Electronic Colloquium on Computational Complexity, Revision 1 of Report No. 66 (2012)

PARALLEL ALGORITHMS FOR MATROID INTERSECTION AND MATROID PARITY

JINYU HUANG *

Abstract. A maximum linear matroid parity set is called a basic matroid parity set, if its size is the rank of the matroid. We show that determining the existence of a common base (basic matroid parity set) for linear matroid intersection (linear matroid parity) is in NC^2, provided that there are polynomial number of common bases (basic matroid parity sets). For graphic matroids, we show that finding a common base for matroid intersection is in NC^2, if the number of common bases is polynomial bounded. To our knowledge, these algorithms are the first deterministic NC algorithms for matroid intersection and matroid parity. We also give a new RNC^2 algorithm that finds a common base for graphic matroid intersection.

Similar to the Tutte's theorem, we derive the determinant criterion for the existence of a common base (basic matroid parity set) for linear matroid intersection (linear matroid parity). Moreover, we prove that if there is a black-box NC algorithm for PIT (Polynomial Identity Testing), then there is an NC algorithm to determine the existence of a common base (basic matroid parity set) for linear matroid intersection (linear matroid parity).

1. Introduction. In the algorithmic view, the problems of linear matroid intersection and linear matroid parity are similar to the graph matching problem. All those three problems are polynomial-time solvable. Thus a question in parallel complexity is that whether all these three problems have NC algorithms. There is an RNC^2 algorithm to find a perfect matching in a general graph [17]. When the graph is planar, Vazirani gives an NC^2 algorithm to determine whether the graph has a perfect matching [21]. In the same paper, an NC^2 algorithm to determine the number of perfect matchings in a planar graph is also presented. When the graph has polynomial number of perfect matchings, Grigoriev and Karpinski give an NC^3 algorithm to find all perfect matchings [7]. Recently, Agrawal, Hoang and Thierauf [1] improve the results of Grigoriev and Karpinski. Specifically, they show that constructing all perfect matchings is in NC^2, provided that the input graph has polynomial number of perfect matchings.

Since there is a strong link between matroids and matchings, it is interesting whether the parallel algorithms for matching can be extended to the parallel algorithms for relevant matroid problems. Based on the Cauchy-Binet theorem and the Isolating Lemma, Narayanan, Saran and Vazirani [18] show that there are RNC^2 algorithms for the problems of linear matroid intersection and linear matroid matching (linear matroid parity). However, whether there are deterministic NC algorithms for the problems of linear matroid parity and linear matroid intersection is still open. Matroid intersection and matroid parity have many applications. For example, they are used in approximation algorithms [2, 3] and network coding [9]. Thus the efficient NC algorithms for matroid intersection and matroid parity are very useful. Moreover, those NC algorithms may also lead to fast sequential algorithms.

Recently, elegant matrix formulations for the problems of linear matroid intersection and linear matroid parity are obtained [8, 10]. Based on these formulations, fast randomized algebraic algorithms for the problems of linear matroid intersection and linear matroid parity are presented [4, 8]. Both of these algorithms are based on the work of Coppersmith and Winograd [6] Mucha and Sankowski [16].

1.1. Main Results. We define the following problems for matroid intersection and matroid parity.

*Department of Applied Mathematics, Illinois Institute of Technology (jyhuangzju@gmail.com)

ISSN 1433-8092

- Existence: determine whether there is a common base (basic matroid parity set) for matroid intersection (matroid parity).
- Enumeration: count the number of common bases (basic matroid parity sets) for matroid intersection (matroid parity).
- Find-One: find a common base for matroid intersection.
- Construct-All: construct all common bases for matroid intersection.

Assume that there are polynomial bounded number of common bases (basic matroid parity sets) for linear matroid intersection (linear matroid parity). We show that the Enumeration for linear matroid intersection and the Enumeration for linear matroid parity are in NC^2. As a conclusion, the Existence for linear matroid intersection and the Existence for linear matroid parity are in NC^2. Moreover, we show that the Construct-All for graphic matroid intersection is in NC^2. As a result, the corresponding Find-One is in NC^2 for graphic matroid intersection. All these algorithms are based on the work of Agrawal, Hoang and Thierauf [1]. The condition that there are polynomial number of common bases (basic matroid parity sets) is nontrivial. Since the Existence for linear matroid intersection and the Existence for linear matroid parity are related to PIT (Polynomial Identity Testing). Kabanets and Impagliazzo [11] show that any deterministic algorithm for PIT implies the arithmetic circuit lower bounds. For the historic reason, it is hard to prove the arithmetic circuit lower bounds and it is still an open problem to design a polynomial time algorithm for PIT. So our algorithms are best possible based on current techniques. In fact, we prove that if there is a black-box NC algorithm for PIT, then there is an NC algorithm of the Existence for linear matroid intersection and there is an NC algorithm of the Existence for linear matroid parity.

In order to obtain the NC algorithms, we relate the number of common bases for linear matroid intersection and basic matroid parity sets for linear matroid parity with the matrix formulations of these problems introduced by Geelen, Iwata [10] and Harvey [8]. To achieve this goal, we use the Theorem 4.1 and the Theorem 4.2 in [8]. Hence we also answer a problem of Harvey [8][1].

Besides, We give a new RNC^2 algorithm for graphic matroid intersection, which is simpler than that in [18].

2. Notations and Preliminaries.

2.1. Linear Algebra. Given a matrix A, let $A_{R,C}$ denote the submatrix induced by rows R and columns C. A submatrix of A containing all rows (columns) is denoted by $A_{*,C}$ ($A_{R,*}$). An entry of A is denoted by $A_{i,j}$. The submatrix $A_{del(i,j)}$ of A denotes the submatrix without row i and column j. The adjoint of A is denoted by $adj(A)$. An $n \times n$ square matrix A is called skew-symmetric if $A = -A^T$. Now assume that n is even for the skew-symmetric matrix A. Let $pf(A)$ denote the Pfaffian of A.

Lemma 2.1.

$$det(A) = (pfA)^2$$

[1]Harvey asked the following problem at the end of the paper. "Do Theorem 4.1 and Theorem 4.2 have other applications?"

