Electronic Colloquium on Computational Complexity, Report No. 100 (2006)

On the Complexity of Rank and Rigidity

Meena Mahajan Jayalal Sarma M.N.

The Institute of Mathematical Sciences,

Chennai 600 113, India.

Email: {meena,jayalal}@imsc.res.in

July 17, 2006

Abstract

Given a matrix M over a ring $\mathbb{K}$, a target rank r and a bound k, we want to decide whether the rank of M can be brought down to below r by changing at most k entries of M. This is a decision version of the well-studied notion of matrix rigidity. We examine the complexity of the following related problems and obtain completeness results for small (counting logspace or smaller) classes: (a) computing the determinant, testing singularity, and computing the rank for matrices with special structure, (b) determining whether $k \in O(1)$ changes to a matrix suffice to bring its rank below a specified value, and (c) constructing a singular matrix *closest* (in a restricted sense) to the given matrix. We then consider bounded rigidity, where the magnitude of individual changes is bounded by a pre-specified value, and show NP hardness in general, and tighter bounds in special cases. We also extend the rigidity lower and upper bounds for the full-1s lower triangular matrices to full-1s extended lower triangular matrices, with a small gap between the two.

1 Introduction

Many problems in linear algebra fall in NC ; see for instance [All04]. However, corresponding optimization search problems can be considerably harder. We consider one such instance in this paper: matrix rank. Over any field, computing rank is known to be in NC [Mul87]. Now consider the following existential search question: Given a matrix M over a ring $\mathbb{K}$, a target rank r and a bound k, decide whether the rank of M can be brought down to below r by changing at most k entries of M. Intuitively, one would expect such a question to be in $\exists \cdot$ NC: guess k locations where M is to be changed, guess the new entries to be inserted there, and compute the rank. However, this intuition, while correct for finite fields, does not directly translate to a proof for $\mathbb{Q}$ and $\mathbb{Z}$, since the required new entries may not have representations polynomially-bounded in the input size. In fact, the best upper bound we can

ISSN 1433-8092

Matrix type (over $\mathbb{Q}$)	RANK BOUND	SINGULAR	DETERMINANT
general	$\mathsf{C_=L}$-complete	$\mathsf{C_=L}$-complete	GapL -complete
non-negative (even 0-1)	$\mathsf{C_=L}$-complete	$\mathsf{C_=L}$-complete	GapL -complete
symmetric non-negative	$\mathsf{C_=L}$-complete	$\mathsf{C_=L}$-complete	?
symmetric non-negative diagonally dominant (d.d.)	L-complete	L-complete	?
symmetric d.d.	L -hard even when det $\in \{0,1\}$		?
diagonal	TC^0-complete	AC^0	TC^0-complete
tridiagonal		$\mathsf{C_=NC}^1$	GapNC^1
tridiagonal non-negative	non-negative perm in planar # BWBP		

Table 1: RANK BOUND, SINGULAR, and DETERMINANT for special matrices

see in the general case is recursive enumerability. In this note, we explore the computational complexity of several variants of this problem.

The above question is directly related to the notion of rigidity of a matrix, which is the smallest value of k for which the answer is yes. The notion of rigidity was introduced by Valiant [Val77] and was independently proposed by Grigoriev [Gri76]. The main motivation for studying rigidity is that good lower bounds on rigidity give important complexity-theoretic results in other computational models, like linear algebraic circuits and communication complexity. Though the question we address is in fact a computational version of rigidity, it has no direct implications for these lower bounds. However, it provides natural complete problems based on linear algebra for important complexity classes.

Our question bears close resemblance to the body of problems considered under *matrix completion*, see for instance [BFS99, Lau01]. Given a matrix with indeterminates in some locations, can we instantiate them in such a way that some desired property (e.g. non-singularity) is achieved? Towards the end of this paper, we briefly discuss how results from matrix completion can yield upper bounds for our question.

In this paper, we restrict our attention to $\mathbb{Z}$ and $\mathbb{Q}$ (some extensions to finite fields are discussed at the end).

Since the computational rigidity question requires a fine understanding of how matrix rank varies with individual entries, we first consider the complexity of computing the rank for matrices with special structure. For general matrices, checking if the rank is at most r is known to be $\mathsf{C_=L}$-complete [ABO99]. We consider restrictions which are combinations of non-negativity, 0-1 entries, symmetry, diagonal dominance, tridiagonal and diagonal support, and we consider the complexities of three problems: computing the rank, computing the determinant and testing singularity. These, though intimately related, can have differing complexities, as Table 1 shows. One of the classes figuring there that needs special explanation is planar # BWBP . Branching programs as a computational model have been shown to be surprisingly powerful in the Boolean context; e.g. bounded-width branching programs (BWBP) capture NC^1, the class of languages with polynomial-size logarithmic depth circuits. However, in the arithmetic context, where we are interested in computing values rather than determining membership, they are not that well understood. It is still open ([All04, CMTV98]) whether the containment $\#\mathsf{BWBP} \subseteq \#\mathsf{NC}^1$ is in fact an equality. It is known that width-2 layered planar $\#\mathsf{BWBP}$ is at least as hard as NC^1 [AAB+99]. Our results concerning tridiagonal and diagonal matrices give a simpler proof of a weaker result: width-2 layered planar $\#\mathsf{BWBP}$ is at least as hard as TC^0 (constant-depth MAJORITY circuits).

Next we address the computational rigidity question. Since even an upper bound of NP is not obvious, we would like to restrict the choice available in changing matrix entries. We consider two variants:

1. In the input, a finite subset $S \subseteq \mathbb{K}$ is given. M has entries over S, and the changed entries must also be from S; rank computation continues to be over $\mathbb{K}$. (For instance, we may consider Boolean matrices, so $S = \{0, 1\}$, while rank computation is over $\mathbb{Z}$.) It is easy to see that this variant is indeed in NP, and in NC if $\mathbb{K}$ is a field and $k \in O(1)$.

2. In the input, a bound θ is given. We require that the changes be bounded by θ; we may apply the bound to each change, or to the total change, or to the total change per row/column. (See for instance [Lok95].) This version has close connections with another well-studied area called linear interval equations which arises naturally in the context of control systems theory (see [Roh96]).

We obtain tighter lower and upper bounds for some of these questions. We obtain a completeness result of $\mathsf{C_=L}$ when $k \in O(1)$ in the first variant, of NP when $r = n$ in the second variant, and of $\mathsf{C_=L}$ when $r = n$ in the general case. The table below summarizes the results.

$\mathbb{K}$, $S \subset \mathbb{K}$ (if $-$, then $S = \mathbb{K}$)	restriction	bound
$\mathbb{Z}$ or $\mathbb{Q}$, {0,1}		in NP
$\mathbb{Z}$ or $\mathbb{Q}$, {0,1}	$k \in O(1)$	$\mathsf{C_=L}$-complete
$\mathbb{Z}$ or $\mathbb{Q}$	$k \in O(1)$	$\mathsf{C_=L}$-hard
$\mathbb{Q}$	$r = n$	$\mathsf{C_=L}$-complete witness-search in $\mathsf{L}^{\mathsf{GapL}}$
$\mathbb{Z}$	$r = n$ and $k = 1$	in $\mathsf{L}^{\mathsf{GapL}}$
$\mathbb{Z}$ or $\mathbb{Q}$	bounded rigidity	NP-hard
$\mathbb{Z}$ or $\mathbb{Q}$	bounded rigidity, $r = n$	NP-complete
$\mathbb{Z}$ or $\mathbb{Q}$	bounded rigidity, $r = n, k = 1$	In PL, and $\mathsf{C_=L}$-hard

Table 2: Bounds on RIGID when $k \in O(1)$ or $r = n$

Obtaining explicit bounds on the rigidity of special matrices is surprisingly elusive, and thus has received a lot of attention. A rare case where a closed-form expression has been obtained for rigidity is full-1s lower triangular matrices [PV91]. (The rareness of matching, or even close, lower and upper bounds, correlates well with the lack of upper bounds on the computational version of rigidity.) In Section 6 we consider an extension of this result to full-1s extended lower triangular (elt) matrices (the first diagonal above the main diagonal can be non-zero). It is worthwhile noting that this is not as restrictive as it sounds: it is known that determinant/permanent computation of elt matrices is as hard as the general case, see [AAM03, Li92]. Even with this small extension in the input structure, we cannot obtain a closed-form expression; however, we show lower and upper bounds differing by a small additive factor.

2 Preliminaries

The rank of a square matrix (we consider only square matrices in this paper) is the maximum number of linearly dependent rows or columns in it. For an integer matrix, its rank is the same whether computed over $\mathbb{Z}$ or $\mathbb{Q}$.

L and NL denote languages accepted by deterministic and nondeterministic logspace classes respectively, and FL is the class of logspace-computable functions. #L is the class of functions that count the number of accepting paths of an NL machine. GapL is the class of functions that count the difference between the number of accepting and rejecting paths of an NL machine. Computing the determinant over $\mathbb{Z}$ or $\mathbb{Q}$ is complete for GapL. In contrast, computing the permanent is complete for # P, the class of functions counting accepting paths of an NP machine.

