes $O(n+m)$ comparisons.

6.3 Graph Cohyperplanes

Let us turn now the hyperplanes of a cographic matroid. Let $G = (V, E)$ be a connected and undirected graph with a total order defined on the set E of edges, and let T the set of edges of a spanning tree of G. Each non-tree edge e determines the cohyperplane $H_e^* = E - C_{\mathcal{M}(G)}(e, T)$. Let $B = E - T$ be the basis of $\mathcal{M}^*(G)$ associated with T. Then, $H_e^* - B = H_e^* \cap T = T - T(e)$ where $e \notin T$.

If G is planar, then we can obtain the hyperplane maxima with algorithm 5 applied on the dual graph of G. If G is not planar, there could be no ordering of $B(\eta)$ that entails that the intersection of $B(\eta)$ with another edge cut $B(f)$ $(f \in T)$ is an interval. Consider for example $K_{3,3}$ as depicted in figure 2 with the spanning tree $T = \{\eta, a, b, c, d\}$.

However, if one endpoint r of η is a leaf of T, it is always possible to determine an ordering on $B(\eta)$ such that for any other $f \in T$, $B(\eta) \cap B(f)$ is an interval. First, notice that $B(\eta) \cap T = \emptyset$ and for $f \in T$, $B(f) = \{e \notin T : f \in T(e)\}$. Regard T as a directed graph with edges directed away from the root r. Visit T in depth first search starting from r and assign to each vertex v the timestamp (as defined in [7]) that records the moment when the inspection of v is completed. Then, the edge $(v, r) \in B(\eta)$ will precede all edges $(u, r) \in B(\eta)$ for which the timestamp of v is smaller than that

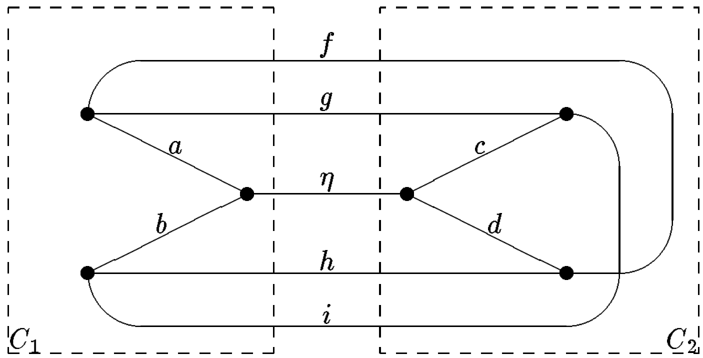

Figure 2: Critical edge cut in $K_{3,3}$

of u. If $f = (x, y)$ is a tree edge, then $B(\eta) \cap B(f)$ consists of all the edges (v, r) where v belongs to the subtree rooted at y, which, by the chosen ordering, is an interval.

By means of this ordering procedure, it is now possible to modify algorithm 5 for non-planar graphs. Indeed, when η has been found, $B(\eta)$ divides the graph G into two connected components C_1 and C_2. Let $N_2(v) = \{\{v, w\} : w \in C_2\}$ for $v \in C_1$. If $N_2(v) \neq \emptyset$, eliminate all edges in $N_2(v)$ but the heaviest. Then contract C_2 in one node r, and apply the ordering procedure above to the edges in $B(\eta)$. In this way, we obtain

$$s_e^1 = \max_{g \in (C_1 \cap T) - T(e)} w(g) .$$

Analogously, by contracting C_1, one obtains

$$s_e^2 = \max_{g \in (C_2 \cap T) - T(e)} w(g) .$$

Eventually, $\max(H_e^* - B) = \max\{s_e^1, s_e^2\}$. Clearly, the algorithm uses $O(n + m)$ comparisons.

6.4 Decomposable Matroids

We can now prove theorem 3.3 by showing an algorithm for the hyperplane maxima problem on decomposable matroids.

If $\mathcal{M}$ is graphic, cographic, or isomorphic to R_{10}, F_7, or F_7^*, then the solution is given by the algorithms in the previous sections or by sorting.

