Learning Read-Once Determinants via the Principal Minor Assignment Problem

A THESIS

SUBMITTED FOR THE DEGREE OF

Master of Technology (Research)

IN THE

Faculty of Engineering

BY

Abhiram Aravind

Computer Science and Automation
Indian Institute of Science
Bangalore – 560 012 (INDIA)

May, 2026

Declaration of Originality

I, **Abhiram Aravind**, with SR No. **04-04-00-10-22-24-1-25238** hereby declare that the material presented in the thesis titled

Learning Read-Once Determinants via the Principal Minor Assignment Problem

represents original work carried out by me in the **Department of Computer Science and Automation** at **Indian Institute of Science** during the years **2024-2026**.
With my signature, I certify that:

- I have not manipulated any of the data or results.
- I have not committed any plagiarism of intellectual property. I have clearly indicated and referenced the contributions of others.
- I have explicitly acknowledged all collaborative research and discussions.
- I have understood that any false claim will result in severe disciplinary action.
- I have understood that the work may be screened for any form of academic misconduct.

Date: 21st May, 2026 Student Signature

In my capacity as supervisor of the above-mentioned work, I certify that the above statements are true to the best of my knowledge, and I have carried out due diligence to ensure the originality of the report.

Advisor Name: Chandan Saha Advisor Signature

DEDICATED TO

My parents and my sister

for their constant support

Acknowledgements

I first thank my advisor Prof Chandan Saha for mentoring me for the two years I have been a Master's student. In addition to teaching me about complexity theory and research, he has also provided me with advice and guidance on my future as well as my character. I am deeply indebted to him for his moral and technical support.

I am grateful for the faculty at Computer Science and Automation at IISc for introducing to me the world of theoretical computer science. The courses I have been taught and the technical guidance I have received have been very helpful.

I thank the complexity theory group in India for exposing me to various areas in complexity theory including areas of research into algebraic complexity theory. I extend my heartfelt gratitude to my collaborators Abhranil Chatterjee, Sumanta Ghosh, Rohit Gurjar, and Roshan Raj without whom this work would not be possible. Working with them on this result was immensely instructive and gratifying.

I am indebted to my family for their constant support throughout my journey at IISc, none of this would have been possible without them.

Finally, I thank my friends at IISc for their support and making my stay here at IISc fun. They have kept me company and have been a reliable source of help. I especially thank my friend and senior Agrim Dewan for lending a hand when I need it, both in and outside complexity theory.

Abstract

A symbolic determinant under rank-one restriction is a polynomial of the form $\det(C + A_0x_0 + \cdots + A_{n-1}x_{n-1})$ where $C, A_0, \ldots, A_{n-1}$ are square matrices over some field $\mathbb{F}$ and $A_0, \ldots, A_{n-1}$ are rank-one. We ask the following problem: given black-box access to the polynomial $\det(C + A_0x_0 + \cdots + A_{n-1}x_{n-1})$, find square matrices $D, B_0, \ldots, B_{n-1}$ with $B_0, \ldots, B_{n-1}$ being rank-one such that $\det(C + A_0x_0 + \cdots + A_{n-1}x_{n-1}) = \det(D + B_0x_0 + \cdots + B_{n-1}x_{n-1})$. Since this polynomial family is known to be equivalent to *read-once determinants (RODs)*, we call this problem *Learning RODs*. A read-once determinant is a determinant whose entries are either field constants or variables, with each variable appearing at most once.

For learning RODs, we give a randomised polynomial time algorithm by connecting it to another well-known problem in linear algebra: the *Principal Minor Assignment Problem (PMAP)*. In PMAP, we are asked to construct a matrix with a given list of principal minors. We show that learning RODs is randomised polynomial time equivalent to a black-box variant of PMAP: given black-box access to $\det(A + X)$ where A is a square matrix over $\mathbb{F}$ and X is a diagonal matrix with variable entries, find a matrix B such that $\det(A+X) = \det(B+X)$. We then solve this black-box PMAP in randomised polynomial time by reducing it to PMAP for a special class of matrices that satisfy what we call *property* $\mathcal{R}$. The randomised polynomial time equivalence between learning RODs and black-box PMAP as well as the reduction from black-box PMAP to PMAP of matrices that satisfy property $\mathcal{R}$ can be derandomised in quasi-polynomial time.

Publications based on this Thesis

- Learning Read-Once Determinants and the Principal Minor Assignment Problem.

  Joint work with Abhranil Chatterjee, Sumanta Ghosh, Rohit Gurjar, Roshan Raj and Chandan Saha.

  To appear in the proceedings of the 58th Annual ACM Symposium on Theory of Computing (STOC), 2026.

Contents

Chapter 1

Introduction

1.1 Motivation and related works

Arithmetic circuits are the standard computational model in algebraic complexity theory. An arithmetic circuit is a directed acyclic graph with input nodes (nodes of in-degree zero) labelled by variables, an output node (node with out-degree zero) and internal nodes labelled either as sum or product nodes. It is easy to see that such circuits compute multivariate polynomials. The *size* of a circuit is the number of edges in it. The maximum distance between an input and an output node is its *depth*. In algebraic complexity, we analyse *classes* of polynomial families. Three important areas in algebraic complexity theory are *circuit lower bounds*, *polynomial identity testing* and *circuit reconstruction problem* or *learning problem*.

Valiant introduced the classes VP and VNP in [Val79] which can be thought of as algebraic analogues of P and NP. A central goal in the area of *circuit lower bounds* is separating VP and VNP, which amounts to finding an explicit circuit family in VNP that does not have polynomial sized circuits. It is considered necessary to separate VP and VNP to separate P/poly and NP/poly (non-uniform versions of P and NP) [Bü00].

Polynomial identity testing (PIT) is the problem of determining if a given multivariate polynomial is the zero polynomial (the polynomial with all of its coefficents being zero after any cancellations). In *white-box PIT*, the polynomial is given explicitly, usually as an arithmetic circuit. In *black-box PIT*, we are given query-access to the polynomial, where we are only allowed to query the polynomial at points of our choosing. PIT is one of the major problems that admit a randomised polynomial time algorithm but no known deterministic ones. Furthermore, efficient deterministic PIT algorithms can result in proving lower bounds for circuits [HS80, KI03, Agr05], so PIT becomes an important problem in algebraic complexity theory.

In the *circuit reconstruction problem* or *learning problem*, we are given black-box access to a

circuit from a circuit class $\mathcal{C}$ that computes a polynomial f, and are required to output a circuit computing f of similar complexity (this is similar to exact learning of Boolean functions). If the output circuit belongs to $\mathcal{C}$, the learning algorithm is said to be *proper*. In the end, we use an efficient black-box PIT algorithm to check if we have reconstructed correctly. It is easy to see that deterministic reconstruction is harder than identity testing. Efficient learning algorithms also imply circuit lower bounds [FK09, Vol16]. Due to this, proper learning for general circuit classes is considered to be a hard problem.

As a result, the learning problem is often considered for restricted models of computation. This can be size or depth constraints, as in constant depth circuits, or a read constraint where we restrict the number of times a variable appears as input. Only a handful of such classes are known to admit efficient proper learning algorithms. These include depth-two circuits (sparse polynomials) [BT88, KS01], read-once formulae (ROFs) [HH91, BHH92, SV14, MV18], and read-once oblivious algebraic branching programs (ROABPs) with known variable ordering [BBB+00, KS06]. *Algebraic Branching Programs (ABPs)*[1], analogous to Boolean branching programs, are a powerful model of computation. Polynomial-sized ABPs, which form the complexity class VBP, capture important polynomials like the determinant polynomials and the iterated matrix multiplication polynomials. Owing to a reduction from ABPs to the determinant polynomial [Val79], we can assume that an ABP computes a polynomial of the form $\det(C + A_0x_0 + \cdots + A_{n-1}x_{n-1})$ where $C, A_0, \ldots, A_{n-1}$ are $r \times r$ matrices over a field $\mathbb{F}$ and $x_0, \ldots, x_{n-1}$ are input variables. VBP is a subset of VP, therefore separating VP from VNP would require separating VBP from VNP, which is also a major open problem in algebraic complexity theory.

1.1.1 Read-Once determinants

One important subclass of ABPs is the *symbolic determinants under rank-one restriction*: this is when the matrices $A_0, \ldots, A_{n-1}$ in $\det(C + A_0x_0 + \cdots + A_{n-1}x_{n-1})$ are rank-one (no such rank restriction is imposed on C). This class of polynomials has been studied in the context of matroid problems, maximum rank matrix completion, and polynomial identity testing [Edm67, Edm68, Edm79, Lov89, Mur93, Gee99, IKS10, IKQS15, GT17]. It is known that polynomials in this class can be expressed as *read-once determinants* of small size [GT17, Lemma 4.3].

A *read-once determinant (ROD)* is the determinant of a matrix whose entries are either field constants from a field $\mathbb{F}$ or variables, with each variable appearing at most once. It is easy to

[1] An ABP can be defined as a directed acyclic graph with a single source and a single sink where the edges are labelled by affine forms in the variables. The weight of a path from source to sink is the product of the affine forms along its edges. The polynomial computed by the ABP is the sum of the weights of all paths.

see that an ROD is a symbolic determinant under rank-once restriction, thus making symbolic determinants under rank-once restriction and RODs essentially equivalent. When compared to ROABPs[1], another 'read-once' restriction on ABPs, it is seen that both ROABPs and RODs capture ROFs [Val79], but unlike ROABPs, RODs are not universal. For example, RODs cannot compute the elementary symmetric polynomial or the permanent polynomial [AJ15]. On the other hand, the determinant polynomial family, which is VBP-complete [Val79] and requires exponential sized ROABPs [Nis91], is easily seen to be computable by small RODs. This makes RODs a rather interesting class of polynomials from a complexity theoretic standpoint.

Thus, a natural question to ask is if this model admits an efficient proper learning algorithm. We shall call this learning problem *learning RODs*.

Problem 1.1 (Learning RODs) *Given black-box access to a polynomial* $f(x_0, \ldots, x_{n-1}) = \det(C + A_0x_0 + \cdots + A_{n-1}x_{n-1})$*, where* $C, A_0, \ldots, A_{n-1}$ *are unknown* $r \times r$ *matrices over a field* $\mathbb{F}$ *and* $A_0, \ldots, A_{n-1}$ *are rank-one, is there an efficient algorithm to output* $r' \times r'$ *matrices* $D, B_0, \ldots, B_{n-1}$ *such that* $f(x_0, \ldots, x_{n-1}) = \det(D + B_0x_0 + \cdots + B_{n-1}x_{n-1})$*?*

Since the determinant family requires exponential sized ROABPs but admits small RODs, as mentioned above, improper learning of RODs via the learning algorithm for ROABPs is not feasible. This makes Problem 1.1 interesting in its own right. Another motivation for solving it comes from its connection to the *principal minor assingment problem*.

1.1.2 Principal minor assignment problem

The *principal minor assignment problem (PMAP)* is a well-known problem in Linear Algebra where given a natural number n and $2^n - 1$ numbers $(p_S)_{S \subseteq \{0,1,\ldots,n-1\}, S \neq \emptyset}$, the goal is to find an $n \times n$ matrix A such that the principal minor of A corresponding to the rows and columns indexed by S equals p_S [HS02, GT06]. Often, it is typically assumed that the given principal minors are consistent – that is, there exists a matrix whose principal minors match the list of numbers. In this case, the principal minors are assumed to be given in the form of an oracle, and one wants to minimise the number of queries to the oracle. Since there are $2^n - 1$ principal minors and the matrix has n^2 entries, this becomes an over-determined problem when $n \geq 5$. So, assuming consistency of principal minors, a question arises if the matrix can be reconstructed with, say, number of queries to the principal minors being polynomial in n. Such efficient algorithms for PMAP are known for special classes of matrices, like symmetric matrices [RKT15], magnitude

[1] An ROABP is a layered ABP with $n+1$ layers numbered 0 to n, the source being layer 0 and the sink being layer n. All edges from layer i to layer $i+1$ are labelled by univariate polynomials in $x_{\sigma(i)}$ where σ is a permutation on $\{0, 1, \ldots, n-1\}$.

symmetric matrices [Bru18, BU24], and a certain subset of matrices with non-zero off-diagonal entries [GT06]. However, an efficient algorithm for general matrices remains elusive.

PMAP becomes especially interesting in the context of learning discrete *determinantal point processes*. A *determinantal point process (DPP)* is a kind of probability distribution defined on subsets of a ground set $\mathcal{Y}$. In the discrete case, $\mathcal{Y}$ is finite and the inclusion probabilities are defined using the principal minors of a $|\mathcal{Y}| \times |\mathcal{Y}|$ positive semi-definite matrix K called the kernel matrix. Specifically, for a random subset $Y \subseteq \mathcal{Y}$, $\Pr[J \subseteq Y]$ equals the principal minor of K corresponding to the rows and columns indexed by J. DPPs see use in random matrix theory and machine learning to model systems where diversity of the predicted sets are important [KT12]. Naturally, given the context of machine learning, the question of learning DPPs become interesting. Efficient algorithms for PMAP can lead to efficient learning algorithms for DPPs, as was the case for symmetric DPPs [UBMR17].

Consider the n-variate polynomial $f(x_0, \ldots, x_{n-1}) = \det(A + X)$ where A is an $n \times n$ matrix and X is a diagonal matrix of the variables $x_0, \ldots, x_{n-1}$. It is easy to see that f is multilinear and the coefficients of the monomials in f are the principal minors of A. This motivates what we shall call the *black-box principal minor assignment problem (black-box PMAP)*: where we need to reconstruct a matrix with the same principal minors as A given black-box access to f.

Problem 1.2 (Black-box PMAP) *Given black-box access to a polynomial $f(x_0, \ldots, x_{n-1}) = \det(A + X)$, where A is an unknown $n \times n$ matrix over a field $\mathbb{F}$ and X is a diagonal matrix of variables, is there an efficient algorithm to output an $n \times n$ matrix B such that $f(x_0, \ldots, x_{n-1}) = \det(B + X)$?*

Note that $\det(A + X) = \det(B + X)$ is equivalent to saying that A and B have the same principal minors.

1.2 Our Results

We give randomised polynomial time algorithms to solve Problem 1.1 and Problem 1.2. The algorithms can be derandomised in quasi-polynomial time. Specifically, we show a randomised polynomial time equivalence between the two problems and solve the latter in randomised polynomial time. Stating our results formally:

Theorem 1.1 *Let $\mathbb{F}$ be a field where square roots can be computed in polynomial time. Let $n \in \mathbb{N}$ and $|\mathbb{F}| > n^6$. Then*

1. *Given black-box access to $f(x_0, \ldots, x_{n-1}) = \det(C + A_0 x_0 + \ldots A_{n-1} x_{n-1})$, where the $r \times r$ matrices $C, A_0, \ldots, A_{n-1}$ are unknown (r is also unknown) and $A_0, \ldots, A_{n-1}$ are rank-one, there is a randomised algorithm that runs in time polynomial in n that can recover*

(with high probability) $n \times n$ matrices $D, B_0, \ldots, B_{n-1}$ with $B_0, \ldots, B_{n-1}$ being rank-one such that $f(x_0, \ldots, x_{n-1}) = \det(D + B_0 x_0 + \cdots + B_{n-1} x_{n-1})$. This algorithm can be derandomised in quasi-polynomial time.

