Multi-Linear Formulas for Permanent and Determinant are of Super-Polynomial Size*

Ran Raz†
Weizmann Institute
ranraz@wisdom.weizmann.ac.il

Abstract

An arithmetic formula is multilinear if the polynomial computed by each of its sub-formulas is multilinear. We prove that any multilinear arithmetic formula for the permanent or the determinant of an $n \times n$ matrix is of size super-polynomial in n. Previously, super-polynomial lower bounds were not known (for any explicit function) even for the special case of multilinear formulas of constant depth.

1 Introduction

Arithmetic formulas for computing the permanent and the determinant of a matrix have been studied since the 19th century. Are there polynomial size formulas for these functions ? Although the permanent and the determinant are among the most extensively studied computational problems, polynomial size formulas for these functions are not known. The smallest known formula for the permanent of an $n \times n$ matrix is of size $O(n^2 2^n)$. The smallest known formula for the determinant of an $n \times n$ matrix is of size $n^{O(\log n)}$. An outstanding open problem in complexity theory is to prove that polynomial size formulas for these functions do not exist. Note, however, that super-polynomial lower bounds for the size of arithmetic formulas are not known for any explicit function and that questions of this type are considered to be among the most challenging open problems in theoretical computer science.

In this paper, we prove super-polynomial lower bounds for the subclass of *multilinear formulas*. An arithmetic formula is *multilinear* if the polynomial computed by each of its sub-formulas is multilinear (as a formal polynomial), that is, in each of its monomials the power of every input variable is at most one.

*A preliminary version of the paper appeared in STOC 2004 [R1]
†Research supported by Israel Science Foundation (ISF) grant.

1.1 Multilinear Formulas

Let F be a field, and let $\{x_1, ..., x_m\}$ be a set of input variables. An *arithmetic formula* is a binary tree whose edges are directed towards the root. Every leaf of the tree is labelled with either an input variable or a field element. Every other node of the tree is labelled with either $+$ or $\times$ (in the first case the node is a *plus gate* and in the second case a *product gate*).

An arithmetic formula computes a polynomial in the ring $F[x_1, ..., x_m]$ in the following way. A leaf just computes the input variable or field element that labels it. A plus gate computes the sum of the two polynomials computed by its sons. A product gate computes the product of the two polynomials computed by its sons. The *output* of the formula is the polynomial computed by the root. For a formula Φ, we denote by $\hat{\Phi}$ the output of the formula, that is, the polynomial computed by the formula. The *size* of a formula Φ is defined to be the number of nodes in the tree, and is denoted by $|\Phi|$.

A polynomial in the ring $F[x_1, ..., x_m]$ is *multilinear* if in each of its monomials the power of every input variable is at most one. An arithmetic formula is *multilinear* if the polynomial computed by each gate of the formula is multilinear.

1.2 Motivation

Multilinear formulas are restricted, as they do not allow the intermediate use of higher powers of variables in order to finally compute a certain multilinear function. Note, however, that for many multilinear functions, formulas that are not multilinear are very counter-intuitive, as they require a "magical" cancellation of all high powers of variables. For many multilinear functions, it seems "obvious" that the smallest formulas should be multilinear. (This is not necessarily the case for every function, and in particular, as noted below, may not be true for the determinant).

Multilinear polynomials are very powerful and are extensively used in theoretical computer science. Hence, the class of multilinear formulas seems to be quite strong and it is very interesting to study its computational power.

Note also that both the permanent and the determinant are multilinear functions in the input variables and that many of the well known formulas for these functions are multilinear formulas. In particular, the smallest known arithmetic formula for the permanent is multilinear.[1]

Finally, we note that several classes of formulas that were studied in the past are subclasses of multilinear formulas. One example is *monotone arithmetic formulas* (see for example [S, SS, V1]). It is easy to see that a monotone arithmetic formula for a multilinear function is always multilinear.

[1]For the determinant, the smallest known formulas are not multilinear. (Sub-exponential size multilinear formulas for the determinant are not known.)

1.3 Our Results

We prove that over any field, any multilinear arithmetic formula for the permanent or the determinant of an $n \times n$ matrix is of size $n^{\Omega(\log n)}$.

An obvious corollary of our result is that over any field, any multilinear arithmetic circuit[2] for the permanent or the determinant of an $n \times n$ matrix is of depth $\Omega(\log^2 n)$.

Our method is quite general and can be applied for many other functions. For simplicity of the presentation, we concentrate in this paper on the permanent and the determinant, as they are the most interesting functions that we are able to apply our bounds for. In a subsequent paper, we use the same methods to prove lower bounds for multilinear formulas for other functions, and we use this to separate multilinear circuit size from multilinear formula size [R2].