The exact counting logspace class $\mathsf{C_=L}$ bears the same relationship to GapL as NL to # L; a language L is in $\mathsf{C_=L}$ iff it consists of exactly those strings where a certain GapL function is zero. The languages

$$\textsc{singular}(\mathbb{K}) = \{M \mid \text{ Over } \mathbb{K}, M \text{ is not full rank}\}$$

$$\textsc{rank bound}(\mathbb{K}) = \{(M, r) \mid \text{ Over } \mathbb{K}, \text{rank}(M) < r\}$$

for $\mathbb{K} = \mathbb{Z}$ or $\mathbb{Q}$ are complete for $\mathsf{C_=L}$ [ABO99]. (Note that for any type of matrices, and any complexity class $\mathcal{C}$, $\mathcal{C}$-hardness of SINGULAR implies $\mathcal{C}$-hardness of RANK BOUND.)

We consider the following circuit classes. NC^1 is the class of languages with polynomial size logarithmic depth Boolean circuits. $\#\mathsf{NC}^1$ is the class of functions computed by similar arithmetic circuits (gates compute $+$ and $\times$.) AC^0 (TC^0) is the class of languages with polynomial size constant depth unbounded fanin Boolean circuits, where gates compute AND, OR, NOT (and MAJORITY). For more details, see [Vol99].

The rigidity function, and its decision version, are as defined below[1]. (Here $\text{support}(N) = \#\{(i,j) \mid N(i,j) \neq 0\}$.) Lemma 1, first observed by Valiant, is folklore.

$$R_M(r) \stackrel{def}{=} \inf_N \ \{\text{support}(N) : rank(M+N) < r\}$$
$$\textsc{rigid}_{\mathbb{K}} = \{(M, r, k) \mid R_M(r) \leq k\}$$

[1]In much of the rigidity literature, $\text{rank}(M+N) \leq r$ is required. We use strict inequality to be consistent with the definition of RANK BOUND from [ABO99].

Lemma 1. *Over any field $\mathbb{F}$, $R_M(r+1) \leq (n-r)^2$.*

3 Computing the rank for special matrices

Computation of rank is intimately related to computation of the determinant. Mulmuley [Mul87] showed that over arbitrary fields (and also over $\mathbb{Z}$), rank can be computed in NC, with the primitives being the field operations.

The following is easy to see: We include a proof for completeness.

Proposition 2. *The languages* RANK BOUND*($\mathbb{Z}$) and* SINGULAR*($\mathbb{Z}$) remain $\mathsf{C_=L}$-hard even if the instances are restricted to be symmetric 0-1 matrices.*

Proof. Let A' be the symmetric matrix $\begin{bmatrix} 0 & A \\ A^T & 0 \end{bmatrix}$. Since $\text{rank}(A') = 2(\text{rank}(A))$, RANK BOUND($\mathbb{Z}$) remains $\mathsf{C_=L}$-hard when restricted to symmetric matrices. Further, DETERMINANT remains GapL hard even the matrices are restricted to be 0-1 (see for instance [Tod91]). Thus SINGULAR remains $\mathsf{C_=L}$-hard even when restricted to 0-1 matrices. Since M is in SINGULAR if and only if (M, n) is in RANK BOUND if and only if $(M', 2n)$ is in RANK BOUND, it follows that RANK BOUND($\mathbb{Z}$) remains $\mathsf{C_=L}$-hard for symmetric 0-1 matrices as well. $\square$

However, we do not know similar hardness for DETERMINANT. While it remains GapL hard for 0-1 matrices, it is not clear that there are GapL -hard symmetric instances.

We now consider an additional restriction. A matrix M is said to be *diagonally dominant* if for every i, $|m_{i,i}| \geq \sum_{j \neq i} |m_{i,j}|$. (If all the inequalities are strict, then M is said to be *strictly diagonally dominant.*) We show:

Theorem 3. SINGULAR*($\mathbb{Z}$) and* RANK BOUND*($\mathbb{Z}$) restricted to non-negative diagonally dominant symmetric matrices are L-complete. The hardness is via uniform TC^0-computable many-one reductions.*

Proof. To show this, we exploit a very nice connection between such matrices and certain associated graphs defined as follows. For a non-negative symmetric diagonally-dominant matrix M, the *support graph* G_M is the undirected graph $G_M = (V, E_M)$ where $V = \{v_1, \ldots v_n\}$, and $E_M = \{(v_i, v_j) \mid i \neq j, \quad m_{i,j} > 0\} \cup \{(v_i, v_i) \mid m_{i,i} > \sum_{i \neq j} m_{i,j}\}$ The following is shown in [Dah99] for $\mathbb{R}$, and clearly also holds for $\mathbb{Q}$.

Lemma 4 ([Dah99]). *Let M be a non-negative symmetric diagonally dominant matrix of order n over $\mathbb{Q}$ or $\mathbb{R}$. Then rank$(M) = n - c$, where c is the number of bipartite components in the support graph G_M.*

Note: the presence of a self-loop means a component is non-bipartite.

Membership in L : Now, given a matrix M satisfying the stated conditions, it is straightforward to construct the support graph G_M. By [AG00, NTS95, Rei05], checking whether two vertices belong to the same component in an undirected graph, counting the number of components, and checking bipartiteness of a named component are all in L. Hence, by Lemma 4, rank(M) can be computed in L.

Hardness: The reduction is from undirected forest accessibility UFA, which is L-complete and remains L-hard even when the graph has exactly 2 components [CM87].

Let G, s, t be an instance of UFA, where G has two trees. We construct a new graph $G' = (V', E')$ as follows: take two disjoint copies of G. Add two new vertices u and v with self-loops. Connect u to both copies of t and v to both copies of s.

If there is an $s \rightsquigarrow t$ path ρ in G, then G' has three components: the copies of the component containing s and t join up, while the copies of the other component remain disconnected (and hence bipartite). The new joined up component is not bipartite due to the loops at u and v. Hence G' has exactly two bipartite components.

If there is no $s \rightsquigarrow t$ path in G, then G' has two components: the copies of the component containing s join up, as also the copies of the component containing t. Both components are non-bipartite due to the loops at u and v.

To complete the proof, we need to produce a matrix M such that G' is its support graph. We construct M as follows:

$$\text{For each } i \neq j \qquad m_{i,j} = \begin{cases} 1 & \text{if } (i,j) \in E' \\ 0 & \text{otherwise} \end{cases}$$

$$\text{For each } i \qquad m_{i,i} = \begin{cases} 1 + \sum_{j \neq i} m_{i,j} & \text{if } (i,i) \in E' \\ \sum_{j \neq i} m_{i,j} & \text{otherwise} \end{cases}$$

M can be constructed from G by a uniform TC^0 circuit. From Lemma 4, M is singular if and only if there is an $s \rightsquigarrow t$ path in G. ☐

Note that in the above construction, $s \rightsquigarrow_G t \implies \text{rank}(M) = n - 2$, and $s \not\rightsquigarrow_G t \implies \text{rank}(M) = n$. By omitting the self-loop at v, we get one bipartite component when there is no s, t path in G, and two if there is a path. Thus M is necessarily singular, and testing whether $(M, n-1)$ is in RANK BOUND($\mathbb{Q}$) is L-hard. This shows that the hardness of RANK BOUND for such matrices is not derived from the hardness of testing singularity alone.

Also note that though rank for these matrices can be computed in L, we do not know how to compute the exact value of the determinant itself. In a brief digression, we note the following: if a matrix is to have no trivial (all-zero) rows, and yet be diagonally dominant, then it cannot have any zeroes on the diagonal. How restrictive is this requirement? In general, it isn't too much so, as we show below. However, we do not know of a many-one reduction.

Lemma 5. *For every* GapL *function f and every input x, $f(x)$ can be expressed as* $\det(M) - 1$, *where M has no zeroes on the diagonal. Further, M can be obtained from x via projections.*

Proof. Consider Toda's proof [Tod91] for showing that DETERMINANT is GapL hard (see also [ABO99, MV97]). Given any GapL function f and input x, it constructs a directed graph G with self-loops at every vertex except a special vertex s. G also has the property that every non-trivial cycle (not a self-loop) in G passes through s. If A is the adjacency matrix of G, then the construction satisfies $f(x) = \det(A)$. Now consider the matrix B obtained by adding a self-loop at s. What additional terms does $\det(B)$ have that were absent in $\det(A)$? Such terms must correspond to cycle covers using the self-loop at s; i.e. cycle covers in $G \setminus \{s\}$. But $G \setminus \{s\}$ has no non-trivial cycles, so the only additional cycle cover is all self-loops, contributing a 1. Thus $\det(B) = 1 + \det(A)$, and B is the required matrix. $\square$

In Theorem 3, if we relax the condition of non-negativity, then the hardness of course continues to hold (but we do not know how to show membership in L). Via a somewhat different reduction, we show that for such matrices, L -hardness of SINGULAR holds even for matrices whose determinant is known to be in $\{0, 1\}$.

