Chapter 42

Randomized Parallel Algorithms for Matroid Union and Intersection, with Applications to Arboresences and Edge-Disjoint Spanning Trees

H. Narayanan[*] Huzur Saran[†] Vijay V. Vazirani[‡]

Abstract

The strong link between matroids and matching is used to extend the ideas that resulted in the design of Random NC algorithms for matching to obtain RNC algorithms for the matroid union, intersection and matching problems, for linearly representable matroids. As a consequence, we obtain RNC algorithms for the well-known problems of finding an arboresence and a maximum cardinality set of edge-disjoint spanning trees in a graph. The key tools used are linear algebra and randomization.

1 Introduction

We obtain Random NC algorithms for the fundamental problems of matroid union and matroid intersection, for linearly representable matroids.

A key step in the matroid union algorithm is a simple method, using randomization, for obtaining a linear representation for the union (it is well known that the union will be a matroid, and will also be linearly representable). Once this is done, it is straightforward to find a maximal independent set in the union matroid. This set is partitioned into independent sets in the corresponding matroids using our RNC matroid intersection algorithm.

An RNC algorithm for the matroid intersection problem, which is a generalization of bipartite matching, is obtained using the Binet-Cauchy Theorem and the Isolating Lemma of [MVV]. Similar techniques, together with a theorem of Lovász [Lo2], yield an RNC algorithm for the matroid matching problem as well. This problem, also called the matroid parity problem, generalizes matroid intersection and general graph matching.

Using the matroid union and intersection algorithms, we obtain RNC algorithms for the matroid covering and packing problems. For the case of a graphic matroid, these correspond to the well-known problems of finding an arboresence in a graph (*i.e.* the minimum number of forests that cover all edges), and of finding a maximum cardinality set of edge-disjoint spanning trees in it. The latter has obvious applications to fault tolerant communications [IR,Gu]. It also has applications in the analysis of electric networks [IF,OIW], and in the study of rigidity of structures [LY].

The first polynomial time algorithms for matroid union and matroid intersection were given by Edmonds [Ed1] , and Lovász [Lo2] gave the first such algorithm for the matroid matching problem on representable matroids.

Our results draw on the ideas used in the design of RNC algorithms for the maximum matching problem [KUW1,MVV]. Broadly speaking, the solution for matching consisted of taking the problem into linear algebra, and the use of randomization for extracting a maximum matching. Indeed, we are also forced to work within the realm of linear algebra; our results hold only for linearly representable matroids. However, since almost all useful matroids have this property, this is not a major restriction. It is interesting to see matroid theory and the matching problem playing their typical roles once again – the former unifies and generalizes concepts, and the latter 'serves as an archetypical example of how a "well-solvable" problem can be studied' (quoted from [LP]).

[*] Dept. of Elect. Engg. I.I.T. Bombay
[†] Dept. of C.S. & Engg. I.I.T. Delhi
[‡] Dept. of C.S. & Engg. I.I.T. Delhi

2 Preliminaries

Let us first review some basic definitions and notation from matroid theory; details may be found in [Ai,La,We].

A *matroid* M is a finite set $S = \{e_1, \ldots, e_n\}$ and a collection $\mathcal{I}$ of subsets of S (called *independent sets*) such that the following hold

1. $\emptyset \in \mathcal{I}$
2. If $X \in \mathcal{I}$ and $Y \subseteq X$ then $Y \in \mathcal{I}$.
3. If $U, V \in \mathcal{I}$ with $|U| = |V|+1$ then there exists $x \in U - V$ such that $V \cup \{x\} \in \mathcal{I}$.

A subset of S which is not independent, is called *dependent*. The *rank* of a set $A \subseteq S$, (denoted by $\rho(A)$) is the cardinality of a maximal independent subset of A. The rank of the matroid M is the rank of S. A maximal independent set of M is called a *base*. A fundamental consequence of the matroid axioms is that all bases of a matroid have the same cardinality, thus every maximal independent set is actually maximum.

A matrix B over a field F is a *linear representation* of M if there is a one-to-one correspondence between the elements of S and the columns of B, such that a subset of S is independent in M if and only if the corresponding set of columns of B are independent over F. If the rank of M is r then B need have only r rows. A matroid M is said to be *linearly representable* over F if their exists such a matrix.

For convenience, we will restrict our attention to matroids that are linearly representable over the rationals; it is easy to check that our results are valid as long as $|F|$ is large enough as a function of n and r. Henceforth, 'representable' will mean 'representable over **Q**'.

Given a graph G, the matroid $\hat{G}$ obtained by taking S to be the set of edges of G and $\mathcal{I}$ to be the forests of G, is called a *graphic* matroid. Every base of $\hat{G}$ corresponds to a spanning forest in G. It is easily seen that if we delete any one row of the vertex-edge incidence matrix of G, we obtain a linear representation of $\hat{G}$ over $GF(2)$. A linear representation of $\hat{G}$ over **Q** is obtained by changing one 1 to a -1 in each column of the above matrix.

