On the Complexity of Matrix Rank and Rigidity

Jayalal Sarma M.N.

The Institute of Mathematical Sciences(IMSc),
Chennai, India

Joint work with

Meena Mahajan(IMSc)

CSR 2007

Ekaterinburg, Russia

September 4-6, 2007

Matrix Rank

Rank of a matrix $M \in \mathbb{F}^{n \times n}$ has the following equivalent definitions.

- The size of the largest submatrix with a non-zero determinant.
- The number of linearly independent rows/columns of a matrix.
- The smallest r such that $M = AB$ where $A \in \mathbb{F}^{n \times r}, B \in \mathbb{F}^{r \times n}$.

RANK BOUND: Given a matrix M and a value r, is $\text{rank}(M) < r$?
SINGULAR: Given a matrix M is $\text{rank}(M) < n$?

Matrix Rank

Rank of a matrix $M \in \mathbb{F}^{n \times n}$ has the following equivalent definitions.

- The size of the largest submatrix with a non-zero determinant.
- The number of linearly independent rows/columns of a matrix.
- The smallest r such that $M = AB$ where $A \in \mathbb{F}^{n \times r}, B \in \mathbb{F}^{r \times n}$.

RANK BOUND: Given a matrix M and a value r, is $\text{rank}(M) < r$?
SINGULAR: Given a matrix M is $\text{rank}(M) < n$?

Motivations from linear algebra, control theory, from algorithmics, complexity theory. In the context of seperating complexity classes, it might facilitate application of well developed algebraic techniques.

Complexity Theoretic Preliminaries

- L : Languages accepted by log-space bounded deterministic Turing machines. [Reachability in Undirected Graphs]
- NL : Languages accepted by log-space bounded non-deterministic Turing machines. [Reachability in Directed Graphs]
- $C_=L$: Languages accepted by a non-deterministic Turing machine such that input is in the language if and only if $\#$ of accepting paths $= \#$ of rejecting paths. [SINGULAR]

Circuits are DAGs with $\wedge$, $\vee$ and $\neg$ gates at the vertices.

- AC^0 : poly size constant depth and unbounded fanin circuits.
- TC^0 : AC^0 with "majority" gates.

$$AC^0 \dashrightarrow TC^0 \longrightarrow NC^1 \longrightarrow L \longrightarrow NL \longrightarrow C_=L$$

Computing the Rank

- The natural approach takes exponential time.
- Can be computed in Polynomial time :
  Gaussian elimination, LU decomposion, SV decomposition.
  But they are inherently sequential.
- Rank can be computed in NC.
  Elegant parallel algorithm (Mulmuley 87) by relating the problem to testing if some coefficients of the characterstic polynomial are zeros. Independently by Chistov(1986).
- Refined complexity bounds by Allender et.al 1996. Upper bound testing exactly characterises $C_=L$.

Computing the rank of special matrices

- Several applications give rise to structured matrices.
- Complexity theoretic characterisations.
- Known result: For symmetric non-negative matrices, RANK BOUND and SINGULAR are $C_=L$-complete (Allender et.al, 1996).

Restrictions we are interested in:

- $M = [a_{i,j}]$ is diagonally dominant if

$$|a_{ii}| \geq \sum_{j \neq i} |a_{ij}|$$

  Fun fact : If dominance is strict for all i, M is non-singular.
- Diagonal matrices : Non-zero entries only on the main diagonal.

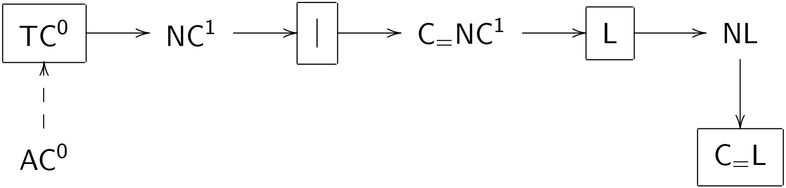
Rank of Restricted Families of Matrices

Matrix type	RANK BOUND	SINGULAR
Sym.Non-neg.	$C_=L$-complete [ABO96]	$C_=L$-complete [ABO96]

$$\begin{tikzcd}
TC^0 \arrow[r] & NC^1 \arrow[r] & C_=NC^1 \arrow[r] & L \arrow[r] & NL \arrow[d] \\
AC^0 \arrow[u, dashed] & & & & C_=L
\end{tikzcd}$$

Rank of Restricted Families of Matrices

Matrix type	RANK BOUND	SINGULAR
Sym.Non-neg.	$C_=L$-complete [ABO96]	$C_=L$-complete [ABO96]
Sym.Non-neg. Diag. Dom.	L-complete	L-complete
Tridiagonal	?	in $C_=NC^1$
Diag	TC^0-complete	in AC^0