Theorem 6. SINGULAR*($\mathbb{Z}$) for symmetric diagonally dominant matrices is* L *-hard, even when restricted to instances with 0-or-1 determinant.*

Proof. As in the proof of Theorem 3, we begin with an instance (G, s, t) of UFA where G has exactly two components. Add edge (s, t) to obtain graph H. By the matrix-tree theorem,if A is the Laplacian matrix of H (defined below), and B is obtained by deleting the topmost row and leftmost column of A, then $\det(B)$ equals the number of spanning trees of H.

$$\begin{aligned} a_{i,i} = &\quad \text{the degree of vertex } i \text{ in } H \\ a_{i,j} = &\quad -1 \text{ if } i \neq j \text{ and } (i,j) \text{ is an edge in } H \\ a_{i,j} = &\quad 0 \text{ if } i \neq j \text{ and } (i,j) \text{ is not an edge in } H \end{aligned}$$

Clearly, A is diagonally dominant (in fact, for each i, the constraint is an equality); also, since H is an undirected graph, A is symmetric.

Now the number of spanning trees in H is 1 if $s \not\rightsquigarrow_G t$ (H itself is a tree) and is 0 if $s \rightsquigarrow_G t$ (H still has two components). $\square$

Continuing further along restricting matrices, we consider the simplest form of the matrices considered in Theorem 3, namely non-negative diagonal matrices. Clearly, the rank is now the number of non-zero entries. Checking whether an entry is zero can be done by a single AND gate which looks at the negated literals in that entry. Since polylog thresholds are in AC^0 [RW91], it follows that not just singularity, but also instances (M, r) of RANK BOUND where r is within a polylog additive (subtractive) factor of 0 (or n, respectively) are in fact in AC^0. RANK BOUND($\mathbb{Z}$) itself, for such matrices, is in TC^0. Also, the determinant can be computed in TC^0 since it merely involves iterated multiplication. On the other hand, an instance $x_1 \dots x_n$ of the TC^0-complete problem co -MAJORITY can be written as the instance $(D(x), n/2)$ of RANK BOUND($\mathbb{Z}$). ($D(x)$ is the matrix obtained by placing the vector x on the diagonal and placing zeroes elsewhere.) Similarly, an instance $a_1, \dots, a_n$ of iterated multiplication (n n-bit numbers) can be recast as such a determinant by placing the numbers on the diagonal. Thus

Theorem 7. RANK BOUND*($\mathbb{Z}$) and* DETERMINANT*, restricted to diagonal matrices, are* TC^0*-complete. The hardness does not require negative entries.*

This is another instance where RANK BOUND does not derive its hardness from the singularity threshold; it is in fact (provably) harder than SINGULAR. (The first instance is in the Note after Theorem 3; however, in that case, SINGULAR is also equally hard.)

Another restriction we consider is tridiagonal matrices: $m_{i,j} \neq 0 \implies |i-j| \leq 1$. While we do not know the complexity of RANK BOUND or SINGULAR here, we can show that DETERMINANT and PERMANENT are in GapNC^1. It is known that $\#\mathsf{BWBP} \subseteq \#\mathsf{NC}^1 \subseteq \mathsf{GapNC}^1 = \mathsf{GapBWBP}$, see [CMTV98]. We obtain bounded width branching programs which are also layered planar, and hence are exactly the G-graphs referred to in [AAB+99]. Counting paths in G-graphs may be simpler than GapNC^1 due to planarity. However [AAB+99] (see also [All04]) shows that it is hard for NC^1.

Theorem 8. *Computing the permanent of a non-negative tridiagonal matrix over $\mathbb{Z}$ can be reduced to counting paths in a planar* BWBP *. The permanent and determinant of a tridiagonal matrix over $\mathbb{Z}$ can be computed in* GapNC^1*.*

Proof. Let A_i be the top-left submatrix of A of order i, and let X_i and Y_i denote its permanent and determinant respectively. We have the following recurrences:

$$\begin{array}{ll} X_0 = Y_0 = 1 & X_1 = Y_1 = a_{1,1} \\ X_i = a_{i,i}X_{i-1} + a_{i-1,i}a_{i,i-1}X_{i-2} & Y_i = a_{i,i}Y_{i-1} - a_{i-1,i}a_{i,i-1}Y_{i-2} \end{array}$$

When all entries are 0-1, then the branching program for X_n has width 2 and can be drawn in a layered planar fashion (see Figure 1). When the entries are larger, each edge here can be replaced by a width-3 gadget with the appropriate number of paths in a standard way, giving width 5. If there are negative entries, or when we are computing the determinant, we need to either allow negative weights, or double the width and lose planarity.

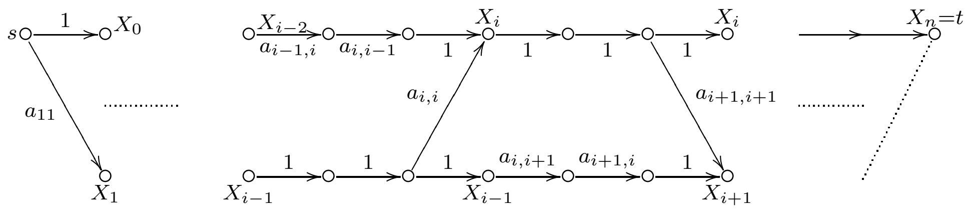

Figure 1: Width-2 branching program for tridiagonal 0-1 permanent

∎

4 Complexity results on rigidity

We now study the decision version of rigidity $\text{RIGID}_{\mathbb{K}}$, and also its restriction $\text{RIGID}_{\mathbb{K},S}$ defined below, where the matrices can have entries only from $S \subseteq \mathbb{K}$.

$$\text{RIGID}_{\mathbb{K},S} = \left\{ (M,r,k) \mid \begin{array}{c} M \text{ over } S, \ \exists M' \text{ over S}: \\ \text{rank}(M') < r \ \wedge \ \text{support}(M - M') \leq k \end{array} \right\}$$

We will mostly consider S to be either all of $\mathbb{K}$, or only $\mathbb{B} = \{0,1\}$. We also consider the complexity of RIGID when k is fixed, via the following language:

$$\text{RIGID}_{\mathbb{K},S}(k) = \{(M,r) \mid (M,r,k) \in \text{RIGID}_{\mathbb{K},S}\}$$

The language $\text{RIGID}_{\mathbb{Z}}(0)$ is nothing but RANK BOUND($\mathbb{Z}$), and hence by [ABO99] is complete for $\mathsf{C_=L}$. When $k > 0$, we can still obtain some bounds provided S is finite. We have the following completeness result for one such case.

Theorem 9. *For each k, $\text{RIGID}_{\mathbb{Z},\mathbb{B}}(k)$ is complete for $\mathsf{C_=L}$.*

Proof. **Membership:** We show that for each k, $\text{RIGID}_{\mathbb{Z},\mathbb{B}}(k)$ is in $\mathsf{C_=L}$. An instance (M,r) is in $\text{RIGID}_{\mathbb{Z},\mathbb{B}}(k)$ if there is a set of $0 \leq s \leq k$ entries of M, which, when flipped, yield a matrix of rank below r. The number of such sets is bounded by $\Sigma_{s=0}^{k}\binom{n}{s} = t \in n^{O(1)}$. Let the corresponding matrices be denoted $M_1, M_2 \dots M_t$; these can be generated from M in logspace. Now $(M,r) \in \text{RIGID}_{\mathbb{Z},\mathbb{B}}(k) \iff \exists i : (M_i, r) \in \text{RANK BOUND}(\mathbb{Z})$. Hence $\text{RIGID}_{\mathbb{Z},\mathbb{B}}(k) \leq_{dtt}^{\log} \text{RANK BOUND}(\mathbb{Z})$. Since RANK BOUND($\mathbb{Z}$) is in $\mathsf{C_=L}$, and since $\mathsf{C_=L}$ is closed under logspace disjunctive truth-table reductions [AO96], it follows that $\text{RIGID}_{\mathbb{Z},\mathbb{B}}(k)$ is in $\mathsf{C_=L}$.
Hardness: To show a corresponding hardness result, we first need the following lemma, which is folklore. we put down a self-contained proof here for completeness.

Lemma 10. *Over any field $\mathbb{F}$, for any two matrices M and N of the same order,*

$$\textit{support}(M - N) = 1 \Longrightarrow |\textit{rank}(M) - \textit{rank}(N)| \leq 1$$

Proof. Let $\text{rank}(M) = r$ and let V be a set of r linearly independent row vectors of M. Let M and N differ at position (i,j). There are two possibilities.
Case ($v_i \notin V$): Changing v_i does not change V, so $\text{rank}(N) \geq k$. The change could leave v_i dependent on V, or make it independent; the dependence of other rows on V is unaffected. So $\text{rank}(N) \leq k+1$.
Case ($v_i \in V$): If all row vectors outside V are spanned by $V \setminus \{v_i\}$, then changing v_i cannot increase rank, and can decrease it by at most 1. So $k - 1 \leq \text{rank}(N) \leq k$. If there is a row vector v outside V that is independent of $V \setminus \{v_i\}$, then $V' = (V \cup \{v\}) \setminus \{v_i\}$ spans all rows outside V'. (This property requires a field as we may need to invert an element to make V' a basis.) Now, $v_i \notin V'$, so by the previous case, $k \leq \text{rank}(N) \leq k+1$. ☐

Though this Lemma may not hold for rings in general, it does hold for $\mathbb{Z}$.

