Algorithms for Enumerating Circuits in Matroids*

Endre Boros[1], Khaled Elbassioni[1], Vladimir Gurvich[1], and Leonid Khachiyan[2]

[1] RUTCOR, Rutgers University, 640 Bartholomew Road, Piscataway NJ 08854-8003; {boros,elbassio,gurvich}@rutcor.rutgers.edu

[2] Department of Computer Science, Rutgers University, 110 Frelinghuysen Road, Piscataway NJ 08854-8003; leonid@cs.rutgers.edu

Abstract. We present an incremental polynomial-time algorithm for enumerating all circuits of a matroid or, more generally, all minimal spanning sets for a flat. This result implies, in particular, that for a given infeasible system of linear equations, all its maximal feasible subsystems, as well as all minimal infeasible subsystems, can be enumerated in incremental polynomial time. We also show the NP-hardness of several related enumeration problems.

1 Introduction

Let M be a matroid on ground set S of cardinality $|S| = n$, i.e. a collection of subsets of S satisfying (i) $\emptyset \in M$, (ii) if $X \in M$ and $Y \subseteq X$ then $Y \in M$, and (iii) if $X, Y \in M$ and $|Y| > |X|$ then there exists an element $y \in Y \setminus X$ such that $X \cup \{y\} \in M$. Elements of M are called the *independent sets* of M. We assume throughout the paper that M is defined by an *independence oracle*, i.e. an algorithm $\mathcal{I}$ which, given a subset X of S, can determine in unit time whether or not X is independent in M. This assumption implies that the rank of any set $X \subseteq S$, $r(X) = \max\{|I| \ : \ I \text{ independent subset of } X\}$, and in particular, the rank of the matroid $r(M) \stackrel{\text{def}}{=} r(S)$ can be determined in $O(n)$ time by the well-known greedy algorithm. Hence the rank of X in the dual matroid M^* (that is, the matroid whose maximal independent sets are the complements of the maximal independent sets of M) $r^*(X) = r(S \setminus X) + |X| - r(M)$, can also be computed in $O(n)$ time. In particular, $\mathcal{I}$ can be used as an independence oracle for the dual matroid.

Let $\mathcal{C}(M)$ be the family of all circuits of M, i.e. the family of all minimal dependent subsets of S, and let $\mathcal{B}(M)$ be the family of all bases of M, i.e., the collection of all maximal independent sets. By definition, $\mathcal{C}(M)$ and the family

* This research was supported in part by the National Science Foundation Grant IIS-0118635. The research of the first and third authors was also supported in part by the Office of Naval Research Grant N00014-92-J-1375. The second and third authors are also grateful for the partial support by DIMACS, the National Science Foundation's Center for Discrete Mathematics and Theoretical Computer Science.

T. Ibaraki, N. Katoh, and H. Ono (Eds.): ISAAC 2003, LNCS 2906, pp. 485–494, 2003.

$\mathcal{B}(M^*) = \{X : S \setminus X \in \mathcal{B}(M)\}$ of bases of the dual matroid M^* are mutually transversal hypergraphs.

It is a folklore result that all bases of a matroid M can be enumerated with polynomial delay, i.e. in $poly(n)$ time per each generated base. This can be done by traversing the connected "metagraph" $\mathcal{G} = (\mathcal{B}(M), \mathcal{E})$ in which two "vertices" $B, B' \in \mathcal{B}(M)$ are connected by an edge in $\mathcal{E}$ iff B and B' can be obtained from each other by exchanging a pair of elements, i.e. when $|B \setminus B'| = |B' \setminus B| = 1$. The connectivity of $\mathcal{G}$ then follows from the well-known *base axiom*:

If $B, B' \in \mathcal{B}(M)$ *and* $x \in B' \setminus B$ *then* $(B \cup y) \setminus x \in \mathcal{B}(M)$ *for some* $y \in B \setminus B'$.

When M is the cycle matroid of a given graph $G = (V, E)$ and $\mathcal{C}(M)$ is the family of all simple cycles of G, all elements of $\mathcal{C}(M)$ can also be enumerated with polynomial delay (see e.g. [9]). This is also true for M^*, the cocycle matroid of G, when each element of $\mathcal{C}(M^*)$ is a minimal set of edges whose removal increases the number of connected components of G (see e.g. [8]). In general, however, we are not aware of any polynomial-delay algorithm for enumerating all circuits of an arbitrary matroid M. Intuitively, the circuit enumeration problem seems to be harder than the base enumeration due to the fact that $|C(M)| \leq (n - r(M))|\mathcal{B}(M)|$, whereas in general, $|\mathcal{B}(M)|$ cannot be bounded by a polynomial in n and $|\mathcal{C}(M)|$. In addition, there is a combinatorial reduction which reduces the enumeration of all bases of a matroid to the enumeration of all circuits of another matroid (see Section 5).