Let us turn now to the simple case of direct sums. Let $\mathcal{M} = \mathcal{M}_1 \oplus \mathcal{M}_2$, $B \in \mathcal{B}(\mathcal{M})$, and λ_i be the exchange closure function of $\mathcal{M}_i$ for $i = 1, 2$. Notice that $B_i = B \cap E(\mathcal{M}_i) \in \mathcal{B}(\mathcal{M}_i)$. So, if $e \in B_i$ $(i = 1, 2)$, then

$$H_e = \lambda(B - \{e\}) = \lambda_i(B_i - \{e\}) \cup \lambda_j(B_j) = H_e^{(i)} \cup E(\mathcal{M}_j) ,$$

where $\{j\} = [2] - \{i\}$ and $H_e^{(i)}$ is the hyperplane $\lambda_i(B_i - \{e\})$. Then,

Algorithm 6 *Hyperplanes of direct sums.*

1. determine the heaviest non-basis element η and assume that $\eta \in E(\mathcal{M}_i) - B_i$; set $\max(H_e - B) \leftarrow w(\eta)$ for any $e \in B - E(\mathcal{M}_i)$;

2. determine recursively the value of s_e for any $e \in B_i = B \cap E(\mathcal{M}_i)$.

We will henceforth assume that $\mathcal{M}$ is connected, and we will decompose the matroid $\mathcal{M}_\eta = \mathcal{M}/(B - B(\eta))$ in a way similar to that of step 2 of algorithm 2. Moreover, we will also maintain a threshold value Θ to account for the elements belonging to a hyperplane that is in some sense far away from the current matroid part; initially, we will assume that $\Theta = \min X$.

Algorithm 7 *Hyperplanes of matroid composition.*

1. If $\mathcal{M}_\eta$ is graphic, cographic, or isomorphic to R_{10}, F_7, solve the hyperplane maxima problem with one of the algorithms in the previous sections and let s_e the maximum over $H_e - B$ thus obtained; assign $\max(H_e - B) \leftarrow \max\{s_e, \Theta\}$;
2. If $\mathcal{M}_\eta$ is not graphic, cographic, or isomorphic to R_{10}, F_7, or F_7^*, find $\mathcal{M}_1$ and $\mathcal{M}_2$ such that
   - $\widetilde{\mathcal{M}_\eta} = \mathcal{M}_1 \oplus_k \mathcal{M}_2$ ($k \in \{2, 3\}$),
   - $\eta \in E(\mathcal{M}_1)$,
   - if $\mathcal{M}_2$ is graphic, there is no other decomposition $\widetilde{\mathcal{M}_\eta} = \mathcal{M}'_1 \oplus_j \mathcal{M}'_2$ with $j \in \{2, 3\}$, $E(\mathcal{M}_2) \subset E(\mathcal{M}'_2)$, and $\mathcal{M}'_2$ graphic,
   - if $\mathcal{M}_2$ is cographic, there is no other decomposition $\widetilde{\mathcal{M}_\eta} = \mathcal{M}'_1 \oplus_j \mathcal{M}'_2$ with $j \in \{2, 3\}$, $E(\mathcal{M}_2) \subset E(\mathcal{M}'_2)$, and $\mathcal{M}'_2$ cographic, and
   - $|E(\mathcal{M}_2)|$ is minimal
3. Define the *entrypoint* p_2 of $\mathcal{M}_2$ as follows:
   - if $k = 2$, then $p \in B_1 \cap E(\mathcal{M}_2)$ is the entrypoint p_2 of $\mathcal{M}_2$;
   - if $k = 3$ and $\{p, q\} = B_1 \cap E(\mathcal{M}_2)$, let $r \in B_1 \cap E(\mathcal{M}_2) - \{p, q\}$ be the entrypoint p_2 of $\mathcal{M}_2$;
   - if $k = 3$ and $\{p\} = B_1 \cap E(\mathcal{M}_2)$, then $p = p_2$ is the entrypoint of $\mathcal{M}_2$;
4. solve the hyperplane maxima problem on $\mathcal{M}_2$ with threshold Θ and
   - if the sum is unbalanced let $\Theta(\mathcal{M}_1)$ be equal to $\max(\{\Theta\} \cup E(\mathcal{M}_2))$;
   - otherwise, let $p \in B_2 \cap E(\mathcal{M}_1)$, and define $\Theta(\mathcal{M}_1)$ be equal to $\max(H_p^{(2)} - B_2)$, assign to p the maximum weight in $E(\mathcal{M}_2)$;
5. solve recursively the hyperplane maxima problem in $\mathcal{M}_1$ with threshold $\Theta(\mathcal{M}_1)$;
6. assign to $\mathcal{M}_2$ the contribution of the elements in $\mathcal{M}_1$ (p_2 is again the entrypoint of $\mathcal{M}_2$):
   - if $k = 2$, assign to any $e \in B_2(p_2)$ the maximum weight in $H_e^{(2)} \cup (H_{p_2}^{(1)} - B_1)$,
   - if $k = 3$ and the sum is unbalanced, let $T = E(\mathcal{M}_1) \cap E(\mathcal{M}_2)$ and for any $p \in T - \{p_2\}$ assign to any $e \in B_2(p)$ the maximum weight in $H_e^{(2)} \cup (H_p^{(1)} - B_1)$;
   - if $k = 3$ and the sum is balanced, assign to any element in $B_2(p_2)$ the maximum weight in $H_e^{(2)} \cup (H_{p_2}^{(1)} - B_1)$.

