ON TENSOR-RANK AND LOWER BOUNDS FOR ARITHMETIC FORMULAS

ALI KARIM LALANI†

Abstract. This write-up explores the properties of tensor rank and its implications for algebraic complexity theory. We discuss the existence of high-rank random tensors and the difficulty of constructing explicit examples with non-trivial lower bounds. Central to this discussion is the breakthrough by Ran Raz, which establishes that sufficiently strong lower bounds for the rank of explicit tensors imply super-polynomial lower bounds for general arithmetic formulas. We provide a sketch of the proof and examine related ideas and open questions at the end.

1. Introduction

A tensor T is essentially a multidimensional array of scalars over a field $\mathbb{F}$. Formally, it is a function $T : [n]^d \to \mathbb{F}$ that maps a d-tuple of indices to a scalar. This provides a "lookup rule": given an address $(i_1, \ldots, i_d)$, the tensor returns the value stored at that location. While cases where $d = 1$ and $d = 2$ represent vectors and matrices, higher-dimensional tensors ($d \geq 3$) exhibit more complex structural properties.

Definition 1.1. The rank of a d-dimensional tensor T, denoted $rk(T)$, is the minimum number of rank-1 tensors required to represent T as their sum:

$$rk(T) = \min \left\{ r : T = \sum_{i=1}^{r} u_i^{(1)} \otimes u_i^{(2)} \otimes \cdots \otimes u_i^{(d)} \right\}$$

where each $u_i^{(j)} \in \mathbb{F}^n$.

This is a natural extension of matrix rank, which can be characterized as the smallest r such that $M = \sum_{i=1}^{r} u_i v_i^T$. However, unlike the matrix case, determining this value for higher-order tensors is computationally difficult.

Theorem 1.2 ([Has90]). *Computing the rank of a tensor is NP-Hard.*

2. Basic Bounds

A natural question is the range of values $rk(T)$ can take for a given dimension d and side length n. While a random tensor is expected to have high rank, establishing even simple upper bounds provides a baseline for complexity.

Date: April 2026.

†University of Toronto. Email: ali@cs.toronto.edu.

Theorem 2.1. *Any tensor* $T : [n]^d \to \mathbb{F}$ *has* $rk(T) \leq n^{d-1}$.

Proof. Let $\{e_1, \dots, e_n\}$ be the standard basis for $\mathbb{F}^n$. We can decompose T by "freezing" the first $d-1$ indices. For each fixed $(d-1)$-tuple of indices $(i_1, \dots, i_{d-1}) \in [n]^{d-1}$, we define a vector $v^{(i_1,\dots,i_{d-1})} \in \mathbb{F}^n$ such that its k-th component is $T(i_1, \dots, i_{d-1}, k)$. The entire tensor can then be written as a sum of n^{d-1} terms:

$$T = \sum_{i_1=1}^{n} \cdots \sum_{i_{d-1}=1}^{n} e_{i_1} \otimes \cdots \otimes e_{i_{d-1}} \otimes v^{(i_1,\dots,i_{d-1})}$$

Since each term $e_{i_1} \otimes \cdots \otimes e_{i_{d-1}} \otimes v^{(i_1,\dots,i_{d-1})}$ is a rank-1 tensor, it follows that $rk(T) \leq n^{d-1}$. □

Despite this upper bound, a simple dimension-counting argument shows that most tensors possess high rank. Heuristically, a generic tensor requires n^d parameters to specify, while a rank-1 tensor is specified by dn parameters. This leads to the heuristic that the rank of a generic tensor is approximately $rk(T) \geq \frac{n^{d-1}}{d}$ [Wig18]. However, as noted by Wigderson, the current state of explicit lower bounds is "pathetic"; we still struggle to construct explicit tensors with rank significantly exceeding n for $d = 3$.

Currently, the most robust method for proving explicit lower bounds involves *flattening* the d-dimensional tensor into a 2D matrix. By partitioning the d indices of a tensor into two disjoint sets, we can view the tensor as a matrix M.