1.4 Previous Work

The best lower bound for the size of general arithmetic formulas for the permanent or the determinant of an $n \times n$ matrix is a lower bound of $\Omega(n^3)$ [K]. Super-polynomial lower bounds for the size of general arithmetic formulas are not known for any explicit function. Such bounds are only known for restricted cases. Among the most interesting results are the exponential lower bound for the *non-commutative* case [N], and the exponential lower bound for formulas of depth 3 (over finite fields) [GK, GR].[3]

Multilinear arithmetic formulas were formally defined in [NW], although they were already considered in earlier works (see for example [V2]). Previous to our result, lower bounds for the size of multilinear formulas were not known even for formulas of constant depth. Exponential lower bounds for a variant of constant depth multilinear formulas were obtained in [NW].[4] Lower bounds for several other restricted subclasses of multilinear formulas were obtained in [N, NW, RS].

For general background on algebraic complexity theory and on the arithmetic complexity of the permanent and the determinant, see [G1, G2, BCS].

[2]An arithmetic circuit is *multilinear* if the polynomial computed by each of its gates is multilinear (as a formal polynomial).

[3]The *depth* of a formula is defined to be the maximal number of changes of labels (i.e., maximal number of changes of gates from $+$ to $\times$ or vice-versa) along a path from a leaf to the root of the formula (where the maximum is taken over all such paths). In this paper we will not need this notion, except for this subsection.

[4]These bounds were obtained for the weaker model of constant depth *set-multilinear* formulas, where the variables are partitioned into sets (e.g., the rows of a matrix) and it is assumed in addition that in the polynomial computed by each gate of the formula each monomial contains at most one variable from every set.

1.5 Methods

The starting point for our proof is the partial derivatives method of Nisan and Wigderson [N, NW]. It was suggested in [NW] that for certain restricted subclasses of arithmetic formulas (and circuits), the dimension of the space spanned by all partial derivatives of the output is quite small. The method was used in [N, NW, RS, SW] to obtain lower bounds for several subclasses of formulas and circuits. Note, however, that for multilinear formulas the dimension of the space spanned by all partial derivatives may be very large, even if the formula is of linear size. In particular, that dimension may be much larger than the dimension of the space spanned by all partial derivatives of the permanent or the determinant. Nevertheless, the set of partial derivatives still plays a crucial roll in our proof.

To handle sets of partial derivatives, we make use of the *partial derivatives matrix*, discussed below in Section 3. The partial derivatives matrix was first used for proving lower bounds by Nisan [N], and was later on used in several other works.

In our proof, we also use rank arguments as well as random restrictions. Both these methods were used for proving lower bounds in numerous of works. However, we use them here in a completely different way. For example, random restrictions were used in many works in order to eliminate gates. Here, we use random restrictions in order to make gates unbalanced (see Section 4) without eliminating even a single gate.

1.6 Discussion

It is still not known whether or not general arithmetic formulas are more powerful than multilinear arithmetic formulas. In order to prove a super-polynomial gap between these two classes, one has to show the existence of a multilinear polynomial $f : \mathrm{F}^n \to \mathrm{F}$ such that:

1. There exists a small (i.e., polynomial size) arithmetic formula for f.
2. Any multilinear arithmetic formula for f is large (i.e., is of super-polynomial size).

Whether or not such a function f exists is an important open problem.[5]

A very interesting open problem in complexity theory is to give a deterministic polynomial time algorithm for identity testing of arithmetic circuits (or formulas). Such an algorithm should get as an input an arithmetic circuit (or formula) and decide in deterministic polynomial time whether or not the output of the circuit is identically 0. A very interesting recent result of Impagliazzo and Kabanets shows tight connections between proving lower bounds for arithmetic circuits and designing such algorithms [IK]. In this work, we prove unconditional

[5] In a recent work [A], Scott Aaronson uses our results to construct a function $f : \{0,1\}^n \to R$, such that: 1) There exists a small (i.e., polynomial size) arithmetic formula for a polynomial $g : R^n \to R$ that extends f. 2) Any multilinear arithmetic formula for a polynomial $h : R^n \to R$ that extends f is large (i.e., is of super-polynomial size). Note, however, that the polynomial g in Aaronson's construction is not multilinear, and hence this construction doesn't settle the above mentioned open problem.

lower bounds for the subclass of multilinear formulas. Can our result be used to give a deterministic sub-exponential time algorithm for identity testing for multilinear formulas ?