2. *Let X be an $n \times n$ diagonal matrix with variable entries $x_0, \ldots, x_{n-1}$. Given black-box access to $f(x_0, \ldots, x_{n-1}) = \det(A + X)$, where A is an unknown $n \times n$ matrix, there is a randomised algorithm that runs in time polynomial in n that can recover (with high probability) an $n \times n$ matrix B such that $f(x_0, \ldots, x_{n-1}) = \det(X + B)$. This algorithm can be derandomised in quasi-polynomial time.*

The algorithm for Problem 1.2 automatically results in a $2^{O(n)}$ algorithm for PMAP: by querying all the principal minors, one can easily construct the polynomial $\det(A + X)$. In the scenario where the principal minors provided as input need not be consistent, this would be near optimal as one needs to query all principal minors to affirm that they are consistent. If the principal minors are provided as a list, then this algorithm would be almost linear in the input size.

The size constraint on the field arises from the need to apply the Polynomial Identity Lemma (see Section 4.1). If one can work with field extensions, this requirement can be dropped. The requirement of an efficient square root finding algorithm comes from the need to solve quadratic equations. This assumption is justified for rational numbers as there exist deterministic square root finding algorithms for it. For finite fields, there exist randomised polynomial time algorithms that can be derandomised in polynomial time if a quadratic non-residue is known (which is justified assuming the Generalised Riemann Hypothesis [Bac90]).

The results come from exploiting the properties of a certain class of matrices satisfying what we call *property* $\mathcal{R}$. An $n \times n$ matrix with $n \geq 4$ is said to satisfy property $\mathcal{R}$ if all its off-diagonal entries are non-zero and, for all distinct indices i, j, k, l, if the submatrix formed by the rows indexed by i, j and columns indexed by k, l is rank-one, then there exists a set of indices S with $i, j \in S$ and $k, l \notin S$ such that the submatrix of A with rows indexed by S and columns indexed by the complement of S is rank-one.

Theorem 1.1 is solved by reducing them to PMAP for matrices with property $\mathcal{R}$. We have the following:

Theorem 1.2 *Let $\mathbb{F}$ be a field such that there is a deterministic polynomial time square root finding algorithm over $\mathbb{F}$. Let A be an unknown $n \times n$ matrix over $\mathbb{F}$ satisfying property $\mathcal{R}$. Then there is a deterministic polynomial time algorithm that, given the principal minors of A of orders at most 4, outputs an $n \times n$ matrix B that is principal minor equivalent to A.*

Note that in a random matrix A over a large enough field, the off-diagonal entries will be non-zero and there will not be a 2×2 submatrix that is rank-one with high probability. Thus, property $\mathcal{R}$ is a genericity condition satisfied by random matrices. While [GT06] also solves PMAP for matrices satisfying a genericity condition they call 'off-diagonal full', it does not seem to correspond to any structural property of a matrix that guarantees a solution for PMAP. Furthermore, off-diagonal full matrices cannot have cuts (see Definition 2.6) while matrices satisfying property $\mathcal{R}$ can. We also demonstrate a reduction from black-box PMAP to PMAP for matrices with property $\mathcal{R}$ (see Chapter 4). It is not clear if such a reduction exists for off-diagonal full matrices.

1.3 Organisation

Chapter 2 establishes definitions and notations that we shall use. Chapter 3 shows a randomised polynomial time equivalence (also, deterministic quasi-polynomial time equivalence) between Problem 1.1 and Problem 1.2. Chapter 4 reduces Problem 1.2 to PMAP of matrices satisfying property $\mathcal{R}$. Chapter 5 lays out a cut detection algorithm given oracle access to 4×4 and smaller principal minors of a matrix satisfying property $\mathcal{R}$. Chapter 6 finally establishes the algorithm that proves Theorem 1.2.

Lemma 2.8, Lemma 5.7 and Lemma 6.3 from the paper [ACG+26] are referred to without proofs; these were proven by Chatterjee, Ghosh, Gurjar and Raj before the author and Chandan Saha began collaborating with them.

Chapter 2

Preliminaries

2.1 Basic notations

We denote by $[0\,..\,n)$ the set $\{0, 1, 2, \ldots, n-1\}$. The $n \times n$ diagonal matrix with entries $z_0, \ldots, z_{n-1}$ is denoted as $\mathrm{diag}(z_0, \ldots, z_{n-1})$. We denote the set of $n \times n$ matrices over the field $\mathbb{F}$ as $\mathbb{F}^{n\times n}$. We denote by I_n the $n \times n$ identity matrix, and by $O_{m,n}$ the $m \times n$ zero matrix.

Let $A \in \mathbb{F}^{n\times n}$. We index the rows and columns of A by the set $[0\,..\,n)$, that is, they are zero-indexed. Let $S, T \subseteq [0\,..\,n)$. Then, $A[S, T]$ represents the sub-matrix of A with rows indexed by S and columns by T. We denote the sub-matrix $A[S, S]$ as $A[S]$. We denote by $\bar{S}$ the set complement of S, that is, $[0\,..\,n) \setminus S$. By $\mathrm{adj}\, A$, we refer to the adjoint of A.

2.2 Read-once determinants

Definition 2.1 *A symbolic determinant under rank-one restriction is a polynomial of the form* $f(x_0, \ldots, x_{n-1}) = \det(C + A_0x_0 + \ldots, A_{n-1}x_{n-1})$ *where* $C, A_0, \ldots, A_{n-1} \in \mathbb{F}^{r\times r}$ *and* $A_0, \ldots, A_{n-1}$ *are rank 1.*

If for $i \in [0\,..\,n)$, $A_i = u_i v_i^T$, then it is not hard to see that we can also express f above as $f(x_0, \ldots, x_{n-1}) = \det(UXV^T + C)$ where U and V are $r \times n$ matrices whose columns are u_i and v_i respectively, and $X = \mathrm{diag}(x_0, \ldots, x_{n-1})$.

We can show that symbolic determinants under rank-one restriction are equivalent to *read-once determinants*.

Definition 2.2 *A read-once determinant (ROD) is a polynomial expressed as a determinant of a matrix whose entries are either field constants or variables, with each variable appearing at most once.*

Lemma 2.1 ([GT17, Lemma 4.3]) *Let* $f(x_0, \ldots, x_{n-1}) = \det(UXV^T + C)$ *where* $U, V \in \mathbb{F}^{r\times n}$, $C \in \mathbb{F}^{r\times r}$ *and* $X = \mathrm{diag}(x_0, \ldots, x_{n-1})$. *Then,*

$$f(x_0, \ldots, x_{n-1}) = \det \begin{bmatrix} I_n & X & O_{n,r} \\ O_{n,n} & I_n & V^T \\ U & O_{r,n} & C \end{bmatrix}.$$

We demonstrate that, given, black-box access to $\det(UXV^T + C)$, we can assume $r \leq n$.

Lemma 2.2 *Let* $f(x_0, \ldots, x_{n-1}) = \det(UXV^T + C)$ *be a non-zero polynomial where* $C \in \mathbb{F}^{r\times r}, U, V \in \mathbb{F}^{r\times n}$ *and* $X = \mathrm{diag}(x_0, \ldots, x_{n-1})$. *Then we can assume, without loss of generality, that* $r \leq n$.

Proof: If $r \leq n$, we are done, so assume $r > n$. Let $k \leq n$ be the rank of the matrix U (The choice of U is arbitrary; we can similarly work with V). By Lemma 2.1,

$$f(x_0, \ldots, x_{n-1}) = \det \begin{bmatrix} I_n & X & O_{n,r} \\ O_{n,n} & I_n & V^T \\ U & O_{r,n} & C \end{bmatrix}.$$

Using row transformations on U, we have

$$f(x_0, \ldots, x_{n-1}) = \det \begin{bmatrix} I_n & X & 0_{n,r} \\ 0_{n,n} & I_n & V^T \\ \hat{U} & 0_{k,n} & C_1 \\ 0_{r-k,n} & 0_{r-k,n} & C_2 \end{bmatrix},$$

where $\hat{U} \in \mathbb{F}^{k\times n}$, $C_1 \in \mathbb{F}^{k\times r}$, $C_2 \in \mathbb{F}^{(r-k)\times r}$ and $\hat{U}$ is full-rank. For f to be non-zero, C_2 must be full rank. We can perform row and column transformations on C_2 and absorb any resulting scalars into a column of $\hat{C}$ to get

$$f(x_0, \ldots, x_{n-1}) = \det \begin{bmatrix} I_n & X & 0_{n,k} & 0_{n,r-k} \\ 0_{n,n} & I_n & \hat{V}^T & \hat{V}_1^T \\ \hat{U} & 0_{k,n} & \hat{C} & C_3 \\ 0_{r-k,n} & 0_{r-k,n} & 0_{r-k,k} & I_{r-k} \end{bmatrix}$$

$$= \det \begin{bmatrix} I_n & X & 0_{n,k} \\ 0_{n,n} & I_n & \hat{V}^T \\ \hat{U} & 0_{k,n} & \hat{C} \end{bmatrix}$$
$$= \det(\hat{U}X\hat{V}^T + \hat{C}),$$

where $\hat{V} \in \mathbb{F}^{k\times n}$ and $\hat{C} \in \mathbb{F}^{k\times k}$. □

The following lemma is useful in converting a read-once determinant back into a symbolic determinant under rank-one restriction.

Lemma 2.3 *Let $A \in \mathbb{F}^{r\times(n-r)}, B \in \mathbb{F}^{r\times r}$ and $C \in \mathbb{F}^{(n-r)\times r}$. Then,*

$$\det \begin{bmatrix} A & B \\ I_{n-r} & C \end{bmatrix} = (-1)^{r(n-r)} \det \left(\begin{bmatrix} A & B \end{bmatrix} \begin{bmatrix} -C \\ I_r \end{bmatrix} \right).$$

Proof: Using block Gaussian elimination,

$$\det \begin{bmatrix} A & B \\ I_{n-r} & C \end{bmatrix} = \det \begin{bmatrix} A & B - AC \\ I_{n-r} & O_{n-r,r} \end{bmatrix}$$
$$= (-1)^{r(n-r)} \det(B - AC)$$
$$= (-1)^{r(n-r)} \det \left(\begin{bmatrix} A & B \end{bmatrix} \begin{bmatrix} -C \\ I_r \end{bmatrix} \right).$$

□

2.3 Principal minor equivalence and cuts in matrices

Definition 2.3 (Reducible and irreducible matrix, irreducible blocks) *A matrix $A \in \mathbb{F}^{n\times n}$ is* reducible *if there exists a permutation matrix $P \in \mathbb{F}^{n\times n}$ such that PAP^T is a block upper triangular matrix with at least two blocks along the diagonal. The blocks along the diagonal of PAP^T are the* irreducible blocks *of A. A matrix that is not reducible is* irreducible.

Equivalently, let G be the directed graph of n vertices with an edge from i to j present if and only if $A[i,j] \neq 0$. Then A is reducible if and only if G is not strongly connected. The irreducible blocks would then correspond to the strongly connected components of G.

Definition 2.4 (Diagonal similarity and diagonal equivalence) *Two matrices A and $B \in \mathbb{F}^{n\times n}$ are said to be* diagonally similar *(denoted as $A \overset{DS}{=} B$) if there exists an invertible diagonal matrix $D \in \mathbb{F}^{n\times n}$ such that $A = D^{-1}BD$. If $A \overset{DS}{=} B$ or $A^T \overset{DS}{=} B$, then A and B are said to be* diagonally equivalent *(denoted as $A \overset{DE}{=} B$).*

Definition 2.5 (Principal minor equivalence) *Two matrices $A, B \in \mathbb{F}^{n\times n}$ are said to be* principal minor equivalent *(denoted as $A \overset{PME}{=} B$) if for all $S \subseteq [0\,..\,n)$, $\det(A[S]) = \det(B[S])$.*

Definition 2.6 (Cut in a matrix) *A set $S \subset [0\,..\,n)$ with $2 \le |S| \le n-2$ is a* cut *in an irreducible matrix $A \in \mathbb{F}^{n\times n}$ if both* $\operatorname{rank} A[S, \bar{S}]$ *and* $\operatorname{rank} A[\bar{S}, S]$ *are* 1.

The following lemma follows from [Ahm23, Corollary 5.4].

Lemma 2.4 *Let A be a reducible matrix. Suppose the irreducible blocks of A are indexed by $T_0, \ldots, T_{k-1}$. Then $A \overset{PME}{=} B$ if and only if:*

- *The irreducible blocks of B are also indexed by the same sets $T_0, \ldots, T_{k-1}$.*
- *For each $i \in [0\,..\,k)$, $A[T_i] \overset{PME}{=} B[T_i]$.*

It is easy to see that $A \overset{\text{DE}}{=} B \implies A \overset{\text{PME}}{=} B$. [HL84, Theorem 3] and [Loe86, Theorem 1] prove a partial converse:

Lemma 2.5 *Let $A, B \in \mathbb{F}^{n\times n}$. Suppose A is irreducible, and $A \overset{PME}{=} B$. Then,*

- *If $n = 2$ or 3, then $A \overset{DE}{=} B$.*
- *Else if $n \ge 4$ and A has no cut, then $A \overset{DE}{=} B$.*

Motivated by [Ahm23, Lemma 4.5], [CGGR25] introduces *cut-transpose* as a principal minor preserving operation.

Definition 2.7 (Cut-transpose) *Let S be a cut in an irreducible matrix $A \in \mathbb{F}^{n\times n}$. Let*

$$A = \begin{array}{c} \\ S \\ \bar{S} \end{array} \begin{array}{c} \begin{array}{cc} S & \bar{S} \end{array} \\ \begin{bmatrix} A[S] & pq^T \\ uv^T & A[\bar{S}] \end{bmatrix} \end{array}$$

for $p, v \in \mathbb{F}^{|S|}$, $u, q \in \mathbb{F}^{|\bar{S}|}$, where q^T is the first non-zero row of $A[S, \bar{S}]$ and u is the first non-zero column of $A[\bar{S}, S]$. Then the cut-transpose *of A with respect to S is*

$$\operatorname{ct}(A, S) = \begin{array}{c} \\ S \\ \bar{S} \end{array} \begin{array}{c} \begin{array}{cc} S & \bar{S} \end{array} \\ \begin{bmatrix} A[S] & pu^T \\ qv^T & A[\bar{S}]^T \end{bmatrix} \end{array}.$$

Lemma 2.6 ([CGGR25, Lemma 2.12]) *Let S be a cut in $A \in \mathbb{F}^{n\times n}$. Then $\operatorname{ct}(A, S) \overset{PME}{=} A$.*

For 4×4 matrices, we have the following:

Lemma 2.7 ([CGGR25, Lemma 3.1]) *Let $A \in \mathbb{F}^{4\times 4}$ have non-zero off-diagonal entries. Let $B \in \mathbb{F}^{4\times 4}$ be such that $A \overset{PME}{=} B$. Then one of the following holds:*

- $A \overset{DE}{=} B$.
- *There exists a common cut in A and B. For any common cut S in A and B, $\mathrm{ct}(A, S) \overset{DE}{=} B$.*

2.4 Property $\mathcal{R}$

Motivated by the proofs of [HL84, Theorem 2] and [Loe86, Lemma 6], we introduce a property of certain matrices called property $\mathcal{R}$.

