Construction of Rigid Matrices from PCPs

Amey Bhangale
(UC Riverside, USA)

Workshop on Matrix Rigidity
FSTTCS, Dec 2020
What are rigid matrices?

- A matrix $A \in \mathbb{F}_{2}^{N \times N}$ is rigid if it is far from low rank matrices.

- Formally, a matrix $A$ is $(\rho, \Delta)$-rigid if

$$\min_{B: rank(B) = \rho} dist(A, B) \geq \Delta$$
Applications

• Circuit lower bound

[Valiant 77]: For any $\epsilon > 0$, if $A$ is $\left(\frac{N}{\log \log N}, N^{1+\epsilon}\right)$-rigid, then $x \rightarrow Ax$ can’t be computed by circuits of size $O(N)$ and depth $O(\log N)$.

[FGHK 16] Current best known (explicit) circuit lower bound: $3.01N$
Applications

• Communication complexity

[Razborov 89] Let \( f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\} \) be a function \( PH^{cc} \), then for every \( \epsilon > 0 \), the communication matrix \( M_f \) is not \( (\rho, \Delta) \)-rigid where

\[
\rho = 2^{poly\left(\frac{\log n}{\epsilon}\right)}, \Delta = \epsilon \cdot 4^n
\]

\[
M_f(x, y) = f(x, y)
\]
Applications

- Approximate probabilistic $\mathbb{F}_2$-degree

If $\epsilon$-approximate probabilistic degree of $f : \{0,1\}^{2n} \rightarrow \{0,1\}$ is at most $\rho$ then $M_f$ is not $(n^{O(\rho)}, \varepsilon 4^n)$-rigid.

[Razborov,Smolensky 89] Approximating Majority needs probabilistic $\mathbb{F}_2$-degree at least $\sqrt{n}$. 
## Previous constructions

<table>
<thead>
<tr>
<th></th>
<th>Rank $\rho$</th>
<th>Distance $\Delta$</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Random Matrix</strong></td>
<td>$O(N)$</td>
<td>0.01 $N^2$</td>
<td>DTIME($2^{O(N^2)}$)</td>
</tr>
<tr>
<td><strong>[Friedman, SSS 90’s]</strong></td>
<td>Any</td>
<td>$\Omega\left(\frac{N^2}{\rho \log \left(\frac{N}{\rho}\right)}\right)$</td>
<td>DTIME($N^{O(1)}$) (Untouched minor argument)</td>
</tr>
<tr>
<td><strong>[Goldreich–Tal 15]</strong></td>
<td>$\geq \sqrt{N}$</td>
<td>$\Omega\left(\frac{N^3}{\rho^2 \log N}\right)$</td>
<td>DTIME($2^{O(N)}$) (Random Toeplitz matrices ($\mathbb{F}_2$))</td>
</tr>
</tbody>
</table>
# Recent constructions

<table>
<thead>
<tr>
<th>Rank $\rho$</th>
<th>Distance $\Delta$</th>
<th>Time $\text{NTIME}(N^{O(1)})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Alman-Chen 19]</td>
<td>$2^O((\log N)^{1/4-\epsilon})$</td>
<td>0.01 $N^2$</td>
</tr>
<tr>
<td>[B., Harsha, Paradise, Tal 20]</td>
<td>$2^O\left(\frac{\log N}{\log \log N}\right)$</td>
<td>0.01 $N^2$</td>
</tr>
</tbody>
</table>
New implications

- \([\text{AC19}] \ \text{TIME}[2^{\log n^{o(1)}}]^{NP} \not\subseteq PH^{cc}\)

- \([\text{AC 19}] \ E^{NP} \not\subseteq (\text{restricted}) AC^0[p] \circ LTF \circ AC^0[p] \circ LTF\)

- \([\text{BHPT 20, Viola 20}] \ \text{Approximate probabilistic } \mathbb{F}_2\text{-degree lower bound of } \Omega \left(\frac{n}{\log^2 n}\right)\) for a function in \(E^{NP}\)

- \([\text{BHPT 20}] \ \text{Simpler (PCPs } \rightarrow \text{Matrix rigidity)}\)

- \([\text{BHPT 20}] \ \text{Tight w.r.t. the PCP parameters!}\)
Probabilistically Checkable Proofs

• Suppose we want to prove a mathematical statement $\phi$ (Think of $\phi$: a given instance of 3SAT is satisfiable)

• PCPs provide a way to verify the claim $\phi$, by reading the proof at a few locations.