In this paper we present a simple algorithm for enumerating all circuits of an arbitrary matroid M in *incremental polynomial time*, i.e. show that for each $k \leq |\mathcal{C}(M)|$, one can compute k circuits of M in $poly(n, k)$ time. This is done in Section 2. By duality, this result also gives an incremental polynomial time algorithm for enumerating all hyperplanes or, more generally, all flats of a given rank in M or M^*. Thus, any level of the lattice of flats of M can be produced in incremental polynomial time.

In Section 3 we consider the enumeration of all circuits of M which contain a given element $a \in S$. Again, we show that all circuits through a can be enumerated in incremental polynomial time, and discuss some dual formulations of this result. We are not aware of any efficient algorithm for enumerating all circuits containing $t \geq 2$ elements of a given matroid M. In Section 4 we argue that this problem can be solved with polynomial delay for each fixed t when M is the cycle or cocycle matroid of a given graph, but becomes NP-hard when t is part of the input. Section 5 deals with the enumeration of all minimal subsets X of a given set $D \subseteq S$ such that X spans a given flat A of M. Examples of such spanning sets include generalized Steiner trees and multiway cuts in graphs. We reduce the enumeration problem for minimal A-spanning sets to the generation of all circuits through a given element in some extended matroid, and hence obtain an incremental polynomial-time algorithm. All maximal subsets of a given set D which do not span A can also be enumerated in incremental polynomial time.

Finally, Section 6 discusses some variants of the circuit enumeration problem for two matroids on S. We also discuss generalized circuits whose definition is obtained by replacing some singletons of S by subsets, i.e., by performing the parallel extension of the rank function $r(X)$ for some sets $A_1, ..., A_n \subseteq S$. We show that the enumeration problems corresponding to these variants and generalizations of circuits are all NP-hard already for graphic and cographic matroids. By duality, this is also true for analogous problems stated in terms of generalized hyperplanes.

2 Enumeration of All Circuits of a Matroid

Let M be a matroid defined by an independence oracle on ground set S of size n, and let $\mathcal{C}(M) \subseteq 2^S$ be the family of all circuits of M.

Theorem 1. *For each $k \leq |\mathcal{C}(M)|$, computing k circuits of M can be carried out in $poly(n, k)$ time.*

Proof. If B is a base of M and $x \in S \setminus B$ then there exists a unique circuit $C = C(B, x)$ such that $x \in C \subseteq B \cup x$. This circuit $C(B, x)$, called the fundamental circuit of x in the base B, can be computed by querying the independence oracle on at most $|B|$ subsets of $B \cup x$. We start by constructing a base B^o of M and the system $\mathcal{F}(B^o) = \{C(B^o, x) \mid x \in S \setminus B^o\}$ of $n - r(M)$ fundamental circuits for B^o. This can be done in $poly(n)$ time. Next, the family $\mathcal{C}(M)$ of circuits of any matroid satisfies the *circuit axiom*:

If C_1 and C_2 are distinct circuits of M and $e \in C_1 \cap C_2$ there exists a circuit C_3 such that $C_3 \subseteq (C_1 \cup C_2) \setminus e$.