Definition 2.8 (Property $\mathcal{R}$) *A matrix $A \in \mathbb{F}^{n\times n}$ satisfies property $\mathcal{R}$ if and only if*

- *All off-diagonal entries of A are non-zero.*
- *For all $\{i, j, k, l\} \subseteq [0\,..\,n)$, if $\operatorname{rank} A[\{i, j\}, \{k, l\}] = 1$, then there exists $S \subset [0\,..\,n)$ with $\{i, j\} \subseteq S$, $\{k, l\} \subseteq \bar{S}$ such that $\operatorname{rank} A[S, \bar{S}] = 1$.*

Property $\mathcal{R}$ allows for quickly testing principal minor equivalence by checking equality of principal minors of order at most 4.

Lemma 2.8 ([ACG+26, Theorem 1.3]) *Let $A \in \mathbb{F}^{n\times n}$ satisfy property $\mathcal{R}$. Let $B \in \mathbb{F}^{n\times n}$ be such that for all $S \subseteq [0\,..\,n)$ with $|S| \leq 4$, $A[S] \overset{PME}{=} B[S]$. Then $A \overset{PME}{=} B$.*

2.5 Black-box with inverted inputs

Given black-box access to a degree d polynomial $f(x_0, \ldots, x_{n-1})$ where some of the inputs are inverted, we can evaluate it at a point $(a_0, \ldots, a_{n-1})$ when some of the coordinates are zero. Say $a_k, \ldots, a_{n-1}$ are zero. Then we see $g(t) = f(a_0, \ldots, a_{k-1}, t, \ldots, t)$ is a polynomial in t. Say $g(t) = \sum_{i\in\{0,1,\ldots,d\}} c_i t^i$ Observe that c_0 is the value we need and if the field is large enough, we can find distinct values $\beta_0, \beta_1, \ldots, \beta_d \in \mathbb{F}$. Then

$$\begin{bmatrix} g(\beta_0) \\ g(\beta_1) \\ \vdots \\ g(\beta_d) \end{bmatrix} = \begin{bmatrix} 1 & \beta_0 & \beta_0^2 & \cdots & \beta_0^d \\ 1 & \beta_1 & \beta_1^2 & \cdots & \beta_1^d \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \beta_d & \beta_d^2 & \cdots & \beta_d^d \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_d \end{bmatrix}.$$

On the right is a Vandermonde matrix that is invertible. Then by inverting the matrix, we can compute the value of c_0.

Chapter 3

Equivalence between Learning RODs and Black-Box PMAP

In this chapter, we establish a randomised polynomial time equivalence between learning RODs and black-box PMAP. We show that given black-box access to a polynomial $f(x_0, \ldots, x_{n-1}) = \det(C + A_0x_0 + \cdots + A_{n-1}x_{n-1})$ where $A_0, \ldots, A_{n-1}$ are rank-one, we can reduce the problem of learning f to learning a polynomial of the form $g(x_0, \ldots, x_{n-1}) = \det(X + B)$ where $X = \operatorname{diag}(x_0, \ldots, x_{n-1})$ and vice-versa. The first section shows a randomised polynomial time reduction from learning RODs to black-box PMAP, a reduction that can be derandomised in quasi-polynomial time. The second section shows a deterministic polynomial time reduction from black-box PMAP to learning RODs.

3.1 Reduction from learning RODs to black-box PMAP

Say $f(x_0, \ldots, x_{n-1}) = \det(UXV^T + C)$ is a non-zero polynomial where $U, V \in \mathbb{F}^{r\times n}$ and $C \in \mathbb{F}^{r\times r}$. We proceed similarly to [GT17, Theorem 4.2]. First, we homogenise the polynomial f, with the help of Section 2.5, as follows:

$$g(x_0, \ldots, x_{n-1}, y_0, \ldots, y_{n-1}) = y_0 \ldots y_{n-1} f(\frac{x_0}{y_0}, \ldots, \frac{x_{n-1}}{y_{n-1}}) = \det(Y)\det(UY^{-1}XV^T + C).$$

By Lemma 2.1,

$$g(x_0, \ldots, x_{n-1}, y_0, \ldots, y_{n-1}) = \det(Y)\det\begin{bmatrix} I_n & Y^{-1}X & O_{n,r} \\ O_{n,n} & I_n & V^T \\ U & O_{r,n} & C \end{bmatrix} = \det\begin{bmatrix} Y & X & O_{n,r} \\ O_{n,n} & I_n & V^T \\ U & O_{r,n} & C \end{bmatrix},$$

where $Y = \mathrm{diag}(y_0, \ldots, y_{n-1})$. Since f is non-zero, g is as well, which means the matrix $\begin{bmatrix} U & C \end{bmatrix}$ is full rank. Performing row transformations and permuting the columns, we get

$$g(x_0, \ldots, x_{n-1}, y_0, \ldots, y_{n-1}) = \alpha \det \begin{bmatrix} \leftarrow & L'W'P & \rightarrow \\ & I_{n+r} \quad H & \end{bmatrix},$$

where

- L' is the $n \times (2n + r)$ matrix $\begin{bmatrix} I_n & I_n & 0_{n,r} \end{bmatrix}$.
- W' is the $(2n + r) \times (2n + r)$ diagonal matrix $\begin{bmatrix} Y & 0_{n,n} & 0_{n,r} \\ 0_{n,n} & X & 0_{n,r} \\ 0_{r,n} & 0_{r,n} & 0_{r,r} \end{bmatrix}$.
- P is the $(2n+r)\times(2n+r)$ permutation matrix corresponding to the column permutations performed.
- H is the $(n + r) \times n$ matrix resulting from the transformations.
- α is the scalar resulting from the transformations.

By Lemma 2.3, $g(x_0, \ldots, x_{n-1}, y_0, \ldots, y_{n-1}) = (-1)^{n(n+r)}\alpha \det(L'W'PR'^T)$ where $R' = \begin{bmatrix} -H^T & I_n \end{bmatrix}$ is a $n \times (2n + r)$ matrix. We can drop the rows of R' and W' as well as the columns of L' and W' corresponding to the zeroes in W', and then absorb $(-1)^{n(n+r)}\alpha$ and P into the matrix R' to get $g(x_0, \ldots, x_{n-1}, y_0, \ldots, y_{n-1}) = \det(LWR^T)$, where

- L is the $n \times 2n$ matrix $\begin{bmatrix} I_n & I_n \end{bmatrix}$.
- W is the $2n \times 2n$ diagonal matrix $\begin{bmatrix} Y & 0_{n,n} \\ 0_{n,n} & X \end{bmatrix}$.
- R is the $n \times 2n$ matrix resulting from all the algebraic manipulations.

Observe that we have black-box access to g. We can now use the Isolation Lemma [MVV87] to find a monomial in g (see [NSV92, Section 4.1], [GT17, Section 2.4]. This reduction can be done in deterministic quasi-polynomial time [GT17]). By the structure of the matrix L, we see that the the monomial must be of the form $z_0 z_1 \ldots z_{n-1}$ with $z_i = x_i$ or y_i. Let $\bar{z}_i = y_i$ if $z_i = x_i$ and vice-versa. Say

$$g(x_0, \ldots, x_{n-1}, y_0, \ldots, y_{n-1}) = \det \left(\begin{bmatrix} I_n & I_n \end{bmatrix} \begin{bmatrix} Z & O_{n,n} \\ O_{n,n} & \bar{Z} \end{bmatrix} \begin{bmatrix} R_0 \\ R_1 \end{bmatrix} \right),$$

where $Z = \text{diag}(z_0, \ldots, z_{n-1}), \bar{Z} = \text{diag}(\bar{z}_0, \ldots, \bar{z}_{n-1})$; and $R_0, R_1 \in \mathbb{F}^{n \times n}$ are the matrices that comprise R^T. It is easy to see that $\det(R_0) \neq 0$ is the coefficient of $z_0 \ldots z_{n-1}$ and is known. Then,

$$\begin{aligned} g(z_0, \ldots, z_{n-1}, \bar{z}_0, \ldots, \bar{z}_{n-1}) &= \det(R_0) \det\left(\begin{bmatrix} I_n & I_n \end{bmatrix} \begin{bmatrix} Z & O_{n,n} \\ O_{n,n} & \bar{Z} \end{bmatrix} \begin{bmatrix} I_n \\ R_1 R_0^{-1} \end{bmatrix} \right) \\ &= \det(R_0) \det(Z + \bar{Z} R_1 R_0^{-1}). \end{aligned}$$

Let $\tilde{g}(z_0, \ldots, z_{n-1}, \bar{z}_0, \ldots, \bar{z}_{n-1}) = \bar{z}_0 \ldots \bar{z}_{n-1} g(z_0, \ldots, z_{n-1}, \frac{1}{\bar{z}_0}, \ldots, \frac{1}{\bar{z}_{n-1}})$. It is easy to see that

$$\begin{aligned} \tilde{g}(z_0, \ldots, z_{n-1}, \bar{z}_0, \ldots, \bar{z}_{n-1}) &= \det(R_0) \det(\bar{Z}) \det(Z + \bar{Z}^{-1} R_1 R_0^{-1}) \\ &= \det(R_0) \det(\bar{Z} Z + R_1 R_0^{-1}). \end{aligned}$$

Observation 3.1 *In a monomial of $\tilde{g}$, $\bar{z}_i$ appears if and only if z_i does. This arose out of the homogenisation of f: y_i appears in a monomial of g if and only if x_i does not.*

Multiplying $\tilde{g}$ by $\det(R_0)^{-1}$, and setting $\bar{z}_i = 1$ for all $i \in [0\,..\,n)$, we get $h(z_0, \ldots, z_{n-1}) = \det(Z + R_1 R_0^{-1})$. Suppose we learn a matrix M such that $h(z_0, \ldots, z_{n-1}) = \det(Z + M)$. By Observation 3.1, we see $\tilde{g}(z_0, \ldots, z_{n-1}, \bar{z}_0, \ldots, \bar{z}_{n-1}) = \det(R_0) \det(\bar{Z} Z + M)$. Thus

$$\begin{aligned} g(z_0, \ldots, z_{n-1}, \bar{z}_0, \ldots, \bar{z}_{n-1}) &= \det(R_0) \det(\bar{Z}) \det(\bar{Z}^{-1} Z + M) \\ &= \det(R_0) \det(Z + \bar{Z} M) \\ &= \det(R_0) \det\left(\begin{bmatrix} I_n & I_n \end{bmatrix} \begin{bmatrix} Z & O_{n,n} \\ O_{n,n} & \bar{Z} \end{bmatrix} \begin{bmatrix} I_n \\ M \end{bmatrix} \right) \\ &= \det\left(\begin{bmatrix} \det(R_0) & O_{1,n-1} & \det(R_0) & O_{1,n-1} \\ O_{n-1,1} & I_{n-1} & O_{n-1,1} & I_{n-1} \end{bmatrix} \begin{bmatrix} Z & O_{n,n} \\ O_{n,n} & \bar{Z} \end{bmatrix} \begin{bmatrix} I_n \\ M \end{bmatrix} \right). \end{aligned}$$

Let l_{x_i} be the column of $\begin{bmatrix} \det(R_0) & O_{1,n-1} & \det(R_0) & O_{1,n-1} \\ O_{n-1,1} & I_{n-1} & O_{n-1,1} & I_{n-1} \end{bmatrix}$ corresponding to x_i and let r_{x_i} be the row of $\begin{bmatrix} I_n \\ M \end{bmatrix}$ corresponding to x_i. Similarly define l_{y_i} and r_{y_i}. We see that the polynomial $g(x_0, \ldots, x_{n-1}, y_0, \ldots, y_{n-1}) = \det(\sum_{i \in [0\,..\,n)} (l_{x_i} r_{x_i} x_i + l_{y_i} r_{y_i} y_i))$. We can now set the y_i variables to 1 and simplify to see that $f(x_0, \ldots, x_{n-1}) = \det(D + B_0 x_0 + \cdots + B_{n-1} x_{n-1})$ where $D = \sum_{i \in [0\,..\,n)} (l_{y_i} r_{y_i})$ and $B_i = l_{x_i} r_{x_i}$.

Algorithm 1 Reduction of ROD Learning to Black-box PMAP

Input: Black-box access to $f = \det(C + \sum_{i\in[0\,..\,n)} A_i x_i)$ for some matrices $C, A_0, \ldots, A_{n-1} \in \mathbb{F}^{r\times r}$ and $A_0, \ldots, A_{n-1}$ are rank one.

Output: Matrices $D, B_0, \ldots, B_{n-1} \in \mathbb{F}^{n\times n}$ with $B_0, \ldots, B_{n-1}$ being rank-one such that $f = \det(D + \sum_{i\in[0\,..\,n)} B_i x_i)$.

Assumption: Oracle access to learning algorithm for $\det(M + Z)$ where $Z = \operatorname{diag}(z_1, \ldots, z_n)$.

1: Homogenise f to get $g(x_0, \ldots, x_{n-1}, y_0, \ldots, y_{n-1}) = y_0 \ldots y_{n-1} f(\frac{x_0}{y_0}, \ldots, \frac{x_{n-1}}{y_{n-1}})$.

2: Use the Isolation Lemma to isolate a monomial of g. Let this monomial be $z_0, \ldots, z_{n-1}$ with coefficient γ.

3: $h(z_0, \ldots, z_{n-1}) = \frac{1}{\gamma} g(z_0, \ldots, z_{n-1}, 1, \ldots, 1)$ will be the input to the black-box principal minor assignment problem.

4: Suppose M is the output of the black-box principal minor assignment problem. Then $g = \gamma \det\left(\begin{bmatrix} I_n & I_n \end{bmatrix} \begin{bmatrix} Z & 0_{n,n} \\ 0_{n,n} & \bar{Z} \end{bmatrix} \begin{bmatrix} I_n \\ M \end{bmatrix}\right)$.

5: Set the y_i variables to 1 and simplify to recover $f = \det(D + \sum_{i\in[0\,..\,n)} B_i x_i)$. Output $D, B_0, \ldots, B_{n-1}$.