A Vandermonde matrix V has the form

$$\begin{pmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^{n-1} \end{pmatrix}$$

We have

LEMMA 2.2.

$$det(V) = \prod_{i \neq j} (x_j - x_i)$$

In particular, if all $x_1, \cdots, x_n$ are distinct, V is nonsingular.

A Tutte matrix T for a simple directed graph G with even number of vertices can be defined as follows

$$T_{ij} = \begin{cases} x_e & \text{if } (v_i, v_j) \in E \\ -x_e & \text{if } (v_j, v_i) \in E \\ 0 & \text{otherwise} \end{cases}$$

where x_e is an indeterminate. If G is an undirected graph, we can first give an arbitrary orientation of G and then define the Tutte matrix as above.

2.2. Matroid. All definitions and facts in this subsection can be found in [20, 22]. A pair $M = (S, \mathcal{I})$ is called a matroid if S is a finite set (called the ground set) and $\mathcal{I}$ is a nonempty collection of subsets of S (called independent sets) satisfying:
1) if $I \subseteq J \in \mathcal{I}$, then $I \in \mathcal{I}$.
2) if $I, J \in \mathcal{I}$ and $|I| < |J|$, then there is an $e \in J - I$ such that $I \cup \{e\} \in \mathcal{I}$.

The rank of $U \subseteq S$ is

$$r_M(U) = \max\{|I| | I \subseteq U, I \in \mathcal{I}\}$$

An independent set $B \in \mathcal{I}$ with maximum rank is called a base of M. All bases have the same size, which is called the rank of M. A graphic matroid for a graph G can be defined as $M = (E(G), \mathcal{I})$ where $I \subseteq E(G)$ is independent if I is acyclic in G.

Let Q be a matrix with n columns. Then we define a matroid $M = (S, \mathcal{I})$ with $S = \{1, 2, \cdots, n\}$ as follows. A set $I \subseteq S$ is independent in M if the columns of Q indexed by I are linearly independent. A matroid obtained in this way is called a linear matroid. If the entries of Q are in a field $\mathbb{F}$, then M is representable over $\mathbb{F}$. Many important matroids are linear representable such as graphic matroids. Without specific statement, all matroids in this paper are linear representable. Suppose that $M = (E(G), \mathcal{I})$ is the graphic matroid of G where M has rank r and $|E| = m$. Further, assume that $\mathbb{F}_p$ is a finite field with p elements where p is a prime.

LEMMA 2.3. *The graphic matroid M has an $r \times m$ matrix representation Q over the field $\mathbb{F}_p$. Each nonzero entry of Q is -1 or 1, and each column of Q has at most two non-zero entries. If a column contains two non-zero entries, their signs are opposite.*

The Lemma 2.3 follows from [22] (page 148, Theorem 1 and 2).

Matroid Intersection. Let the pair of linear matroids be represented over the same field. Given two maroids $M_1 = (S, \mathcal{I}_1)$ and $M_2 = (S, \mathcal{I}_2)$, the matroid

intersection problem is to find a maximum common independent set $I \in \mathcal{I}_1 \cap \mathcal{I}_2$. There is a matrix formulation for the linear matroid intersection problem. More details can be found in [8]. Let Q^1 be an $r \times n$ matrix whose columns represent M_1 and let Q^2 be an $n \times r$ matrix whose rows represent M_2. Let T be a diagonal matrix where $T_{i,i}$ is an indeterminate t_i for $1 \leq i \leq n$. Given $J \subseteq S$, define the matrix

$$Z(J) := \begin{pmatrix} & Q^1_{*,J} & Q^1_{*,\bar{J}} \\ Q^2_{J,*} & & \\ Q^2_{\bar{J},*} & & T_{del(J,J)} \end{pmatrix}$$

where $\bar{J} = [n] - J$. Let $\bar{r}(J)$ denote the maximum size of an intersection between M_1/J and M_2/J. We have

LEMMA 2.4. *Given $J \subseteq S$, $rank(Z(J)) = n + r_1(J) + r_2(J) - |J| + \bar{r}(J)$. Let M_1 and M_2 have a common base. The matrix $Z(J)$ is nonsingular if and only if J is a subset of a common base.*

The proof is in [8](Theorem 4.1 and 4.2). Let $J := \phi$, then

$$Z(\phi) = \begin{pmatrix} & Q^1 \\ Q^2 & T \end{pmatrix}$$

Assume that both M_1 and M_2 have rank r. A corollary from the Lemma 2.4 is as follows, which is of independent interest.

COROLLARY 2.5. *Let Z denote $Z(\phi)$. The matroids M_1 and M_2 have a common base $B \in \mathcal{I}_1 \cap \mathcal{I}_2$ if and only if $det(Z) \not\equiv 0$.*

Proof. If M_1 and M_2 have a common base, then $det(Z) \not\equiv 0$ directly follows from the second part of the Lemma 2.4. Now suppose $det(Z) \not\equiv 0$, then $rank(Z) = n + r$. From the Lemma 2.4, we have $rank(Z) = n + \bar{r}(\phi)$. So we have $\bar{r}(\phi) = r$. Since $\bar{r}(\phi)$ is the maximum size of a common independent set between M_1 and M_2, M_1 and M_2 have a common base. □

Similar to the Tutte's theorem, the Corollary 2.5 is a determinant criterion for the existence of a common base for matroid intersection.

Matroid Parity. Let $M = (S, \mathcal{I})$ be a matroid and let $S_1, S_2, \cdots, S_m$ be a partition of S into pairs where $S = S_1 \cup \cdots \cup S_m$. The matroid parity problem is to find a maximum size collection $\{S_{i_1}, \cdots, S_{i_k}\}$ such that $S_{i_1} \cup S_{i_2} \cup \cdots \cup S_{i_k}$ is independent in M. Sometimes the matroid parity problem is also called the matroid matching problem. There are polynomial time algorithms for the linear matroid parity problem [4, 14, 20]. Now we give a matrix formulation for the linear matroid parity problem introduced by Geelen and Iwata [10]. Let Q be an $r \times 2m$ matrix whose columns represent the matroid M. We construct a graph $G = (S, E)$ as follows. The vertex set S is the ground set of M. The edge set E consists of all the pairs $S_1, S_2, \cdots, S_m$. As a result, there are exactly m edges in G and those m edges correspond to the partition of S. Let T be the Tutte matrix of G and let $\mathcal{V}(M)$ denote the cardinality of the maximum matroid parity set.