Given an arbitrary collection $\mathcal{C}'$ of k circuits of M we can check in $poly(n, k)$ time whether or not $\mathcal{C}'$ is closed with respect to the circuit axiom, i.e., for any two distinct circuits $C_1, C_2 \in \mathcal{C}'$ with a common element $e \in C_1 \cap C_2$ the given collection $\mathcal{C}'$ also contains a circuit $C_3 \subseteq (C_1 \cup C_2) \setminus e$. To enumerate all circuits in M we start with the fundamental system of circuits $\mathcal{C}' = \mathcal{F}(B^o)$ and repeatedly check whether $\mathcal{C}'$ is closed with respect to the circuit axiom. Since each violation of the circuit axiom produces a new circuit, it remains to show that if some system $\mathcal{C}'$ of circuits is closed with respect to the circuit axiom and $\mathcal{F}(B^o) \subseteq \mathcal{C}'$ then $\mathcal{C}' = \mathcal{C}(M)$. This follows from the fact that any set system $\mathcal{C}' \subseteq 2^S$ satisfying the circuit axiom and the Sperner condition $C_1, C_2 \in \mathcal{C}',\ C_1 \neq C_2 \Longrightarrow C_1 \not\subseteq C_2$ defines a matroid M' on S, see [6,11]. By definition, the bases of M' are all maximal independent sets for $\mathcal{C}'$, i.e. all those maximal subsets of S which contain no set in $\mathcal{C}'$. In our case $\mathcal{C}' \subseteq \mathcal{C}(M)$ and hence $\mathcal{C}'$ is Sperner by definition. Furthermore, since $\mathcal{C}'$ contains the fundamental system of circuits for $B^o \in \mathcal{B}(M)$, it follows that B^o is also a base of M', implying that the ranks of M and M' are equal. Let $C \in \mathcal{C}(M)$ be an arbitrary circuit of M, then C is the fundamental circuit for some base $B \in \mathcal{B}(M)$ and some element $x \in S \setminus B$, i.e. $C = C(B, x)$. Since B is independent in M' and $|B| = r(M) = r(M')$, we

conclude that $B \in \mathcal{B}(M')$. Now M' must also contain a unique fundamental circuit $C' = C'(B, x)$. Since any circuit of M' is also a circuit of M, we conclude that $C = C(B, x) = C'(B, x)$, which shows that $C \in \mathcal{C}' = \mathcal{C}(M')$. □

Let $\mathcal{H}_t(M) = \{X \ : \ X \text{ maximal subset of } S \text{ such that } r(X) \leq t\}$ be the family of all flats of rank t in M, where t is an integer threshold. In particular, when $t = rank(M) - 1$ the family $\mathcal{H}_t(M)$ consists of all hyperplanes of M. Let also $\mathcal{C}_t(M) = \{X \ : \ X \text{ minimal subset of } S \text{ such that } r(X) \leq |X| - t\}$, so that $\mathcal{C}_1(M) = \mathcal{C}(M)$ is exactly the family of all circuits of M.

Corollary 1. *Given an integer parameter t, all flats in $\mathcal{H}_t(M)$ can be enumerated in incremental polynomial time. Similarly, all elements of $\mathcal{C}_t(M)$ can also be enumerated in incremental polynomial time.*

Proof. Since each hyperplane of M is the complement of a cocircuit of M and vice versa, the enumeration of all hyperplanes of M is equivalent with the circuit enumeration for the dual matroid M^*. Hence by Theorem 1 all hyperplanes of M can be enumerated in incremental polynomial time. Furthermore, the corollary also holds for the family $\mathcal{H}_t(M)$ of all flats of rank t, because $\mathcal{H}_t(M)$ consists of all hyperplanes of the truncated matroid M_{t+1} whose rank function is defined by $r_{t+1}(X) = \min\{r(X), t+1\}$. Finally, let $\tau = |S| - r(M) - t$ then enumerating all flats of rank τ for M^* is equivalent with the enumeration of all maximal solutions $Y \subseteq S$ to the inequality $r^*(Y) = r(S \setminus Y) + |Y| - r(M) \leq \tau$. The latter problem is in turn equivalent with the enumeration of all minimal solutions $X = S \setminus Y$ to the inequality $r(X) \leq |X| - t$. □

By Corollary 1 the lattice $\mathcal{L}(M)$ of flats of any matroid M can be computed in incremental polynomial time. It is known [5] that $|\mathcal{L}(M)| \geq 2^{r(M)}$.

3 Circuits through a Given Element

An important open question in linear programming is whether there exists an efficient way to enumerate all vertices of a given polytope

$$P = \{x = (x_1, \ldots, x_n) \in \Re^n \ : \ \sum_{i=1}^{n} a_i x_i = a, \quad x_1, \ldots, x_n \geq 0 \ \},$$

where $a, a_1, \ldots, a_n$ are given d-dimensional vectors. Each vertex of P can be identified with a minimal supporting set I of coordinates $[n] = \{1, \ldots, n\}$ for which the system of linear equations

$$\sum_{i \in I} a_i x_i = a \tag{1}$$

has a positive real solution. Dropping the non-negativity conditions we arrive at the problem of enumerating all minimal sets $I \subseteq [n]$ for which (1) has a real solution. This is equivalent with the enumeration of all those circuits of the vectorial matroid $M = \{a, a_1, \ldots, a_n\} \subseteq \Re^d$ that contain a. When M is the

cycle or cocycle matroid of some connected graph $G = (V, E)$ and $a = (uv) \in E$ is an edge with endpoints $u, v \in V$, enumerating all circuits through a calls for computing all simple uv-paths or all minimal uv-cuts in G, which can be done with polynomial delay [9]. The following result indicates that all circuits through a given element a can be efficiently enumerated for any matroid M.