Lemma 6.1 *After step 2 of the algorithm above, $\mathcal{M}_2$ is graphic or cographic or isomorphic to R_{10}, F_7, or F_7^*.*

Proof. Same as that of lemma 5.1. $\square$

Moreover,

Lemma 6.2 *After step 2 of the algorithm above, the sum is not unbalanced from $\mathcal{M}_2$ to $\mathcal{M}_1$.*

Proof. By induction on the number of recursive calls to the algorithm. Suppose that there is a sum $\widetilde{\mathcal{M}_\eta} = \mathcal{M}_1 \oplus_k \mathcal{M}_2$ that is unbalanced from $\mathcal{M}_2$ to $\mathcal{M}_1$. Then $B(\eta) \subseteq E(\mathcal{M}_1)$ and $B_2 \cap B(\eta) = \emptyset$, so $E(\mathcal{M}_2) \subseteq \lambda_2(T)$ where $T = E(\mathcal{M}_1) \cap E(\mathcal{M}_2)$. If $k = 2$, then $T = \{p\}$, $E(\mathcal{M}_2)$ is the parallel class of p, and $|E(\mathcal{M}_2)| \geq 3$, so that $\widetilde{\mathcal{M}_\eta}$ is not simple. Analogously, if $k = 3$, since $E(\mathcal{M}_2) \subseteq \lambda_2(T)$ and $|E(\mathcal{M}_2)| \geq 7$, it easy to see that $\widetilde{\mathcal{M}_\eta}$ is not simple. $\square$

The correctness of the algorithm easily follows from theorem 4.17 and lemma 6.2. It remains to estimate the number of comparisons executed.

Lemma 6.3 *Step 2 of algorithm 2 is executed at most $m - 7$ times.*

Proof. Same as that of lemma 5.3. $\square$

The complexity of the algorithm follows from an argument similar to that in section 5. The total number of elements in the matroids at the bottom of the recursion is $O(m)$, and so the maximum weight element of each part of $\mathcal{M}_\eta$ can be computed in linear time. Analogously, the updates in step 4 throughout all executions of the algorithm can be executed by examining each element at most once. It follows that the total number of comparisons is $O(n + m)$.

7 Open Problems

Because of the striking success of Komlós' algorithm in the solution of the minimum spanning tree problem, an application of our result on fundamental circuits would be to extend it to the sensitivity analysis and to the construction of optimum bases for classes of matroids more general than graphs.

The obvious open question is to find an algorithm to solve the set maxima problem in its generality. As a more realistic objective, we would like to find an algorithm for the maxima of the hyperplanes of $PG(d, q)$ when both the parameters q and d vary (the only algorithm known so far assumes that d is a constant [2]), and for the maxima of nearly disjoint hyperplanes.