The hardness for $\text{RIGID}_{\mathbb{Z},\mathbb{B}}(0)$ holds because SINGULAR remains $\mathsf{C_=L}$-hard even when restricted to 0-1 matrices. Hardness for all the languages mentioned in the lemma also follows from this fact, and from the following claim:

$$\begin{aligned}
&(1)\ M \in \text{SINGULAR}(\mathbb{Z}) &\Longrightarrow\ & (M \otimes I_{k+1}, n(k+1) - k) \in \text{RIGID}_{\mathbb{Z},\mathbb{B}}(0) \subseteq \text{RIGID}_{\mathbb{Z},\mathbb{B}}(k) \\
&(2)\ M \notin \text{SINGULAR}(\mathbb{Z}) &\Longrightarrow\ & (M \otimes I_{k+1}, n(k+1) - k) \notin \text{RIGID}_{\mathbb{Z}}(k)
\end{aligned}$$

Here $\otimes$ denotes tensor product. Note that $\text{rank}(M \otimes I_{k+1}) = (k+1)\text{rank}(M)$. To see the claim, observe that if $M \in \text{SINGULAR}(\mathbb{Z})$, then $\text{rank}(M) \leq n-1$ and so $\text{rank}(M \otimes I_{k+1}) \leq (k+1)(n-1) < n(k+1) - k$. If $M \notin \text{SINGULAR}(\mathbb{Z})$, then $\text{rank}(M \otimes I_{k+1}) = n(k+1)$. Thus we want to reduce the rank by at least $k+1$. By Lemma 10, we need to change at least $k+1$ entries. $\square$

Observe that this result also holds for any finite S, even if S is not fixed a priori but supplied explicitly as part of the input. The $\leq_{dtt}^{\log}$ reduction will range over, for each choice of s changes as described above, a set of s choices from S, and produce the corresponding matrix. (For implicit representations of S, this may not work, even if S is finite, since we need the changed entries to have polynomial sized representation in terms of the input size.)

The hardness results above were obtained essentially by exploiting the hardness of testing singularity. Therefore we now consider the complexity of RIGID at the singular-vs-non-singular threshold, i.e. when $r = n$.

From Lemma 1 we know that over any field $\mathbb{F}$, (M, n, k) is in RIGID whenever $k \geq 1$. And $(M, n, 0)$ is in RIGID if and only if $M \in \text{SINGULAR}(\mathbb{F})$. So the complexity of deciding this predicate over $\mathbb{Q}$ is already well understood. We then address the question of how difficult it is to come up with a witnessing matrix.

Theorem 11. *Given a non-singular matrix M over $\mathbb{Q}$, a singular matrix N satisfying support$(M - N) = 1$ can be constructed in L^{GapL}.*

Proof. For each (i, j), let $M(i, j)$ be the matrix obtained from M by replacing $m_{i,j}$ with an indeterminate x. Then $\det(M(i, j))$ is of the form $ax + b$, and a and b can be determined in GapL (see for instance [AAM03]). Since $R_M(n) = 1$, there is at least one position (i, j) where the determinant is sensitive to the entry, and hence $a \neq 0$. Setting $m_{i,j}$ to be $-b/a$ gives the desired N. $\square$

Another question that arises naturally is the complexity of RIGID at the singularity threshold over rings. Note that Lemma 1 does not necessarily hold for rings. For instance, changing one entry of a non-singular rational matrix M suffices to make it singular. But even if M is integral, the changed matrix may not be integral, and over $\mathbb{Z}$, $R_M(n)$ may well exceed 1. (It does, for the matrix $\begin{bmatrix} 2 & 3 \\ 5 & 7 \end{bmatrix}$.) Thus, the question of deciding $R_M(n)$ over $\mathbb{Z}$ is non-trivial. We show:

Theorem 12. *Given $M \in \mathbb{Z}^{n \times n}$, deciding if (M, n, k) is in RIGID$(\mathbb{Z})$ is (1) trivial for $k \geq n$, (2) $\mathsf{C_=L}$ complete for $k = 0$, and (3) in $\mathsf{L}^{\mathsf{GapL}}$ for $k = 1$.*

Proof. The first part holds because zeroing out an entire row always gets singularity. The second part merely says that $\text{SINGULAR}(\mathbb{Z})$ is $\mathsf{C_=L}$-complete. The third part follows from the proof of Theorem 11 and additionally checking the integrality of b/a. $\square$

In particular, the third result in this theorem implies that if over $\mathbb{Z}$, $R_M(n) = 1$ for a non-singular matrix M, and if N is the witnessing matrix, then the single non-zero entry in N has size polynomially bounded in that of M. However, if $R_M(n) > 1$ we do not know such a size bound.

5 Computing Bounded Rigidity

We now consider the bounded norm variant of rigidity described in Section 1. Namely, changed matrix entries can differ from the original entries by at most a pre-specified amount θ. Note that over $\mathbb{Q}$, this still does not imply an a priori polynomial-size bound on the changed entries.

First, we formally define the functions / languages of interest. The *norm rigidity* $\Delta_M(r)$ and *bounded rigidity* $R_M(r,\theta)$,as defined in [Lok95], and the decision version, are

$$\Delta_M(r) \stackrel{def}{=} \inf_N \left\{ \sum_{i,j} |n_{i,j}|^2 : rank(M+N) < r \right\}$$

$$R_M(r,\theta) \stackrel{def}{=} \inf_N \ \{support(N) : rank(M+N) < r, \forall i,j : |n_{i,j}| \leq \theta\}$$

$$\text{B-RIGID}_{\mathbb{K}} = \{(M,r,k,\theta) \mid R_M(r,\theta) \leq k\}$$

The following lemma shows that the bounded rigidity functions can behave very differently from the standard rigidity function.

Lemma 13. *For any $\epsilon \geq 0$, any n, and any (sufficiently large) m, there is an $n \times n$ matrix with m-bit entries such that $R_M(n) = 1$, $\Delta_M(n) = \Theta(4^m)$, and the bounded rigidity $R_M(n, m^\epsilon)$ is undefined.*

Proof. Let M be an $n \times n$ diagonal matrix M with $m_{i,i} = 2^m$ and $m_{i,j} = 0$ for $i \neq j$. Clearly, $R_M(n) = 1$; just zero out any diagonal entry. This involves a norm change of 4^m. Can M be made singular by a smaller norm-change, even allowing more entries to be changed? Recall the definition of strict diagonal dominance from Section 3. We invoke the Levy-Desplanques theorem (see for instance Theorem 2.1 in [MM64]) that says that the determinant of a strictly diagonally dominant matrix is non-zero. Now, a total norm-change less than 4^m will not destroy strict diagonally dominance, and the matrix will remain non-singular. Hence $\Delta_M(n) = 4^m$, and $R_M(n, m^\epsilon)$ is undefined. ☐

Thus there are cases when $R_M(r,\theta)$ is undefined. This motivates the following question. Given a matrix M, a rank r and θ how difficult is it to check whether $R_M(r,\theta)$ is defined? We show the following:

Theorem 14. *Given a matrix $M \in \mathbb{Q}^{n\times n}$, and a rational number $\theta > 0$,*

1. *Testing if $R_M(n,\theta)$ is defined is* NP*-complete.*

2. *Given k, testing if $R_M(n,\theta)$ is at most k is* NP*-complete.*

(In other words, deciding $\text{B-RIGID}_{\mathbb{K}}$ is NP -hard, even when $r = n$ and $k = n^2$, and is in NP when $r = n$.)

Proof. To begin with, notice that, $R_M(r,\theta)$ is defined if and only if $R_M(r,\theta) \le n^2$. Therefore it suffices to show hardness for (1) and an NP algorithm for (2).

We use notation and some results from the *linear interval equations* literature. For two matrices A and B, we say that $A \le B$ if for each i,j, $A_{ij} \le B_{ij}$. For $A \le B$, the interval of matrices $[A, B]$ is the set of all matrices C such that $A \le C \le B$. An interval is said to be singular if it contains at least one singular matrix; otherwise it is said to be regular.

Membership: Theorem 2.8 of [PR93] establishes that checking singularity of a given interval matrix is in NP.

Now given M, θ and k, we want to test whether $R_M(n,\theta)$ is at most k. In NP, we guess k positions $(p_1,q_1),(p_2,q_2),\ldots(p_k,q_k)$ and define $\Delta_{p_m q_m} = \theta$ for all $1 \le m \le k$ and 0 elsewhere. Now let $\underline{A} = M - \Delta$ and $\overline{A} = M + \Delta$. Then the interval $[\underline{A}, \overline{A}]$ is singular iff $R_M(n,\theta) \le k$, and this singularity can be tested in NP.

Hardness: We start with the *maximum bipartite subgraph problem*: Given an undirected graph $G = (V,E)$, with n vertices and m edges and a number k, check whether there is bipartite subgraph with at least k edges. This problem is known to be NP-complete (see [GJ79]). In [PR93], there is a reduction from this problem to computing the *radius of non-singularity*, defined as follows: Given a matrix A, its radius of non-singularity $d(A)$ is the minimum $\epsilon > 0$ such that the interval $[A - \epsilon J, A + \epsilon J]$ is singular, where J is the all-1s matrix. We sketch the reduction of [PR93] below and observe that it yields NP-hardness for our problem as well.