3.2 Reduction from black-box PMAP to learning RODs

Say we have $f(x_0, \ldots, x_{n-1}) = \det(X + A)$, $X = \operatorname{diag}(x_0, \ldots, x_{n-1})$, which we wish to learn via reduction to learning RODs. Note that $\det(X + A)$ is by itself an ROD which we can learn. Say $f(x_0, \ldots, x_{n-1}) = \det(UXV^T + C)$ where $U, V \in \mathbb{F}^{r\times n}$ and $C \in \mathbb{F}^{r\times r}$. By Lemma 2.2, we can assume $r \leq n$. Then by Lemma 2.1,

$$f(x_0, \ldots, x_{n-1}) = \begin{bmatrix} I_n & X & O_{n,r} \\ O_{n,n} & I_n & V^T \\ U & O_{r,n} & C \end{bmatrix}.$$

The coefficient of $x_0 \ldots x_{n-1}$ in f is $(-1)^n \det \begin{bmatrix} O_{n,n} & V^T \\ U & C \end{bmatrix} = 1$. Thus, we must have $r = n$ and $\det(U)\det(V) = 1$, making U and V invertible. Then

$$f(x_0, \ldots, x_{n-1}) = \det(U^{-1})\det(UXV^T + C)\det((V^T)^{-1}) = \det(X + U^{-1}C(V^T)^{-1}).$$

Chapter 4

Reduction from Black-Box PMAP to PMAP with Property $\mathcal{R}$

In this chapter, we demonstrate how Black-box PMAP reduces to PMAP of matrices with property $\mathcal{R}$. We first demonstrate that for a matrix A and a diagonal matrix D of randomly chosen entries, the irreducible blocks of $(A+D)^{-1}$ satisfy property $\mathcal{R}$ with high probability. Then, we show how to get black-box access to $\det((A+D)^{-1}+X)$ from black-box access to $\det(A+X)$, and how to find the irreducible blocks of $(A+D)^{-1}$. Finally, given a matrix $C \overset{\text{PME}}{=} (A+D)^{-1}$, we recover a matrix principal minor equivalent to A. We finally describe how to derandomise the reduction in quasi-polynomial time.

4.1 Irreducible blocks of $(A+D)^{-1}$ satisfy property $\mathcal{R}$

We demonstrate that for an arbitrary matrix A, the irreducible blocks of $(A+D)^{-1}$ satisfy property $\mathcal{R}$ with high probability, where D is a diagonal matrix of random field constants. We first state the Jacobi identities relating the minors of a matrix with the minors of its inverse (see [Gan60, Page 21]) which we shall use in this section:

Lemma 4.1 *Let $A \in \mathbb{F}^{n\times n}$ be an invertible matrix. Then for all $S,T \subseteq [0\,..\,n), |S| = |T|$, $\det(A[S,T]) = (-1)^r \det(A)\det(A^{-1}[\bar{T},\bar{S}])$ where $r = (\sum_{i\in S} i) + (\sum_{i\in T} i)$.*

We shall also require the following results:

Lemma 4.2 ([HL84, Theorem 1]) *Let $A \in \mathbb{F}^{n\times n}$ be an irreducible matrix and let $Y = \operatorname{diag}(y_0,\ldots,y_{n-1})$ with $y_0,\ldots,y_{n-1}$ being algebraically independent elements. Then all the entries of $(A+Y)^{-1}$ are non-zero.*

Lemma 4.3 ([HL84, Theorem 2]) *Let $A \in \mathbb{F}^{n\times n}$ and let $Y = \mathrm{diag}(0, 0, y_2, \ldots, y_{n-1})$ with $y_2, \ldots, y_{n-1}$ being algebraically independent elements. If $\det(A + Y) = 0$, then there exists a partition $S \sqcup T$ of $[2\,..\,n)$ such that $\mathrm{rank}\, A[\{0, 1\} \cup S, \{0, 1\} \cup T] \leq 1$.*

Lemma 4.4 ([CGGR25, Claim A.1]) *Let $A \in \mathbb{F}^{n\times n}$ be an irreducible matrix and let $S \subseteq [0\,..\,n)$ be such that $2 \leq |S| \leq n - 2$ and $\mathrm{rank}\, A[S, \bar{S}] = 1$. Then $\mathrm{rank}\, A^{-1}[S, \bar{S}] = 1$.*

We now show the following:

Lemma 4.5 *Let $Y = \mathrm{diag}(y_0, \ldots, y_{n-1})$ with $y_0, \ldots, y_{n-1}$ being algebraically independent elements. Then for an $n \times n$ irreducible matrix A, $(A + Y)^{-1}$ satisfies property $\mathcal{R}$.*

Proof: From Lemma 4.2, all the off-diagonal entries of $(A + Y)^{-1}$ are non-zero. Now, let $S_0, S_1 \subseteq [0\,..\,n)$ with $|S_0| = |S_1| = 2$ and $S_0 \cap S_1 = \emptyset$. Suppose $(A+Y)^{-1}[S_0, S_1]$ is rank 1, that is, $\det((A + Y)^{-1}[S_0, S_1]) = 0$. By Lemma 4.1, we have $\det((A + Y)[\bar{S_1}, \bar{S_0}]) = 0$. Then, by Lemma 4.3, there exists a partition $T_0 \sqcup T_1$ of $[0\,..\,n)\setminus(S_0 \cup S_1)$ such that $\mathrm{rank}\, A[S_0 \cup T_0, S_1 \cup T_1] = 1$ and as a result, $\mathrm{rank}(A + Y)[S_0 \cup T_0, S_1 \cup T_1] = 1$ (Note the rank cannot be zero, otherwise A becomes reducible). Thus, by Lemma 4.4, $\mathrm{rank}(A + Y)^{-1}[S_0 \cup T_0, S_1 \cup T_1] = 1$. □

Observation 4.1 *Let $D \in \mathbb{F}^{n\times n}$ be a diagonal matrix such that*

1. *$(A + D)$ is invertible.*
2. *Off-diagonal entries of $(A + D)^{-1}$ are non-zero.*
3. *For all $S_0, S_1 \subseteq [0\,..\,n)$ with $|S_0| = |S_1| = 2$, $S_0 \cap S_1 = \emptyset$, $\det((A + D)^{-1}[S_0, S_1]) = 0 \implies \det((A + Y)^{-1}[S_0, S_1]) = 0$.*

then the matrix $(A + D)^{-1}$ satisfies property $\mathcal{R}$.

Proof: Since off-diagonal entries of both $(A+D)^{-1}$ and $(A+Y)^{-1}$ are non-zero, and $(A+Y)^{-1}$ satisfies property $\mathcal{R}$, we have

$$\begin{aligned}
&\mathrm{rank}(A + D)^{-1}[S_0, S_1] = 1 \implies \det((A + D)^{-1}[S_0, S_1]) = 0 \\
\implies &\det((A + Y)^{-1}[S_0, S_1]) = 0 \implies \mathrm{rank}(A + Y)^{-1}[S_0, S_1] = 1 \\
\implies &\mathrm{rank}(A + Y)^{-1}[T_0, T_1] = 1 \implies \mathrm{rank}(A + D)^{-1}[T_0, T_1] = 1
\end{aligned}$$

for some partition $T_0 \sqcup T_1$ of $[0\,..\,n)$ with $S_0 \subseteq T_0, S_1 \subseteq T_1$. Thus, $(A + D)^{-1}$ satisfies property $\mathcal{R}$. □

The above observation implies the following:

Observation 4.2 *Let $A \in \mathbb{F}^{n\times n}$ be an irreducible matrix. Consider the following polynomials in $\mathbb{F}[y_0, \ldots, y_{n-1}]$:*

1. $\det(A + Y)$.
2. *For all $\{i, j\} \subset [0\,..\,n)$,* $\operatorname{adj}(A + Y)[i, j] = (-1)^{i+j} \det((A + Y)[[0\,..\,n) \setminus \{j\}, [0\,..\,n) \setminus \{i\}])$.
3. *For all $S_0, S_1 \subseteq [0\,..\,n)$ with $|S_0| = |S_1| = 2$, $S_0 \cap S_1 = \emptyset$,* $\det(\operatorname{adj}(A + Y)[S_0, S_1])$.

If there exists an assignment of values $d_0, \ldots, d_{n-1}$ to $y_0, \ldots, y_{n-1}$ such that all the polynomials above are simultaneously non-zero, then $(A + D)^{-1}$ with $D = \operatorname{diag}(d_0, \ldots, d_{n-1})$ would satisfy property $\mathcal{R}$.

If $|\mathbb{F}| > n^6$, then by the Polynomial Identity Lemma [DL78, Sch80, Zip79], a random assignment would leave all of the polynomials above non-zero with high probability. Hence, for a random diagonal matrix D, $(A + D)^{-1}$ would satisfy property $\mathcal{R}$ with high probability.

Now if A is reducible, then the argument above would apply to the irreducible blocks of A (which are the same as that of $(A + D)^{-1}$). This is easy to see if we assume without loss of generality that A is a block-diagonal matrix.

4.2 Reduction to PMAP with property $\mathcal{R}$

4.2.1 Obtaining black-box access to $\det(X + (A + D)^{-1})$

Given black-box access to $\det(A + X)$ and a diagonal matrix D of random field entries, where $X = \operatorname{diag}(x_0, \ldots, x_{n-1})$, we have $\det(A+D)^{-1}$ (since D is random, $\det(A+D)$ is non-zero with high probability) as well as black-box access to

$$\det(A + D)^{-1} \det(A + D + X) = \det(I_n + (A + D)^{-1}X).$$

Then by Section 2.5, we can have black-box access to

$$\det(I_n + (A + D)^{-1}X^{-1}) \det(X) = \det(X + (A + D)^{-1}),$$

which is what we need.

4.2.2 Finding the irreducible blocks

We first see that via interpolation, we can query the principal minors of $(A+D)^{-1}$ with black-box access to $\det((A + D)^{-1} + X)$. We make the following observation

Observation 4.3 *Let the irreducible blocks of a matrix A have non-zero off-diagonal entries. Then, two indices i, j belong to the same irreducible block if and only if $A[i,j]A[j,i] \neq 0$, that is, $A[\{i,j\}]$ is irreducible.*

We can check if a two-by-two submatrix is irreducible using the following observation.

Observation 4.4 $A[i,j]A[j,i] = \det(A[\{i,j\}]) - A[i]A[j]$.

Since by Section 4.1, the irreducible blocks of $(A+D)^{-1}$ satisfy property $\mathcal{R}$ with high probability, we can query one-by-one and two-by-two principal minors to check which indices belong to which irreducible blocks. Thus, we can find the indices to the irreducible blocks of $\det((A+D)^{-1}+X)$. Say the indices are $T_0 \sqcup T_1 \sqcup \cdots \sqcup T_{k-1} = [0\,..\,n)$. Since the principal minors of the irreducible blocks are the same as that of $(A+D)^{-1}$, we have black-box access to $\det((A+D)^{-1}[T_i] + X[T_i])$ for each $i \in [0\,..\,k)$, which we learn separately.

4.2.3 Recovering a matrix PME to the original matrix

Say we have $C_i \overset{\text{PME}}{=} (A+D)^{-1}[T_i]$ for each irreducible block $(A+D)^{-1}[T_i]$ of $(A+D)^{-1}$. Since we know the indices in T_i, we can put together the C_is to get a block diagonal matrix C with $C[T_i] = C_i$. Then by Lemma 2.4, $C \overset{\text{PME}}{=} (A+D)^{-1}$. Then $\det(C+X) = \det((A+D)^{-1}+X)$, which means

$$\begin{aligned}\det(C^{-1})\det(C+X) &= \det(I_n + C^{-1}X)\\ = \det(A+D)\det((A+D)^{-1}+X) &= \det(I_n + (A+D)X).\end{aligned}$$

Inverting the variables and multiplying by $x_0 \ldots x_{n-1}$, we get

$$\begin{aligned}\det(I_n + C^{-1}X^{-1})\det(X) &= \det(X + C^{-1})\\ = \det(I_n + (A+D)X^{-1})\det(X) &= \det(X + (A+D)).\end{aligned}$$

This means $\det(C^{-1} - D + X) = \det(A+X)$, thus $C^{-1} - D \overset{\text{PME}}{=} A$.

4.3 Derandomising the reduction

The reduction above can be derandomised in quasi-polynomial time as follows:

- Note that in Item 3 of Observation 4.2, by Lemma 4.1, $\det(\operatorname{adj}(A+Y)[S_0, S_1]) \neq 0 \iff \det((A+Y)[\bar{S}_1, \bar{S}_0]) \neq 0$. Thus, all items in Observation 4.2 are read-once determinants, and we can use the [GT17] hitting set to come up with a quasi-polynomially large set of

diagonal matrices $D_0, \ldots, D_{k-1}$ containing at least one matrix D_i such that the irreducible blocks of $(A + D_i)^{-1}$ satisfy property $\mathcal{R}$.

- Say for some $j \in [0\,..\,k)$, D_j is a 'bad' matrix, that is, the off-diagonal entries of $(A+D_j)^{-1}$ are not guaranteed to be non-zero. Then for two indices $p, q \in [0\,..\,n)$, if they belong to different irreducible blocks, the algorithm in section 4.2.2 still identifies them as belonging to different blocks. It is when p and q belong to the same irreducible block that the algorithm may fail: it might put p and q as being in separate blocks. This means that for the 'correct' diagonal matrix D_i, the number of irreducible blocks returned by the algorithm will be minimal. By running it for all diagonal matrices $D_0, \ldots, D_{k-1}$ and picking the matrix which returns the least number of irreducible blocks, we can find the indices corresponding to the irreducible blocks of A.

- We can run the property $\mathcal{R}$ reconstruction algorithm for all matrices $(A + D_i)^{-1}$. For a diagonal matrix D_i, let C_i be the matrix reconstructed, and let $B_i = C_i^{-1} - D_i$. We want to check if $B_i \overset{\text{PME}}{=} A$. To do so, we first find another diagonal matrix D_i' such that $(B_i+D_i')^{-1}$ satisfies property $\mathcal{R}$ (Since B_i is known explicitly, this can be done by checking if the polynomials in Observation 4.2 are non-zero). Then, we can check if the principal minors of $(A + D_i')^{-1}$ and $(B_i + D_i')^{-1}$ of order at most 4 are equal. If they are, then by Lemma 2.8, $A \overset{\text{PME}}{=} B_i$.

Chapter 5

Cut Discovery Algorithm

In this chapter, given a matrix A that satisfies property $\mathcal{R}$, we describe how to find a cut in a matrix $C \overset{\text{PME}}{=} A$ given oracle access to the principal minors of A of order 4 and smaller. We denote by $\mathsf{PM4}_A$ the oracle for querying the principal minors of A of order at most 4. We define property $\mathcal{P}$ as follows:

Definition 5.1 (Property $\mathcal{P}$) *Given $A \in \mathbb{F}^{n\times n}$ with non-zero off-diagonal entries and $n \geq 4$, we say a tuple $\{\{i,j\},\{k,l\}\}$ with $\{i,j,k,l\} \subseteq [0\,..\,n)$ and i,j,k,l being distinct satisfies property $\mathcal{P}$ if there exists a matrix $C \in \mathbb{F}^{4\times 4}$ such that $C \overset{PME}{=} A[\{i,j,k,l\}]$ and $\{i,j\}$ is a cut in C.*

Lemma 5.1 *Let $A \in \mathbb{F}^{n\times n}$ satisfy property $\mathcal{R}$ with $n \geq 4$. Suppose $\{\{i,j\},\{k,l_0\}\}$ and $\{\{i,j\},\{k,l_1\}\}$ satisfy property $\mathcal{P}$. Then $\{\{i,j\},\{l_0,l_1\}\}$ satisfies property $\mathcal{P}$.*

Given access to $\mathsf{PM4}_A$, we can check if a tuple $\{\{i,j\},\{k,l\}\}$ satisfies property $\mathcal{P}$ (We describe how in Section 6.2).