Acknowledgments

We are indebted with Mike Fredman for having suggested us the problem, for countless hours of discussion, and for proofreading an earlier draft of this paper. We also thank Sarmad Abbasi, Jeff Kahn, S. Muthukrishnan, and Lorant Porkolab for helpful discussions.

References

[1] Ahuja, R. K., Magnanti, T. L., and Orlin, J. B. *Network Flows.* Prentice-Hall, Englewood Cliff, NJ, 1993.

[2] Bar-Noy, A., Motwani, R., and Naor, J. A linear time approach to the set maxima problem. *SIAM J. Disc. Math. 5*, 1 (Feb. 1992), 1–9.

[3] Bixby, R. E. Kuratowski's and Wagner's theorems for matroids. *Journal of Combinatorial Theory, Series B 22*, 1 (Feb. 1977), 31–53.

[4] Brylawski, T. A combinatorial model for series-parallel networks. *Transactions of the American Mathematical Society 154* (Feb. 1971), 1–22.

[5] Brylawski, T. Modular construction for combinatorial geometries. *Transactions of the American Mathematical Society 203* (Mar. 1975), 1–44.

[6] Brylawski, T. Constructions. In *Theory of Matroids*, N. L. White, Ed., vol. 26 of *Encyclopedia of Mathematics and Its Applications*. Cambridge University Press, 1986, ch. 7.

[7] Cormen, T. H., Leiserson, C. E., and Rivest, R. L. *Introduction to Algorithms.* MIT Press and Mc-Graw Hill, 1990.

[8] Desper, R. M. The set-maxima problem: an overview. Master's thesis, Rutgers University, Mar. 1994.

[9] Goddard, W., Kenyon, C., King, V., and Schulman, L. J. Optimal randomized algorithms for local sorting and set maxima. *SIAM J. Comput. 22*, 2 (Apr. 1993), 272–283.

[10] Golumbic, M. C. *Algorithmic Graph Theory and Perfect Graphs.* Computer Science and Applied Mathematics. Academic Press, New York, 1980.

[11] Graham, R. L., Yao, F. F., and Yao, A. C. Information bounds are weak in the shortest distance problem. *Journal of the Association for Computing Machinery 27*, 3 (July 1980), 428–444.

[12] Karger, D. R., Klein, P. N., and Tarjan, R. E. A randomized linear-time algorithm to find minimum spanning trees. *Journal of the Association for Computing Machinery 42*, 2 (Mar. 1995), 321–328.

[13] Komlós, J. Linear verification for spanning trees. *Combinatorica 5*, 1 (1985), 57–65.

[14] Oxley, J. G. *Matroid Theory.* Oxford University Press, 1992.

[15] Seymour, P. D. The matroids with the max-flow min-cut property. *Journal of Combinatorial Theory, Series B 23*, 2–3 (October–December 1977), 189–222.

[16] Seymour, P. D. Decomposition of regular matroids. *Journal of Combinatorial Theory, Series B 28*, 3 (June 1980), 305–359.

[17] Tarjan, R. E. Applications of path compressions on balanced trees. *Journal of the Association for Computing Machinery 26*, 4 (Oct. 1979), 690–715.

[18] Tarjan, R. E. Sensitivity analysis of minimum spanning trees and shortest path trees. *Information Processing Letters 14*, 1 (Mar. 1982), 30–33.

[19] Truemper, K. *Matroid Decomposition.* Academic Press, Boston, 1992.

[20] Tseng, F. T., and Truemper, K. A decomposition of the matroids with the max-flow min-cut property. *Discrete Applied Mathematics 15*, 2, 3 (Nov. 1986), 329–364.

[21] Welsh, D. J. A. *Matroid Theory.* Academic Press, London, 1976.