Given an instance G, k of the maximum bipartite subgraph problem, we define the matrix N as,

$$N_{ij} = \begin{cases} -1 & \text{if } i \ne j \text{ and } i \text{ and } j \text{ are adjacent in } G \\ 2m+1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

Notice that since N is diagonally dominant, by Levy-Desplanques theorem (see for instance Theorem 2.1 in [MM64]), N is invertible. Let $M = N^{-1}$.

By Theorems 2.6 and 2.2 of [PR93],

$$\begin{aligned} (G,k) \text{ is a Yes instance} &\iff 1/d(M) \ge (2m+1)n + 4k - 2m \\ &\iff d(M) \le \theta = \tfrac{1}{(2m+1)n+4k-2m} \\ &\iff \text{the interval } [M - \theta J, M + \theta J] \text{ is singular} \\ &\iff R_M(n,\theta) \text{ is defined.} \end{aligned}$$

In the appendix, we give a self-contained proof of these implications, specialising the proofs of Theorems 2.1, 2.2 and 2.6 from [PR93] to this case. $\square$

Unravelling the NP algorithm described in the membership part above, and its proof of correctness, is illuminating. Essentially, what is established in [Roh94] and used in [PR93] is the following:

Lemma 15 ([Roh94]). *If an interval $[A, B]$ is singular, i.e. the determinant vanishes for some matrix C within the bounds $A \le C \le B$, then the determinant vanishes for a matrix $D \in [A, B]$ which, at all but at most one position, takes an extreme value (d_{ij} is either a_{ij} or b_{ij}).*

In particular, this implies that there is a matrix in the interval whose entries have representations polynomially long in that of A and B. This is because let D be the matrix claimed to exist as above, and let k, l be the (only) position where $a_{kl} < d_{kl} < b_{kl}$. The other entries of D match those of A or B and hence are polynomially bounded anyway. Now put a variable x at k, l to get matrix D_x. Its determinant is a univariate linear polynomial $\alpha x + \beta$ which vanishes at $x = d_{kl}$. Now α and β can be computed from D_x in GapL , and hence are polynomially bounded. If $\alpha = 0$, then $\beta = 0$ and the polynomial is identically zero. Otherwise, the zero of the polynomial is $-\beta/\alpha$. Either way, there is a zero with a polynomially long representation.

In [Roh94], the above lemma is established as part of a long chain of equivalences concerning determinant polynomials. However, it is in fact a general property of arbitrary multilinear polynomials, as we show below.

Lemma 16 (Zero-on-an-Edge Lemma). *Let $p(x_1 \dots x_t)$ be a multilinear polynomial over $\mathbb{Q}$. If it has a zero in the hypercube H defined by $[\ell_1, u_1], \dots [\ell_t, u_t]$, then it has a zero on an edge of H, i.e. a zero $(a_1, \dots, a_t)$ such that for some k, $\forall (i \neq k)$, $a_i \in \{\ell_i, u_i\}$.*

Proof. The proof is by induction on the dimension of the hypercube. The case when $t = 1$ is vacuously true, since H is itself an edge. Consider the case $t = 2$. Let $p(x_1, x_2)$ be the multilinear polynomial which has a zero (z_1, z_2) in the hypercube H; $\ell_i \leq z_i \leq u_i$ for $i = 1, 2$. Assume, to the contrary, that p has no zero on any edge of H. Define the univariate polynomial $q(x_1) = p(x_1, z_2)$. Since $q(x_1)$ is linear and vanishes at z_1, $p(\ell_1, z_2)$ and $p(u_1, z_2)$ must be of opposite sign. But the univariate linear polynomials $p(\ell_1, x_2)$ and $p(u_1, x_2)$ do not change signs on the edges either, and so $p(\ell_1, u_2)$ and $p(u_1, u_2)$ also have opposite sign. By linearity of $p(x_1, u_2)$, there must be a zero on the edge $x_2 = u_2$, contradicting our assumption.

Let us assume the statement for hypercubes of dimension less than t. Consider the hypercube of dimension t and the polynomial $p(x_1, \dots x_t)$. Let $(z_1 \dots z_t)$ be the zero inside the hypercube. The multilinear polynomial r corresponding to $p(x_1, \dots x_{n-1}, z_t)$ has a zero inside the $(t-1)$-dimensional hypecube H' defined by intervals $[\ell_1, u_1], \dots [\ell_{t-1}, u_{t-1}]$. By induction, r has a zero on an edge of H'. Without loss of generality, assume that this zero is $(z'_1, \alpha_2 \dots \alpha_{t-1})$ where $\alpha_i \in \{\ell_i, u_i\}$. Thus the polynomial $q(x_1, x_t) = p(x_1, \alpha_2 \dots \alpha_{t-1}, x_t)$ has a zero in the hypercube defined by intervals $[\ell_1, u_1], [\ell_t, u_t]$. Hence the base case applies again, completing the induction. $\square$

The hard instance that we get in Theorem 14 is a matrix with a rational entries and the bound θ is also a rational number. If M is such a matrix, we can produce a matrix N with the same rank by multiplying each entry by ℓ where ℓ is the lcm of the denominators of the entries. $R_M(r, \theta) = R_N(r, \ell\theta)$. Thus, theorem 14 hold for integer matrices too.

The matrices that are produced in the above reduction are all symmetric. Rohn [Roh94] considered the case when the interval of matrices under consideration is symmetric; that is both the boundary matrices are symmetric. Notice that the interval can still contain non-symmetric matrices. He proved that in such an interval, if there is a singular matrix, then there must be a symmetric singular matrix too.

Analogous to Theorem 9, we consider the complexity of $\textsc{b-rigid}_{\mathbb{K}}$ when $k \in O(1)$.

Theorem 17. B-RIGID$_{\mathbb{K}}$ *is* $\mathsf{C_=L}$*-hard for each fixed choice of* k*, and remains hard when* r *is restricted to be* n*. When* $k = 1$ *and* $r = n$*, it is in* PL *.*

Proof. For any k, $(M, n, k, 0) \in$ B-RIGID$_{\mathbb{K}}$ $\iff$ M is singular; hence $\mathsf{C_=L}$-hardness.

To see the PL upper bound, let $\theta = \frac{p}{q}$. For each element (i, j), define the the $(i, j)^{th}$ element as variable x and then write the determinant as $ax + b$. Thus, if $|x| = |\frac{b}{a}| \le \frac{p}{q}$ for at least one such (i, j) pair, we are done. This is equivalent to checking if $(bq)^2 \le (ap)^2$. a and b can be written as determinants, hence $(ap)^2$ and $(bq)^2$ are GapL functions, and comparison of two GapL functions can be done in PL. Since PL is closed under disjunction, the entire computation can be done in PL. □

6 Rigidity of full-1s ELT matrices

In this section, we consider the result of Pudlak and Vavrin [PV91] giving a closed-form expression for the rigidity of full-1s lower triangular matrices, and attempt to extend it to full-1s *extended lower triangular* (elt) matrices. (An elt matrix is one where $m_{i,j} \neq 0 \implies j \le i+1$. A full-1s elt matrix EL_m of order m is an elt matrix satisfying $j \le i + 1 \implies m_{i,j} = 1$.) While our upper and lower bounds do not match, they differ only by a small additive factor.

Theorem 18. *Given* n *and* r *such that* $r \le n - 2$*, define the following quantities:* $k = \lfloor \frac{n-r-1}{2r+1} \rfloor$*;* $\delta = n - r - k(2r + 1)$*;* $\Gamma = \frac{(k+1)}{2}(n - r + \delta)$*;* $\ell = \lfloor \frac{n-r}{2r+1} \rfloor$*. Now, over any field* $\mathbb{F}$*,*

1. *If* $n \le 3r$*, then* $R_{EL_n}(r + 1) = n - r - 1$*.*
2. *If* $n \ge 3r + 1$*, then* $\Gamma \le R_{EL_n}(r + 1) \le \Gamma + \ell - 1$*.*

Proof. Our lower bound proof has two phases: the first follows the lower bound proof of [PV91], using the fact that deleting row i and column $i + 1$ of EL_m yields EL_{m-1} provided $i \neq m$, and the second phase raises this lower bound by 1 when $n = 3r + 1$. Our upper bound proof directly mimics that of [PV91].

Upper Bound: Recall that $\ell = \lfloor \frac{n-r}{2r+1} \rfloor$; now define $\tau = n - r - (2r + 1)\ell$. We will show that

$$R_M(r) \le \frac{(\ell + 1)}{2}(n - r + \tau) + \ell - 1$$

From this, we can obtain the upper bound claimed in the theorem. When $n \le 3r$, it is obvious. When $n \ge 3r + 1$, consider two cases:

Case 1: $\ell = k$. In this case, $\tau = \delta$. Then $\Gamma = \frac{(k+1)}{2}(n - r + \delta) = \frac{(\ell+1)}{2}(n - r + \tau)$.