The *dual* of a matroid M, denoted by M^*, is the matroid whose independent sets are $\{X : \exists B$ a base of M, and $X \subseteq S - B\}$.

Given a set S and a partition $S_1, S_2, \ldots S_k$ of S, the matroid obtained by taking $\mathcal{I} = \{I \subseteq S : |I \cap S_j| \leq 1, 1 \leq j \leq k\}$ is called a *partition* matroid. It is easily seen that partition matroids are linearly representable (pick k linearly independent vectors and let all elements in S_i be represented by the i^{th} vector).

Let $M_1, M_2, \ldots M_k$ be matroids on S. Let $\mathcal{I} = \{X : X = X_1 \cup \ldots \cup X_k$ such that X_i is independent in $M_i, 1 \leq i \leq k\}$. Then $\mathcal{I}$ is the collection of independent sets of a matroid on S. We call this matroid the *union* of $M_1, \ldots, M_k$ and denote it by $\bigvee_{i=1}^{k} M_i$. If X is an independent set in $\bigvee_{i=1}^{k} M_i$, then a partition of X into sets $X_1, \ldots, X_k$ such that X_i is independent in M_i is called a *proper* partition of X.

The intersection of matroids can be defined analogously. However, the collection of sets so obtained does not form the independent sets of a matroid. As a consequence, a maximal set in this collection need not be maximum. The problem of finding a maximum cardinality set in the intersection of two matroids is called the *matroid intersection* problem.

The *matroid matching problem* is: given matroid $M = (S, \mathcal{I})$, and a partition of S into pairs $(x_1, x_2) \ldots (x_{2n-1}, x_{2n})$, pick the largest number of pairs so that the picked elements form an independent set.

The $(i,j)^{th}$ entry of a $n \times n$ matrix A will be denoted by $A(i,j)$ and its determinant will be denoted by $|A|$.

$$|A| = \sum_{\sigma} sign(\sigma) value(\sigma)$$

Here, the sum is taken over all permutations over $\{1, \ldots, n\}$, $sign(\sigma)$ is $+1$ if σ is even and -1 if σ is odd, and $value(\sigma) = \prod_{i=1}^{n} A(i, \sigma(i))$.

In this paper, we state all our results assuming the arithmetic CREW PRAM model of computation [KR]. The arithmetic operations we require are addition, multiplication and subtraction of $O(n \log n)$ bit numbers, where n is the size of the ground set. We use $M(n)$ to denote the number of processors required to multiply two $n \times n$ matrices in $O(\log n)$ time, $M(n) = o(n^{2.5})$.

3 Obtaining a representation for the union matroid

If matroids $M_1, M_2, \ldots, M_k$ over set S are linearly representable, then the union $\bigvee_{i=1}^k M_i$ is also linearly representable. However, the standard way of constructing such a representation does not seem to be parallelizable. We first give an *RNC* algorithm for constructing a linear representation of the union of k matroids, for any $k \geq 1$.

We will give details for $k = 2$; the generalization will be obvious. Let the ranks of M_1 and M_2 be r_1 and r_2 respectively. Let A_1 be a $r_1 \times n$ matrix representing M_1 and A_2 be a $r_2 \times n$ matrix representing M_2. Obtain A'_2 from A_2 by multiplying the columns of A_2 by distinct variables $x_1, x_2, \ldots, x_n$, *i.e.* $A'_2(i,j) = A_2(i,j) \cdot x_j$. It is easy to verify that a set of columns of A'_2 is independent if and only if the corresponding set of columns of A_2 is independent.

Define B, an $(r_1 + r_2) \times n$ matrix, whose first r_1 rows are the same as the rows of A_1 and the last r_2 rows are the same as the rows of A'_2, *i.e.*

$$B(i,j) = \begin{cases} A_1(i,j) & \text{if } i \leq r_1 \\ A'_2(i - r_1, j) & \text{otherwise} \end{cases}$$

Lemma 3.1. *A set $S' \subseteq S$ is independent in $M_1 \vee M_2$ if and only if the corresponding columns of B are linearly independent.*

Proof. Let us first state a key property of matrix B that makes the lemma hold. Let C be a $(k+l) \times (k+l)$ submatrix of B that picks k rows from A_1 and l rows from A'_2, and some set of $k + l$ columns. Corresponding to a set of l of these columns, we can define two submatrices C_1 and C_2, of C as follows: C_2 consists of l rows of A'_2 and these l columns, and C_1 consists of the unpicked rows and columns. Let the indices of these columns be $i_1, \ldots, i_l$ in B. Consider $|C| = \sum_\pi sign(\pi) value(\pi)$, where as usual, the sum runs over all permutations π mapping rows of C onto columns of C. Let P be the set of permutations mapping rows of C_1 onto columns of C_1, and rows of C_2 onto columns of C_2. Then it is easy to see that

$$\sum_{\pi \in P} sign(\pi) value(\pi) = |C_1||C_2|;$$

this term is of the form $c x_{i_1} \ldots x_{i_l}, c \in \mathbf{Q}$. Furthermore, a monomial of this form is not generated by any other permutation π. Therefore, if C_1 and C_2 are both non-singular then so is C. Similarly, if C is non-singular then for a suitable choice of $i_1 \ldots i_l$ (corresponding to a monomial with non-zero coefficient in $|C|$), C_1 and C_2 must be non-singular.