Theorem 2. *Let M be a matroid with ground set S, let $a \in S$, and let $\mathcal{C}(M, a)$ the set of circuits C of M such that $a \in C$. Assuming that M is defined by an independence oracle, all elements of $\mathcal{C}(M, a)$ can be enumerated in incremental polynomial time.*

Proof. Two elements $x, y \in S$ are said to be *connected* in M if either $x = y$ or there is a circuit $C \in \mathcal{C}(M)$ containing both x and y. We may assume w.l.o.g. that M is connected. Given a set $X \subseteq S$, let $D(X) = X \setminus \bigcap\{C \in \mathcal{C}(M, a) \,:\, C \subseteq X\}$, where as before $\mathcal{C}(M, a)$ denotes the set of all circuits containing a. Lehman's theorem [6,11] asserts that for any connected matroid M the circuits of M not containing a are precisely the minimal sets of the form $D(C_1 \cup C_2)$ where C_1 and C_2 are distinct members of $\mathcal{C}(M, a)$. Hence for any connected matroid M:

$$|\mathcal{C}(M)| \leq |\mathcal{C}(M, a)|\big(|\mathcal{C}(M, a)| + 1\big)/2.$$

This bound and Theorem 1 readily imply that all circuits in $\mathcal{C}(M, a)$ can be enumerated in output polynomial time $poly(|\mathcal{C}(M, a)|)$ by simply generating all circuits in $\mathcal{C}(M)$ and discarding those of them that do not pass through a. In fact, since our enumeration problem is self-reducible, the above bound also implies an incremental polynomial-time algorithm. To see this, assume that we wish to enumerate a given number k of circuits in $\mathcal{C}(M, a)$, or list all of them if $k \geq |\mathcal{C}(M, a)|$. Since for each integer $k' \leq |\mathcal{C}(M)|$ we can obtain k' circuits in $\mathcal{C}(M)$ in $poly(n, k')$ time, we can decide whether or not $k \geq |\mathcal{C}(M, a)|$ by attempting to generate $k' = k(k+1)/2$ circuits in $\mathcal{C}(M)$, in time bounded by a polynomial in n and k. If we discover that $|\mathcal{C}(M)| \leq k(k+1)/2$ by producing all circuits in $\mathcal{C}(M)$ then we also have the entire set $\mathcal{C}(M, a)$. Suppose now that we have computed $k(k+1)/2$ circuits in $\mathcal{C}(M)$ but fewer than k of them pass through a. Let $b \neq a$ be another element of S. Delete b and compute the connected component S' which contains a in the matroid M restricted to $S \setminus b$. Note that any circuit of $\mathcal{C}(M, a)$ which does not contain b must belong to S'. So we may apply the same procedure to the connected matroid M' obtained by restricting M on S', and either obtain all circuits of $\mathcal{C}(M, a)$ which avoid b, or conclude that the number of such circuits exceeds k. Since in the latter case we can reduce the size of S by removing b for good (as long as we are not required to produce more than k circuits of $\mathcal{C}(M, a)$), we may now assume w.l.o.g. that for each element $b \neq a$ we have obtained all the circuits in $\mathcal{C}(M, a)$ which avoid b. This means that in time polynomial in n and k we can produce all circuits in $\mathcal{C}(M, a)$ which skip some element of S. Unless S itself is the only element of $\mathcal{C}(M, a)$, this gives the entire set $\mathcal{C}(M, a)$. □

By duality, Theorem 2 gives an incremental polynomial-time algorithm for enumerating all hyperplanes (or, more generally, all flats of a given rank t) which

do not contain a. Needless to say that all hyperplanes (or flats of rank t) which *contain* an arbitrary set of elements $A \subseteq S$ can be enumerated in incremental polynomial time because this is equivalent with enumerating all circuits of the (truncated) matroid M restricted to $S \setminus A$.