Case 2: $\ell = k + 1$. In this case, $\tau = 0$, $\delta = 2r + 1$, and $n = 2r\ell + r + \ell = \delta\ell + r$. Then

$$\begin{aligned}
\Gamma &= \tfrac{(k+1)}{2}(n - r + \delta) \\
&= \tfrac{(\ell+1)}{2}(n - r + \delta) - \tfrac{1}{2}(n - r + \delta) \\
&= \tfrac{(\ell+1)}{2}(n - r + \tau) + \tfrac{(\ell+1)}{2}(\delta) - \tfrac{1}{2}(\delta\ell + \delta) \\
&= \tfrac{(\ell+1)}{2}(n - r + \tau)
\end{aligned}$$

Now we establish the upper bound in terms of ℓ and τ.

We start with a full-1 ELT matrix $EL_{n\times n}$. The rank of the matrix is $n-1$ to begin with. We identify r linearly independent rows $R_{j_1}, \ldots R_{j_r}$ which we will keep intact, so the rank of the resulting matrix is still at least r. We will change each of the other rows to one of these rows by changing some entries. But to minimise the number of entries changed, we adopt the following general strategy used in [PV91] for T_n. Let n_0 be the first set of rows which we will explicitly make zero. Similarly, n_{2i-1} is the number of rows just above R_{j_i} which are changed to R_{j_i} by changing the appropriate 0s to 1s, and n_{2i} is the number of rows below the row R_{j_i} which are changed to R_{j_i} by changing the appropriate 1s to 0s. Now the total number of changes is a function of these n_i's, as described below, and the natural idea for minimizing the number of changes is to make the contribution of each n_i roughly equal. In particular, this evenly spaces out the rows to be preserved. In detail:

$$\begin{array}{rcl@{\quad}c@{\quad}l}
\#\text{ of changes in } n_0\text{-block} & = & \sum_{t=1}^{n_0}(t+1) & = & \frac{n_0(n_0+3)}{2} \\
\#\text{ of changes in } n_{2i-1}\text{-block} & = & \sum_{t=1}^{n_{2i-1}} t & = & \frac{n_{2i-1}(n_{2i-1}+1)}{2} \\
\#\text{ of changes in } n_{2i}\text{-block} & = & \sum_{t=1}^{n_{2i}} t & = & \frac{n_{2i}(n_{2i}+1)}{2}) \\
\#\text{ of changes in } n_{2r}\text{-block} & = & n_{2r}-1+\sum_{t=1}^{n_{2r}-1} t & = & \frac{(n_{2r}+2)(n_{2r}-1)}{2}
\end{array}$$

and we want to minimize the total number of changes.

It can be seen that the optimal choice to achieve this would be to make all the n_i's equal, except n_0 which should be one less. This can happen when $\tau = 2r$; we set $n_0 = \ell$, $n_i = \ell+1$ for $i \geq 1$. When $\tau < 2r$, some of the blocks other than n_0 will also have size ℓ rather than $\ell+1$. Thus the last τ blocks will have size $\ell+1$, and the first $(2r+1-\tau)$ will be of size ℓ. Thus,

$$\begin{array}{rcl}
\text{Total number of changes} & = & \frac{\ell(\ell+1)}{2}(2r+1)+\ell-1+(\ell+1)\tau \\
& = & \frac{(\ell+1)}{2}\left[n-r+\tau\right]+\ell-1
\end{array}$$

Lower Bound: The lower bound of (1) (when $n \leq 3r$) is easy to see. For decreasing the rank of any matrix, at least one entry has to be changed.

The lower bound when $n \geq 3r+1$ is a little more tricky. In [PV91], the corresponding lower bound for lower triangular matrices T_n is obtained by first showing that if $T_n + B_n$ has rank bounded by r, then some row of B_n has at least $k+1$ non-zero entries. Deleting this row and column yields $T_{n-1}+B_{n-1}$ also of rank bounded by r. Applying this argument repeatedly, the total number of changes is bounded by a certain sum, yielding the result. Our proof follows the same outline, and differs in essentially two places: (a) deleting any row i and column $i+1$ of EL_n yields EL_{n-1}. (b) at $n = 3r+1$ a tighter bound is possible.

Given r, for each n we define

$$\begin{array}{rcl}
k_{n,r} & = & \left\lfloor \dfrac{n-r-1}{2r+1} \right\rfloor \\
\delta_{n,r} & = & n-r-k_{n,r}(2r+1)
\end{array}$$

Thus $k_{n,r}(2r+1)+r+1 \leq n \leq k_{n,r}(2r+1)+3r+1$. The value of $k_{n,r}$ remains unchanged for $2r+1$ successive values of m, during which $\delta_{m,r}$ ranges over 1 to $2r+1$.

Notice that if $r+2 \leq n \leq 3r+1$, there is a row with at least 1 change. Now, for a general n, assuming that $EL_n + B_n$ has rank bounded by r, the following lemma shows that B_n has reasonable row-wise density.

Lemma 19. *Let $r \leq n-2$, and let B_n be a matrix such that rank$(EL_n + B_n) \leq r$. Let $k = k_{n,r}$, $\delta = \delta_{n,r}$. Then there exists a row in B_n with at least $(k+1)$ non-zeroes.*

Proof. Assume to the contrary that every row of B_n (possibly other than row n) has fewer non-zeroes than required. Let $A_n = EL_n + B_n$. The idea is to choose a set S of $r+1$ rows which exclude row n, are linearly independent in EL_n, and are linearly dependent in A_n, and to then show that one of the rows from S in B_n has many non-zeroes. We choose S as follows:

$$S = \{k, k+(2k+1), \ldots, k+r(2k+1)\}$$

Since rank$(A_n) \leq r$, the rows indexed by S are linearly dependent in A_n; hence for some non-empty subset S' of S, we have non-zero α_j's satisfying

$$\sum_{j \in S'} \alpha_j a_j = 0$$

and hence

$$\sum_{j \in S'} \alpha_j l_j = -\sum_{j \in S'} \alpha_j b_j$$

Here a_j, l_j, b_j refer to the jth row vectors of A_n, EL_n and B_n respectively. By our assumption, the vector on the RHS has at most $s'k$ non-zero entries ($s' = |S'|$). Exploiting the special structure of the matrix, we show that the LHS has more non-zero terms than the RHS and get a contradiction. Due to the structure of EL_n, the LHS is of the form $(c_1, c_1 \ldots c_1, c_2, c_2 \ldots c_2, \ldots c_{s'} \ldots c_{\ s'}, 0 \ldots 0)$. Each section c_i is of size at least $2k+1$, except the c_1 section, which has size at least $k+1$. Two consecutive sections cannot be zeros since $\alpha_j \neq 0$ for all j. And the last section necessarily has $c_{s'} \neq 0$.

Case 1: s' is odd; $s' = 2\ell+1$. Now consider the LHS. There are at least $\ell+1$ blocks of non-zeroes. At most one of these (the first) is of size $k+1$; all the rest have size $2k+1$. Hence the number of non-zero elements is at least $(2k+1)\ell+k+1 = (2\ell+1)k+\ell+1 > (2\ell+1)k$.

Case 2: s' is even; $s' = 2\ell$ with $\ell \neq 0$. There are at least ℓ blocks of non-zeros. Furthermore, if the first block is a non-zero block, then in fact there must be $\ell+1$ non-zero blocks. Thus there are at least ℓ blocks of non-zeros of size $2k+1$. Thus the number of non-zeroes is at least $\ell(2k+1) > 2\ell k$.

$\square$

Lemma 20. $R_{EL_n}(r+1) \geq 2r+1$ *when* $n = 3r+1$.

Proof. Suppose not; assume that $2r$ changes suffice to bring the rank of $E = EL_{3r+1}$ to r or less. That is, there is a matrix B with at most $2r$ non-zero entries such that $A = B + E$ has rank r or less. Since there are $3r+1$ rows, at least $r+1$ of them are left unchanged. These must be linearly dependent to achieve $\text{rank}(A) \leq r$, so they must include rows $n-1$ and n of E (all other rows of E are linearly independent) and exactly $r-1$ other rows.

Let S be the set of preserved rows; $|S| = r+1$ and $\{n-1, n\} \subseteq S$. Let $S' = [n] \setminus S$; then $|S'| = 2r$. Each row of B in S' has at least one non-zero. But since there are only $2r$ non-zeroes overall, each row in S' has, in fact, exactly one non-zero.