LEMMA 2.6. *Define*

$$K := \begin{pmatrix} & Q \\ -Q^T & T \end{pmatrix}$$

Then $2\mathcal{V}(M) = rank(K) - 2m$.

The proof of the Lemma 2.6 follows from [10] (Theorem 4.1). Since T is a skew-symmetric matrix, K is also a skew-symmetric matrix.

2.3. Parallel Complexity. Most results in this subsection follows from Karp and Ramachandran [12] and Papadimitriou [19]. Let $\mathcal{C} = (C_0, C_1, \cdots)$ be a uniform family of Boolean circuits. The class NC^k where $k > 1$ is the class of problems that are solvable by a uniform family of Boolean circuits with $O(\log^k(n))$ depth and $poly(n)$ size where $poly(n) = \bigcup_{k\geq 1} O(n^k)$. We define NC^1 to be the class of problems that are solvable by alternating Turing machines in $O(\log(n))$ time. The class NC is defined to be $\bigcup_{k\geq 1} NC^k$. The class RNC is the randomized version of the class NC. The formal definition of RNC can be found in [19]. Given an n-bit integer x and an integer i where $1 \leq i \leq n$, the Powering problem is to compute x^i.

LEMMA 2.7. *Addition, subtraction, multiplication and division of integers are solvable in NC^1. Moreover, the Powering can be computed in NC^1.*

The proof is in [5, 12]. We assume that binary arithmetic operations in a field take unit time. Then we have

LEMMA 2.8. *Let A, B be $n \times n$ matrices with entries in a field $\mathbb{F}$. Then $det(A)$, A^{-1}, $rank(A)$ and AB can be computed in NC^2. If A is skew-symmetric, $pf(A)$ can be computed in NC^2.*

Above results can be found in [12, 15].

3. NC Algorithms for Matroid Intersection. Given two maroids $M_1 = (S, \mathcal{I}_1)$ and $M_2 = (S, \mathcal{I}_2)$ with $|S| = n$, let Q^1 be an $r \times n$ matrix whose columns represent M_1 and let Q^2 be an $n \times r$ matrix whose rows represent M_2. Let T be a diagonal matrix $diag(t_1, \cdots, t_n)$ where t_i is an indeterminate for $1 \leq i \leq n$. Define

$$Z := \begin{pmatrix} & Q^1 \\ Q^2 & T \end{pmatrix}$$

Let π and π^i with an index i denote the subsets of $\{1, 2, \cdots, n\}$ with $n - r$ distinct elements. Further, $\pi(j)$ and $\pi^i(j)$ represent the jth element of π and π^i respectively (the elements of π and π^i are listed in the nondecreasing order). It can be observed that $\det(Z)$ is a multilinear polynomial such that

$$\det(Z) = C_1T_1 + \cdots + C_kT_k$$

where C_i is a constant and $T_i = \prod_{j=1}^{n-r} t_{\pi^i(j)}$ is a monomial. Informally, $\det(Z)$ consists of k nonzero terms such that each term $C_i \prod_{j=1}^{n-r} t_{\pi^i(j)}$ corresponds to a common base $B = S - \{\pi^i(1), \cdots, \pi^i(n-r)\}$.

THEOREM 3.1. *There is a bijection between the nonzero terms (monomials) of $\det(Z)$ and the set of common bases. Specifically, each term $C_i \prod_{j=1}^{n-r} t_{\pi^i(j)}$ of $\det(Z)$ is nonzero if and only if $B = S - \{\pi^i(1), \cdots, \pi^i(n-r)\}$ is a common base.*

Proof. Let $S = \{1, \cdots, n\}$ be the ground set of two matroids. Let $B = \{1, 2, \cdots, r\}$ be a subset of S where r is the rank of two matroids and let B^1 (B^2) be the first r columns (rows) of Q^1 (Q^2). Let $C_1 \prod_{i=r+1}^{n} t_i$ be a term in $\det(Z)$. It is sufficient to show that B is a common base if and only if $C_1 \neq 0$. We set

$$t_i := \begin{cases} 0 & 1 \leq i \leq r \\ 1 & \text{otherwise} \end{cases} \tag{3.1}$$

Since $\det(Z)$ is a multi-linear polynomial such that each term consists of $n-r$ different variables, $\det(Z) = C_1$ after the assignment of t_i in (3.1). On the other hand, the

matrix Z has the form (after the assignment in (3.1))

$$Z = \begin{pmatrix} & B^1 & \bar{B^1} \\ B^2 & & \\ \bar{B^2} & & I \end{pmatrix} \tag{3.2}$$

Then $|\det(Z)| = |\det(B^1)||\det(B^2)|$. Thus we have $|C_1| = |\det(B^1)||\det(B^2)|$. From be definition of B, B_1 and B_2, we can conclude that B is a common base if and only if $\det(B^1) \neq 0$ and $\det(B^2) \neq 0$. As a consequence, the set B is a common base if and only if $C_1 \neq 0$. In other words, B is a common base if and only if $C_1 \prod_{i=r+1}^{n} t_i$ is a nonzero term in $\det(Z)$. For any other subset B' of S with r elements, we can interchange rows and columns of Z such that the first r columns(rows) of $Q^1(Q^2)$ in Z represent B'. Since interchanging rows and columns only change the sign of the determinant, we can apply the same argument as above to B' after change. Similarly, for any other term $C_i \prod_{j=1}^{n-r} t_{\pi^i(j)}$ in $\det(Z)$, we can interchange rows and columns of Z such that the last $n-r$ entries of T are $t_{\pi^i(1)}, t_{\pi^i(2)}, \cdots, t_{\pi^i(n-r)}$. Then we can apply the same argument as above.