Characterising Log space

Theorem

Computing the rank of symmetric non-negative diagonally dominant matrices is L*-complete.*

Characterising Log space

Theorem

Computing the rank of symmetric non-negative diagonally dominant matrices is L*-complete.*

MEMBERSHIP: For a non-neg. sym. dd matrix $M \in \mathbb{Q}^{n\times n}$, define the support graph $G_M = (V, E_M)$ has $V = \{v_1, \ldots v_n\}$, and

$$E_M = \{(v_i, v_j) \mid i \neq j \quad m_{i,j} > 0\} \cup \left\{(v_i, v_i) \mid m_{i,i} > \sum_{i \neq j} m_{i,j}\right\}$$

c : Number of bipartite components of G_M.

Characterising Log space

Theorem

Computing the rank of symmetric non-negative diagonally dominant matrices is L*-complete.*

MEMBERSHIP: For a non-neg. sym. dd matrix $M \in \mathbb{Q}^{n \times n}$, define the support graph $G_M = (V, E_M)$ has $V = \{v_1, \ldots v_n\}$, and

$$E_M = \{(v_i, v_j) \mid i \neq j \quad m_{i,j} > 0\} \cup \left\{ (v_i, v_i) \mid m_{i,i} > \sum_{i \neq j} m_{i,j} \right\}$$

c : Number of bipartite components of G_M.

Claim [Dah99]: $\text{rank}(M) = n - c$

Using this we can reduce the problem to counting the number of bipartite components in a graph. This can be computed in L.

Characterising Log space

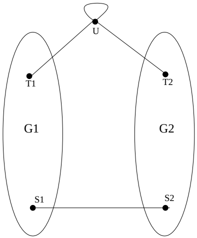
Hardness :

The problem of testing reachability in undirected forests where there are exactly two components is L-complete [CM87]. Given an instance, $(G(V, E), s, t)$, define $G'(V \times \{0, 1\}) \cup \{u\}, E')$:

Characterising Log space

HARDNESS :

The problem of testing reachability in undirected forests where there are exactly two components is L-complete [CM87]. Given an instance, $(G(V,E),s,t)$, define $G'(V \times \{0,1\}) \cup \{u\}, E')$:



Claim :

G' has two bipartite components $\iff$ t is reachable from s in G

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

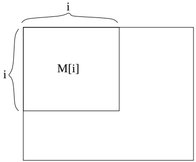
$$\text{For each } i \qquad m_{i,i} = \begin{cases} 1 + \sum_{j \neq i} m_{i,j} & \text{if } (i,i) \in E' \\ \sum_{j \neq i} m_{i,j} & \text{otherwise} \end{cases}$$

For tri-diagonal matrices

Theorem

SINGULAR *for tri-diagonal matrices is in* $C_=NC^1$. *Computing the determinant of these matrices is in* GapNC^1, *hard for* NC^1.

DETERMINANT:



$$P_i = Perm(M[i])$$

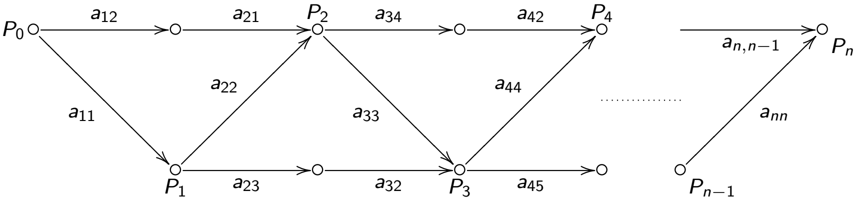
$$D_i = Derm(M[i])$$

We have the following recurrences:

$$P_0 = D_0 = 1 \qquad P_1 = D_1 = a_{1,1}$$

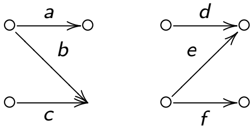
$$P_i = a_{i,i}P_{i-1} + a_{i-1,i}a_{i,i-1}P_{i-2} \qquad D_i = a_{i,i}D_{i-1} - a_{i-1,i}a_{i,i-1}D_{i-2}$$

Planar Branching Program for P_i



Similar graphs have been studied earlier as G-graphs [AAB$^+$99]. where they show that counting the number of s-t paths in such graphs is hard for NC^1.

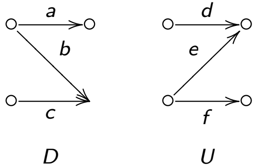
G-graphs are those layered graphs which can be decomposed into the following components.