Definition 5.2 (Plausible set) *Given a matrix $A \in \mathbb{F}^{n\times n}$ with non-zero off-diagonal entries, $n \geq 4$, and a set $S \subset [0\,..\,n)$ with $2 \leq |S| \leq n-2$, we say S is a* plausible set *for A if for all $\{i,j\} \subseteq S$, $\{k,l\} \subseteq \bar{S}$, $\{\{i,j\},\{k,l\}\}$ satisfies property $\mathcal{P}$. We say S is* minimal *if no proper subset of S is a plausible set.*

Lemma 5.2 *Let $A \in \mathbb{F}^{n\times n}$ satisfy property $\mathcal{R}$ with $n \geq 4$, and let $S \subset [0\,..\,n)$ with $2 \leq |S| \leq n-2$. Then S is a plausible set for A if and only if there exists a matrix $C \overset{PME}{=} A$ such that S is a cut in C.*

We shall prove Lemma 5.1 and Lemma 5.2 later. We first present an algorithm that finds a cut in a matrix PME to A satisfying property $\mathcal{R}$, given oracle access to its principal minors of order at most 4.

Algorithm 2 Black-box cut discovery

Input: Access to $\mathsf{PM4}_A$ for a matrix A satisfying property $\mathcal{R}$.

Output: A minimal cut S in some matrix $C \overset{\text{PME}}{=} A$ if one exists, else output `'No cut'`.

Assumption: Access to a 2-SAT solver.

function FindCut($\mathsf{PM4}_A$)
  if $n \leq 3$ **then**
    return `'No cut'`
  end if
  for $(i, j, k, l) \in [0\,..\,n)^4$, with i, j, k, l being distinct, **do**
    if $\{\{i, j\}, \{k, l\}\}$ satisfies property $\mathcal{P}$ **then**
      Let Φ_t be an empty 2-CNF.
      For each $e \in [0\,..\,n) \setminus \{i, j, k, l\}$, x_e is a Boolean variable that can appear in Φ_t.
      for $e \in [0\,..\,n) \setminus \{i, j, k, l\}$ **do**
        if one of $\{\{i, j\}, \{k, e\}\}$ or $\{\{i, j\}, \{l, e\}\}$ does not satisfy property $\mathcal{P}$ **then**
          Add clause x_e to Φ_t.
        end if
        if one of $\{\{i, e\}, \{k, l\}\}$ or $\{\{j, e\}, \{k, l\}\}$ does not satisfy property $\mathcal{P}$ **then**
          Add clause $\neg x_e$ to Φ_t.
        end if
      end for
      for $(p, q) \in ([0\,..\,n) \setminus \{i, j, k, l\})^2, p \neq q$, **do**
        if one of $\{\{i, p\}, \{k, q\}\}$, $\{\{j, p\}, \{k, q\}\}$, $\{\{i, p\}, \{l, q\}\}$ or $\{\{j, p\}, \{l, q\}\}$ does not satisfy property $\mathcal{P}$ **then**
          Add clause $\neg x_p \vee x_q$ to Φ_t.
        end if
      end for
      if Φ_t is satisfiable **then**
        Let α be a minimal satisfying assignment for Φ_t.
        return $\{i, j\} \cup \{e \mid x_e = 1 \text{ in } \alpha\}$.
      end if
    end if
  end for
  return `'No cut'`
end function

Lemma 5.3 *Algorithm 2 outputs a minimal cut S in some $C \overset{PME}{=} A$ if and only if a cut exists in A.*

Proof: Let S be a cut in some $C \overset{\text{PME}}{=} A$. Then it is easy to see that for all $e \in [0\,..\,n) \setminus \{i,j,k,l\}$, $x_e = 1 \iff e \in S$ must satisfy Φ_t. Now suppose the algorithm outputs a satisfying assignment for some $\{i,j,k,l\}$. Let $S = \{e \mid x_e = 1\}$. Then

- $\forall e \in S$, $\neg x_e$ cannot have been a clause in Φ_t, so both $\{\{i,j\},\{k,e\}\}$ and $\{\{i,j\},\{l,e\}\}$ satisfy property $\mathcal{P}$. Similarly, $\forall e \in \bar{S}$, x_e cannot have been a clause in Φ_t, so both $\{\{i,e\},\{k,l\}\}$ and $\{\{j,e\},\{k,l\}\}$ satisfy property $\mathcal{P}$.
- $\forall p \in S, q \in \bar{S}$, $\neg x_p \vee x_q$ cannot have been a clause in Φ_t, which means all of $\{\{i,p\},\{k,q\}\}$, $\{\{j,p\},\{k,q\}\}$, $\{\{i,p\},\{l,q\}\}$ and $\{\{j,p\},\{l,q\}\}$ satisfy property $\mathcal{P}$.
- $\forall p,q \in S, r \in \bar{S}$, all of $\{\{i,p\},\{k,r\}\}$, $\{\{i,q\},\{k,r\}\}$, $\{\{i,p\},\{l,r\}\}$ and $\{\{i,q\},\{l,r\}\}$ satisfy property $\mathcal{P}$, so by Lemma 5.1, both $\{\{p,q\},\{k,r\}\}$ and $\{\{p,q\},\{l,r\}\}$, satisfy property $\mathcal{P}$. Similarly, we can see that $\forall p \in S, q, r \in \bar{S}$, both $\{\{p,i\},\{q,r\}\}$ and $\{\{p,j\},\{q,r\}\}$ satisfy property $\mathcal{P}$.
- $\forall p,q \in S, r,t \in \bar{S}$, both $\{\{p,i\},\{r,t\}\}$ and $\{\{q,i\},\{r,t\}\}$ satisfy property $\mathcal{P}$, thus by Lemma 5.1, $\{\{p,q\},\{r,t\}\}$ satisfies property $\mathcal{P}$.

This means that $\{i,j\} \cup S$ is a plausible set for A, which, by Lemma 5.2, is a cut in some $C \overset{\text{PME}}{=} A$. Since at Line 23, we insist on a minimal satisfying assignment (an assignment where flipping any 1s to 0s causes the CNF to be unsatisfied), the cut output must be a minimal cut. □

5.1 Proof of Lemma 5.1

Lemma 5.1 (restated). *Let $A \in \mathbb{F}^{n \times n}$ satisfy property $\mathcal{R}$ with $n \geq 4$. Suppose $\{\{i,j\},\{k,l_0\}\}$ and $\{\{i,j\},\{k,l_1\}\}$ satisfy property $\mathcal{P}$. Then $\{\{i,j\},\{l_0,l_1\}\}$ satisfies property $\mathcal{P}$.*

Proof: We use the following lemma:

Lemma 5.4 ([CGGR25, Lemma 3.2]) *Let A be an $n \times n$ matrix over $\mathbb{F}$ with nonzero off-diagonal entries. Let $S \subseteq [0\,..\,n)$ be a cut in A. Then, for any $T \subseteq [0\,..\,n)$ the following holds.*

1. *If T or $\bar{T}$ is a subset of S or $\bar{S}$, then T is a cut in A if and only if T is a cut in $\mathrm{ct}(A,S)$.*
2. *Otherwise, T is a cut in A if and only if $T \Delta S$ is a cut in $\mathrm{ct}(A,S)$.*

Observation 5.1 *Let $A \in \mathbb{F}^{n\times n}$ and let $\{\{i,j\},\{k,l\}\}$ satisfy property $\mathcal{P}$. Then either $\{i,j\}$ is a cut in $A[\{i,j,k,l\}]$ or both $\{i,k\}$ and $\{j,k\}$ are cuts in $A[\{i,j,k,l\}]$ (while $\{i,j\}$ is not).*

Proof: By definition, there exists $C \overset{\text{PME}}{=} A[\{i,j,k,l\}]$ such that $\{i,j\}$ is a cut in C. If $\{i,j\}$ is a cut in A, we are done. Suppose it is not, then by Lemma 2.7, there exists a cut in $A[\{i,j,k,l\}]$, which we assume without loss of generality is $\{i,k\}$, such that $\{i,j\}$ is a cut in $\text{ct}(A[\{i,j,k,l\}],\{i,k\})$. Since both $\{i,j\}$ and $\{i,k\}$ are cuts in $\text{ct}(A[\{i,j,k,l\}],\{i,k\})$, by Lemma 5.4, $\{i,j\}\Delta\{i,k\} = \{j,k\}$ is a cut in $A[\{i,j,k,l\}]$. □

Now we prove Lemma 5.1. Assume without loss of generality that $\{\{0,1\},\{2,3\}\}$ and $\{\{0,1\},\{2,4\}\}$ satisfy property $\mathcal{P}$. We look at the following cases:

1. $\{0,1\}$ **is a cut in both** $A[\{0,1,2,3\}]$ **and** $A[\{0,1,2,4\}]$**:** $\{0,1\}$ would be a cut in $A[\{0,1,2,3,4\}]$ and hence, in $A[\{0,1,3,4\}]$.

2. $\{0,1\}$ **is not a cut in either** $A[\{0,1,2,3\}]$ **or** $A[\{0,1,2,4\}]$**:** By Observation 5.1, $\{0,2\}$ and $\{1,2\}$ are cuts in both $A[\{0,1,2,3\}]$ and $A[\{0,1,2,4\}]$. This means $\{0,2\}$ and $\{1,2\}$ are cuts in $A[\{0,1,2,3,4\}]$, and by Lemma 5.4, $\{0,2\}\Delta\{1,2\} = \{0,1\}$ is a cut in $C = \text{ct}(A[\{0,1,2,3,4\}],\{1,2\})$. Thus, $\{0,1\}$ is a cut in $C[\{0,1,3,4\}] \overset{\text{PME}}{=} A[\{0,1,3,4\}]$.

3. $\{0,1\}$ **is a cut in** $A[\{0,1,2,3\}]$ **but not in** $A[\{0,1,2,4\}]$**:** By Observation 5.1, $\{0,4\}$ and $\{1,4\}$ are cuts in $A[\{0,1,2,4\}]$. Since A satisfies property $\mathcal{R}$ and $\{0,1\}$ is a cut in $A[\{0,1,2,3\}]$, either $A[\{0,1\},\{2,3,4\}]$ or $A[\{0,1,4\},\{2,3\}]$ is rank one; and either $A[\{2,3\},\{0,1,4\}]$ or $A[\{2,3,4\},\{0,1\}]$ is rank one.

   (a) If $A[\{0,1\},\{2,3,4\}]$ and $A[\{2,3,4\},\{0,1\}]$ are rank one, then $\{0,1\}$ is a cut in $A[\{0,1,3,4\}]$.

   (b) If $A[\{0,1,4\},\{2,3\}]$ and $A[\{2,3\},\{0,1,4\}]$ are rank one, then $\{0,4\}$ is a cut in $A[\{0,2,3,4\}]$. But $\{0,4\}$ is a cut in $A[\{0,1,2,4\}]$, which means $\{0,4\}$ is a cut in $A[\{0,1,3,4\}]$. Similarly, $\{1,4\}$ is a cut in $A[\{1,2,3,4\}]$ and $A[\{0,1,2,4\}]$, and thus in $A[\{0,1,3,4\}]$. Thus, by Lemma 5.4, $\{0,4\}\Delta\{1,4\} = \{0,1\}$ is a cut in $\text{ct}(A[\{0,1,3,4\}],\{1,4\})$.

   (c) If $A[\{0,1,4\},\{2,3\}]$ and $A[\{2,3,4\},\{0,1\}]$ are rank one, then,

$$A[\{0,1,2,3,4\}] = \begin{bmatrix} * & * & a & b & * \\ * & * & \alpha a & \alpha b & * \\ c & \gamma c & * & * & * \\ d & \gamma d & * & * & * \\ e & \gamma e & \beta a & \beta b & * \end{bmatrix}.$$

If we consider the fact that $\{0,4\}$ and $\{1,4\}$ are cuts in $A[\{0,1,2,4\}]$, we can see that $\{0,1\}$ must be a cut in $A[\{0,1,3,4\}]$.

$$A[\{0,1,2,3,4\}] = \begin{bmatrix} * & \frac{\gamma e}{\beta} & a & b & \frac{\delta e}{\beta} \\ \frac{\alpha e}{\beta} & * & \alpha a & \alpha b & \frac{\alpha\delta e}{\beta} \\ c & \gamma c & * & * & \delta c \\ d & \gamma d & * & * & * \\ e & \gamma e & \beta a & \beta b & * \end{bmatrix}.$$

If $A[\{0,1\},\{2,3,4\}]$ and $A[\{2,3\},\{0,1,4\}]$ are rank one, a similar analysis with the transpose holds.

□

5.2 Proof of Lemma 5.2

Lemma 5.2 (restated). *Let $A \in \mathbb{F}^{n\times n}$ satisfy property $\mathcal{R}$ with $n \geq 4$, and let $S \subset [0\,..\,n)$ with $2 \leq |S| \leq n-2$. Then S is a plausible set for A if and only if there exists a matrix $C \overset{PME}{=} A$ such that S is a cut in C.*

Proof: The reverse direction trivially follows from the definition of a plausible set, so we prove the forwards direction. We prove it in three stages: when S is of size 2, when S is minimal and finally the general case.

5.2.1 When S is of size 2

Suppose S is a plausible set for A of size 2. Define a binary relation $\sim$ on $\bar{S}$ as follows:

$$e \sim f \iff e = f \text{ or } A[S \cup \{e,f\}] \text{ has cut } S.$$

It is easy to see that $\sim$ is an equivalence relation on $\bar{S}$. For all $e \in \bar{S}$, let T_e be the equivalence class of e, and $\bar{T_e} = \bar{S} \setminus T_e$.

Claim 5.1 *Given an $n \times n$ matrix A with non-zero off-diagonal entries, suppose $S = \{s_0, s_1\}$ is a plausible set for A. Then either S is a cut in A or for all $e \in \bar{S}$, both $\{s_0\} \cup T_e$ and $\{s_1\} \cup T_e$ are cuts in A.*

Proof:

We prove by induction on n. Assume without loss of generality $S = \{0,1\}$. If $n = 4$, we have two cases:

1. If $2 \sim 3$, then $\{0,1\}$ is a cut in A.

2. If $2 \not\sim 3$, then by Observation 5.1, $\{0,2\}$,$\{1,2\}$, $\{0,3\}$ and $\{1,3\}$ are all cuts in A.

Now we handle the induction case for $n \geq 5$.

Claim 5.2 *If for some $e \in [n] \setminus \{0,1\}$, $|\bar{T}_e| \geq 2$, then both $\{0\} \cup T_e$ and $\{1\} \cup T_e$ are cuts in A.*

Proof: For each $f \in \bar{T}_e$, note that $\{0,1\}$ remains a plausible set for $A[[0\,..\,n) \setminus \{f\}]$ and since $f \in \bar{T}_e$, T_e remains the same for $A[[0\,..\,n) \setminus \{f\}]$ as well. By induction, either $\{0,1\}$ is a cut of $A[[0\,..\,n) \setminus \{f\}]$ or both $\{0\} \cup T_e$ and $\{1\} \cup T_e$ are cuts of $A[[0\,..\,n) \setminus \{f\}]$.