For each $i \in S'$, row i is dependent on S and on $S \setminus \{n\}$. (With a single change per row, no row cannot be zeroed out.) Let $U = S \setminus \{n\} \cup \{i\}$. Then, as in the proof of Lemma 19, there exists $U' \subseteq U$: $i \in U'$, and for each $u \in U'$, $\exists \alpha_u \neq 0$ such that

$$\sum_{u \in U'} \alpha_u e_u = - \sum_{u \in U'} \alpha_u b_u.$$

The RHS has a single non-zero in row i since rows of B from S are zero. The LHS is of the form is of the form $(c_1, c_1 \dots c_1, c_2, c_2 \dots c_2, \dots c_{u'} \dots c_{u'}, 0 \dots 0)$ where $c_{u'} \neq 0$. To get just one non-zero on the LHS, $c_{u'}$ must be a block of size 1, and all other c_j's must be zero. Thus $\exists k : U' = \{k-1, k\}$, and $\alpha_k + \alpha_{k-1} = 0$. But, we know that α_i must be non-zero, since this is the row we are expressing as a combination of rows in S. Hence U' must be either $\{i-1, i\}$ or $\{i, i+1\}$. Thus, for each row $i \in S'$, either row $i-1$ or row $i+1$ is in S. So rows in S can be separated by at most 2 rows of S'. Since rows $n = 3r+1$ and $n-1 = 3r$ are in S, the 3rd last row of S is at least $3r-3$, the 4th last row of S is at least $3r-6$, and so on; the first row of S is at least row 3. But then row 1 does not have a neighbouring row in S, a contradiction. $\square$

Using these lemmas we can establish the lower bound. When $n \geq 3r+2$, apply Lemma 19 repeatedly, eliminating one dense row each time, preserving the ELT structure, until n comes down to $3r+1$. Now Lemma 20 says that $2r+1$ more changes are necessary. Thus the total number of changes is at least $\delta(k+1) + (2r+1)k + (2r+1)(k-1) + \ldots + (2r+1)3 + (2r+1)2 + (2r+1) = \frac{(k+1)}{2}(n - r + \delta)$, giving the lower bound.

$\square$

7 Discussion

The matrix rigidity problem over $\mathbb{Z}$ and $\mathbb{Q}$ is not even known to be in NP in general; we have looked at various restricted cases. Over finite fields, however, as mentioned in the introduction, even the general problem is in NP. One can also prove analogous results for the restricted cases over finite fields. It is known [BDHM92] that SINGULAR($\mathbb{F}_p$) is complete for $\mathsf{Mod}_p\mathsf{L}$ (In fact, computing the exact value of the determinant over $\mathbb{F}_p$ is in $\mathsf{Mod}_p\mathsf{L}$.), and that (see for instance [All04]), for any prime p, RANK BOUND($\mathbb{F}_p$) is in $\mathsf{Mod}_p\mathsf{L}$. Using this, and closure properties of $\mathsf{Mod}_p\mathsf{L}$, we can obtain analogues of Theorem 9 and 11 for finite fields; (1) for each k, and each prime p, $\text{RIGID}_{\mathbb{F}_p}(k)$ is complete for $\mathsf{Mod}_p\mathsf{L}$, and (2) given a non-singular matrix, a singular matrix can be obtained by changing just one entry, and the change can be computed in $\mathsf{Mod}_p\mathsf{L}$.

A major open problem here is to obtain a better upper bound (e.g., even decidability is not known) for the computational rigidity question. A related question is that of *minimum rank completion*, see for instance [BFS99]. Given a matrix with indeterminates at some positions, find the smallest rank achievable under all possible instantiations of these variables. 1-MinRank is the special case of MinRank where each variable occurs at most once. The rigidity question is easily seen to lie in NP(1-MinRank). However, the best known upper bound for 1-MinRank is recursive enumerability.

We can also consider the complementary question to matrix rigidity, namely, computing the number of entries that need to be changed to increase the rank above a given value. Using arguments similar to the case of decreasing rank, we can obtain similar complexity results in this case also. However, notably in this case, we not only have decidability, we also have an upper bound of NP. This follows from the framework of *maximum rank matrix completion*, which is known to be in P [Gee99, Mur93].

8 Acknowledgements

We thank V. Arvind, N.S. Narayanaswamy, R. Balasubramanian and Kapil Paranjape for many insightful discussions.

References

[AAB+99] E. Allender, A. Ambainis, D. A. Mix Barrington, S. Datta, and H. LeThanh. Bounded-depth arithmetic circuits: counting and closure. In *Proceedings of 26th International Colloquium on Automata, Languages and Programming (ICALP)*, volume 1644 of *Lecture Notes in Computer Science*, pages 149–158. Springer-Verlag, 1999.

[AAM03] E. Allender, V. Arvind, and M. Mahajan. Arithmetic complexity, kleene closure, and formal power series. *Theory of Computing Systems*, 36(4):303–328, 2003.

[ABO99] E. Allender, R. Beals, and M. Ogihara. The complexity of matrix rank and feasible systems of linear equations. *Computational Complexity*, 8(2):99–126, 1999.

[AG00] C Àlvarez and R Greenlaw. A compendium of problems complete for symmetric logarithmic space. *Computational Complexity*, 9:73–95, 2000.

[All04] E. Allender. Arithmetic circuits and counting complexity classes. In Jan Krajicek, editor, *Complexity of Computations and Proofs*, Quaderni di Matematica Vol. 13, pages 33–72. Seconda Universita di Napoli, 2004. An earlier version appeared in the Complexity Theory Column, SIGACT News 28, 4 (Dec. 1997) pp. 2-15.

[AO96] E. Allender and M. Ogihara. Relationships among PL, #l, and the determinant. *RAIRO Theoretical Information and Applications*, 30:1–21, 1996. Conference version in *Proc. 9th IEEE Structure in Complexity Theory Conference* (1994), 267–278.

[BDHM92] G. Buntrock, C. Damm, U. Hertrampf, and C. Meinel. Structure and importance of logspace MOD-classes. *Math. Systems Theory*, 25:223–237, 1992.

[BFS99] Jonathan F. Buss, Gudmund Skovbjerg Frandsen, and Jeffrey Shallit. The computational complexity of some problems of linear algebra. *Journal of Computer and System Science*, 58:572–596, 1999.

[CM87] S A Cook and P McKenzie. Problems complete for L. *Journal of Algorithms*, 8:385–394, 1987.

[CMTV98] H. Caussinus, P. McKenzie, D. Thérien, and H. Vollmer. Nondeterministic NC^1 computation. *Journal of Computer and System Sciences*, 57:200–212, 1998.

[Dah99] G. Dahl. A note on nonnegative diagonally dominant matrices. *Linear Algebra and Applications*, 317:217–224, April 1999.

[Gee99] James F. Geelen. Maximum rank matrix completion. *Linear Algebra and its Applications*, 288(1–3):211–217, 1999.

[GJ79] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W.H.Freeman and Co., 1979.

[Gri76] D. Yu Grigoriev. Using the notions of seperability and independence for proving the lower bounds on the circuit complexity. Notes of the Leningrad branch of the Steklov Mathematical Institute, Nauka, 1976. in Russian.

[Lau01] M. Laurent. Matrix completion problems. In C.A. Floudas and P.M. Pardalos, editors, *The Encyclopedia of Optimization*, volume 3, pages 221–229. Kluwer, 2001.

[Li92] L. Li. Formal power series: An algebraic approach to the GapP and #P functions. In *Proc. 7th Structure in Complexity Theory Conference*, pages 144–154, 1992.

[Lok95] Satyanarayana V. Lokam. Spectral methods for matrix rigidity with applications to size-depth tradeoffs and communication complexity. In *Proc. 36th FOCS*, pages 6 – 15, 1995.

[MM64] Marvin Marcus and Henryk Minc. *A Survey of Matrix Theorey and Matrix inequalities*, volume 14 of *The Prindle, Weber and Schmidt Complementary Series in Mathematics*. Allyn and Bacon, Boston, 1964.

[Mul87] K. Mulmuley. A fast parallel algorithm to compute the rank of a matrix over an arbitrary field. *Combinatorica*, 7:101–104, 1987.

[Mur93] K. Murota. Mixed matrices - Irreducibility and decomposition. In R. A. Brualdi, S. Friedland, and V. Klee, editors, *Combinatorial and Graph-Theoretic Problems in Linear Algebra*, volume 50 of *The IMA Volumes in Mathematics and Its Applications*, pages 39 – 71. Springer, 1993.

[MV97] M. Mahajan and V Vinay. Determinant: combinatorics, algorithms, complexity. *Chicago Journal of Theoretical Computer Science*, 1997:5, dec 1997.

[NTS95] N. Nisan and A. Ta-Shma. Symmetric Logspace is closed under complement. *Chicago Journal of Theoretical Computer Science*, 1995.

[PR93] S. Poljak and J. Rohn. Checking robust nonsingularity is NP-hard. *Math. Control Signals Systems*, 6:1–9, 1993.

[PV91] P. Pudlak and Z. Vavrin. Computation of rigidity of order n^2/r for one simple matrix. *Comment. Nath. Univ. Carolinae.*, 32(2):213–218, 1991.

[Rei05] O. Reingold. Undirected *st*-conenctivity in logspace. In *Proc. 37th Annual ACM Symposium on Theory of Computing STOC*, pages 376–385, 2005.

[Roh94] J. Rohn. Checking positive definiteness or stability of symmetric interval matrices is NP-hard. *Commentationes Mathematicae Universitatis Carolinae*, 35:795–797, 1994.

[Roh96] Jiri Rohn. Checking Properties of Interval Matrices. Technical Report 686, Institute of Computer Science, Academy of Sciences of the Czech Republic, 1996. 36 p.

[RW91] P. Radge and A. Wigderson. Linear-size constant-depth polylog-threshold circuits. *Information Processing Letters*, 39:143–146, 3 1991.

[Tod91] S. Toda. Counting problems computationally equivalent to the determinant. Technical Report CSIM 91-07, Dept of Comp Sc & Information Mathematics, Univ of Electro-Communications, Chofu-shi, Tokyo, 1991.