Counting paths in G-graphs to Tridiagonal determinant :

- First suppose that the encoded string has alternate DU. Just read off the weights on the corresponding edges in the graph, produce matrix M_1 such that,

$$Perm(M_1) = \text{the number of weighted } s\text{-}t \text{ paths in the graph}$$



- Any BWBP can be transformed to this form : If the string does not start with a D we will just put in a prefix D with $def = 101$
- When there are UU or DD, Simply put in a D with $def = 101$ in between two U and a U with $abc = 101$ in between two Ds.

How close is M to a rank r matrix?

Definition (Rigidity)

Given a matrix M and $r \leq n$, rigidity of the matrix M ($R_M(r)$) is the number of entries of the matrix that we need to change to bring the rank below r.

[Val77] Interesting in a circuit complexity theory setting. If for some $\epsilon > 0$ there exists a $\delta > 0$ such that an $n \times n$ matrix M_n has rigidity $R_{M_n}(\epsilon n) \geq n^{1+\delta}$ over a field $\mathbb{F}$, then the transformation $x \rightarrow Mx$ cannot be computed by linear size logarithmic depth linear circuits.

[Raz89] For an explicit infinite sequence of (0,1)-matrices $\{M_n\}$ over a finite field $\mathbb{F}$, if $R_M(r) \geq \frac{n^2}{2^{(\log r)^{o(1)}}}$ for some $r \geq 2^{(\log \log n)^{\omega(1)}}$, then there is an explicit language $L_M \notin \mathrm{PH}^{cc}$, where PH^{cc} is the analog of PH in the communication complexity setting.

Computing Rigidity - Why could that be interesting?

$\text{RIGID}(M, r, k)$: Given a matrix M, values r and k, is $R_M(r) \leq k$?

- Natural optimisation problem related to rank.
- Valiant's reduction [Val77] identifies "high rigidity" as a a combinatorial property of the matrices (which defines the function computed) based on which he proves linear size lower bounds for log-depth circuits. Among the $n \times n$ matrices, the density of "rigid" matrices is high.
- Practical Applications : Optimisation in control theory.

Computing Rigidity

$\textsc{rigid}(M, r, k)$: Given a matrix M, values r and k, is $R_M(r) \leq k$?

Field $\mathbb{F}$	restriction	bound
$\mathbb{F}$	-	in NP
$\mathbb{F}_2$	-	NP -complete [Des07]
$\mathbb{Z}$ or $\mathbb{Q}$	Boolean, constant k	$\mathrm{C_=L}$-complete
$\mathbb{Z}$ or $\mathbb{Q}$	constant k	$\mathrm{C_=L}$-hard
$\mathbb{F}_p$	constant k	$\mathrm{Mod}_p\mathrm{L}$-complete
$\mathbb{Q}$	$r = n$	$\mathrm{C_=L}$-complete witness-search in $\mathrm{L}^{\mathrm{GapL}}$
$\mathbb{Z}$	$r = n$ and $k = 1$	in $\mathrm{L}^{\mathrm{GapL}}$

For constant k, for 0-1 matrices, RIGID is $\mathrm{C}_{=}\mathrm{L}$-complete

MEMBERSHIP: we need to test if 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 $\Sigma_{s=0}^{k}\binom{n}{s} = t \in n^{O(1)}$.

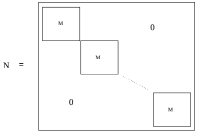
Let the corresponding matrices be $M_1, M_2 \ldots M_t$; these can be generated from M in logspace. Now,

$$\begin{aligned}(M, r) \in \text{RIGID}(k) &\iff \exists i : (M_i, r) \in \text{RANK BOUND}(\mathbb{Z}) \\ &\iff (N', r') \in \text{RANK BOUND}(\mathbb{Z})\end{aligned}$$

where N' and r' can be generated in L using standard techniques.

For constant k, for 0-1 matrices, RIGID is $C_=L$-complete

For 0-1 matrices, for $k - 0$, the problem is $C_=L$-hard, since $\textsc{rigid}(M, n, 0)$ tests if the matrix is singular.
To prove it for arbitrary k, tensor it with I_{k+1}, the rigidity gets amplified by a factor of k.



$$\begin{aligned}
M \in \textsc{singular}(\mathbb{Z}) \quad &\Longrightarrow \\
&(N, n(k+1) - k) \in \textsc{rigid}(N, n(k+1) - k, 0) \\
&\subseteq \textsc{rigid}(N, n(k+1) - k, k) \\
M \notin \textsc{singular}(\mathbb{Z}) \quad &\Longrightarrow \\
&(N, n(k+1) - k) \notin \textsc{rigid}(N, n(k+1) - k, k)
\end{aligned}$$

Bounded Rigidity

Definition (Bounded Rigidity)

Given a matrix M and $r < n$, bounded rigidity of the matrix M ($R_M(b, r)$) is the number of entries of the matrix that we need to change to bring the rank below r, if the change allowed per entry is atmost b.