• The prover needs to write the proof in a specific format.
Probabilistically Checkable Proofs

• **Completeness:** “Correct claims can always be proven”
  
  If $\phi$ is true, then there exists $\pi^*$
  
  $$\Pr_R[V^{\pi^*}(\phi, R) = 1] \geq c$$

• **Soundness:** “Incorrect claims cannot be proven”
  
  If $\phi$ is false, then for all $\pi$
  
  $$\Pr_R[V^\pi(\phi, R) = 1] \leq s$$
Probabilistically Checkable Proofs

• Other important parameters
  • **Size** of the proof
  • Number of **queries**
  • **Gap** between the completeness (c) and soundness (s)
  • Verifier’s **running time** $V(\phi, R)$
  • **Smoothness**: all locations from $\pi$ are equally likely to be queried
Probabilistically Checkable Proofs

- For any language in NTIME($T(n)$), there exist PCPs with the following parameters:
  - **Size** of the proof: $T \cdot \text{poly}(\log T)$ [BGHSV 1]
  - **Number of queries**: $q = O(1)$
  - **Gap** between the completeness and soundness (1 vs. $\epsilon$) for any $\epsilon > 0$
  - **Verifier’s running time** $V(\phi, R)$: $\text{poly}(\log T)$ [BGHSV 2]
  - **Smoothness**: all locations from $\pi$ are equally likely to be queried
PCPs to Rigid Matrix (Overview) [Alman-Chen 19]

• L be any unary language in NTIME($2^n$) \ NTIME($2^n/n$). Given $x = 1^n$

• Let $\pi$ be the proof of “$x \in L$” written in a matrix form

• $\pi$ cannot be $\delta$—approximated (hamming distance) by a low rank matrix

• If it were then we will put $L \in$ NTIME($2^n/n$), a CONTRADICTION!
Overview

• L be any unary language in NTIME($2^n$) \ NTIME($2^n/n$). Given $x = 1^n$

• Let $\pi$ be the proof of “$x \in L$” written in a matrix form

• $\pi$ cannot be $\delta$—approximated (hamming distance) by a low rank matrix

  • If it were then let $A, B$ be the low rank decomposition

  • (Guess $A, B$) Simulate verifier on $A \cdot B$

    • Completeness: Accepted with probability $1 - q\delta$

    • Soundness : Accepted with probability < 0.0001

  • If the overall verification is done in time < $2^n/n$, then CONTRADICTION!

Since $|\pi| = 2^n \cdot \text{poly}(n), |A| + |B| \ll 2^n/n$

Smoothness and $\pi^* \approx_\delta A \cdot B$
Overview

• L be any unary language in $\text{NTIME}(2^n) \setminus \text{NTIME}(2^n/n)$. Given $x = 1^n$

• Let $\pi$ be the proof of “$x \in L$” written in a matrix form

• $\pi$ cannot be $\delta$—approximated (hamming distance) by a low rank matrix

• If it were then let $A, B$ be the low rank decomposition

• (Guess $A, B$) Simulate verifier on $A \cdot B$
  
  • Completeness: Accepted with probability $1 - q\delta$
  
  • Soundness: Accepted with probability $< 0.0001$

• If the overall verification is done in time $< 2^n/n$, then CONTRADICTION!
NTIME machine outputting rigid matrices

- On input $1^N$, the machine does the following:
  - Let $L$ be the language from the previous slide
  - Set $x = 1^n$ ($N = 2^n$)
  - Guess the “proof” $\pi$ of the statement “$x \in L$” ($\pi \in \mathbb{F}_2^{2^{n/2} \times 2^{n/2}}$)
  - Output the matrix $\pi$

Claim: For infinitely many $N$, the machine outputs a rigid matrix.
Overview

• L be any unary language in NTIME($2^n$) \ NTIME($2^n/n$). Given $x = 1^n$

• Let $\pi$ be the proof of “$x \in L$” written in a matrix form

• $\pi$ cannot be $\delta$—approximated (hamming distance) by a low rank matrix

  • If it were then let $A, B$ be the low rank decomposition

  • (Guess $A, B$) Simulate verifier on $A \cdot B$ (needs to be done in $< 2^n/n$ time)

    • Completeness: Accepted with probability $1 - q\delta$

    • Soundness: Accepted with probability $< 0.0001$

  • If the overall verification is done in time $< 2^n/n$, then CONTRADICTION!
Overview

• L be any unary language in NTIME($2^n$) \ NTIME($2^n/n$). Given $x = 1^n$

• Let $\pi$ be the proof of “$x \in L$” written in a matrix form

• $\pi$ cannot be $\delta$—approximated (hamming distance) by a low rank matrix

  • If it were then let $A, B$ be the low rank decomposition

  • (Guess $A, B$) Simulate verifier on $A \cdot B$ \ needs to be done in $< 2^n/n$ time\)

    • Completeness: Accepted with probability $1 - q\delta$

    • Soundness : Accepted with probability $< 0.0001$

  • If the overall verification is done in time $< 2^n/n$, then CONTRADICTION!

\[\frac{\pi}{\pi - \delta} - A, B < 2^n/n\]

\[\text{[AC 19] Boils down to fast counting \#1s in a product of low rank matrices}\]
Rest of the talk

• Introduce rectangular PCPs

• Convince that the simulation can be done in \( < \frac{2^n}{n} \) non-deterministic time

• How the fast counting is used in this process
Each query location depends on the full randomness \( R \)
Rectangular PCPs

1. Proof as a matrix

2. Row and column indices depend on the first and the second half of the randomness, respectively.
\( \tau \)-almost-rectangular PCPs

1. Proof as a matrix

2. Row index depends on \((R_{\text{row}}, R_{\text{shared}})\) and the column index depends on \((R_{\text{col}}, R_{\text{shared}})\).