Theorem 2.2. *For any flattening* M *of a tensor* T, $rk(T) \geq rank(M)$.

This theorem is a folklore result (see, e.g., [Raz10] and [AFT11]) that enables the use of standard linear algebra by partitioning the k dimensions of a tensor T into two disjoint sets to construct an $n^{\lfloor k/2 \rfloor} \times n^{\lceil k/2 \rceil}$ matrix M. The proof sketch relies on the observation that if T has rank one, then M must also have rank one, which implies that $rk(T) \geq rank(M)$. Consequently, if M is full rank, this yields an explicit lower bound of $\Omega(n^{\lfloor k/2 \rfloor})$ [NW96]. For example, a symmetric tensor defined by $T(i_1, \dots, i_k) = 1$ if and only if $|\{i_1, \dots, i_k\}| = k$ achieves a lower bound of $n^{\lfloor k/2 \rfloor}$. While this remains the best currently known explicit lower bound, significantly tighter bounds are required to prove results regarding the formula size of the permanent.

3. Tensors and Algebraic Complexity

The study of tensor rank is motivated by its connection to algebraic complexity theory. Any d-dimensional tensor can be represented as a set-multilinear polynomial, allowing us to translate questions of multilinear algebra into questions about circuit complexity.

Definition 3.1. Let $T : [k]^d \to \mathbb{F}$ be a d-dimensional tensor. The *associated polynomial* f_T in variables $\{X_{i,j} : i \in [d], j \in [k]\}$ is defined as:

$$f_T(X_{1,1}, \dots, X_{d,k}) = \sum_{(i_1,\dots,i_d) \in [k]^d} T(i_1, \dots, i_d) \cdot \prod_{j=1}^{d} X_{j,i_j}$$

Each monomial in f_T contains exactly one variable from each of the d sets of variables. Crucially, the tensor rank of T corresponds exactly to the size of the smallest depth-3 ($\Sigma\Pi\Sigma$) set-multilinear formula required to compute f_T.

An earlier result by Strassen provides a more general lower bound for circuit size based on tensor rank.

Theorem 3.2 ([Str73]). *Let T be a three-dimensional tensor of tensor rank at least R. Then the polynomial $F(x, y, z) = \sum_{i,j,k\in[n]} T(i, j, k)x_i y_j z_k$ requires circuits of size at least $\Omega(R)$ over $\mathbb{R}$.*

The significance of these lower bounds was realized by the breakthrough result of Raz [Raz10]. Raz showed that sufficiently strong lower bounds on tensor rank can be lifted to prove super-polynomial lower bounds for general arithmetic formulas.

Theorem 3.3 ([Raz10]). *Any explicit example of a tensor $A : [n]^r \to \mathbb{F}$ with tensor-rank $\geq n^{r\cdot(1-o(1))}$, where $r \leq O(\log n/\log\log n)$ is super-constant, implies an explicit super-polynomial lower bound for the size of general arithmetic formulas over $\mathbb{F}$.*

This result shows that just finding a single tensor with sufficiently high rank would resolve a major open problem in algebraic complexity: finding a polynomial that is hard for any general formula.

4. Homogenization of Arithmetic Formulas

Raz begins by proving this homogenization result. This theorem, alongside the multilinearization theorem, is used at the outset of the proof of the main theorem.

Historically, Strassen [Str73] showed that any arithmetic circuit of size s computing a degree-r homogeneous polynomial can be converted into a homogeneous circuit of size $poly(s, r)$. However, for formulas, the standard approach suggests an exponential blowup of $s^{\Omega(r)}$, which led Nisan and Wigderson [NW96] to conjecture that homogenization is inherently expensive for formulas. Raz refutes this for small degrees, providing the necessary first step to lift lower bounds from tensors to general formulas.