Now we map each common base $B = S - \{\pi^i(1), \cdots, \pi^i(n-r)\}$ to a nonzero term $C_i \prod_{j=1}^{n-r} t_{\pi^i(j)}$. This map is injective, since $S - \{\pi^i(1), \cdots, \pi^i(n-r)\}$ is unique for $\prod_{j=1}^{n-r} t_{\pi^i(j)}$. It is also surjective, since each nonzero term $C_i \prod_{j=1}^{n-r} t_{\pi^i(j)}$ corresponds to a common base $B = S - \{\pi^i(1), \cdots, \pi^i(n-r)\}$, which is proved before. □

If M_1 and M_2 are the graphic matroids, then each nonzero term's coefficient is -1 or $+1$ in $\det(Z)$. So we have

COROLLARY 3.2. *Let Q^1 be an $r \times n$ matrix whose columns represent the graphic matroid M_1 and let Q^2 be an $n \times r$ matrix whose rows represent the graphic matroid M_2. Suppose that there are k common bases. We have*

$$\det(Z) = C_1 \prod_{j=1}^{n-r} t_{\pi^1(j)} + \cdots + C_k \prod_{j=1}^{n-r} t_{\pi^k(j)}$$

where $C_i \neq 0$ for each i. Each term $C_i \prod_{j=1}^{n-r} t_{\pi^i(j)}$ of $\det(Z)$ is nonzero if and only if $B = S - \{\pi^i(1), \cdots, \pi^i(n-r)\}$ is a common base. Moreover, every coefficient C_i is either -1 or 1 in $\det(Z)$.

Proof. Suppose that a nonzero term in $\det(Z)$ is $C_1 \prod_{i=r+1}^{n} t_i$. It is sufficient to show that C_1 is either -1 or 1. We set

$$t_i := \begin{cases} 0 & 1 \leq i \leq r \\ 1 & \text{otherwise} \end{cases} \tag{3.3}$$

Then $|\det(Z)| = |C_1|$. Just as in the proof of the Theorem 3.1, the matrix Z has the form

$$Z := \begin{pmatrix} & B^1 & \bar{B^1} \\ B^2 & & \\ \bar{B^2} & & I \end{pmatrix} \tag{3.4}$$

where B^i is a nonsingular $r \times r$ matrix for $i = 1, 2$. From the Lemma 2.3, we know that each column(row) of $B^1(B^2)$ consists of only one nonzero entry (-1 or 1) or consists of two nonzero entries -1 and 1. Thus we can apply following two types elementary row(column) operations so that $B_1(B_2)$ becomes an identity matrix I:

(a) Adding one row(column) to the other row(column).
(b) Multiplying -1 to each entry of a row(column).
After those operations, Z becomes

$$Z' := \begin{pmatrix} & I & \bar{B}^1 \\ I & & \\ \bar{B}^2 & & I \end{pmatrix} \tag{3.5}$$

We have $|\det(Z')| = 1$. Since operations (a) and (b) do not change the absolute value of the determinant, we have $|C_1| = 1$. For other nonzero terms, we can apply the similar argument. □

Let P be a polynomial such that M_1 and M_2 have at most $P(n)$ common bases. Assume that there is an oracle O that computes P with input n. Define an $n \times n$ matrix $T_m(t)$ as

$$T_m(t)_{ij} := \begin{cases} t^{(m^i \mod q)} & i = j \\ 0 & \text{otherwise} \end{cases}$$

where t is a variable, $q > nP^2(n)$ is a prime and $m \in \mathbb{F}_q$. Define matrices $Z_m(t)$ for $1 \leq m < q$ as

$$Z_m(t) := \begin{pmatrix} & Q^1 \\ Q^2 & T_m(t) \end{pmatrix}$$

Let $D_m(t)$ denote the determinant of $Z^{(m)}(t)$, then we have

$$\begin{aligned} D_m(t) &= \det(Z_m(t)) \\ &= \sum_i C_m(i) t^{e_m(\pi^i)} \end{aligned}$$

where $e_m(\pi^i) = \sum_{j=1}^{n-r} (m^{\pi^i(j)} \mod q)$. Now suppose that there are k common bases.

LEMMA 3.3. *Let $\pi^1, \cdots, \pi^k$ denote k subsets of $\{1, 2, \cdots, n\}$ such that each π^i consists of $n - r$ distinct elements. There is a m with $1 \leq m < q$ such that $e_m(\pi^i) \neq e_m(\pi^j)$ for all i and j with $i \neq j$ in $D_m(t)$.*

Proof. Define a polynomial

$$p_\pi(x) = \sum_{i=1}^{n-r} x^{\pi(i)}$$

Thus $e_m(\pi^i) \neq e_m(\pi^j)$ is equivalent to $p_{\pi^i}(m) \neq p_{\pi^j}(m) \mod q$. Since $\pi^i \neq \pi^j$ for each $i \neq j$, we have $p_{\pi^i} \not\equiv p_{\pi^j}$. Moreover, the degree of each polynomial p_π is bounded by n. Thus $p_{\pi^i} - p_{\pi^j}$ can have at most n roots for each $i \neq j$. Since $\binom{k}{2} n < P^2(n)n < q$, there is an element $m \in \mathbb{F}_q$ such that $p_{\pi^i}(m) - p_{\pi^j}(m) \neq 0$ for all pairs of polynomials p_{π^i} and p_{π^j} with $i \neq j$. Then there is a $m \in \mathbb{F}_q$ such that $e_m(\pi^i) \neq e_m(\pi^j)$ for each $i \neq j$. □

It is not hard to see that $U = n(q-1)$ is the upper bound for the degree of $D_m(t)$. Then we can write $D_m(t)$ as

$$D_m(t) = \sum_{i=0}^{U} C_m(i) t^i$$

From the Theorem 3.1 and the Lemma 3.3, there is a $m \in \mathbb{F}_q$ such that $D_m(t)$ consists of exactly k nonzero terms. So the number of common bases is the number of nonzero terms in $D_m(t)$. Define two vectors

$$D_m = (D_m(0), D_m(1), \cdots, D_m(U))^T$$
$$C_m = (C_m(0), C_m(1), \cdots, C_m(U))^T$$

Further, define a matrix $\bar{V}$ as

$$\begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 1^2 & \cdots & 1^U \\ 1 & 2 & 2^2 & \cdots & 2^U \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & U & U^2 & \cdots & U^U \end{pmatrix}$$

The matrix $\bar{V}$ is nonsingular, since $\bar{V}$ is the Vandermonde matrix with $x_i = i$ for $0 \leq i \leq U$. Thus we have

$$D_m = \bar{V} C_m$$

Now the Algorithm 1 is an NC^2 algorithm of the Enumeration for matroid intersection.