The rest of the proof is now straightforward. Suppose $S' \subseteq S$ is independent in $M_1 \vee M_2$. Let (S_1, S_2) be a proper partition of S', and let $|S_1| = k$ and $|S_2| = l$. Then, the columns of $A_1(A_2)$ corresponding to $S_1(S_2)$ are linearly independent, and one can pick $k(l)$ rows to get a $k \times k(l \times l)$ non-singular matrix $C_1(C_2)$. Then by the above stated property, the $(k+l) \times (k+l)$ matrix C, consisting of all the picked rows and columns is non-singular.

To prove the other direction, pick a maximal non-singular square submatrix, C, of the linearly linearly independent columns of B. These columns correspond to some set $S' \subseteq S$. By the above-stated property, there exist non-singular submatrices C_1 and C_2 of C. These yield a partition of S' into (S_1, S_2) such that S_i is independent in M_i, $i = 1, 2$. This implies that S' is independent in $M_1 \vee M_2$. ■

The construction given above clearly generalizes to k matroids $M_1, \ldots, M_k$ of rank $r_1, \ldots, r_k$: multiply the columns of $M_2, \ldots, M_k$ with distinct variables, $x_1, \ldots, x_{(k-1)n}$, and 'stack up' the resulting matrices, together with M_1, to obtain an $(r_1 + \ldots + r_k) \times n$ matrix B. Since matroid union is associative, the proof that this construction works follows immediately using induction.

Lemma 3.2. *Given matroids $M_1 \ldots M_k$, the matrix B obtained above is a linear representation for $\bigvee_{i=1}^k M_i$.*

We next show, using randomization, how to obtain a linear representation of $\bigvee_{i=1}^k M_i$ using only rational entries, *i.e.* with no variables.

Lemma 3.3. *Let $\tilde{B}$ be obtained from B by substituting each indeterminate randomly and independently from $\{1, \ldots, 2n\binom{n}{n/2}\}$. Then*

$$Pr[\tilde{B} \textit{ is a linear representation of } \bigvee_{i=1}^{k} M_i] > \frac{1}{2}$$

Proof. Note that it is enough to show that our substitution maintains the independence of every base of B. Let the rank of the union matroid be r. Then, it can have at most $\binom{n}{r}$ distinct bases. Corresponding to each base β there is an $r \times r$ non-singular submatrix of B, C_β. Consider the polynomial

$$p = \Pi_\beta |C_\beta|,$$

where the product is taken over all bases of B. The degree of p is no more than $\binom{n}{r}r$. Now, it is enough to obtain a substitution for the indeterminates such that p evaluates to a non-zero value. By lemma 3.4, it is sufficient to substitute for the variables randomly and independently from the integers $\{1, 2, \ldots, 2r\binom{n}{r}\}$, *i.e.* using at most $n \log n$ bits for each number, since $2r\binom{n}{r} \leq 2n\binom{n}{n/2}$ for $n \geq r \geq 0$. ∎

LEMMA 3.4. ([SC,ZI]) *Let $p(x_1, x_2, \ldots, x_n)$ be a non-trivial polynomial of degree d over field F and let $S \subseteq F$. If $x_1, x_2, \ldots, x_n$ are randomly and independently chosen from S then,*

$$Pr[\textit{the substitution is a zero of } p] \leq d/|S|$$

Hence we get:

THEOREM 3.1. *Given the linear representations of k matroids, $M_1, M_2, \ldots, M_k$ with ranks $r_1, r_2, \ldots, r_k$, there is a RNC^0 algorithm for obtaining a representation of the union using $n(\sum_{i=1}^k r_i) \leq n^2 k$ processors.*

Once we get a representation, $\tilde{B}$, for the union matroid $M = \bigvee_{i=1}^k M_i$, we can find the lexicographically first base, I, for M by finding the lexicographically first base of $\tilde{B}$. Since the lexicographically first base of a matrix can be obtained in NC^2 we get:

COROLLARY 3.1. *Given k matroids, $M_1, \ldots, M_k$ via their linear representations over $\mathbf{Q}$, there is an RNC^2 algorithm that uses $O(nM(n))$ processors for obtaining the lexicographically first maximal independent set in the union $\bigvee_{i=1}^k M_i$.*

REMARK 3.1. Notice that even if the random substitution was a zero of p, if a set of columns are independent in $\tilde{B}$ then they must be independent in B. Therefore, I is guaranteed to be independent in M.