Theorem 4.1 (Homogenization). *Let Φ be a fan-in 2 formula of size s and product-depth d that computes a homogeneous polynomial of degree r. Then there exists a fan-in 2 homogeneous formula Ψ of size $O\left(\binom{d+r+1}{r} \cdot s\right)$ and product-depth d that computes the same polynomial.*

Proof Sketch. The construction relies on tracking the degree progression along the unique path from any node to the root. For each node $u \in \Phi$, we define a state space N_u consisting of functions $D : path(u) \to \{0, \dots, r\}$ such that degrees only change at product gates and remain constant across sum gates.

We build Ψ by creating nodes (u, D) for every $u \in \Phi$ and $D \in N_u$, where each node computes the homogeneous part of degree $D(u)$ of the original node u. At a product gate $u = v \cdot w$, the formula performs a discrete convolution:

$$\hat{\Psi}_{(u,D)} = \sum_{i=0}^{D(u)} \hat{\Psi}_{(v,D_{v,i})} \cdot \hat{\Psi}_{(w,D_{w,D(u)-i})}$$

The size blowup is determined by $|N_u|$. Since there are at most d product gates on any path, we are essentially distributing a degree budget of r into $d + 1$ slots. By the stars and bars formula, this is exactly $\binom{d+r+1}{r}$. This construction was verified as part of HW1 Q5. $\square$

Corollary 4.2. Let $f \in \mathbb{F}[x_1, \dots, x_n]$ be a homogeneous polynomial of degree $r \leq O(\log n)$. If there exists a polynomial size formula for f, then there exists a polynomial size homogeneous formula for f.

5. Multilinearization

To now consider set-multilinear polynomials, we track which variable sets are being used. Let $X_1, \dots, X_r$ be disjoint sets of variables.

Definition 5.1. For vectors $a, b \in \{0,1\}^r$, we say $b \leq a$ if $b(i) \leq a(i)$ for all $i \in \{1, \dots, r\}$. A monomial q is **set-multilinear of type** a if it is multilinear and contains exactly one variable from X_i if $a(i) = 1$, and no variables from X_i if $a(i) = 0$.

We let f_a denote the set-multilinear part of polynomial f of type a. Similarly, for an arithmetic formula Φ, let $\hat{\Phi}_{u,a}$ denote the type-a part of the polynomial computed at node u. Just as Theorem 1 homogenized a formula, Theorem 3 extracts its set-multilinear part.

Theorem 5.2 (Multilinearization)**.** *Let Φ be a fan-in 2 formula of size s and product-depth d that computes a set-multilinear polynomial over the disjoint sets $X_1, \dots, X_r$. Then there exists a fan-in 2 set-multilinear formula Ψ of size*

$$\mathit{Size}(\Psi) = O((d+2)^r \cdot s)$$

and product-depth d that computes the same polynomial.

Contrast with Homogenization: The blowup is now bounded by $(d+2)^r$ instead of $\binom{d+r+1}{r}$.

Proof Sketch. The proof structure is identical to the homogenization theorem, but instead of tracking an integer degree, we track the **type vector** $a \in \{0,1\}^r$.

- **State Space** N_u**:** The set of functions $D : \mathit{path}(u) \to \{0,1\}^r$.
- **Constraints:**
  - Sum gate v: $D(v) = D(\mathit{child})$.
  - Product gate v: $D(v) \geq D(\mathit{child})$ (bitwise).
- **Inductive Step:** We construct nodes (u, D) computing $\hat{\Psi}_{(u,D)} = \hat{\Phi}_{u,D(u)}$. For a product gate $u = v \cdot w$, we sum over all valid type partitions:
$$\hat{\Psi}_{(u,D)} = \sum_{a \leq D(u)} \hat{\Psi}_{(v,D_{v,a})} \cdot \hat{\Psi}_{(w,D_{w,D(u)-a})}$$
- **Blowup:** Each of the r coordinates can flip from 0 to 1 at most once along the $\leq d$ product gates. This bounds $|N_u|$ by $(d+2)^r$.