Algorithm 1: A parallel algorithm of the Enumeration for matroid intersection

input : An $r \times n$ matrix Q_1 and an $n \times r$ matrix Q_2 that represent the matroids M_1 and M_2 respectively.

begin
  query the oracle O to obtain $P(n)$;
  $q := FindPrime(nP^2(n))$;
  for $1 \leq m \leq q-1$
    construct the matrix $Z_m(t)$;
    construct the matrix $\bar{V}$;
    compute the vector D_m;
    $C_m := \bar{V}^{-1} D_m$;
    for $0 \leq i \leq (q-1)n$
      if $C_m(i) \neq 0$, then $C_m(i) := 1$;
    $N_m := \sum_{i=0}^{n(q-1)} C_m(i)$;
  $N := \max_m(N_m)$;
  return N;
end

THEOREM 3.4. *Assume that there are polynomial bounded number of common bases. Then the Algorithm 1 is an NC^2 algorithm of the Enumeration for linear matroid intersection. As a consequence, the Existence for linear matroid intersection is also solvable in NC^2.*

Proof. We first prove the correctness of the algorithm. By the Theorem 3.1 and the Lemma 3.3, the number of nonzero coefficients $C_m(i)$ in $D_m(t)$ is the number of common bases for some $m < q$. Since we do not know m, we can compute the number of nonzero coefficients $C_m(i)$ for each m with $1 \leq m < q$. The largest one is the number of common bases.

Next, we show that the Algorithm 1 is in NC^2. The algorithm needs a procedure $FindPrime()$ to find a prime q. Since $P^2(n)$ is a polynomial of n, without loss of generality, assume $nP^2(n) = n^k$ for some k. By the prime number theorem, there is a prime between n^k and n^{k+1}. In $FindPrime()$, we can in parallel test whether q is a prime for each $n^k \leq q \leq n^{k+1}$. The test can be done by trial division. In other words, try dividing q by each integer $2, \cdots, \lceil \sqrt{q} \rceil$ in parallel. Thus the procedure $FindPrime()$ can be done with $poly(n)$ processors and $O(\log(n))$ parallel time. Finding the maximum value among m elements can be done in NC^2, then the step 12 of the algorithm is in NC^2. So we need only focus on the steps from 5 to 11 in the algorithm. We can run the steps from 5 to 11 in parallel for each $1 \leq m \leq q-1$. For each m, the steps from 5 to 11 can be computed in NC^2. The reason is as follows. The matrices $Z^{(m)}(t)$ and the matrix $\bar{V} = (v_{ij})$ with $v_{ij} = i^j$(assume $0^0 = 1$) can be constructed in NC^1, since the Powering can be done in NC^1. Compute each entry of the vector D_m in parallel. Since the determinant is solvable in NC^2, D_m can be computed in NC^2. Compute each entry of C_m in parallel, which can be done in NC^2.

Thus the Enumeration for matroid intersection is in NC^2. Since $N \neq 0$ if and only if there is a common base, the Existence for matroid intersection is in NC^2. □

Next, we show that constructing all common bases for graphic matroid intersection is in NC^2. At first, we revise the definition of $T_m(t)$. Find n distinct prime numbers $q_1, \cdots, q_n$ such that $\max_i q_i = O(n^2)$. Define

$$T_m(t)_{ij} := \begin{cases} q_i t^{(m^i \mod q)} & i = j \\ 0 & \text{otherwise} \end{cases}$$

By the Corollary 3.2, the determinant $D_m(t)$ becomes

$$D_m(t) = \sum_i C_m(i) t^{e_m(\pi^i)}$$

where each nonzero coefficient satisfies

$$|C_m(i)| = \prod_{j=1}^{n-r} q_{\pi^i(j)}$$

Thus we can design a parallel algorithm to construct all common bases from $C_m(i)$. Specifically, given a nonzero coefficient $C_m(i)$, test whether $C_m(i) \not\equiv 0(\mod q_j)$ for $1 \leq j \leq n$. If $C_m(i) \not\equiv 0(\mod q_j)$, then t_j does not appear in the nonzero term with the coefficient $C_m(i)$. Thus j is in the common base by the Theorem 3.1. Since testing whether $C_m(i) \equiv 0(\mod q_j)$ can be done in NC^1, we can construct all common bases in NC^2. The pseudocode is as follows.

Algorithm 2: A parallel algorithm of the Construct-All for graphic matroid intersection

input : An $r \times n$ matrix Q_1 that represents the graphic matroid M_1 and an $n \times r$ matrix Q_2 that represents the graphic matroid M_2.

begin
  query the oracle O to obtain $P(n)$;
  $q := FindPrime(nP^2(n))$;
  for $1 \leq m \leq q-1$
    $I_m := \emptyset$;
    construct the matrix $Z^{(m)}(t)$;
    construct the matrix $\bar{V}$;
    compute the vector D_m;
    $C_m := \bar{V}^{-1} D_m$;
    for $0 \leq i \leq n(q-1)$
      $B := \emptyset$;
      for $1 \leq j \leq n$
        if $C_m(i) \not\equiv 0 \mod q_j$
          $B := B \cup \{j\}$;
      $I_m := I_m \cup \{B\}$;
  $I := I_{m_0}$ where $|I_{m_0}| = \max_m(|I_m|)$;
  return I;
end

THEOREM 3.5. *The Construct-All for graphic matroid intersection can be done in NC^2, provided that there are polynomial bounded number of common bases. Thus, the corresponding Find-One for graphic matroid intersection is also in NC^2.*

4. RNC Algorithm for Graphic Matroid Intersection. In this section, we give an RNC algorithm for graphic matroid intersection. Our algorithm is "simpler" than the algorithm in [18]. Since our algorithm of finding a common base is nearly identical to the algorithm of finding a perfect matching in a bipartite graph that is introduced in [17], which is easy to be understood and programmed. Recall that the matrix Z and T defined in the Section 2.2, we have

$$Z = \begin{pmatrix} & Q^1 \\ Q^2 & T \end{pmatrix}$$

where $T = \mathrm{diag}(t_1, \cdots, t_n)$. From the definition of Z, we have $Z_{(r+i)(r+i)} = t_i$ for $1 \leq i \leq n$. The RNC^2 algorithm to find a common base for graphic matroid intersection is as follows.