It is also worth mentioning that $\{C \setminus \{a\} \mid C \in \mathcal{C}(M, a)\}$ and $\{C' \setminus \{a\} \mid C' \in \mathcal{C}(M^*, a)\}$ form a pair of mutually transversal Sperner hypergraphs. For instance, these hypergraphs consist of all uv-paths and all uv-cuts respectively, when M is a cycle matroid of a connected graph $G = (V, E)$ in which edge $a = (uv)$ connects vertices $u, v \in V$.

4 Circuits through t Elements

It is natural to ask what is the complexity of enumerating all circuits of M which contain a given set $A = \{a_1, \ldots, a_t\}$ of $t \geq 2$ elements of S. As we argue below, this problem is NP-hard when t is part of the input but can be solved with polynomial delay if $t = |A|$ is fixed and M is the cycle or cocycle matroid of a given graph $G = (V, E)$. However, we are not aware of an efficient algorithm for listing all circuits through $t = const \geq 2$ elements of arbitrary matroids.

Let M be the cycle matroid of G so that the circuits of M are the simple cycles of G. An edge set A may be contained in a simple cycle only if A itself is a simple cycle or A is a union of k pairwise vertex disjoint simple paths $P_1, \ldots, P_k$ for some integer positive $k \leq t$. All simple cycles containing $P_1, \ldots, P_k$ can be enumerated with polynomial delay via lexicographic backtracking [9] by growing and merging these partial paths (so that their number continually decreases). Hence backtracking listing algorithms reduce the enumeration of simple cycles containing $a_1, \ldots, a_t$ to the following decision problem:

> Does there exist a simple cycle in G which contains k given disjoint paths $P_1, \ldots, P_k$?

When k is fixed, by considering all possible permutations and reversals of $P_1, ..., P_k$ the latter problem can in turn be polynomially reduced to the well-known *disjoint-path problem*:

> Given k pairs of vertices $\{u_i, v_i\}$, $i = 1, \ldots, k$ of a graph, can these pairs be connected by k pairwise vertex disjoint paths?

Even though the disjoint path problem is NP-complete when k is part of the input (see [4]), it is known [10] to be solvable in polynomial time for each fixed k. Hence all simple cycles through $t = const$ edges can be enumerated with delay bounded by a polynomial in the size of the input graph.

As we mentioned earlier, if $t = |A|$ is part of the input then the problem of enumerating all simple cycles through t edges of a graph becomes NP-complete. In fact, given a graph $G = (V, E)$ and a (large) matching $A \subset E$ it is NP-hard to decide whether G has *any* simple cycle containing A. This can be seen from the following argument. Given a graph $H = (U, E)$, substitute an edge e_u for each vertex $u \in U$. Then, unless G consists of a single edge, the resulting graph

$G = P_2 \times H$ has a simple cycle through the matching $A = \{e_u : u \in U\}$ iff the original graph H is Hamiltonian, a condition which is NP-complete to verify.

Now, let M be the cocycle matroid of a connected graph $G = (V, E)$ and accordingly, let the circuits of M be the minimal cuts of G. It is well-known and easy to see that an edge set $C \subseteq E$ forms a minimal cut in G iff there is a partition $V = U \cup W$ such that C is the set of all edges between U and W and the induced subgraphs $G[U]$ and $G[W]$ are both connected. In particular, this means that C (and each subset of C) must form a bipartite graph. Given an edge set $A = \{a_1, \ldots, a_t\} \subseteq E$ which forms a bipartite graph $G_A = (V_A, A)$, let us split G_A into connected components $G_{A_i} = (V_{A_i}, A_i)$, $i = 1, \ldots, k$ for some $k \leq t$. Then the problem of enumerating all minimal cuts containing A can be solved with polynomial delay via lexicographic backtracking [9] by growing and merging these connected components in all possible ways (so that their number can only decrease). Specifically, backtracking listing algorithms reduce the enumeration of minimal cuts containing $a_1, \ldots, a_t$ to the following decision problem:

> Given two disjoint vertex sets $U', W' \subseteq V$, can they be extended to a partition U, W which defines a minimal cut, that is $U' \subseteq U$, $W' \subseteq W$, $U \cap W = \emptyset$, $U \cup W = V$, and the induced subgraphs $G[U]$, $G[W]$ are both connected ?