[Val77] L. G. Valiant. Graph theoretic arguments in low-level complexity. In *Proceedings of 6th International Symposium on Mathematical Foundations of Computer Science (MFCS)*, volume 53 of *LNCS*, pages 162–176. Springer, Berlin, 1977.

[Vol99] H. Vollmer. *Introduction to Circuit Complexity: A Uniform Approach*. Springer, 1999.

Appendix

Detailed Proof for Hardness in Theorem 14

The hardness result in Theorem 14 is proved by using Theorems 2.1, 2.2 and 2.6 of [PR93]. For clarity, we present here a self-contained proof, obtained by specialising the results of [PR93].

We want to show that deciding whether $R_M(r,\theta) \leq k$, i.e. testing whether it is possible to bring the rank of a matrix M below r by changing at most k entries, each by at most θ, is NP-hard. The reduction is from the *maximum bipartite subgraph problem* which is known to be NP-complete (see [GJ79]), and which is stated as follows: Given an undirected graph $G = (V, E)$, with n vertices and m edges, decide whether G has a bipartite subgraph with at least k edges. Without loss of generality, assume that G has no isolated vertices.

Define the matrix N as,

$$N_{ij} = \begin{cases} -1 & \text{if } i \neq j \text{ and } i \text{ and } j \text{ are adjacent in } G \\ 2m+1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

Notice that since N is diagonally dominant, by Levy-Desplanques theorem (see for instance Theorem 2.1 in [MM64]), N is invertible. Let $M = N^{-1}$. We establish hardness by showing the following:

Lemma 21. *G has a bipartite subgraph of size k $\iff$ $R_M(n, \frac{1}{(2m+1)n+4k-2m}) \leq n^2$.*

We first show the $\Longrightarrow$ direction.

For any $y, z \in \{-1,+1\}^n$, let $N' = Nyz^T$. Let λ be a non-zero eigen-value of N', with a corresponding eigenvector x; thus $N'x = Nyz^Tx = \lambda x$.

Claim 1: $\lambda = z^TNy$. Thus if N is rational, so is λ.
Proof: Since $\lambda x \neq 0$, the scalar z^Tx is also non-zero. Premultiplying both sides by z^T and dividing by z^Tx, we get $\lambda = z^TNy$.

Claim 2: $R_M(n, 1/\lambda) \leq n^2$, where $M = N^{-1}$.
Proof: By Claim 1, $(\lambda I - Nyz^T)x = 0$. Premultiplying by $N^{-1} = M$ gives $(\lambda M - yz^T)x = 0$ and hence $(M - (1/\lambda)yz^T)x = 0$. Since y, z have only ± 1 entries, changing each entry of M by at most $1/\lambda$ suffices to reach the singular matrix $M - (1/\lambda)yz^T$.

Claim 3: Vectors $y \in \{-1,+1\}^n$ are in bijection with cuts $(S, \overline{S})$ in G, where $S = \{i \mid y_i = +1\}$. When N is obtained from G as above, then $y^TNy = 4\delta(S) + (2m+1)n - 2m$, where $\delta(S)$ is the size of the cut $(S, \overline{S})$.
Proof:

$$\begin{aligned} y^TNy &= \textstyle\sum_{ij} n_{ij}y_iy_j \\ &= \textstyle\sum_{ij}(-\frac{1}{2}n_{ij})((y_i - y_j)^2 - 2) \\ &= -\frac{1}{2}\textstyle\sum_{ij} n_{ij}(y_i - y_j)^2 + \sum_{ij} n_{ij} \\ &= 4(\delta(S)) + (2m+1)n - 2m \end{aligned}$$

The $\Longrightarrow$ direction of the lemma is immediate from Claims 1,2,3: a bipartite subgraph or a cut with at least k edges corresponds to a vector y with $y^T Ny \geq 4k + (2m+1)n - 2m$ (Claim 3), $y^T Ny$ is an eigenvalue of Nyy^T (Claim 1), so $R_M(n, 1/(y^T Ny)) \leq n^2$ (Claim 2). But $R_M(r,\alpha) \leq R_M(r,\beta)$ for all $\alpha \geq \beta$, giving the implication.

Now we consider the $\Longleftarrow$ direction. Again, the proof is broken into several steps. Let $\alpha = \frac{1}{4k+(2m+1)n-2m}$.

Claim 4: If $R_M(n,\alpha) \leq n^2$, then there exist $t \in [-1,+1]^n$, $z \in \{-1,+1\}^n$ such that $M - \alpha t z^T$ is singular.
Proof: Since $R_M(n,\alpha) \leq n^2$, there is a singular matrix A in the interval $[M - J\alpha, M + J\alpha]$, where J is the all-ones matrix. And there is a non-zero vector x such that $Ax = 0$. Define vector z as $z_i = sgn(x_i)$. ($sgn(a) = +1$ if $a > 0$, -1 otherwise.) Define vector t as follows: $t_i = (Mx)_i/\alpha X$ where $X = \sum_j |x_j| = \sum_j z_j x_j$. To see that $M - \alpha t z^T$ is singular, note that for each i, the ith element of $(M - \alpha t z^T)x$ is $(Mx)_i - \alpha t_i \sum_j z_j x_j = 0$. Next, let us see why each $|t_i| \leq 1$. Since A is in the interval $[M - J\alpha, M + J\alpha]$, there is a B such that $A = M - B$ and each $|b_{ij}| \leq \alpha$. Since $Ax = 0$, $Mx = Bx$. Now $t_i = (Mx)_i/\alpha X = (Bx)_i/\alpha X = \sum_j b_{ij}x_j/\alpha X$. So $|t_i| \leq \sum_j |b_{ij}||x_j|/\alpha X \leq \sum_j \alpha|x_j|/\alpha X = 1$.

Claim 5: If $M - \alpha t z^T$ is singular for t, z as in Claim 4, then there is a $y \in \{-1,+1\}^n$ and a $\beta \in (0,1]$ such that $M - \alpha\beta y z^T$ is singular.
Proof: Consider n variables $w_1, \ldots, w_n$ and consider the the symbolic matrix $M - \alpha w z^T$. Its determinant is a multilinear polynomial $p(w_1, \ldots, w_n)$, with a zero at t. Since $t_i \in [-1,+1]$ for each i, we can use the Zero-on-an-Edge lemma (Lemma 16) to conclude that $p(.)$ has a zero $a = (a_1, \ldots, a_n)$ on an edge of the corresponding hypercube. i.e. there is a $k \in [n]$ such that for each $i \neq k$, $a_i \in \{-1,+1\}$, and $a_k \in [-1,+1]$. Restricted to this edge, $p(.)$ is linear in w_k, and so is either 0 everywhere on this edge or flips sign exactly once, at a. Thus $p(y)p(y') \leq 0$, where y, y' are the the endpoints of this edge in the hypercube (and hence in $\{-1,+1\}^n$).

Since M is invertible, $det(M) \neq 0$. Hence either $p(y)det(M) \leq 0$ or $p(y')det(M) \leq 0$; w.l.o.g. assume that $p(y)det(M) \leq 0$. Now define a function ϕ as follows: $\phi(s) = det(M - s\alpha y z^T)$. Clearly, $\phi(0) = det(M) \neq 0$ and $\phi(1) = p(y)$, so $\phi(0)\phi(1) \leq 0$. But ϕ is a continuous function, so it must have a zero, say β, in $(0,1]$. Then $0 = \phi(\beta) = det(M - \beta\alpha y z^T)$.

Claim 6: If $M = N^{-1}$ where N is constructed from G as described in the reduction, then the maximum eigen-value of Nyz^T, over all choices of $y, z \in \{-1,+1\}^n$, is achieved when $y = z$.
Proof: Let $y, z \in \{-1,+1\}^n$ with $y \neq z$. Let i be an index where y, z differ; w.l.o.g., let $y_i = -1$, $z_i = +1$. Let y' be obtained from y by flipping y_i to $+1$. We claim that $Ny'z^T$ has a larger eigenvalue than Nyz^T. By Claim 1; the eigenvalues in question are $z^T Ny$ and $z^T Ny'$. By the special structure of N, we can see that changing y_i from -1 to $+1$ gives an increase of $2m+1$ from the diagonal entry, and a decrease of at most

$2(n-1)$ from the off-diagonal entry. Since G has no isolated vertices, $m \geq n/2$ and so there is an overall gain in going from y to y'.

Now we can establish the $\Longleftarrow$ direction: By Claims 4 and 5, $M - \alpha\beta y^T z$ is singular, hence $(1/(\alpha\beta))NM - Ny^T z$ is singular, hence there is a non-zero vector x such that $\frac{1}{\alpha\beta}x = Nyz^T x$ and $\frac{1}{\alpha\beta}$ is an eigenvalue of Nyz^T. Since $0 < \beta \leq 1$, this eigenvalue is at least $1/\alpha$. By Claim 6, there is a y' such that $y'^T Ny \geq 1/\alpha = 4k + (2m+1)n - 2m$. Now Claim 3 yields the desired bipartite subgraph.

Note that Claims 1,2,4,5, hold for any invertible matrix.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de/