Algorithm 3: An RNC^2 algorithm of the Find-One for graphic matroid intersection

input : An $r \times n$ matrix Q_1 and an $n \times r$ matrix Q_2 that represent the graphic matroids M_1 and M_2 respectively.

1 **begin**
2 In parallel, assign a random integer weight w_i from $[1, 2n]$ to each t_i for $1 \leq i \leq n$, each weight is chosen uniformly and independently;
3 Replace t_i by 2^{w_i} in the matrix Z;
4 Compute $\det(Z)$;
5 If $\det(Z) = 0$, return False;
6 Else, obtain the highest power w of 2 such that $2^w \mid \det(Z)$;
7 Compute $adj(A)$, its (j,i)-th entry is $\det(Z_{del(i,j)})$;
8 Let $B := \{1, 2, \cdots, n\}$;
9 For each $Z_{(r+i)(r+i)}$ where $1 \leq i \leq n$ do in parallel: Compute $\det(Z_{del(r+i,r+i)})2^{w_i}/2^w$. If this quantity is odd, set $B := B - \{i\}$;
10 Return B;
11 **end**

THEOREM 4.1. *The Algorithm 3 is an RNC^2 algorithm of the Find-One for graphic matroid intersection.*

In order to prove the theorem, we need the following lemma, which is called the Isolating Lemma.

LEMMA 4.2. *Let $S = \{x_1, \cdots, x_n\}$ be a set with n elements and let F be a family of subsets of S. If each element x_i of S is assigned an integer weight w_i that is chosen uniformly and independently from $[1, 2n]$, then*

$$Pr(\textit{there is a unique minimum weight set in } F) \geq 1/2$$

where the weight of the set $S_j \in F$ is $\sum_{x_i \in S_j} w_i$.

The proof is in [17]. Now we prove the correctness of the algorithm.

Proof. (Theorem 4.1) If there is no common base, then $\det(Z) \equiv 0$ by the Corollary 2.5. So the step 5 correctly returns false when there is no common base. Suppose that there is at least one common base, we show that with probability at least $1/2$, the algorithm returns a common base. Let $U := \{t_1, \cdots, t_n\}$. From the Corollary 3.2, we know that each common base for graphic matroid intersection corresponds to a subset of U. Let F be a family of all those subsets of U that correspond to common bases for graphic matroid intersection. By the Isolating Lemma, there is a unique minimum weight subset after the step 3 of the algorithm with probability at least $1/2$. Assume that C is the unique minimum weight subset in F after the step 3 of the algorithm. Without loss of generality, let $C := \{t_{r+1}, \cdots, t_n\}$ and let $w = \sum_{i=r+1}^{n} w_i$, which is the weight of C. Then C corresponds to the common base $B := \{1, \cdots, r\}$. From the Corollary 3.2, we have

$$\det(Z) = C_1 \cdot T_1 + \cdots + C_k \cdot T_k$$

where $|C_i| = 1$ for $1 \leq i \leq k$. After the assignment of t_i by 2^{w_i} for $1 \leq i \leq n$, there is a term $C_i \cdot T_i = \pm 2^w$. Moreover, every other term $C_j \cdot T_j$ satisfies $2^w \mid C_j \cdot T_j$, since w is the minimum weight. So $\det(Z) \neq 0$ and 2^w is the highest power of 2 that divides $\det(Z)$. Next, we show that the algorithm returns the common base $B := \{1, \cdots, r\}$ that corresponds to the subset C. Notice that the entry $Z_{(r+i)(r+i)}$ of the matrix Z is

t_i. If $t_i \in C$, $\det(Z_{del(r+i,r+i)}) \cdot 2^{w_i}/2^w$ is odd. Otherwise $\det(Z_{del(r+i,r+i)}) \cdot 2^{w_i}/2^w$ is even. Since $t_i \in C$ if and only if $i \notin B$, the set B that is returned by the algorithm is the common base $\{1, \cdots, r\}$. Thus the Algorithm 2 is an RNC^2-algorithm that finds a common base for graphic matroid intersection. □

If the maximum common independent set for graphic matroid intersection is not a base, it is possible to use a similar technique that finds a maximum matching of a bipartite graph in [17] to obtain an RNC^2 algorithm of finding a maximum common independent set.

5. NC Algorithms for Matroid Parity. Let Q be an $r \times 2m$ matrix whose columns represent the matroid $M = (S, \mathcal{I})$ and let T be the Tutte matrix of $G = (S, E)$. The unique perfect matching $\{S_1, \cdots, S_m\}$ of G is the partition of S into pairs. Assume that the maximum parity set has size r. Then $r = 2n$ for some integer n. Let

$$K := \begin{pmatrix} & Q \\ -Q^T & T \end{pmatrix}$$

Since K is skew-symmetric, we can compute the Pfaffian $pf(K)$ of K. Each nonzero term of $pf(K)$ contains $(2m + r)/2 = m + n$ entries of K. Because Q is an $r \times 2m$ matrix and the north-west submatrix of K is the zero matrix, each nonzero term of $pf(K)$ contains r elements from Q and $m - r/2 = m - n$ elements from T. Thus,

$$pf(K) = C_1 T_1 + \cdots + C_k T_k$$

where $C_i \neq 0$ is a constant and $T_i = T_{i_1 j_1} T_{i_2 j_2} \cdots T_{i_{m-n} j_{m-n}}$ for each $1 \leq i \leq k$. The monomial T_i in $pf(K)$ corresponds to a matching $\{(i_1, j_1), \cdots, (i_{m-n}, j_{m-n})\}$ in G. Let B_i denote the set of pairs $\{S_1, \cdots, S_m\} - \{(i_1, j_1), \cdots, (i_{m-n}, j_{m-n})\}$. Similar to the matroid intersection problem, we have the following theorem for the basic matroid parity problem.