We now show that $\{0,1\}$ cannot be a cut in $A[[0\,..\,n) \setminus \{f\}]$ for any $f \in \bar{T}_e$; suppose $\{0,1\}$ were a cut of $A[[0\,..\,n) \setminus \{f\}]$. Without loss of generality, let $\{f,g\} \subseteq \bar{T}_e$. Then $\{0,1\}$ would be a cut in $A[\{0,1,e,g\}]$, contradicting the fact that $g \in \bar{T}_e$. Thus, both $\{0\} \cup T_e$ and $\{1\} \cup T_e$ are cuts of $A[[0\,..\,n) \setminus \{f\}]$ and $A[[0\,..\,n) \setminus \{g\}]$, making both $\{0\} \cup T_e$ and $\{1\} \cup T_e$ cuts of A.

□

If there exists $e \in [0\,..\,n) \setminus \{0,1\}$ such that $|\bar{T}_e| = 0$, then for all $f \in [0\,..\,n) \setminus \{0,1,e\}$, $A[\{0,1,e,f\}]$ has cut $\{0,1\}$. Thus, $\{0,1\}$ is a cut in A. Now suppose $|\bar{T}_e| \geq 1$ for all $e \in [0\,..\,n) \setminus \{0,1\}$. Then for each $e \in [0\,..\,n) \setminus \{0,1\}$,

1. If $|\bar{T}_e| \geq 2$, then by Claim 5.2, both $\{0\} \cup T_e$ and $\{1\} \cup T_e$ cuts of A.

2. If $|\bar{T}_e| = 1$, let $\bar{T}_e = \{f\}$. Then $\bar{T}_f = T_e = [0\,..\,n) \setminus \{0,1,f\}$ and $|\bar{T}_f| = n - 3 \geq 2$. Since $T_f = \{f\}$, by Claim 5.2, both $\{0,f\}$ and $\{1,f\}$ are cuts in A, which means both $\{1\} \cup T_e$ and $\{0\} \cup T_e$ are cuts in A.

□

By the above claim, either S is a cut in A or for all $e \in [0\,..\,n) \setminus S$, $\{s_0\} \cup T_e$ and $\{s_1\} \cup T_e$ are cuts. If the latter is true, then by Lemma 5.4, $(\{s_0\} \cup T_e)\Delta(\{s_1\} \cup T_e) = S$ is a cut in $\mathrm{ct}(A, \{s_0\} \cup T_e)$.

□

5.2.2 When S is minimal

Claim 5.3 *Let $A \in \mathbb{F}^{n \times n}$ satisfy property $\mathcal{R}$. If S is a minimal plausible set for A with $|S| < n-2$, then S is a minimal plausible set for $A[[0\,..\,n) \setminus \{e\}]$ for all $e \in \bar{S}$.*

Proof: We see that S is a plausible set for $A[[0\,..\,n) \setminus \{e\}]$. Suppose S is not minimal for $A[[0\,..\,n) \setminus \{e\}]$, which means there exists $T \subset S$ that is plausible for $A[[0\,..\,n) \setminus \{e\}]$. Since T

is not a plausible set for A, there exists $\{i,j\} \subseteq T$, $k \in S \setminus T$ such that $\{\{i,j\},\{k,e\}\}$ does not satisfy property $\mathcal{P}$. Let $l \in \bar{S} \setminus \{e\}$. Since S is a plausible set for A, $\{\{i,j\},\{l,e\}\}$ satisfies property $\mathcal{P}$. Since T is a plausible set for $A[[0\,..\,n) \setminus \{e\}]$, $\{\{i,j\},\{k,l\}\}$ satisfies property $\mathcal{P}$. But by Lemma 5.1, $\{\{i,j\},\{k,e\}\}$ must satisfy property $\mathcal{P}$, a contradiction. Thus S must be a minimal plausible set for $A[[0\,..\,n) \setminus \{e\}]$. □

We also require the following lemma:

Lemma 5.5 ([CGGR25, Lemma 3.6]) *Let A and B be two $n \times n$ matrices with non-zero off-diagonal entries and $A \overset{PME}{=} B$. Let S be a minimal cut of A of size greater than 2. Then, S is also a cut of B.*

We now prove Lemma 5.2 for minimal plausible sets by induction on n. The base case of $n = 4$ is already handled by the previous subsection as $|S| = 2$. Now let $n > 4$, and S be a minimal plausible set for A. Since $|S| = 2$ is already proven, we assume $|S|, |\bar{S}| > 2$. Let $\{f,g\} \subset \bar{S}$. By Claim 5.3, S is a minimal plausible set for both $A[[0\,..\,n) \setminus \{f\}]$ and $A[[0\,..\,n) \setminus \{g\}]$. By induction there exist $C_f \overset{\text{PME}}{=} A[[0\,..\,n) \setminus \{f\}]$ and $C_g \overset{\text{PME}}{=} A[[0\,..\,n) \setminus \{g\}]$ with S as a cut. Since S is a minimal plausible set for $A[[0\,..\,n) \setminus \{f\}]$ and $A[[0\,..\,n) \setminus \{g\}]$, it must be a minimal cut for C_f and C_g. Thus, by Lemma 5.5, S is a cut in $A[[0\,..\,n) \setminus \{f\}]$ and $A[[0\,..\,n) \setminus \{g\}]$, and as a result, S is a cut in A.

5.2.3 General S

First, we state a few results that we need.

Lemma 5.6 ([CGGR25, Lemma 3.4]) *Let $A \in \mathbb{F}^{n\times n}$ with non-zero off-diagonal entries. Let $S \subset [0\,..\,n)$ be a cut in the matrix A and $e \in S$, and suppose $T \subseteq \bar{S}$ is a cut in $A[\bar{S} \cup \{e\}]$. Then, T is also a cut in the matrix A.*

Lemma 5.7 ([ACG+26, Lemma 5.7]) *Let $A, B \in \mathbb{F}^{n\times n}$ with non-zero off-diagonal entries. Let $S \subset [0\,..\,n)$ be a cut in both A and B. Let $s \in S$ and $t \in \bar{S}$. Then $A \overset{PME}{=} B$ if and only if $A[S \cup \{t\}] \overset{PME}{=} B[S \cup \{t\}]$ and $A[\bar{S} \cup \{s\}] \overset{PME}{=} B[\bar{S} \cup \{s\}]$. Furthermore, if S is a minimal cut of A, then $A[S \cup \{t\}]$ has no cut.*

We now prove the following:

Lemma 5.8 *Let $A \in \mathbb{F}^{n\times n}$ matrix with non-zero off diagonal entries and $S \subset [0\,..\,n)$ be a cut of A. Let $s \in S, t \in \bar{S}$ and M and N be two matrices such that $M \overset{PME}{=} A[S \cup \{t\}]$ and*

$N \overset{PME}{=} A[\bar{S} \cup \{s\}]$. *Then, the matrix* B *defined as follows is principal minor equivalent to* A.

$$B = \begin{array}{c} \\ S \\ \\ \bar{S} \end{array} \begin{array}{c} \begin{array}{cc} S & \bar{S} \end{array} \\ \begin{bmatrix} M[S] & M[S,t] \cdot \frac{N[s,\bar{S}]}{N[s,t]} \\ \\ \frac{N[\bar{S},s]}{N[t,s]} \cdot M[t,S] & N[\bar{S}] \end{bmatrix} \end{array}.$$

Proof: It is easy to see that $B[S \cup \{t\}] = M$. We claim that $B[\bar{S} \cup \{s\}] = DND^{-1}$ where $D \in \mathbb{F}^{(|\bar{S}|+1)\times(|\bar{S}|+1)}$ is a diagonal matrix such that $D[s] = \frac{M[s,t]}{N[s,t]}$ and $\forall\, e \in \bar{S}, D[e] = 1$.

Observe that since $A[s,t]$ and $A[t,s]$ are non-zero, $M[s,t], M[t,s], N[s,t], N[t,s]$ are all non-zero as well. Also, since $M \overset{\text{PME}}{=} A[S \cup \{t\}]$ and $N \overset{\text{PME}}{=} A[\bar{S} \cup \{s\}]$, we have $A[s,t]A[t,s] = M[s,t]M[t,s] = N[s,t]N[t,s]$. This means $\frac{N[s,t]}{M[s,t]} = \frac{M[t,s]}{N[t,s]}$. Then

$$\begin{aligned} DND^{-1} &= \begin{bmatrix} \frac{M[s,t]}{N[s,t]} & O_{1,|\bar{S}|} \\ O_{|\bar{S}|,1} & I_{|\bar{S}|} \end{bmatrix} \begin{bmatrix} N[s] & N[s,\bar{S}] \\ N[\bar{S},s] & N[\bar{S}] \end{bmatrix} \begin{bmatrix} \frac{N[s,t]}{M[s,t]} & O_{1,|\bar{S}|} \\ O_{|\bar{S}|,1} & I_{|\bar{S}|} \end{bmatrix} \\ &= \begin{bmatrix} \frac{M[s,t]}{N[s,t]}N[s] & \frac{M[s,t]}{N[s,t]}N[s,\bar{S}] \\ N[\bar{S},s] & N[\bar{S}] \end{bmatrix} \begin{bmatrix} \frac{N[s,t]}{M[s,t]} & O_{1,|\bar{S}|} \\ O_{|\bar{S}|,1} & I_{|\bar{S}|} \end{bmatrix} \\ &= \begin{bmatrix} N[s] & \frac{M[s,t]}{N[s,t]}N[s,\bar{S}] \\ \frac{N[s,t]}{M[s,t]}N[\bar{S},s] & N[\bar{S}] \end{bmatrix} = \begin{bmatrix} N[s] & \frac{M[s,t]}{N[s,t]}N[s,\bar{S}] \\ \frac{M[t,s]}{N[t,s]}N[\bar{S},s] & N[\bar{S}] \end{bmatrix}. \end{aligned}$$

Thus, $B[\bar{S} \cup \{s\}] = DND^{-1}$, which implies $B[\bar{S} \cup \{s\}] \overset{\text{PME}}{=} N \overset{\text{PME}}{=} A[\bar{S} \cup \{s\}]$. By Lemma 5.7, $A \overset{\text{PME}}{=} B$ □

Now we prove Lemma 5.2 for a general plausible set S by induction on n. The base case $n = 4$ follows from the $|S| = 2$ case. Assume $n > 4$. Since we have already proven the lemma when S is a minimal plausible set, we assume S is not minimal. Let $T \subset S$ be a minimal plausible set. Then, there exists a matrix $C \overset{\text{PME}}{=} A$ such that T is a cut in C. Let $t \in T$. Observe that $\bar{S}$ is a plausible set for $A[\bar{T} \cup \{t\}]$. Since $A[\bar{T} \cup \{t\}]$ satisfies property $\mathcal{R}$, by induction, there exists a matrix $N \overset{\text{PME}}{=} C[\bar{T} \cup \{t\}]$ such that $\bar{S}$ is a cut in N. Let $s \in S \setminus T$ and $M = C[T \cup \{s\}]$. Let $C' \overset{\text{PME}}{=} C$ be the matrix constructed from Lemma 5.8 using $M \overset{\text{PME}}{=} C[T \cup \{s\}]$ and $N \overset{\text{PME}}{=} C[\bar{T} \cup \{t\}]$. By construction, T is a cut in C' and $C'[\bar{T} \cup \{t\}] \overset{\text{DS}}{=} N$, thus $C'[\bar{T} \cup \{t\}]$ has $\bar{S} \subseteq \bar{T}$ as a cut. By Lemma 5.6, $\bar{S}$ and thus S is a cut in C'.

Chapter 6

Reconstruction of Matrices with Property $\mathcal{R}$

In this chapter, we demonstrate how to reconstruct matrices satisfying property $\mathcal{R}$ given oracle access to their principal minors of order at most 4. We first demonstrate how to reconstruct matrices for $n = 2, 3$ and 4, then we present the reconstruction algorithm for matrices satisfying property $\mathcal{R}$ without a cut. Finally, we extend the algorithm to general matrices satisfying property $\mathcal{R}$, irrespective of the presence of a cut.

We describe a 'canonical' form for matrices with non-zero off-diagonal entries that helps simplify analyses of diagonally similar matrices.

Definition 6.1 (The function $\mathcal{N}$) *$\mathcal{N}$ is a function defined on $n \times n$ matrices with non-zero off-diagonal entries as*

$$\mathcal{N}(B) = \text{diag}(1, B[0,1], \ldots, B[0,n-1]) \cdot B \cdot \text{diag}(1, B[0,1]^{-1}, \ldots, B[0,n-1]^{-1}).$$

Lemma 6.1 *For any matrices $B, C \in \mathbb{F}^{n \times n}$ with non-zero off-diagonal entries,*

- $\mathcal{N}(B) \stackrel{DS}{=} B$.
- *For each $i \in [1\,..\,n)$, $\mathcal{N}(B)[0,i] = 1$.*
- *If $C \stackrel{DS}{=} B$ and $\forall\, i \in [1\,..\,n)$, $B[0,i] = C[0,i] = 1$, then $C = B$.*
- $B \stackrel{DS}{=} C \iff \mathcal{N}(B) = \mathcal{N}(C)$.

Proof: The first two statements follow trivially from the definition of $\mathcal{N}$. The fourth follows from the first three, so we prove the third statement. Let $D = \text{diag}(d_0, \ldots, d_{n-1})$ be the such that $C = DBD^{-1}$. Then for all $i \in [1\,..\,n)$, we have $C[0,i] = d_0 d_i^{-1} B[0,i]$. We infer that $d_0 = d_i$ for all i, which makes D a scalar multiple of identity. Thus, $C = DBD^{-1} = BDD^{-1} = B$. $\square$

6.1 Reconstruction of 2×2 and 3×3 matrices

The following observation allows us to reconstruct 2×2 matrices.

Observation 6.1 *Let $A \in \mathbb{F}^{2\times 2}$. Then, with 3 queries to $\mathsf{PM4}_A$, we can reconstruct the following matrix that is principal minor equivalent to A.*

$$\begin{bmatrix} A[0] & 1 \\ A[0,1]A[1,0] & A[1] \end{bmatrix}$$

We now deal with reconstruction of 3×3 matrices.

Lemma 6.2 *Let $B \in \mathbb{F}^{3\times 3}$ with non-zero off-diagonal entries. Then with 7 queries to $\mathsf{PM4}_A$ and an algorithm to compute square roots, we can reconstruct the following matrix that is principal minor equivalent to A*

$$\begin{bmatrix} A[0] & 1 & 1 \\ A[0,1]A[1,0] & A[1] & \gamma \\ A[0,2]A[2,0] & \frac{A[1,2]A[2,1]}{\gamma} & A[2] \end{bmatrix}$$

where γ is a root of the quadratic equation $az^2 - bz + c$ in z, with

- $a = A[0,2]A[2,0]$.
- $b = A[0,1]A[1,2]A[2,0] + A[0,2]A[2,1]A[1,0]$.
- $c = A[0,1]A[1,0]A[1,2]A[2,1]$.