Next, we would like to obtain a proper partition of I. For this we need a parallel algorithm for the matroid intersection problem.

4 Matroid intersection and matroid matching

4.1 Matroid intersection

We use the Isolating Lemma of [MVV] and the Binet-Cauchy Theorem to obtain an *RNC* algorithm to compute the maximum cardinality set in the intersection of two matroids. Recall that the Binet-Cauchy Theorem states that, given two $n \times m$ matrices A and B,

$$|AB^T| = \sum_\alpha |A_\alpha||B_\alpha|$$

where the sum is taken over all possible ways, α, of choosing n columns out of m, and by $A_\alpha(B_\alpha)$ we mean the $n \times n$ submatrix of $A(B)$ consisting of columns chosen by α.

We shall need the following slightly modified version of the Isolating Lemma (this is a straightforward extension of the original Lemma).

LEMMA 4.1. (ISOLATING LEMMA [MVV]) *Let $(X, \mathcal{F})$ be a set system, where $X = \{x_1, x_2, \ldots, x_n\}$ and $\mathcal{F} \subseteq 2^X$ is a family of subsets of X. Let $\{c_1, c_2, \ldots, c_n\}, c_i \in \mathbf{Q}$ be a set of initial weights on the x's. Pick $w_1, w_2, \ldots, w_n$ randomly and independently from $[1 \ldots 2n]$. Define the weight of x_i to be $c_i + w_i$, and let the weight of a set be the sum of the weights of its elements. Then,*

$$Pr[\exists! \textit{ minimum weight set in } \mathcal{F}] \geq 1/2.$$

THEOREM 4.1. *There is an RNC^2 algorithm using $O(n^{4.5})$ processors for the matroid intersection problem, for linearly representable matroids.*

Proof. Let the two matroids M_1 and M_2 be represented by the matrices A and B respectively. Without loss of generality assume that A and B have the same number of rows, say r, (since the smaller matrix can be padded with rows of zeroes). In general, the required maximum cardinality subset of S may have fewer than r elements; if so, by the Binet-Cauchy Theorem $|AB^T| = 0$, and we get no information from this determinant. We will get around this by first augmenting A and B with extra columns. Let $e_1, e_2, \ldots, e_r$ be the unit vectors over $\mathbf{Q}^r$. Obtain $r \times r^2$ matrices C and D such that for each $1 \leq i, j \leq r$, there is an index $1 \leq k \leq r^2$, such that the k^{th} column of C is e_i and the k^{th} column of D is e_j.

Let $[A, C]$ represent the $r \times (n + r^2)$ matrix whose first n columns are the same as those of A, and the last r^2 columns are the same as those of C. Similarly, obtain $[B, D]$. The augmented matrices have the following property: any subset of indices from $[1 \ldots n]$ such that the corresponding columns are linearly independent both in A and in B can be extended to a set of r indices from $[1 \ldots n + r^2]$ such that the corresponding columns are linearly independent both in $[A, C]$ and $[B, D]$.

Randomly and independently pick $w_1, \ldots w_{n+r^2}$ from $[1 \ldots 2(n+r^2)]$. For $1 \leq i \leq n$, multiply the i^{th} column of $[A, C]$ by x^{w_i}, and for $n+1 \leq i \leq r^2+n$, multiply the i^{th} column of $[A, C]$ by $x^{w_i+2r(n+r^2)}$ to obtain a new $r \times (n + r^2)$ matrix E. Compute $|E[B, D]^T|$. This can be done in NC^2 using $O(n^{4.5})$ processors (see [BCP]).

We now apply the Isolating Lemma. Here, the set system, X consists of the set of $n + r^2$ indices and $\mathcal{F}$ consists of all choices of r indices such that the corresponding columns are independent both in $[A, C]$ and in $[B, D]$. Then with high probability, the minimum weight set of indices, α, is unique for the chosen weight distribution, and has weight w_α. By the Binet-Cauchy Theorem $|E[B, D]^T| \neq 0$, since the coefficient of the minimum power of x (i.e. x^{w_α}) is non-zero. The minimum weight set of indices α can be obtained as follows:

```
    For $i = 1$ to $n + r^2$, in parallel do:
        decrease $w_i$ by 1, keeping the
        rest of the weights unchanged,
        and compute $|E[B, D]^T|$.  If the
        coefficient of $x^{w_\alpha}$ changes,
        then pick index $i$.
    end;
```

Because of the choice of weights, the subset of these indices in the range $[1 \ldots n]$ will constitute a maximum cardinality subset of S that is independent in both matroids. ∎

REMARK 4.1. The algorithm given above extends to the weighted matroid intersection problem, provided the weights are given in unary. Let the i^{th} element of the ground set have weight W_i; find a minimum weight maximum cardinality set in the intersection of the two matroids. In this case, we multiply the i^{th} column of $[A, C]$ by $x^{w_i+2(n+r^2)W_i}$ for $1 \leq i \leq n$, and by $x^{w_i+2r(n+r^2)W}$ for $n < i \leq n + r^2$, where W is the weight of the heaviest ground set element. Extending this to the case of binary weights is left open.