Finally, we mention a beautiful and surprising application of our results in the field of quantum computation. In [A], Scott Aaronson defines a class of quantum states, called *tree states*, and argues that this is the class of states accepted by people who are currently skeptic about the possibility of constructing quantum computers that are able to be in arbitrary quantum states. Using our results, Aaronson gives explicit examples for quantum states that are not tree states and challenges experimental physicists to construct such quantum states in their labs. States that are not tree states seem to be essential for quantum computation.

2 Syntactic Multilinear Formulas

Let Φ be an arithmetic formula over the set of variables $\{x_1, ..., x_m\}$. For every node v in the formula, denote by Φ_v the sub-formula with root v, and denote by X_v the set of variables that appear in the formula Φ_v. We say that an arithmetic formula Φ is *syntactic multilinear* if for every product gate v of Φ, with sons v_1, v_2, the sets of variables X_{v_1} and X_{v_2} are disjoint.

Note that any syntactic multilinear formula is clearly multilinear. At the other hand, a multilinear formula is not necessarily syntactic multilinear. Nevertheless, the following proposition shows that without loss of generality we can assume that a multilinear formula is syntactic multilinear.

Proposition 2.1 *For any multilinear formula, there exists a syntactic multilinear formula of the same size that computes the same polynomial.*

Proof:
Let Φ be a multilinear formula. Let v be a product gate in Φ, with sons v_1, v_2, and assume that X_{v_1} and X_{v_2} both contain the same variable x_i. Since Φ is multilinear, $\hat{\Phi}_v$ is a multilinear polynomial and hence in at least one of the polynomials $\hat{\Phi}_{v_1}, \hat{\Phi}_{v_2}$ the variable x_i doesn't appear. W.l.o.g. assume that in the polynomial $\hat{\Phi}_{v_1}$ the variable x_i doesn't appear. Then every appearance of x_i in Φ_{v_1} can be replaced by the constant 0. By repeating this for every product gate in the formula, as many times as needed, we obtain a syntactic multilinear formula that computes the same polynomial. $\square$

3 The Partial-Derivatives Matrix

Let f be a multilinear polynomial over the set of variables $\{y_1, ..., y_m\} \cup \{z_1, ..., z_m\}$. For a multilinear monomial p in the set of variables $\{y_1, ..., y_m\}$ and a multilinear monomial q in the set of variables $\{z_1, ..., z_m\}$, denote by $M_f(p, q)$ the coefficient of the monomial pq in the polynomial f. Since the number of multilinear monomials in a set of m variables[6] is 2^m, we

[6] We only consider monomials with coefficient 1.

can think of M_f as a $2^m \times 2^m$ matrix, with entries in the field F. We will be interested in the rank of the matrix M_f over the field F.

Let Φ be a multilinear arithmetic formula over the set of variables $\{y_1, ..., y_m\} \cup \{z_1, ..., z_m\}$. For every node v in the formula, denote by Y_v the set of variables in $\{y_1, ..., y_m\}$ that appear in the formula Φ_v, and denote by Z_v the set of variables in $\{z_1, ..., z_m\}$ that appear in the formula Φ_v. Denote by $b(v)$ the average of $|Y_v|$ and $|Z_v|$ and denote by $a(v)$ their minimum. Denote, $d(v) = b(v) - a(v)$.

Recall that the output $\hat{\Phi}$ of the formula Φ is a multilinear polynomial over $\{y_1, ..., y_m\} \cup \{z_1, ..., z_m\}$. For simplicity, we denote the matrix $M_{\hat{\Phi}}$ also by M_Φ. For a node v in the formula, we denote the matrix M_{Φ_v} also by M_v. Note that in M_v all rows that do not correspond to monomials in the set of variables Y_v and all columns that do not correspond to monomials in the set of variables Z_v are identically zero. We will be interested in bounding the rank of the matrix M_v over the field F. The following propositions give basic tools to bound that rank. (Note, however, that the rank of M_v may be as large as 2^m (i.e., full rank), even if the formula Φ is of linear size).