Proof: Let B be the reconstructed matrix. It is easy to see that $A[S] \overset{\text{PME}}{=} B[S]$ for all $|S| \leq 2$. It is also easy to verify that

- $a = \det(A[\{0,2\}]) - A[0]A[2]$.
- $b = \det(A) - A[0]\det(A[\{1,2\}]) - A[1]\det(A[\{0,2\}]) - A[2]\det(A[\{0,1\}]) + 2A[0]A[1]A[2]$.
- $c = (\det(A[\{0,1\}]) - A[0]A[1])(\det(A[\{1,2\}]) - A[1]A[2])$

Thus, we can form and solve the equation $az^2 - bz + c$. Since $A[S] \overset{\text{PME}}{=} B[S]$ for all $|S| \leq 2$, we have

$$\begin{aligned} \det(A) - \det(B) &= A[0,1]A[1,2]A[2,0] + A[0,2]A[2,1]A[1,0] \\ &\quad - B[0,1]B[1,2]B[2,0] - B[0,2]B[2,1]B[1,0] \\ &= b - \gamma a - \frac{c}{\gamma} = 0. \end{aligned}$$

Thus, $\det(A) = \det(B)$. □

Corollary 6.1 *Let $A \in \mathbb{F}^{3\times 3}$ with non-zero off-diagonal entries and let $B \overset{PME}{=} A$ such that $B[0,1] = B[0,2] = 1$. Then B must be one of the two possible matrices that Lemma 6.2 can output.*

Proof: Equating the order 1 and 2 principal minors of A and B, it can be easily seen that B must have the same structure as in Lemma 6.2 for some value of γ. Equating $\det(A)$ and $\det(B)$, we see that γ must satisfy $a\gamma^2 - b\gamma + c$ with a, b, c as in Lemma 6.2. □

6.2 Reconstruction of 4×4 matrices

As a requirement of Algorithm 2, instead of reconstructing a single matrix, we discuss generating a list of principal minor equivalent matrices which we define as follows:

Definition 6.2 (The set $\mathcal{S}_A$) *Given $A \in \mathbb{F}^{4\times 4}$, $\mathcal{S}_A$ is a finite set of 4×4 matrices defined as follows:*

$$\mathcal{S}_A = \{\mathcal{N}(A),\ \mathcal{N}(A^T)\} \cup \{\mathcal{N}(\mathrm{ct}(A,T)) :\ T \subset [0\,..\,4)\ \textit{is a cut in } A.\}$$

Claim 6.1 *Given a matrix $A \in \mathbb{F}^{4\times 4}$ with non-zero off-diagonal entries, for all matrices $C \overset{PME}{=} A$, there exists a $B \in \mathcal{S}_A$ such that $C \overset{DS}{=} B$.*

Proof: Suppose $B \overset{\text{PME}}{=} A$. Then by Lemma 2.7, either $B \overset{\text{DE}}{=} A$ or there exists a cut T in A such that $B \overset{\text{DE}}{=} \mathrm{ct}(A,T)$. If $B \overset{\text{DE}}{=} A$, then either $B \overset{\text{DS}}{=} \mathcal{N}(A)$ or $B \overset{\text{DS}}{=} \mathcal{N}(A^T)$. If $B \overset{\text{DE}}{=} \mathrm{ct}(A,T)$, then either $B \overset{\text{DS}}{=} \mathrm{ct}(A,T)$ or $B \overset{\text{DS}}{=} \mathrm{ct}(A,T)^T = \mathrm{ct}(A,\bar{T})$, implying $B \overset{\text{DS}}{=} \mathcal{N}(\mathrm{ct}(A,T))$ or $B \overset{\text{DS}}{=} \mathcal{N}(\mathrm{ct}(A,\bar{T}))$ respectively. □

The following algorithm constructs the list $\mathcal{S}_A$ given $\mathsf{PM4}_A$.

Algorithm 3 Reconstruction of 4×4 matrices

Input: $\mathsf{PM4}_A$ where $A \in \mathbb{F}^{4\times 4}$ has non-zero off-diagonal entries.

Output: The set $\mathcal{S}_A$.

Assumption: Access to an oracle to compute square roots in $\mathbb{F}$.

1: $\mathcal{S}_A, R_{1,2}, R_{2,3}, R_{1,3} \leftarrow \emptyset$.
2: **for** $(i,j) \in \{(1,2),(2,3),(1,3)\}$ **do**
3: $\quad a \leftarrow A[0,j]A[j,0]$.
4: $\quad b \leftarrow A[0,i]A[i,j]A[j,0] + A[0,j]A[j,i]A[i,0]$.
5: $\quad c \leftarrow A[0,i][i,0]A[i,j][j,i]$.
6: $\quad R_{i,j} \leftarrow \{z \in \mathbb{F} : az^2 - bz + c = 0\}$.
7: **end for**
8: **for** $(\alpha,\beta,\gamma) \in R_{1,2} \times R_{2,3} \times R_{1,3}$ **do**
9: $\quad B \leftarrow \begin{bmatrix} A[0] & 1 & 1 & 1 \\ A[0,1]A[1,0] & A[1] & \alpha & \gamma \\ A[0,2]A[2,0] & \frac{A[1,2]A[2,1]}{\alpha} & A[2] & \beta \\ A[0,3]A[3,0] & \frac{A[1,3]A[3,1]}{\gamma} & \frac{A[2,3]A[3,2]}{\beta} & A[3] \end{bmatrix}$.
10: $\quad$ **if** $B \overset{\text{PME}}{=} A$ **then**
11: $\quad\quad \mathcal{S}_A \leftarrow \mathcal{S}_A \cup \{B\}$.
12: $\quad$ **end if**
13: **end for**
14: **return** $\mathcal{S}_A$.

Claim 6.2 *Algorithm 3 outputs the set $\mathcal{S}_A$ as defined in Definition 6.2.*

Proof: Let $\mathcal{S}'$ be the set output by Algorithm 3. By Claim 6.1 and Lemma 6.1, it is easy to see that $\mathcal{S}' \subseteq \mathcal{S}_A$. Let $B \in \mathcal{S}_A$. Since $B[0,1] = B[0,2] = B[0,3] = 1$ and $B[T] \overset{\text{PME}}{=} A[T]$ for $T \in \{\{0,1,2\},\{0,2,3\},\{0,1,3\}\}$, by Corollary 6.1, B must have the same structure as in Line 9. Thus, $B \in \mathcal{S}'$ which means $\mathcal{S}' = \mathcal{S}_A$. □

6.3 Reconstruction of cut-free matrices

In this subsection, we focus on reconstruction of matrices satisfying property $\mathcal{R}$ and without cuts. To do so, we rely on the following lemma:

Lemma 6.3 ([ACG+26, Corollary 5.1]) *Let $n \geq 5$ and $A \in \mathbb{F}^{n\times n}$ be a matrix satisfying property $\mathcal{R}$ and has no cut. Then there exists $S \subseteq [0\,..\,n)$ with $|S| \geq 3$ such that for each $i \in S$, $A[[0\,..\,n) \setminus \{i\}]$ has no cut.*

Algorithm 4 Finding a sequence of indices satisfying no cut property

Input: $\mathsf{PM4}_A$ where $A \in \mathbb{F}^{n\times n}$ satisfies property $\mathcal{R}$ and has no cut. Two indices $i_0, i_1 \in [0\,..\,n)$.

Output: A sequence of indices $(i_{n-1}, i_{n-2}, \ldots, i_0)$ such that $\forall\ k \in [3\,..\,n), A[\{i_0, \ldots, i_k\}]$ has no cut.

1: **function** NoCutSequence($\mathsf{PM4}_A$,i_0,i_1)
2: $Q \leftarrow ()$
3: $J \leftarrow [0\,..\,n) \setminus \{i_0, i_1\}$
4: **for** $k \in [0\,..\,n-2)$ **do**
5: **for** $j \in J$ **do**
6: **if** FindCut($\mathsf{PM4}_{A[(J\cup\{i_0,i_1\})\setminus\{j\}]}$) = `'No cut'` **then**
7: Append j to Q.
8: $J \leftarrow J \setminus \{j\}$.
9: Break out of inner for-loop.
10: **end if**
11: **end for**
12: **end for**
13: Append (i_1, i_0) to Q.
14: **return** Q.
15: **end function**

The correctness of Algorithm 4 follows trivially from Lemma 6.3.

Algorithm 5 Reconstruction of matrices with property $\mathcal{R}$ and no cuts

Input: $\mathsf{PM4}_A$ where $A \in \mathbb{F}^{n\times n}$ satisfies property $\mathcal{R}$ and has no cut.

Output: A matrix $B \overset{\text{PME}}{=} A$.

Assumption: Access to an oracle to compute square roots in $\mathbb{F}$.

1: **function** NoCutReconstruct($\mathsf{PM4}_A$)
2: $(i_{n-1}, i_{n-2}, \ldots, i_2, 1, 0) \leftarrow$ NoCutSequence($\mathsf{PM4}_A$,0,1).
3: Let B be an uninitialised $n \times n$ matrix.
4: $B[\{0,1\}] \leftarrow \begin{bmatrix} A[0] & 1 \\ A[0,1]A[1,0] & A[1] \end{bmatrix}$.
5: **for** $j \in [2\,..\,n)$ **do**
6: $B[\{0,i_j\}] \leftarrow \begin{bmatrix} A[0] & 1 \\ A[0,i_j]A[i_j,0] & A[i_j] \end{bmatrix}$.
7: $(k_j, k_{j-1}, \ldots, k_2, k_1 = i_j, 0) \leftarrow$ NoCutSequence($\mathsf{PM4}_{A[\{0,1,i_2,\ldots,i_j\}]}, 0, i_j$).

Algorithm 5 Algorithm to reconstruct matrices with property $\mathcal{R}$ and no cuts (continued)

for $\ell \in \{2, 3, \ldots, j\}$ **do**

Let B_0, B_1 be uninitialised matrices. Let $T = \{0, k_2, \ldots, k_{\ell-1}\}$.

$B_0 \leftarrow$

	T	k_1	k_ℓ
T	$B[T \cup \{k_1\}]$		$B[T, k_\ell]$
k_1			$*$
k_ℓ	$B[k_\ell, T]$	$*$	$B[k_\ell]$

($*$ denotes uninitialised entry).

$B_1 \leftarrow$

	T	k_1	k_ℓ
T	$\mathcal{N}(B^T[T \cup \{k_1\}])$		$B[T, k_\ell]$
k_1			$*$
k_ℓ	$B[k_\ell, T]$	$*$	$B[k_\ell]$

($\mathcal{N}$ as defined in Definition 6.1).

$a \leftarrow A[0, k_\ell]A[k_\ell, 0]$.

$b \leftarrow A[0, k_1]A[k_1, k_\ell]A[k_\ell, 0] + A[0, k_\ell]A[k_\ell, k_1]A[k_1, 0]$.

$c \leftarrow A[0, k_1][k_1, 0]A[k_1, k_\ell][k_\ell, k_1]$.

Let $\{\gamma_0, \gamma_1\}$ be the roots of $a\gamma^2 - b\gamma + c$.

for $t \in \{0, 1\}$ **do**

$B_0[k_1, k_\ell], B_1[k_1, k_\ell] \leftarrow \gamma_t$.

$B_0[k_\ell, k_1], B_1[k_\ell, k_1] \leftarrow \frac{A[k_1, k_\ell]A[k_\ell, k_1]}{\gamma_t}$.

if $\forall\, S \subseteq \{0, k_1, \ldots, k_\ell\}, |S| \leq 4, B_0[S] \overset{\text{PME}}{=} A[S]$ **then**

$B[\{0, k_1, \ldots, k_\ell\}] \leftarrow B_0$.

break for-loop at Line 16.

else if $\forall\, S \subseteq \{0, k_1, \ldots, k_\ell\}, |S| \leq 4, B_1[S] \overset{\text{PME}}{=} A[S]$ **then**

$B[\{0, k_1, \ldots, k_\ell\}] \leftarrow B_1$.

break for-loop at Line 16.

end if

end for

end for

return B.

end function

The correctness of Algorithm 5 follows by setting $j = n - 1$ in the following lemma:

Lemma 6.4 *The following loop invariants hold for Algorithm 5:*

1. *After each iteration of the loop at Line 8,* $B[\{0, k_1, \ldots, k_\ell\}] \overset{DE}{=} A[\{0, k_1, \ldots, k_\ell\}]$.

2. *After each iteration of the loop at Line 5,* $B[\{0, 1, i_2, \ldots, i_j\}] \overset{DE}{=} A[\{0, 1, i_2 \ldots, i_j\}]$.

Proof: We prove it by a nested induction on j and ℓ. The base case of $j = 3, \ell = 3$ follows trivially from Lemma 6.2 and Lemma 2.5. Note that in this case, $B_0 = B_1$.

For ease of notation, let $Q_t = \{0, 1, \ldots, i_t\}$ and $K_t = \{0, k_1, k_2, \ldots, k_t\}$. Assume Item 2 holds for $j - 1$. Once again, the base case of $\ell = 3$ follows from Lemma 6.2 and Lemma 2.5. We now assume Item 1 holds for $\ell - 1$. Recall that $i_j = k_1$, which means $K_\ell \setminus \{k_1\} \subseteq Q_{j-1}$. Furthermore, due to Algorithm 4, neither $A[Q_t]$ nor $A[K_t]$ have any cuts for all $2 \leq t < n$. From the induction hypothesis, one of the four cases can occur:

- $B[Q_{j-1}] \overset{\text{DS}}{=} A[Q_{j-1}]$ and $B[K_{\ell-1}] \overset{\text{DS}}{=} A[K_{\ell-1}]$: By Lemma 6.1, $B[Q_{j-1}] = \mathcal{N}(A)[Q_{j-1}]$ and $B[K_{\ell-1}] = \mathcal{N}(A)[K_{\ell-1}]$. This would mean that at Line 10, $B_0 = \mathcal{N}(A)[K_\ell]$ except for entries $B_0[k_1, k_\ell]$ and $B_0[k_\ell, k_1]$, which are still uninitialised. Since $\mathcal{N}(A) \overset{\text{DS}}{=} A$, $\mathcal{N}(A)[k_1, k_\ell]$ must be one of the roots of the equation at Line 15, which means that there is at least one value of γ for which the condition at Line 19 holds. Since $A[K_\ell]$ satisfies property $\mathcal{R}$ and has no cut, by Lemma 2.8 and Lemma 2.5, at the end of the iteration of the for-loop at Line 8, we would have $B[K_\ell] \overset{\text{DE}}{=} A[K_\ell]$. Note that after the update, $B[Q_{j-1}] = \mathcal{N}(A)[Q_{j-1}]$, so $B[Q_{j-1}] \overset{\text{DS}}{=} A[Q_{j-1}]$ remains true.

- $B[Q_{j-1}] \overset{\text{DS}}{=} A^T[Q_{j-1}]$ and $B[K_{\ell-1}] \overset{\text{DS}}{=} A^T[K_{\ell-1}]$: Given $\mathsf{PM4}_A$, it is impossible to distinguish between A and A^T. Thus, the above analysis with A replaced with A^T holds in this case.