4.2 Matroid matching

In this section we shall give an *RNC* algorithm for the *matroid matching problem* for representable matroids: given m pairs of vectors over $\mathbf{Q}^{2n}, \{a^{(1)}, b^{(1)}\}, \ldots, \{a^{(m)}, b^{(m)}\}$, pick the largest number of pairs so that the picked vectors are linearly independent. Let us first define the *wedge product*, of vectors a, b over $\mathbf{Q}^{2n}$ as an $2n \times 2n$ matrix A such that

$$A(i, j) = a_i b_j - a_j b_i.$$

Denote this by $(a \wedge b)$. We will need the following theorem of Lovász:

THEOREM 4.2. ([Lo2,LP]) *Let $\{a^{(1)}, b^{(1)}\}, \ldots, \{a^{(m)}, b^{(m)}\}$ be pairs of vectors over $\mathbf{Q}^{2n}$, then there exists a set of n pairs whose union is a basis iff $|B| \neq 0$, where*

$$B = \sum_{i=1}^{m} (a^{(i)} \wedge b^{(i)}) x_i.$$

Here the x_i's are distinct indeterminates.

THEOREM 4.3. *There is an RNC^2 algorithm using $O(n^{4.5})$ processors for the matroid matching problem, for linearly representable matroids.*

Proof. As in Theorem 4.1, we will first deal with the issue that there may not be n pairs (a_i, b_i) whose union is a basis, by throwing in $\binom{2n}{2}$ additional pairs of vectors. Let $e_1, \ldots, e_{2n}$ be the unit vectors in $\mathbf{Q}^{2n}$. The added vectors are chosen in such a way that for every pair of indices $(i, j), 1 \leq i < j \leq 2n$, (e_i, e_j) is included.

As before, as a result of the augmentation, any subset of the original m pairs that are linearly independent can be extended to n pairs using the added pairs. Let

$$B = \sum_{i=1}^{m+\binom{2n}{2}} (a^{(i)} \wedge b^{(i)})x_i.$$

Randomly and independently pick $w_1, \ldots w_{m+\binom{2n}{2}}$ from $[1 \ldots 2(m + \binom{2n}{2})]$. For $1 \leq i \leq m$, substitute $x_i = x^{w_i}$, and for $m + 1 \leq i \leq m + \binom{2n}{2}$, substitute $x_i = x^{w_i + 2n(m+\binom{2n}{2})}$, and compute $|B|$.

Note that B is skew-symmetric, and hence $|B| = (pf(B))^2$, where $pf(B)$ is the pfaffian of B. Lovàsz [Lo2] shows that $pf(B)$ is linear in each of the variables x_i and each monomial in the polynomial $pf(B)$ is the product of n distinct variables x_i. We will now apply the Isolating Lemma with X as the set of $m + \binom{2n}{2}$ indices, and $\mathcal{F}$ consisting of all choices of n indices such that the corresponding monomial has non-zero coefficient in $pf(B)$. Then, after the random substitution given above, with high probability, there is a unique term having the minimum power of x (say x^w) in $pf(B)$. Consequently, the polynomial $|B|$ will also contain a unique term having minimum power of x, x^{2w}. Hence $|B| \neq 0$. The indices contributing to this term can be obtained as in Theorem 4.1, by decreasing w_i by 1 in parallel for each i. By the choice of weights, the number of indices in the range $[1 \ldots m]$ is maximized. Hence, this set of indices is a solution to the matroid matching problem. ∎

As in the case of matroid intersection, the above algorithm extends to the weighted problem, if the weights are given in unary. Once again, the weighted matroid matching problem for binary weights is left open.

5 Partitioning the independent set

Suppose set I is independent in $M_1 \vee M_2$. In this section, we show how to obtain a proper partition of I.

5.1 Obtaining a representation for the dual matroid

First, we need a parallel algorithm for obtaining a representation of the dual matroid. The standard method parallelizes in a straightforward manner; essentially, it involves finding a base for the null-space of A, where A is a $r \times n$ matrix representing M. Denote the submatrix of A consisting of the first r (last $n - r$) columns of A by $A_1(A_2)$. The columns of A can be permuted in NC to ensure that A_1 is non-singular.

```
For each $n-r$ dimensional unit vector $e_i$,
$1 \leq i \leq n-r$ in parallel do:
    Compute $x_i = -A_1^{-1} A_2 e_i$.
    Output the $n$ dimensional vector
      whose first $r$ components
      consist of $x_i$ and the remaining
      $n-r$ components consist of $e_i$.
end;
```

Let C be the matrix consisting of these $n - r$ n-dimensional column vectors (in any order). Then $B = C^T$ is a linear representation for M^*. Since matrix inversion is in NC^2 (see [Cs]) we get:

LEMMA 5.1. *There is an NC^2 algorithm using $O(nM(n))$ processors for obtaining the linear representation of the dual, M^*, for a linearly representable matroid M.*

5.2 The partitioning algorithm

We may assume without loss of generality that I is the ground set for M_1 and M_2 (if not, we can restrict M_1 and M_2 to I, and pick only the corresponding columns of matrices A and B). The partitioning algorithm is as follows:

```
1. Find a representation $B^*$ for $M_2^*$.

2. Apply the matroid intersection
   algorithm on $A$ and $B^*$.

3. Let $I_1$ be the set found in step 2.
   Output $(I_1, I_2)$ where $I_2 = I - I_1$.
```

LEMMA 5.2. *Assuming that the matroid intersection algorithm in step 2 is successful, (I_1, I_2) is a proper partition of I.*

Proof. It is sufficient to prove that I_2 is independent in M_2, since clearly I_1 is independent in M_1. Since I_1 is independent in M_2^*, I_2 contains a base of M_2. We will finish the proof by showing that $|I_2|$ must equal the cardinality of a base of M_2.

Let (I_1', I_2') be a proper partition of I that maximizes $|I_2|'$. Then I_2' must be a base of M_2; if not, we should be able to move an element from I_1' to I_2' (note that $I_1' \cup I_2'$ is the ground set of M_2). Clearly, I_1' is independent in M_1 and M_2^*. Therefore $|I_1| \geq |I_1'|$, and so $|I_2| \leq |I_2'|$. The lemma follows. ■

When we are given I over $M_1 \vee M_2 \vee \ldots \vee M_k$, using the previous algorithm, we can first partition it into I_1 and I_2 such that I_1 is independent in $M_1 \vee \ldots \vee M_{k/2}$ and I_2 is independent in $M_{k/2+1} \vee \ldots \vee M_k$. For this, we need to construct representations for the two matroids $M_1 \vee \ldots \vee M_{k/2}$ and $M_{k/2+1} \vee \ldots \vee M_k$. Clearly, this just involves choosing the appropriate rows of the representation of $M_1 \vee M_2 \vee \ldots \vee M_k$. Now recursively, in parallel, solve the problem for I_1 and I_2. Note that this will take at most $\log n$ iterations of the above algorithm.

THEOREM 5.1. *There is an RNC^3 algorithm using $O(n^{4.5})$ processors for the following problem: given matroids $M_1, \ldots, M_k$, via their linear representations and an independent set I in $\bigvee_{i=1}^{k} M_i$, find a proper partition of I.*

6 Covering and packing problems

We give *RNC* algorithms for solving the covering and packing problems for a linearly representable matroid M. Let M^k denote $\bigvee_{i=1}^{k} M_i$.

The *matroid covering problem* is: find a minimum cardinality collection $\mathcal{C}$ of independent sets in M such that $\bigcup_{X \in \mathcal{C}} X = S$. Let $|\mathcal{C}| = k$, then k can be at most n, assuming that there are no trivial elements in S (*i.e.* elements which participate in no independent set). Clearly $k = \min\{i : \rho(M^i) = n\}$. Thus, we can carry out a binary search in the interval $[1 \ldots n]$ to obtain k. This would involve at most $\log n$ iterations of the matroid union algorithm. We can then partition S into k independent sets using the algorithm of Theorem 5.1 to get the required cover.

THEOREM 6.1. *There is an RNC^3 algorithm using $O(n^{4.5})$ processors for obtaining a minimum cardinality cover by independent sets of a linearly representable matroid M.*

Since the graphic matroid of a graph is linearly representable (see section 2), we get:

COROLLARY 6.1. *There is an RNC^3 algorithm using $O(n^{4.5})$ processors for obtaining an arboresence of a graph G.*

The *matroid packing problem* is: find a maximum cardinality set of mutually disjoint bases of M. Let k be the cardinality of this set and r be the rank of M. Clearly, $k = \max\{i : \rho(M^i) = ri\}$, and $k \leq \lfloor n/r \rfloor$. To find k, we can do a binary search over the range $1 \ldots \lfloor n/r \rfloor$. A partition of any base of M^k will then give the required set of bases.

THEOREM 6.2. *There is an RNC^3 algorithm using $O(n^{4.5})$ processors for obtaining a maximum cardinality set of disjoint bases of a linearly representable matroid M.*

COROLLARY 6.2. *There is an RNC^3 algorithm using $O(n^{4.5})$ processors for finding a maximum cardinality set of edge-disjoint spanning trees of a graph G.*

7 Las Vegas extensions

The algorithms presented so far have all been Monte Carlo, *i.e.* they work with high probability. We now give parallel algorithms for verifying the solutions obtained, thereby giving their Las Vegas extensions, *i.e.* the running time of the algorithm is probabilistic; however, it is guaranteed to produce the correct solution.