If t, and hence $|U'| + |W'|$, is bounded this problem can be solved in polynomial time. In fact, this is true for the following more general problem:

> Given a graph $G = (V, E)$ and r pairwise disjoint vertex sets $U'_1, \ldots U'_r \subseteq V$, are there vertex sets $U_1 \ldots U_r \subseteq V$ which are still pairwise disjoint, $U'_i \subseteq U_i$ and the induced subgraph $G[U_i]$ is connected (i.e. spans U_i) for each $i = 1, \ldots, r$?

Robertson and Seymour [10] proved that for bounded $|U'_1| + \ldots + |U'_r|$ the above problem can be solved in polynomial time. Obviously, w.l.o.g. one can assume that the extended sets $U_1 \ldots U_r$ form a partition of V and hence for $r = 2$ the above problem includes the previous one.

Finally, similarly to minimal cycles, the enumeration of all minimal cuts through t edges becomes NP-hard when t is part of the input. Indeed, given a graph $G = (V, E)$ and a matching $A = \{a_1 = (u_1, w_1), \ldots, a_t = (u_t, w_t)\} \subseteq E$, it may be NP-hard to tell whether G has a minimal cut containing A. This claim can be shown as follows. Let $U' = \{u_1, \ldots, u_t\}$, $W' = \{w_1, \ldots, w_t\}$, and $V' = V \setminus (U' \cup W')$. Consider the set $\mathcal{G}$ of all graphs $G = (V, E)$ such that (i) the induced subgraph $G[V']$ is complete, (ii) $G[U']$ and $G[W']$ are edge-free, (iii) there are no edges between U' and W' except A, and (iv) the edges between V' and U' and between V' and W' are symmetric in the sense that $(v, u_i) \in E \quad \Leftrightarrow \quad (v, w_i) \in E$ for all $v \in V'$ and all $i = 1, \ldots, t$. Note that condition (iv) makes irrelevant any reversals in A, and that the decision problem:

> Given a graph $G \in \mathcal{G}$, is it possible to split V' between U' and W' to obtain two connected induced subgraphs ?

is polynomially equivalent with the special case of the CNF satisfiability problem in which all clauses are either strictly positive or strictly negative, and the clauses

in the positive and negative halves are symmetric. It remains to notice that this special case of the satisfiability problem is NP-complete since it is equivalent to the identification of *self-compliment saturated hypergraphs*, a problem whose NP-completeness was shown in [3].

We mention in closing that the results of this section also indicate that it may be NP-complete to decide whether a cycle or cocycle matroid M has a hyperplane avoiding a given set A of elements.

5 Minimal Spanning Sets for a Flat

Let M be a matroid on S. Each circuit C containing a given element $a \in S$ can be identified with a minimal (independent) set I such that $a \in Span(I)$, where

$$Span(I) = \{x \in S \ : \ r(I \cup x) = r(I)\}$$

is the closure operator. In this section we consider the problem of enumerating all minimal sets I spanning a given collection of elements $A \subseteq S$. In fact, we will consider a slightly more general problem of generating all minimal subsets $I \subseteq D$ which span A, where D and A are two given nonempty (and not necessarily disjoint) sets of elements of M. We denote the family of all such minimal spanning sets I by $\mathcal{SPAN}(D, A)$. Note that since $A \subseteq Span(I)$ implies $Span(A) \subseteq Span(I)$, we could assume that A is a flat, i.e. $A = Span(A)$.

Example 1 (Generalized Steiner trees and point-to-point connections) Let $G = (V, E)$ be a graph with k given disjoint vertex sets $V_1, \ldots, V_k \subseteq V$. A *generalized Steiner tree* is a minimal set of edges $I \subseteq E$ connecting all vertices within each set V_i, i.e., for each $i = 1, \ldots, k$, all vertices of V_i must belong to a single connected component of (V, I). In particular, for $k = 1$ we obtain the usual definition of Steiner trees. When each set V_i consists of two vertices $\{u_i, v_i\}$, generalized Steiner trees are called *point-to-point* connections. Let $T_1, \ldots, T_k$ be arbitrary spanning trees on $V_1, \ldots, V_k$ composed of "new" edges, and let M the cycle matroid of the multigraph $(V, E \cup T_1 \cup \ldots \cup T_k)$ with a total of $|E| + |V_1| + \ldots + |V_k| - k$ edges. Then $\mathcal{SPAN}(E, T_1 \cup \ldots \cup T_k)$ is the family of all generalized Steiner trees for $V_1, \ldots, V_k$.