- $B[Q_{j-1}] \overset{\text{DS}}{=} A[Q_{j-1}]$ and $B[K_{\ell-1}] \overset{\text{DS}}{=} A^T[K_{\ell-1}]$: By Lemma 6.1, $B[Q_{j-1}] = \mathcal{N}(A)[Q_{j-1}]$ and $B[K_{\ell-1}] = \mathcal{N}(A^T)[K_{\ell-1}]$. Since $K_{\ell-1} \setminus \{k_1\} \subseteq Q_{j-1}$, we must have $\mathcal{N}(A)[K_{\ell-1} \setminus \{k_1\}] = \mathcal{N}(A^T)[K_{\ell-1} \setminus \{k_1\}]$. This would mean that at Line 11, $B_1 = \mathcal{N}(A)[K_\ell]$ except for entries $B_1[k_1, k_\ell]$ and $B_1[k_\ell, k_1]$, which are still uninitialised. Since $\mathcal{N}(A) \overset{\text{DS}}{=} A$, $\mathcal{N}(A)[k_1, k_\ell]$ must be one of the roots of the equation at Line 15, which means that there is at least one value of γ for which the condition at Line 22 holds. Since $A[K_\ell]$ satisfies property $\mathcal{R}$ and has no cut, by Lemma 2.8 and Lemma 2.5, at the end of the iteration of the for-loop at Line 8, we would have $B[K_\ell] \overset{\text{DE}}{=} A[K_\ell]$. Note that after the update, $B[Q_{j-1}] = \mathcal{N}(A)[Q_{j-1}]$, so $B[Q_{j-1}] \overset{\text{DS}}{=} A[Q_{j-1}]$ remains true.

- $B[Q_{j-1}] \overset{\text{DS}}{=} A^T[Q_{j-1}]$ and $B[K_{\ell-1}] \overset{\text{DS}}{=} A[K_{\ell-1}]$: The above analysis with A replaced with A^T holds in this case.

Upon exiting the for-loop at Line 8, since $Q_j = K_j$, $B[Q_j] \overset{\text{DE}}{=} A[Q_j]$. □

6.4 Reconstruction of matrices with cuts

For the general reconstruction algorithm, we rely on Lemma 5.7 to recurse on smaller submatrices.

Algorithm 6 Reconstruction of matrices with property $\mathcal{R}$

Input: $\mathsf{PM4}_A$ where $A \in \mathbb{F}^{n\times n}$ satisfies property $\mathcal{R}$ and has no cut.

Output: A matrix $B \overset{\text{PME}}{=} A$.

1: **function** RECONSTRUCT($\mathsf{PM4}_A$)
2: $S \leftarrow$ FINDCUT($\mathsf{PM4}_A$).
3: **if** $S =$ `'No cut'` **then**
4: **return** NOCUTRECONSTRUCT($\mathsf{PM4}_A$)
5: **else** ▷ S is a minimal cut in some $C \overset{\text{PME}}{=} A$
6: Let $t \in \bar{S}$, $s \in S$.
7: $B_0 \leftarrow$ NOCUTRECONSTRUCT($\mathsf{PM4}_{A[S\cup\{t\}]}$).
8: $B_1 \leftarrow$ RECONSTRUCT($\mathsf{PM4}_{A[\bar{S}\cup\{s\}]}$).
9: **return**
$$\begin{array}{c} \\ S \\ \bar{S} \end{array}\begin{array}{c} \begin{array}{cc} S & \bar{S} \end{array} \\ \begin{bmatrix} B_0[S] & B_0[S,t]\cdot\frac{B_1[s,\bar{S}]}{B_1[s,t]} \\ \frac{B_1[\bar{S},s]}{B_1[t,s]}\cdot B_0[t,S] & B_1[\bar{S}] \end{bmatrix} \end{array}.$$
10: **end if**
11: **end function**

Since we recurse on smaller sub-matrices, the above algorithm runs in polynomial time. Its correctness follows easily from Lemma 5.7 and the fact that Algorithm 2 outputs a minimal cut in some $C \overset{\text{PME}}{=} A$.

Chapter 7

Conclusion

We demonstrated an algorithm to solve PMAP for matrices with property $\mathcal{R}$, and used it to find randomised polynomial time algorithms, that can be derandomised in quasi-polynomial time, for learning read-once determinants and black-box PMAP. Some possible avenues for future work include:

- Given a polynomial class $\mathcal{C}$ and a multivariate polynomial f, *equivalence testing* is the problem of checking if there exists a polynomial $g \in \mathcal{C}$ and an invertible matrix A such that $f(\mathbf{x}) = g(A\mathbf{x})$. Equivalence testing of polynomials of the black-box PMAP form $\det(A+X)$ easily follows from a suitable black-box factorisation algorithm [KT90]. What about equivalence testing of read-once determinants? The reduction from learning RODs to black-box PMAP fails to work for this case. It is not known if efficient algorithms exist for equivalence testing of RODs, nor if it is NP-hard.

- Adding to the previous problem, we can also ask about the existence of a sub-exponential deterministic black-box PIT algorithm for the *orbit* of RODs: a polynomial g is in the *orbit* of a polynomial f if there exists an invertible matrix A such that $g(\mathbf{x}) = f(A\mathbf{x})$. No sub-exponential deterministic black-box PIT is known for the orbit of RODs. Even sub-exponential deterministic black-box PIT for the orbit of the determinant polynomial is an open question.

References

[ACG+26] Abhiram Aravind, Abhranil Chatterjee, Sumanta Ghosh, Rohit Gurjar, Roshan Raj, and Chandan Saha. Learning read-once determinants and the principal minor assignment problem, 2026. Preprint available at arXiv:2603.04255 [v1], https://arxiv.org/abs/2603.04255. 6, 11, 27, 32

[Agr05] Manindra Agrawal. Proving lower bounds via pseudo-random generators. In Sundar Sarukkai and Sandeep Sen, editors, *FSTTCS 2005: Foundations of Software Technology and Theoretical Computer Science*, pages 92–105, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg. 1

[Ahm23] Abeer Al Ahmadieh. The fiber of the principal minor map, 2023. Available at arXiv:2309.00806 [v2], https://arxiv.org/abs/2309.00806. 10

[AJ15] N. R. Aravind and Pushkar S. Joglekar. On the expressive power of read-once determinants. In Adrian Kosowski and Igor Walukiewicz, editors, *Fundamentals of Computation Theory*, pages 95–105, Cham, 2015. Springer International Publishing. 3

[Bac90] Eric Bach. Explicit bounds for primality testing and related problems. *Mathematics of Computation*, 55(191):355–380, 1990. 5

[BBB+00] Amos Beimel, Francesco Bergadano, Nader H. Bshouty, Eyal Kushilevitz, and Stefano Varricchio. Learning functions represented as multiplicity automata. *J. ACM*, 47(3):506–530, May 2000. 2

[BHH92] Nader H. Bshouty, Thomas R. Hancock, and Lisa Hellerstein. Learning arithmetic read-once formulas. In *Proceedings of the Twenty-Fourth Annual ACM Symposium on Theory of Computing*, STOC '92, page 370–381, New York, NY, USA, 1992. Association for Computing Machinery. 2

[Bru18] Victor-Emmanuel Brunel. Learning signed determinantal point processes through the principal minor assignment problem. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 7376–7385, Red Hook, NY, USA, 2018. Curran Associates Inc. 4

[BT88] Michael Ben-Or and Prasoon Tiwari. A deterministic algorithm for sparse multivariate polynomial interpolation. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, STOC '88, page 301–309, New York, NY, USA, 1988. Association for Computing Machinery. 2

[BU24] Victor-Emmanuel Brunel and John Urschel. Recovering a magnitude-symmetric matrix from its principal minors. *Linear Algebra and its Applications*, 703:232–267, 2024. 4

[Bü00] Peter Bürgisser. Cook's versus valiant's hypothesis. *Theoretical Computer Science*, 235(1):71–88, 2000. 1

[CGGR25] Abhranil Chatterjee, Sumanta Ghosh, Rohit Gurjar, and Roshan Raj. Characterizing and testing principal minor equivalence of matrices. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*, STOC '25, page 1067–1078, New York, NY, USA, 2025. Association for Computing Machinery. 10, 11, 17, 23, 27

[DL78] Richard A. Demillo and Richard J. Lipton. A probabilistic remark on algebraic program testing. *Information Processing Letters*, 7(4):193 – 195, 1978. 18

[Edm67] Jack Edmonds. Systems of distinct representatives and linear algebra. *Journal of research of the National Bureau of Standards*, 71:241–245, 1967. 2

[Edm68] Jack Edmonds. Matroid partition. *Mathematics of the Decision Sciences*, 11:335–345, 1968. 2

[Edm79] Jack Edmonds. Matroid intersection. In P.L. Hammer, E.L. Johnson, and B.H. Korte, editors, *Discrete Optimization I*, volume 4 of *Annals of Discrete Mathematics*, pages 39–49. Elsevier, 1979. 2

[FK09] Lance Fortnow and Adam R. Klivans. Efficient learning algorithms yield circuit lower bounds. *Journal of Computer and System Sciences*, 75(1):27–36, 2009. Learning Theory 2006. 2

[Gan60] F. R. Gantmacher. *The Theory of Matrices, Volume I.* Chelsea Publishing Company, New York, 1960. Originally published in Russian; translated by K. A. Hirsch. 16

[Gee99] James F. Geelen. Maximum rank matrix completion. *Linear Algebra and its Applications*, 288:211–217, 1999. 2

[GT06] Kent Griffin and Michael J. Tsatsomeros. Principal minors, part ii: The principal minor assignment problem. *Linear Algebra and its Applications*, 419(1):125–171, 2006. 3, 4, 6

[GT17] Rohit Gurjar and Thomas Thierauf. Linear matroid intersection is in quasi-nc. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2017, page 821–830, New York, NY, USA, 2017. Association for Computing Machinery. 2, 8, 12, 13, 19

[HH91] Thomas Hancock and Lisa Hellerstein. Learning read-once formulas over fields and extended bases. In *Proceedings of the Fourth Annual Workshop on Computational Learning Theory*, COLT '91, page 326–336, San Francisco, CA, USA, 1991. Morgan Kaufmann Publishers Inc. 2

[HL84] D.J. Hartfiel and R. Leowy. On matrices having equal corresponding principal minors. *Linear Algebra and its Applications*, 58:147–167, 1984. 10, 11, 16, 17

[HS80] J. Heintz and C. P. Schnorr. Testing polynomials which are easy to compute (extended abstract). In *Proceedings of the Twelfth Annual ACM Symposium on Theory of Computing*, STOC '80, page 262–272, New York, NY, USA, 1980. Association for Computing Machinery. 1

[HS02] Olga Holtz and Hans Schneider. Open problems on gkk τ-matrices. *Linear Algebra and its Applications*, 345(1):263–267, 2002. 3

[IKQS15] Gábor Ivanyos, Marek Karpinski, Youming Qiao, and Miklos Santha. Generalized wong sequences and their applications to edmonds' problems. *Journal of Computer and System Sciences*, 81(7):1373–1386, 2015. 2

[IKS10] Gábor Ivanyos, Marek Karpinski, and Nitin Saxena. Deterministic polynomial time algorithms for matrix completion problems. *SIAM Journal on Computing*, 39(8):3736–3751, 2010. 2

[KI03] Valentine Kabanets and Russell Impagliazzo. Derandomizing polynomial identity tests means proving circuit lower bounds. In *Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing*, STOC '03, page 355–364, New York, NY, USA, 2003. Association for Computing Machinery. 1

[KS01] Adam R. Klivans and Daniel Spielman. Randomness efficient identity testing of multivariate polynomials. In *Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing*, STOC '01, page 216–223, New York, NY, USA, 2001. Association for Computing Machinery. 2

[KS06] Adam Klivans and Amir Shpilka. Learning restricted models of arithmetic circuits. *Theory of Computing*, 2(10):185–206, 2006. 2

[KT90] Erich Kaltofen and Barry M. Trager. Computing with polynomials given by black boxes for their evaluations: Greatest common divisors, factorization, separation of numerators and denominators. *Journal of Symbolic Computation*, 9(3):301–320, 1990. Computational algebraic complexity editorial. 37

[KT12] Alex Kulesza and Ben Taskar. Determinantal point processes for machine learning. *Foundations and Trends in Machine Learning*, 5(2-3):123–286, 12 2012. 4

[Loe86] Raphael Loewy. Principal minors and diagonal similarity of matrices. *Linear Algebra and its Applications*, 78:23–64, 1986. 10, 11

[Lov89] László Lovász. Singular spaces of matrices and their application in combinatorics. *Boletim da Sociedade Brasileira de Matemática - Bulletin/Brazilian Mathematical Society*, 20(1):87–99, 1989. 2

[Mur93] Kazuo Murota. Mixed matrices: Irreducibility and decomposition. In Richard A. Brualdi, Shmuel Friedland, and Victor Klee, editors, *Combinatorial and Graph-Theoretical Problems in Linear Algebra*, pages 39–71, New York, NY, 1993. Springer New York. 2

[MV18] Daniel Minahan and Ilya Volkovich. Complete derandomization of identity testing and reconstruction of read-once formulas. *ACM Trans. Comput. Theory*, 10(3), May 2018. 2

[MVV87] Ketan Mulmuley, Umesh V. Vazirani, and Vijay V. Vazirani. Matching is as easy as matrix inversion. In *Proceedings of the Nineteenth Annual ACM Symposium*

on Theory of Computing, STOC '87, page 345–354, New York, NY, USA, 1987. Association for Computing Machinery. 13

[Nis91] Noam Nisan. Lower bounds for non-commutative computation. In *Proceedings of the Twenty-Third Annual ACM Symposium on Theory of Computing*, STOC '91, page 410–418, New York, NY, USA, 1991. Association for Computing Machinery. 3

[NSV92] H. Narayanan, Huzur Saran, and Vijay V. Vazirani. Randomized parallel algorithms for matroid union and intersection, with applications to arboresences and edge-disjoint spanning trees. In *Proceedings of the Third Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '92, page 357–366, USA, 1992. Society for Industrial and Applied Mathematics. 13

[RKT15] Justin Rising, Alex Kulesza, and Ben Taskar. An efficient algorithm for the symmetric principal minor assignment problem. *Linear Algebra and its Applications*, 473:126–144, 2015. Special issue on Statistics. 3

[Sch80] Jacob T. Schwartz. Fast probabilistic algorithms for verification of polynomial identities. *Journal of the ACM*, 27(4):701–717, October 1980. 18

[SV14] Amir Shpilka and Ilya Volkovich. On reconstruction and testing of read-once formulas. *Theory of Computing*, 10(18):465–514, 2014. 2

[UBMR17] John Urschel, Victor-Emmanuel Brunel, Ankur Moitra, and Philippe Rigollet. Learning determinantal point processes with moments and cycles. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 3511–3520. JMLR.org, 2017. 4

[Val79] L. G. Valiant. Completeness classes in algebra. In *Proceedings of the Eleventh Annual ACM Symposium on Theory of Computing*, STOC '79, page 249–261, New York, NY, USA, 1979. Association for Computing Machinery. 1, 2, 3

[Vol16] Ilya Volkovich. A guide to learning arithmetic circuits. In Vitaly Feldman, Alexander Rakhlin, and Ohad Shamir, editors, *29th Annual Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pages 1540–1561, Columbia University, New York, New York, USA, 23–26 Jun 2016. PMLR. 2

[Zip79] Richard Zippel. Probabilistic algorithms for sparse polynomials. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation (EUROSAM)*, pages 216–226. Springer-Verlag, 1979. 18