The matroid intersection algorithm can be made Las Vegas as follows: suppose the intersection computed is I. Now, for each element $e \in S - I$, check using Edmonds' matroid intersection algorithm whether the set can be augmented with e (this will be a transitive closure computation in an appropriately defined graph, see [PS]). If this fails for each element, then I is the largest set in this intersection.

Next, we make the matroid union algorithm Las Vegas. Let I be the lexicographically first maximal independent set found in $M_1 \vee \ldots \vee M_k$, and let $I_1, \ldots, I_k$ be the partition of I obtained.

Notice that randomization is used at two points: in finding I, and in partitioning it. Because of Remark 3.1, I must be independent, though it may not be the lexicographically first maximal independent set. We can verify this solution as follows:

1. Verify that I_i is independent in M_i, $1 \leq i \leq k$; if not, halt. This can be done easily since linear representations of M_i are available. (Notice that a failure here indicates a bad choice in the randomization done in the partitioning phase. Hence, if this test is passed, then the partitioning of I is correct.)

2. Verify that I is the lexicographically first maximal independent set of $M_1 \vee \ldots \vee M_k$ as follows: Let $e_1, \ldots, e_n$ be the ordering on the elements of S, and let $I^{(i)}$ denote the restriction of I to $\{e_1, \ldots, e_i\}$. It is sufficient to verify that for each $i, 1 \leq i < n$: if $e_{i+1} \notin I$ then $I^{(i)} \cup \{e_{i+1}\}$ is dependent in the union matroid. This can be done in parallel for each i, as in the matroid intersection case given above. Notice that a partition of $I^{(i)}$ is available.

We next turn to covering. Suppose our algorithm found that $k = \min\{i : S \text{ is independent in } M^i\}$. Now we only need to verify that S is not independent in M^{k-1}. This can easily be done by finding a maximal independent set in M^{k-1}. Finally, for packing, the k found can be verified by checking if $\rho(M^{k+1}) < r(k+1)$, where r is the rank of M. We leave the open problem of obtaining a Las Vegas extension for the matroid matching problem.

8 Discussion and open problems

1) Our results hold only for linearly representable matroids. Although almost all interesting matroids have this property, it is still interesting to check whether there exist fast parallel algorithms for the matroid union and intersection problems when the matroids are not linearly representable. We may assume that a rank oracle or an independence oracle is given for the matroids. In the sequential setting, either of these oracles suffices for running Edmonds' algorithms. The importance of linear representability to algorithm design has been noted previously. For example, Lovász [Lo1] has given a polynomial time algorithm for the matroid matching problem if the given matroid has a linear representation; however, the general problem is not polynomial time solvable [Lo1,JK] (interestingly, this result does not depend on the $P \neq NP$ hypothesis). In the spirit of [Lo1,JK] can one prove negative results in the parallel setting? For other work along these lines see [KUW2].

2) The ideas in this paper can also be applied to finding branching(s) in digraphs. In a directed graph $G = (V, E)$, a *branching rooted at vertex* v is a set of edges that are acyclic, vertex v has indegree zero, and every other vertex has indegree one.

Lovász [Lo3] gives an NC^2 algorithm for finding a branching in a directed graph. It is well-known that sequentially a branching can be found using matroid intersection. Let us first remark that our parallel matroid intersection algorithm gives an RNC^2 algorithm for finding a branching, although the algorithm in [Lo3] is superior not only because it is deterministic, but also because it uses fewer processors. The advantage of the matroid approach is that it extends to the problem of obtaining k edge-disjoint branchings rooted at v, using:

THEOREM 8.1. ([ED2]) *A set $E' \subseteq E$ can be partitioned into k edge-disjoint branchings rooted at v iff*

(i) When considered as a set of undirected edges, E' can be partitioned into k spanning trees.

(ii) Every vertex other than v has indegree k.

Let M be the graphic matroid on E obtained by ignoring edge directions, and let P be the partition matroid on E where the elements of the i^{th} partition are the edges that point into the vertex i. To find a set E' (if it exists) of edges which can be partitioned into k edge-disjoint branchings rooted at v, obtain a maximum cardinality set in the intersection of M^k and P^k. The problems of partitioning E' into k branchings, and of making this algorithm Las Vegas are left as open problems.

3) What is the parallel complexity of finding

a) a lexicographically first intersection of two matroids

b) a lexicographically first maximum cardinality intersection of two matroids.

Special cases of (a) and (b) are lexicographically first maximal and maximum matching respectively, in bipartite graphs. Analogous problems can also be stated for the graphic matroid. Finding a lexicographically first maximal matching is known to be CC-complete. On the other hand, the parallel complexity of lexicographically first maximum matching is unresolved. Is problem (a) CC-complete?

9 Acknowledgements

We wish to thank Prasoon Tiwari and Umesh Vazirani for valuable discussions during which a restricted version of Theorem 4.1 was obtained. Also, we would like to thank Naveen Garg, Samir Khullar, Shachin Maheshwari, Sachin Patkar, M. Sohoni and Prakash Sunderasan for useful discussions.