□

Corollary 5.3. Let f be a set-multilinear polynomial over sets $X_1, \dots, X_r$ of size n each. If $r \leq O(\log n / \log\log n)$ and f has a poly-size formula, then it has a poly-size **set-multilinear** formula.

6. Tensor-Rank and Formula Size

Raz's main breakthrough is the formal connection between arithmetic formula size and the rank of its associated tensor. This result lifts depth-3 lower bounds to general formula lower bounds.

Theorem 6.1 (The Main Connection)**.** *Let $n > 1$ and let $A : [n]^r \to \mathbb{F}$ be a tensor where $r \leq O(\log n / \log \log n)$. If there exists a formula of size n^c for the associated polynomial f_A, then the tensor-rank of A is at most:*

$$rk(A) \leq n^{r\cdot(1-2^{-O(c)})}$$

The proof proceeds in three main stages: multilinearization, the notion of syntactic rank, and a global optimization argument.

1. Normal Form and Multilinearization: We begin by assuming the existence of a formula Φ of size n^c. By the Multilinearization Theorem, we can assume without loss of generality that Φ is a set-multilinear formula of size $n^{O(c)}$. We further restrict Φ to the notion of a **normal form** where the formula is a strict binary tree, sum gates are collapsed, and every leaf computes a product of a scalar and a single variable.

2. Syntactic-Rank: Raz defines the **syntactic-rank** (*syn-rank*) inductively based on the formula's topology:

- For a leaf u, $\textit{syn-rank}(\Psi_u) = 1$.
- For a sum gate u with children u_i, $\textit{syn-rank}(\Psi_u) = \min(n^{r'-1}, \sum \textit{syn-rank}(\Psi_{u_i}))$.
- For a product gate u with children u_1, u_2, $\textit{syn-rank}(\Psi_u) = \textit{syn-rank}(\Psi_{u_1}) \cdot \textit{syn-rank}(\Psi_{u_2})$.

This inductive measure uses the subadditivity and submultiplicativity of tensor rank to provide a provable upper bound: $rk(A) \leq \textit{syn-rank}(\Phi)$.

3. Extended-Formulas and Optimization: To analyze the relationship between size and rank, Raz introduces **extended-formulas**, which assign real-valued weights $w \geq 1$ to nodes to track sub-formula duplication. The core of the proof is a topological optimization argument:

- The space of extended-formulas of a fixed size s is compact, ensuring the existence of a formula that maximizes syntactic-rank.
- In an optimal formula, all sum gates have fan-in 1 and are effectively removed, leaving only product gates.
- A structural analysis reveals that the number of full-rank nodes (those with weight n) along any path is strictly bounded by $O(\log_n s)$.
- This forces a significant portion of the formula to consist of low-rank components, leading to the final upper bound of $n^{r-r/2^{O(c)}}$.

The main result is basically summarized by the contrapositive, which gives a concrete target for proving formula lower bounds.

Corollary 6.2. Let $A : [n]^r \to \mathbb{F}$ be a tensor such that $r \leq O(\log n / \log \log n)$ is super-constant. If the tensor-rank of A satisfies $rk(A) \geq n^{r\cdot(1-o(1))}$, then there is no polynomial-size formula for the polynomial f_A.

7. Aside : Border Rank

While the connection between tensor rank and arithmetic formulas provides the primary motivation for this write-up, the geometry of tensors behaves quite differently from standard matrix algebra.