A An Example

In this appendix, we will describe an application of algorithm 2. Let $X = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$ with $x_1 < x_2 < x_3 < x_4 < x_5 < x_6 < x_7$, and $\mathcal{S} = \{S_1, S_2, S_3, S_4, S_5, S_6\}$ with

$$\begin{aligned} S_1 &= \{x_1, x_4, x_5, x_6\}, \\ S_2 &= \{x_1, x_2, x_6, x_7\}, \\ S_3 &= \{x_2, x_3, x_6, x_7\}, \\ S_4 &= \{x_3, x_4, x_5, x_6\}, \\ S_5 &= \{x_5, x_6\}, \\ S_6 &= \{x_4, x_7\}\ . \end{aligned}$$

The set system $\mathcal{S}$ is the set of fundamental circuit of the binary matroid $\mathcal{M}$ whose standard representative is

$$\left(\begin{array}{ccccccc|cccccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{array}\right) .$$

Execute now the following elementary row operations: exchange row 5 and 7, add the 5th row to the 6th, then add the 4th row to the 5th, 6th, and 7th, and finally rearrange the columns so that in the following matrix the column correspond from left to right to x_1, x_2, x_3, S_1, S_2, S_3, S_4, x_6, x_5, x_7, S_5, S_6, x_4:

$$\left(\begin{array}{ccccccc|cccccc} 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \cline{8-13} 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ \cline{1-7} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{array}\right) .$$

It follows that $\mathcal{M} = \mathcal{M}(G) \oplus_3 \mathcal{M}^*(K_{3,3})$, where G is the graph in figure 3 and $E(\mathcal{M}(G)) \cap E(\mathcal{M}^*(K_{3,3})) = \{p, q, r\}$. The basis of $\mathcal{M}^*[K_{3,3}]$ induced by the basis of B is $\{x_4, x_5, x_6, x_7\}$, the basis of $\mathcal{M}(G)$ induced by the basis of B is $\{x_1, x_2, x_3, p, q\}$, and the sum is unbalanced from $\mathcal{M}(G)$ to $\mathcal{M}^*(K_{3,3})$. So, we solve the fundamental circuit maxima on $\mathcal{M}^*[K_{3,3}]$ first by visiting Komlós' transmuter in reverse order and we obtain that

$$\begin{aligned} \max B(p) &= x_7 \\ \max B(q) &= x_7 \\ \max B(r) &= x_6 \end{aligned}$$

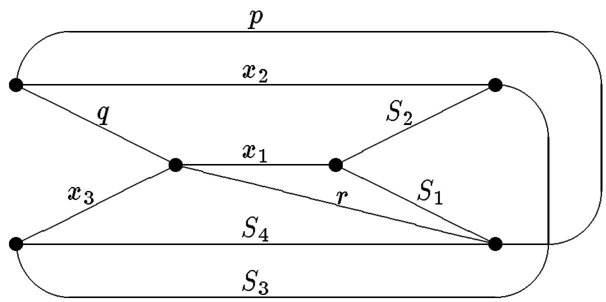

Figure 3: Sample graph G

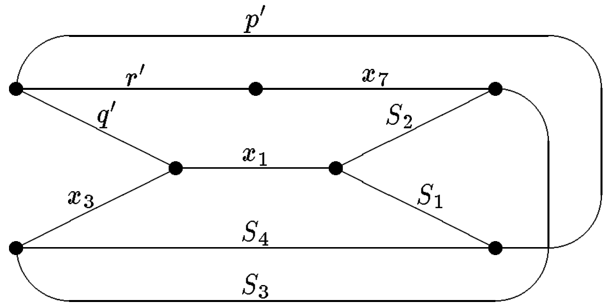

Figure 4: G' obtained with a ΔY exchange

$$
\begin{aligned}
\max B(S_5) &= x_6 \\
\max B(S_6) &= x_7
\end{aligned}
$$

Since $\max B(p) = \max B(q) > \max B(r)$, we have to execute a ΔY exchange on $\mathcal{M}(G)$ along p, q, and r, which yields the graph G' in figure 4. Assign the value of p' and q' to be x_6 and the value of r' to be x_7, and apply Komlòs' algorithm with respect to the basis $\{x_1, x_2, x_3, x_4, p', q', r'\}$ to obtain

$$
\begin{aligned}
\max B(S_1) &= x_6 \\
\max B(S_2) &= x_7 \\
\max B(S_3) &= x_7 \\
\max B(S_4) &= x_6
\end{aligned}
$$