THEOREM 5.1. *There is a bijection between the nonzero terms of $pf(K)$ and the set of basic matroid parity sets. Moreover, every nonzero term $C_i T_i$ maps to a basic matroid parity set B_i.*

Proof. Let $T_d = T_{r+1,r+2} \cdots T_{2m-1,2m}$ and $\bar{B}_d = \{(r+1, r+2), (r+3, r+4), \cdots, (2m-1, 2m)\}$. We show that $B_d = \{S_1, \cdots, S_m\} - \bar{B}$ is the basic matroid parity set if and only if $C_d \neq 0$. The monomial T_d corresponds to the $(2m-r) \times (2m-r)$ south-east submatrix $T_{(2m-r)(2m-r)}$ of K where

$$K = \begin{pmatrix} & Q_1 & \bar{Q}_1 \\ -Q_1^T & T_{rr} & \\ -\bar{Q}_1^T & & T_{(2m-r)(2m-r)} \end{pmatrix}$$

Let K_{NW} be the north-west submatrix of K, which is

$$K_{NW} = \begin{pmatrix} & Q_1 \\ -Q_1^T & T_{rr} \end{pmatrix}$$

Next, we set each T_{ij} appearing in T_d to be 1. So the corresponding $T_{ji} = -1$. Further, we set any other T_{ij} in the matrix T to be 0. Since the matrix K_{NW} is skew-symmetric, we have (after the assignment of T_{ij})

$$|pf(K_{NW})| = |pf(K)| = |C_d|$$

Then we have

$$|\det(Q_1)|^2 = |\det(K_{NW})| = |pf(K_{NW})|^2 = |C_d|^2$$

Thus $1, 2, \cdots, r$ are linear independent columns of Q if and only if $C_d \neq 0$. As a result, B_d is a basic matroid parity set of M if and only if $C_d \neq 0$. Since each row or column of T consists of only one indeterminate, the argument for the general case is similar by permutating rows and columns of K. □

There is a corollary of the Theorem 5.1, which is a determinant criterion for the existence of a basic matroid parity set.

COROLLARY 5.2. *Let the linear matroid M and the matrix K for matroid parity be defined as above. There is a basic matroid parity set of M if and only if* $\det(K)$ *is not identically zero.*

Proof. From the Theorem 5.1, we know that $pf(K) \equiv 0$ if and only if the linear matroid M has no basic matroid parity set. Since $\det(K) = (pfK)^2$, the statement follows. □

Let $A = (a_{ij})$ be the adjacency matrix of $G = (S, E)$. Suppose that $P(2m)$ is the upper bound of the number of basic matroid parity sets. Define matrices $T_d(x) = (t_{ij}(x))$ as

$$t_{ij}(x) := \begin{cases} a_{ij}x^{(d^{(2mi+j)} \mod q)} & i \leq j \\ -a_{ij}x^{(d^{(2mi+j)} \mod q)} & \text{otherwise} \end{cases}$$

where q is a prime such that $q \geq (2m)^2P^2(2m)$ and $d \in \mathbb{F}_q$. Then define

$$K_d(x) := \begin{pmatrix} Q & \\ -Q^T & T_d(x) \end{pmatrix}$$

The Pfaffian $D_d(x)$ of $K_d(x)$ is a polynomial that satisfies

$$D_d(x) = pf(K_d(x)) = \sum_i C_i x^{e_d(i)}$$

where $e_d(i) = (d^{2mi_1+j_1} + \cdots + d^{2mi_{m-n}+j_{m-n}}) \mod q$. Similar to the Lemma 3.3, there is a d with $1 \leq d < q$ such that every $e_d(i)$ differs in $D_d(x)$. Since $e_d(i) \leq (m-n)q$ for each i, the upper bound of the exponent is $U = (m-n)q$. Thus $D_d(x)$ has the form

$$D_d(x) = \sum_{i=0}^{U} C_d(i)x^i$$

where $C_d(i)$ is a constant for each i. Define vectors

$$D_d = (D_d(0), D_d(1), \cdots, D_d(U))^T$$
$$C_d = (C_d(0), C_d(1), \cdots, C_d(U))^T$$

Then there exists a d such that the number of nonzero entries of C_d is the number of basic matroid parity sets. Each entry of D_d is a Pfaffian, which can be computed in NC^2 by the Lemma 2.8. The algorithm is as follows.

Algorithm 4: A parallel algorithm of the Enumeration for linear matroid parity

input : An $r \times 2m$ matrix Q that represents the matroid M and a set of pairs $\{S_1, \cdots, S_m\}$ for S such that $S = S_1 \cup S_2 \cup \cdots \cup S_m$.

begin
  query the oracle O to obtain $P(2m)$;
  $q := FindPrime((2m)^2 P^2(2m))$;
  for $1 \leq d \leq q-1$
    construct the matrix $K_d(x)$;
    construct the matrix $\bar{V}$;
    compute the vector D_d;
    $C_d := V^{-1} D_d$;
    for $0 \leq i \leq (q(m-n))$
      if $C_d(i) \neq 0$, $C_d(i) := 1$;
    $N_d := \sum_{i=0}^{(m-n)q} C_m(i)$;
  $N := \max_d(N_d)$;
  return N;
end

THEOREM 5.3. *Suppose that there are polynomial bounded number of basic matroid parity sets. Then the Enumeration for linear matroid parity is in NC^2. As a consequence, the Existence for linear matroid parity is in NC^2.*

6. NC Algorithms for PIT Implies NC Algorithms for Matroid Intersection and Matroid Parity. The problem of PIT (Polynomial Identity Testing) can be defined as follows. Given an arithmetic circuit C computing a multivariate polynomial $P(x_1, \cdots, x_n)$ over a field $\mathbb{F}$, decide if the polynomial $P(x_1, \cdots, x_n)$ is identically zero. The black-box model of PIT can be defined as a model in which the only access to the circuit is by asking for its value on inputs. In this model, every algorithm of PIT must produce a set of points $H = \{(a_1, \cdots, a_n) | a_1, \cdots, a_n \in \mathbb{F}\}$ such that if $P(a_1, \cdots, a_n) = 0$ for each $(a_1, \cdots, a_n) \in H$ then the circuit computes the zero polynomial. We call this set of points H a hitting set for PIT.