7.1. **Border Rank.** For matrices ($d = 2$), the set of tensors with rank at most r is a variety defined by the vanishing of minors. However, for higher-order tensors ($d \geq 3$), the set $V_r = \{T : rk(T) \leq r\}$ is not closed. Instead, we consider the **Zariski closure** $\overline{V}_r$, which gives rise to the notion of **border rank**, denoted $brk(T)$. This is the smallest r such that T is a limit of rank-r tensors:

$$\overline{V}_r = \{T : \exists\{T_\epsilon\} \text{ s.t. } T_\epsilon \xrightarrow{\epsilon\to 0} T \text{ and } rk(T_\epsilon) \leq r\}$$

A concrete example of this phenomenon occurs in $2 \times 2 \times 2$ tensors. We can represent such a tensor using **slices**, which are 2D cross-sections obtained by fixing one of the three indices. For example, we can view a $2 \times 2 \times 2$ tensor as a pair of 2×2 matrices, M_1 and M_2, representing the front and back layers of the grid. Consider the tensor T defined by the two slices:

$$M_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad M_2 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$$

The true tensor rank of this specific tensor is $rk(T) = 3$. However, it can be represented as the limit of the difference of two rank-1 tensors:

$$T = \lim_{\epsilon\to 0} \left[\underbrace{\left(\begin{pmatrix} 1/\epsilon & 1 \\ 1 & \epsilon \end{pmatrix} \middle| \begin{pmatrix} 1 & \epsilon \\ \epsilon & \epsilon^2 \end{pmatrix} \right)}_{\text{Rank 1}} - \underbrace{\left(\begin{pmatrix} 1/\epsilon & 0 \\ 0 & 0 \end{pmatrix} \middle| \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \right)}_{\text{Rank 1}} \right]$$

As $\epsilon \to 0$, the first slice becomes $\begin{pmatrix} 0 & 1 \\ 1 & \epsilon \end{pmatrix} \to M_1$ and the second becomes $\begin{pmatrix} 1 & \epsilon \\ \epsilon & \epsilon^2 \end{pmatrix} \to M_2$. Since T is the limit of a sum of two rank-1 tensors, $brk(T) \leq 2$.

This gap remains an open question; we still seek a function $f(d)$ that bounds the ratio $rk(T)/brk(T)$. Understanding border rank is important for the complexity of matrix multiplication, as shown in the work of Bini [Bin80] and Strassen [Str87]. This builds on Strassen's classical divide and conquer algorithm, which demonstrated that the 2×2 matrix multiplication tensor has rank (and border rank) at most 7, yielding a complexity of $O(n^{\log_2 7})$.

Acknowledgements

The author would like to thank Professor Shubhangi Saraf for her very engaging instruction and for the opportunity to participate in this course. Her guidance in both undergraduate and algebraic complexity theory has been invaluable to my development in the field.

The author is also grateful to Avi Wigderson; his lecture played a pivotal role in my understanding of tensor rank. Furthermore, the opportunity to meet and speak with him during the Blyth Lecture series was a profound source of motivation and inspiration.

Bibliography

[Has90] J. Håstad. *Tensor Rank is NP-complete*. Journal of Algorithms, 11(4):644–654, 1990.

[Raz10] R. Raz. *Tensor-rank and lower bounds for arithmetic formulas*. STOC '10: Proceedings of the forty-second ACM symposium on Theory of computing, 2010.

[Wig18] A. Wigderson. *Tensor Rank*. IAS Lecture (YouTube), Available at https://youtu.be/9ccuJWoIHrw, 2018.

[AFT11] B. Alexeev, M. Forbes, and J. Tsimerman. *Tensor Rank: Some Lower and Upper Bounds*. IEEE Conference on Computational Complexity, pages 283–291, 2011.

[NW96] N. Nisan and A. Wigderson. *Lower bound on arithmetic circuits via partial derivatives*. Computational Complexity, 6:217–234, 1996.

[Str73] V. Strassen. *Vermeidung von Divisionen*. Journal für die reine und angewandte Mathematik, 264:184–202, 1973.

[Bin80] D. Bini. *Border rank of a* $p \times q \times 2$ *tensor and the optimal approximation of a pair of bilinear forms*. Automata, Languages and Programming (ICALP), Lecture Notes in Computer Science, vol. 85, Springer, Berlin, Heidelberg, 1980.

[Str87] V. Strassen. *Relative bilinear complexity and matrix multiplication*. Journal für die reine und angewandte Mathematik, 375/376:406–443, 1987.