References

[Ai] M. Aigner, *Combinatorial Theory*, Springer-Verlag, New York, 1979.

[Cs] L. Csanky, "Fast parallel matrix inversion algorithms", *SIAM J. Computing*, vol. 5, 1976, pp. 618-623.

[BCP] A. Borodin, S.A. Cook and N. Pippinger, "Parallel computation for well-endowed rings and space bounded probabilistic machines", *Inform. and Control* 58, No. 1-3, pp. 113-136.

[Ed1] J. Edmonds, "Minimum Partition of a Matroid into independent subsets", *J. Res. National Bureau of Standards*, 69B, 1965, pp. 67-72.

[Ed2] J. Edmonds, "Edge-disjoint branchings", *Combinatorial Algorithms*, R. Rustin, Ed., Algorithmics Press, New york, 1972, pp. 91-96.

[Gu] D. Gusfield, "Connectivity and edge-disjoint spanning trees", *Inf. Proc. Letters*, 16, 1983, pp. 87-89.

[IF] M. Iri and S. Fujishige, "Use of matroid theory in operations research, circuits, and systems theory", *Int. J. Systems Sci.*, 12, 1, 1981, pp. 27-54.

[IR] A. Itai and M. Rodeh, "The multi-tree approach to reliability in distributed networks", *Proc.* 25^{th} *annual Symp. on Foundations of Comp. Sci.*, 1984, pp. 137-147.

[JK] P.M. Jensen and B. Korte, "Complexity of matroid property algorithms", *SIAM J. Computing*, 11, 1983, pp. 184- 190.

[KK] G. Kishi and Y. Kajitani, "Maximally distant trees and principal partition of a linear graph", *IEEE Trans. Circuit Theory*, CT-16, **3**, 1969, pp. 323-330.

[KR] R.M. Karp and V. Ramachandran, "A survey of Parallel Algorithms for Shared-Memory Machines", UC. Berkeley Tech. Report. To appear in the *Handbook of Theoretical Computer Science*, to be published by North-Holland.

[KUW1] R.M. Karp, E. Upfal and A. Wigderson, "Constructing a Maximum Matching is in Random NC", *Combinatorica*, 6(1), 1986, pp. 35-48. 475.

[KUW2] R.M. Karp, E. Upfal and A. Wigderson, "Are search and decision problems computationally equivalent?", *Proc.* 17^{th} *annual Symp. on Theory of Computing*, 1985, pp. 464- 475.

[La] E.L. Lawler, *Combinatorial Optimization: Networks and Matroids*, Holt, Rhinehart and Winston, New York, 1976.

[Lo1] L. Lovász, "The matroid matching problem", *Algebraic Methods in Graph Theory, II*, Eds.: L. Lovász and V.T. Sós, *Colloq. Math. Soc. János Bolyai*, 25, North-Holland, Amsterdam, 1981, pp. 495-517.

[Lo2] L. Lovász, "On determinants, matchings, and random algorithms", *Fundamentals of Computation Theory, FCT'79*, ed. L. Budach, *Math Research 2*, Akademie-Verlag, Berlin, 1979, pp. 565-574.

[Lo3] L. Lovász, "Computing ears and branchings in parallel", *Proc.* 26^{th} *annual Symp. on Foundations of Computer Science*,1985, pp. 464-467.

[LP] L. Lovász and M. Plummer, *Matching Theory*, Academic Press, Budapest, Hungary, 1986.

[LY] L. Lovász and Y. Yemini, "On generic rigidity in the plane", *SIAM J. Alg. Disc. Meth.*, 3, 1982, pp. 91-98.

[MS] E.W. Mayr and A. Subramanian, "The complexity of Circuit Value and Network Stability", submitted to *J. of Computer and System Sciences.*

[MVV] K. Mulmuley, U.V. Vazirani and V.V. Vazirani, "Matching is as easy as matrix inversion", *Combinatorica*,7, 1987, pp. 105-114.

[Nw] C. St. J. A. Nash-Williams, "Edge-disjoint spanning trees of finite graphs", *J. London Math. Soc.*, 36, 1961, pp. 445-450.

[PS] C.H. Papadimitriou and K.Steiglitz, "Combinatorial Optimization: Algorithms & Computing". Prentice Hall, 1982. , .

[OIW] T. Ohtsuki, Y. Ishizaki and H. Watanabe, "Topological degrees of freedom and mixed analysis of electrical networks", *IEEE Trans. Circuit Theory*, CT-17, 4, 1970, pp. 491- 499.

[Sc] J. T. Schwartz, "Fast probabilistic algorithms for verification of polynomial identities", *JACM*, vol.

27, 1980, pp. 281-292.

[We] D.J.A. Welsh. *Matroid theory.* Academic Press, New York, 1976.

[Zi] R.E. Zippel, "Probabilistic algorithms for sparse polynomials", *Proc. EUROSAM 79* ed. Ng, *Springer Lecture Notes in Computer Sci.* 72 (1979), 216-226.