THEOREM 6.1. *Suppose that there is a black-box NC algorithm for PIT. Then there is an NC algorithm of the Existence for linear matroid intersection and there is an NC algorithm of the Existence for linear matroid parity.*

Proof. Since there is a black-box algorithm for PIT, there is an NC algorithm that can produce a hitting set. From the Corollary 2.5 we know that there is a common base for linear matroid intersection if and only if $\det(Z)$ is not identically zero. From the definition of the matrix Z, $\det(Z)$ is a multilinear polynomial. Thus we can first generate a hitting set H of $\det(Z)$, which can be done by an NC algorithm for PIT. Then we test whether $\det(Z)$ is zero for all points of H in parallel. Since $\det(Z)$ can be computed in NC^2, we have an NC algorithm of the Existence for matroid intersection. A similar argument can be applied to the Existence for linear matroid parity. Since there is a basic matroid parity set if and only if $\det(K)$ is not identically zero from the Corollary 5.2. □

7. Conclusion and Future Work. Suppose that there are polynomial bounded number of common bases (basic matroid parity sets). We show that the Existence for linear matroid intersection and the Existence for linear matroid parity are in NC^2. Moreover, the Find-One for graphic matroid intersection is in NC^2. All presented

algorithms are oracle algorithms. We also give a new RNC^2 algorithm to find a common base for graphic matroid intersection. This randomized parallel algorithm is simpler than that in [18]. Finally, we show that a black-box NC algorithm for PIT implies an NC algorithm of the Existence for linear matroid intersection and an NC algorithm of the Existence for linear matroid parity.

Our results may be extended to the weighted version. Techniques in [1] may be applicable under the appropriate assumption about the weights.

REFERENCES

[1] Agrawal, M., Hoang, T.M., Thierauf, T.: The polynomially bounded perfect matching problem is in NC^2. In: Proceedings of the 24th annual conference on Theoretical aspects of computer science (STACS), pp. 489-499 (2007)

[2] Berman, P., *Fürer*, M., Zelikovsky, A.: Applications of the Linear Matroid Parity Algorithm to Approximating Steiner Trees. In: Proceedings of International Computer Science Symposium in Russia (CSR), pp. 70-79 (2006)

[3] *Călinescu*, G., Fernandes, C.G., Finkler, U., Karloff, H.: A better approximation algorithm for finding planar subgraphs. In: Proceedings of the 7th annual ACM-SIAM symposium on Discrete algorithms (SODA), pp. 16-25 (1996)

[4] Cheung, H.Y., Lau, L.C., Leung, K.M.: Algebraic algorithms for linear matroid parity problems. In: Proceedings of the 22nd ACM-SIAM Symposium on Discrete Algorithms (SODA), 2011

[5] Chiu, A., Davida, G., Litow, B.: Division in logspace-uniform NC^1. RAIRO - Theoretical Informatics and Applications. 35, 259-275 (2001)

[6] Coppersmith, D., Winograd, S.: Matrix multiplication via arithmetic progressions. In: Proceedings of the 19th annual ACM conference on Theory of computing (STOC), pp. 1-6 (1987)

[7] Grigoriev, D.Y., Karpinski, M.: The matching problem for bipartite graphs with polynomially bounded permanents is in NC. In: Proceedings of the 28th Annual Symposium on Foundations of Computer Science (SFCS), pp. 166-172 (1987)

[8] Harvey, N.J.A.: Algebraic Algorithms for Matching and Matroid Problems. `http://www.cs.ubc.ca/~nickhar/Publications/AlgebraicMatching/Algebraic-SIAM.pdf`

[9] Harvey, N.J.A., Karger, D.R., Murota K.: Deterministic network coding by matrix completion. In: Proceedings of the 16th annual ACM-SIAM symposium on Discrete algorithms (SODA), pp. 489-498 (2005)

[10] James, G., Satoru, I.: Matroid Matching Via Mixed Skew-Symmetric Matrices. Combinatorica 25(2), 187-215 (2005)

[11] Kabanets, V., Impagliazzo, R.: Derandomizing polynomial identity tests means proving circuit lower bounds. Computational Complexity, 13(1-2), 1-46 (2004)

[12] Karp, R.M., Ramachandran, V.: Parallel algorithms for shared-memory machines. In: Handbook of Theoretical Computer Science, Vol A: Algorithms and Complexity. MIT Press 1994

[13] Knuth, D.E.:The Art of Computer Programming vol 1: Fundamental Algorithms (3rd edition). Addison-Wesley, 1997

[14] Lovász, L., Plummer, M.D.: Matching Theory. Elsevier, 1986

[15] Meena, M., Subramanya, P.R., Vinay, V.: A Combinatorial Algorithm for Pfaffians. In: 5th Annual International Conference on Computing and Combinatorics (COCOON). pp. 134-143 (1999)

[16] Mucha, M., Sankowski, P.: Maximum matchings via Gaussian elimination. In: Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science (FOCS) pp. 248- 255 (2004)

[17] Mulmuley, M., Vazirani, U.V., Vazirani, V.V.: Matching is as easy as matrix inversion. In: Proceedings of the 19th annual ACM symposium on Theory of computing (STOC), pp. 345-354 (1987)

[18] Narayanan, H., Saran, H., Vazirani V.V.: Randomized parallel algorithms for matroid union and intersection, with applications to arboresences and edge-disjoint spanning trees. In: Proceedings of the 3rd annual ACM-SIAM symposium on Discrete algorithms (SODA), pp. 357-366 (1992)

[19] Papadimitriou, C.H. Computational complexity. Addison-Wesley, 1994

[20] Schrijver, A.: Combinatorial Optimization: Polyhedra and Efficiency. Springer-Verlag, 2003
[21] Vazirani, V.V.: NC algorithms for computing the number of perfect matchings in $K_{3,3}$-free graphs and related problems. Information and computation 80(2), 152-164 (1989)
[22] Welsh, D.J.A.: Matroid Theory. Academic Press, 1976

ECCC ISSN 1433-8092

http://eccc.hpi-web.de