Example 2 (Multiway cuts) For a connected graph $G = (V, E)$ with k pairs of vertices $\{u_i, v_i\}$, $i = 1, \ldots, k$, a *multiway cut* is a minimal collection of edges whose removal disconnects each u_i from v_i. Letting $A = \{(u_i v_i) \ : \ i = 1, \ldots, t\}$ and assuming w.l.o.g. that $A \cap E = \emptyset$, the family of all multicuts of G can be identified with the family $\mathcal{SPAN}(E, A)$ for the cocycle matroid of $(V, E \cup A)$.

Theorem 3. *Given a matroid M with ground set S and two non-empty sets $D, A \subseteq V$, all elements of $\mathcal{SPAN}(D, A)$ can be enumerated in incremental polynomial time. All maximal subsets of D which do not span A can also be enumerated in incremental polynomial time.*

Proof. Let α be a new element representing A, and let M_α be the matroid on $D \cup \alpha$ with the following rank function:

$$\rho(X) = \begin{cases} r(X), \text{ if } \alpha \notin X \\ \max\{r((X \setminus \alpha) \cup a) \ : \ a \in A\}, \quad \text{otherwise.} \end{cases} \tag{2}$$

It is easy to check that M_α is indeed a matroid. When M is a vectorial matroid over a large field, α can be interpreted as the "general linear combination" of all elements of A; in general, $\rho(X)$ is the so-called *principal extension of* $r(X)$ *on* A *with value* 1 (see e.g. [7]).

When $I \in \mathcal{SPAN}(D, A)$ then $I \cup \alpha$ is a circuit in M_α and conversely, for any circuit C in M_α containing α, the set $C \setminus \alpha$ belongs to $\mathcal{SPAN}(D, A)$. Hence the enumeration problem for $\mathcal{SPAN}(D, A)$ is equivalent with that for the set of all circuits through α in M_α. Given an independence oracle for M, the rank function (2) of the extended matroid can be trivially evaluated in oracle-polynomial time. Therefore the first claim of Theorem 3 directly follows from Theorem 2. To see the second claim note that the maximal subsets of D which do not span A are in one-to-one correspondence with the hyperplanes of M_α which avoid α. □

Finally, let us note that since $\mathcal{SPAN}(S, S)$ is the set of bases of M, the proofs of Theorems 2 and 3 show that the enumeration of all bases of a matroid can be reduced to the enumeration of all circuits of another matroid.

6 Circuits in Two Matroids, Generalized Circuits

Let M_1 and M_2 be two matroids on S, with rank functions $r_1(X)$ and $r_2(X)$. It is known that the minimum of the submodular function $r_1(X) + r_2(S \setminus X)$ for all $X \subseteq S$ gives the maximum cardinality of a set I independent in both M_1 and M_2, and that this minimum can be computed in polynomial time [2]. In particular, when the ranks of M_1 and M_2 are equal one can determine in polynomial time whether M_1 and M_2 share a common base, i.e. $\mathcal{B}(M_1) \cap \mathcal{B}(M_2) \neq \emptyset$. In fact, using this as a subroutine for backtracking on matroids obtained by deleting and contracting elements of S, all bases in $\mathcal{B}(M_1) \cap \mathcal{B}(M_2)$ can be enumerated with polynomial delay.

In contrast to this result, deciding whether M_1 and M_2 contain a common circuit is NP-hard already when M_1 is the cycle matroid of some graph $G = (V, E)$ and M_2 is the uniform matroid on E whose bases are all subsets of size $r = |V| - 1$. In this case, $\mathcal{C}(M_1) \cap \mathcal{C}(M_2) \neq \emptyset$ iff G is Hamiltonian. A similar argument for the NP-complete maximum cut problem shows that testing if $\mathcal{C}(M_1) \cap \mathcal{C}(M_2) \neq \emptyset$ remains NP-hard when M_1 is the cocycle matroid of a graph $G = (V, E)$ and M_2 is again a uniform matroid on E.