- B-RIGID(M, r, k, b): Given a matrix M, values b, r and k, is $R_M(b, r) \leq k$?
- Another formulation : Define an interval of matrices $[A]$ where

$$m_{ij} - b \leq a_{ij} \leq m_{ij} + b$$

Question : Is there a rank r matrix $B \in [A]$ such that $M - B$ has atmost k non-zero entries?

Why should there be?

Consider the matrix

$$\begin{bmatrix} 2^k & 0 & 0 & 0 & 0 \\ 0 & 2^k & 0 & 0 & 0 \\ 0 & 0 & 2^k & 0 & 0 \\ 0 & 0 & 0 & 2^k & 0 \\ 0 & 0 & 0 & 0 & 2^k \end{bmatrix}$$

- $R_M(b, n-1)$ is undefined unless $b \geq \frac{2^k}{n}$.
- Question : For a given matrix M, bound b, target rank r, can we efficiently test whether $R_M(b, r)$ is defined ?

Why should there be?

Consider the matrix

$$\begin{bmatrix} 2^k & 0 & 0 & 0 & 0 \\ 0 & 2^k & 0 & 0 & 0 \\ 0 & 0 & 2^k & 0 & 0 \\ 0 & 0 & 0 & 2^k & 0 \\ 0 & 0 & 0 & 0 & 2^k \end{bmatrix}$$

- $R_M(b, n-1)$ is undefined unless $b \geq \frac{2^k}{n}$.
- Question : For a given matrix M, bound b, target rank r, can we efficiently test whether $R_M(b, r)$ is defined ?

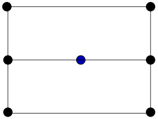
It is NP-hard.

NP-completeness for a restricted case

For a given matrix M, bound b, testing whether $R_M(b, n-1)$ is defined, is NP-complete.

MEMBERSHIP:

- The bound b defines an interval for each entry of the matrix.
- Determinant: a multlilinear polynomial in the entries of M.
- ZERO-ON-AN-EDGE LEMMA: For a multilinear polynomial $p(x_1, x_2 \ldots x_t)$, consider the hypercube defined by the interval of each of the x_is. If there is a zero of the polynomial in the hypercube then there is a zero on an edge of the hypercube

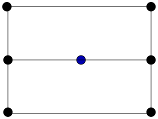


NP-completeness for a restricted case

For a given matrix M, bound b, testing whether $R_M(b, n-1)$ is defined, is NP-complete.

MEMBERSHIP:

- The bound b defines an interval for each entry of the matrix.
- Determinant: a multlilinear polynomial in the entries of M.
- ZERO-ON-AN-EDGE LEMMA: For a multilinear polynomial $p(x_1, x_2 \ldots x_t)$, consider the hypercube defined by the interval of each of the x_is. If there is a zero of the polynomial in the hypercube then there is a zero on an edge of the hypercube



- NP algorithm : Guess the edge of the hypercube where the zero occurs and verify if the sign of determinant at each end point are opposite.

NP-completeness for a restricted case

HARDNESS: The interval $[M - \theta J, M + \theta J]$ is singular if and only if $R_M(n, \theta)$ is defined.
By a reduction from MAXCUT problem, [PR93] showed that that checking interval singularity is NP-hard. Hence the hardness follows in our case too.

Open Problems

- Is there a characterisation of other small complexity classes (like NC^1, NL) using the rank/determinant computation?
- A better upper bound for computing rigidity over $\mathbb{Q}$.
- Is there an efficient algorithm when r is a constant?
- An NP upper bound for bounded rigidity - a generalisation of the zero-on-an-edge lemma to arbitrary rank.

Thank You

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.

Eric Allender, Robert Beals, and Mitsunori Ogihara.
The complexity of matrix rank and feasible systems of linear equations.
In *Proc. 28th ACMSTOC*, pages 161–167, 1996.
Appears in Computational Complexity, 8(2), 99–126, 1999.

S A Cook and P McKenzie.
Problems complete for L.
Jl. of Algorithms, 8:385–394, 1987.

G. Dahl.
A note on nonnegative diagonally dominant matrices.

Linear Algebra and Applications, 317:217–224, April 1999.

Amit Deshpande.
Sampling-based dimension reduction algorithms.
PhD thesis, MIT, May 2007.

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

A. A. Razborov.
On rigid matrices.
manuscript in russian, 1989.

L. G. Valiant.
Graph theoretic arguments in low-level complexity.
In *Proc. 6th MFCS*, volume 53 of *LNCS*, pages 162–176.
Springer, Berlin, 1977.