3. \(|R_{\text{shared}}| = \tau \cdot |R|\)

4. \(|R_{\text{row}}| = |R_{\text{col}}| = \frac{1 - \tau}{2} |R|\)
Main Theorem

• [BHPT 20] Fix any $\epsilon, \tau > 0$. For every language $L \in \text{NTIME}(2^n)$, there exists a rectangular PCP with the following parameters:

  • Completeness 1 and soundness $\epsilon$

  • Query complexity $O(1)$

  • Proof size $2^n \cdot \text{poly}(n)$ (Randomness complexity = $n + O(\log n)$)

  • Verifier’s run-time $2^{en}$

  • Smooth and $\tau$—almost rectangular
Almost-rectangular PCP $\rightarrow$ Rigid Matrices

- L be any unary language in $\text{NTIME}(2^n) \setminus \text{NTIME}(2^n/n)$. Given $x = 1^n$

- Let $\pi$ be the (almost-rectangular) proof.

- $\pi$ cannot be $\delta$—approximated (hamming distance) by a low rank matrix

  - If it were then let $A, B$ be the low rank decomposition of $\pi$

  - (Guess $A,B$) **Simulate verifier on** $A \cdot B$

    - Completeness: Accepted with probability $1 - q\delta$

    - Soundness: Accepted with probability $< 0.0001$

  - If the overall verification is done in time $< 2^n/n$, then **CONTRADICTION!**
Simulate verifier on $A \cdot B$

For simplicity, assume that the verifier is querying 3 bits and accepting iff the parity of the three bits is 1.
Counting #1s in a prod. of low rank matrices

Fix $R_{\text{shared}} = z$

$A_k := i$-th row of $A_k = q_k^{\text{row}}(i, z)$-th row of $A$

$B_k := j$-th col of $B_k = q_k^{\text{col}}(j, z)$-th col of $B$

$M^z(i, j) = \text{parity of the 3 bits queried by the verifier on randomness } (i, R_{\text{shared}}, j)$
Simulation

Simulate verifier on $A \cdot B$

- For every $z \in \{0,1\}^{|R_{shared}|}$
  - Calculate the fraction of 1s in $M^z$. Let the fraction be $p_z$
  - Acceptance probability on the “proof” $A \cdot B$ is $E_z[p_z]$

Total running time of the simulation: $2^{\tau n} \cdot (2^{n/2-\tau/2} \cdot \rho \cdot 6 \cdot 2^{en} + \text{calculate } p_z)$
Fast counting

Total running time of the simulation: \( 2^{\tau n} \cdot \left( 2^{n/2 - \tau/2} \cdot \rho \cdot 6 \cdot 2^{en} + \text{calculate } p_z \right) \)

- Given two matrices \( X \in \mathbb{F}_2^{N \times r} \) and \( Y \in \mathbb{F}_2^{r \times N} \), compute the number of 1s in \( X \cdot Y \)

- [Chan-Williams 16] Can be done in time roughly \( N^{2 - \frac{1}{\log r}} \) (provided \( r = N^{o(1)} \))
Fast counting

Total running time of the simulation:

\[
2^{\tau n} \cdot \left( 2^{n/2 - \tau/2} \cdot \rho \cdot 6 \cdot 2^{en} + 2^{(1-\tau)n - \frac{n}{\log \rho}} \right)
\]

- Given two matrices \(X \in \mathbb{F}_2^{N \times r}\) and \(Y \in \mathbb{F}_2^{r \times N}\), compute the number of 1s in \(X \cdot Y\)

- [Chan-Williams 16] Can be done in time roughly \(N^{2 - \frac{1}{\log r}}\) (provided \(r = N^{o(1)}\))
Finishing the proof

Total running time of the simulation: \(\frac{2^n}{2^{\log \rho}} = \frac{2^n}{n}\) (if \(\rho \approx 2^{\log n}\))

- Given two matrices \(X \in \mathbb{F}_2^{N \times r}\) and \(Y \in \mathbb{F}_2^{r \times N}\), calculate the number of 1s in \(X \cdot Y\)

- [Chan-Williams 16] Can be done in time roughly \(N^{2 - \frac{1}{\log r}}\) (provided \(r = N^{o(1)}\))
Open questions

• Even faster algorithm for counting #1s in a product of low rank matrices

• Algorithm for higher ranks ($r = N^c$)

• Other complexity implications of this framework? e.g. Rectangular rigidity?