Of course, given two matroids M_1 and M_2 on S one can always enumerate all elements of $\mathcal{C}(M_1) \cup \mathcal{C}(M_2)$ in incremental polynomial time due to Theorem 1. Note, however, that deciding whether a given set $C \in \mathcal{C}(M_1) \cup \mathcal{C}(M_2)$ is inclusion-wise *maximal* in $\mathcal{C}(M_1) \cup \mathcal{C}(M_2)$ may be NP-hard. This is because for any set $A \subseteq S$ we may choose M_2 to be the matroids for which A is the only circuit, and then deciding whether A is maximal becomes equivalent with determining if M_1

has a circuit containing A (see Section 4). Perhaps more surprisingly, for two matroids M_1 and M_2 on S enumerating all *minimal* elements of $\mathcal{C}(M_1) \cup \mathcal{C}(M_2)$ may also be hard.

Proposition 1. *Let M_1 and M_2 be the cycle matroids of two graphs $G_1 = (V_1, E)$ and $G_2 = (V_2, E)$ with identical sets of edges, and let $\mathcal{MIN}\{\mathcal{C}(M_1) \cup \mathcal{C}(M_2)\}$ be the collection of all minimal edge sets which form a cycle in G_1 or G_2. Then, given a set family $\mathcal{M} \subseteq \mathcal{MIN}\{\mathcal{C}(M_1) \cup \mathcal{C}(M_2)\}$, it is NP-complete to tell whether $\mathcal{M}$ can be extended, i.e. $\mathcal{M} \neq \mathcal{MIN}\{\mathcal{C}(M_1) \cup \mathcal{C}(M_2)\}$.*

We conclude with yet another generalization of the notion of circuit in a matroid. Let M be a matroid defined by an independence oracle on some ground set U, and let $A_1, \ldots, A_n$ be given (not necessarily disjoint) subsets of U. We define a *generalized circuit* as a minimal subset X of $S = [n]$ such that $\bigcup_{i \in X} A_i$ is a dependent set in M.

Proposition 2. *Enumerating all generalized circuits for the cycle matroid of a graph is NP-hard when $A_1, \ldots, A_n$ are disjoint sets of edges of size 2 each.*

It is easy to see that in Propositions 1 and 2 the cycle matroids of G_1 and G_2 can be replaced by the cocycle matroids of some graphs (e.g., the planar duals of G_1 and G_2). Also, by matroid duality, Proposition 1 shows that it may be NP-hard to enumerate all *generalized hyperplanes* of M, i.e., all those maximal subsets X of $S = [n]$ for which $Span(\cup_{i \in X} A_i) \neq S$.

In contrast to Proposition 2, all *generalized bases* of M, i.e. all minimal sets $X \subseteq [n]$ for which $Span(\cup_{i \in X} A_i) = S$ can be generated in incremental quasi-polynomial time regardless of the sizes of $A_1, \ldots, A_n$, see [1] for more detail.

References

1. E. Boros, K. Elbassioni, V. Gurvich and L. Khachiyan, Matroid intersections, polymatroid inequalities, and related problems, in *Proc. 27th Intl. Symp. on Mathematical Foundations of Computer Science, (MFCS) 2002*, LNCS 2420, pp. 143–154.
2. J. Edmonds, Submodular functions, matroids, and certain polyhedra, in *Combinatorial structures and their applications*, Gordon and Breach, 69–87.
3. T. Eiter and G. Gottlob, Identifying the minimal transversals of a hypergraph and related problems, *SIAM J. Comput.*, 24 (1995) 1278–1304.
4. R. Karp, On the complexity of combinatorial problems, *Networks* 5 (1975) 45–68.
5. T. Lazarson, Independence functions in algebra, (Thesis), Univ. London (1957).
6. A. Lehman, A solution of the Shannon switching game, *J. Soc. Indust. Appl. Math.* 12 (1964) 687–725.
7. L. Lovász, Submodular functions and convexity, in *Mathematical Programming: The State of the Art*, Bonn 1982, pp. 235–257, (Springer Verlag, 1983).
8. J. S. Provan and D. R. Shier, A paradigm for listing (s,t)-cuts in graphs, *Algorithmica*, 15(4) (1996) 357–372.
9. R. C. Read and R. E. Tarjan, Bounds on backtrack algorithms for listing cycles, paths, and spanning trees, *Networks*, 5 (1975) 237–252.
10. N. Robertson and P. D. Seymour, Graph minors, XIII, The disjoint path problem, *J. Comb. Th.*, Ser. B 63 (1995) 65–110.
11. D.J.A. Welsh, Matroid Theory, Academic Press, 1976.