Proposition 3.1 *Let Φ be a multilinear arithmetic formula over the set of variables $\{y_1, ..., y_m\} \cup \{z_1, ..., z_m\}$, and let v be a node in Φ. Then*

$$\mathbf{Rank}(M_v) \leq 2^{a(v)}.$$

Proof:
W.l.o.g. assume that $|Z_v| = a(v)$. The matrix M_v has at most $2^{|Z_v|}$ non-zero columns and hence its rank is at most $2^{a(v)}$. □

Proposition 3.2 *Let Φ be a multilinear arithmetic formula over the set of variables $\{y_1, ..., y_m\} \cup \{z_1, ..., z_m\}$, and let v be a plus gate in Φ. Denote the sons of v by v_1, v_2. Then*

$$\mathbf{Rank}(M_v) \leq \mathbf{Rank}(M_{v_1}) + \mathbf{Rank}(M_{v_2}).$$

Proof:
Follows since M_v is the sum of M_{v_1} and M_{v_2}. □

Proposition 3.3 *Let Φ be a syntactic multilinear arithmetic formula over the set of variables $\{y_1, ..., y_m\} \cup \{z_1, ..., z_m\}$, and let v be a product gate in Φ. Denote the sons of v by v_1, v_2. Then*

$$\mathbf{Rank}(M_v) = \mathbf{Rank}(M_{v_1}) \cdot \mathbf{Rank}(M_{v_2}).$$

Proof:
Since Φ is syntactic multilinear, Y_{v_1}, Y_{v_2} are disjoint and so are Z_{v_1}, Z_{v_2}. Hence, the matrix M_v (restricted to rows that represent monomials in Y_v and columns that represent monomials in Z_v) is the tensor product of M_{v_1} (restricted to monomials in Y_{v_1} and monomials in Z_{v_1})

and M_{v_2} (restricted to monomials in Y_{v_2} and monomials in Z_{v_2}). Hence, the rank of M_v is the product of the rank of M_{v_1} and the rank of M_{v_2}. $\square$

4 Unbalanced Nodes

Let Φ be a syntactic multilinear arithmetic formula over the set of variables $\{y_1, ..., y_m\} \cup \{z_1, ..., z_m\}$. We say that a node v is *k-unbalanced* if $d(v) \geq k$.

Let γ be a simple path from a leaf w to a node v of the formula Φ. We say that γ is *k-unbalanced* if it contains at least one k-unbalanced node. We say that γ is *central* if for every u, u_1 on the path γ, such that u_1 is a direct son of u (i.e., there is an edge from u_1 to u), we have $b(u) \leq 2b(u_1)$. Note that for every node u in the formula, with sons u_1, u_2, we have $b(u) \leq b(u_1) + b(u_2)$. Hence, by induction, for every node u in the formula, there exists at least one central path that reaches u. In particular, at least one central path reaches the root.

We say that a node v of the formula is *k-weak* if every central path that reaches v is k-unbalanced. We say that the formula Φ is *k-weak* if the root of the formula is k-weak, that is, every central path that reaches the root contains at least one k-unbalanced node. The following lemma shows that if a node v is k-weak then the rank of the matrix M_v can be bounded.

Lemma 4.1 *Let Φ be a syntactic multilinear arithmetic formula over the set of variables $\{y_1, ..., y_m\} \cup \{z_1, ..., z_m\}$, and let v be a node in Φ. If v is k-weak then*

$$\mathbf{Rank}(M_v) \leq |\Phi_v| \cdot 2^{b(v)-k/2}.$$

Proof:
First note that if v is k-weak then $b(v) \geq k$, since otherwise a path that reaches v cannot contain a k-unbalanced node. (Recall that there exists at least one central path that reaches v). The proof of the lemma is by induction on $|\Phi_v|$.

Case 1: v is a leaf.
In this case, the rank of M_v is at most 1, and the lemma follows since $b(v) \geq k$.

Case 2: v is k-unbalanced.
In this case, $a(v) = b(v) - d(v) \leq b(v) - k$. Hence, by Proposition 3.1, the rank of M_v is at most $2^{b(v)-k}$.

Case 3: v is a product gate.
Denote the sons of v by v_1, v_2. W.l.o.g. assume that v is not k-unbalanced (otherwise we can apply Case 2). Since the formula is syntactic multilinear, $b(v) = b(v_1) + b(v_2)$. W.l.o.g. assume that $b(v) \leq 2b(v_1)$. Since every central path that reaches v is k-unbalanced and since

v is not k-unbalanced, every central path that reaches v_1 is k-unbalanced. Hence, v_1 is k-weak and by induction,

$$\mathbf{Rank}(M_{v_1}) \leq |\Phi_{v_1}| \cdot 2^{b(v_1)-k/2}.$$

By Proposition 3.1,

$$\mathbf{Rank}(M_{v_2}) \leq 2^{a(v_2)}.$$

Hence by Proposition 3.3,

$$\mathbf{Rank}(M_v) \leq |\Phi_{v_1}| \cdot 2^{b(v_1)+a(v_2)-k/2}.$$

The lemma follows since $b(v_1) + a(v_2) \leq b(v_1) + b(v_2) = b(v)$ and since $|\Phi_{v_1}| \leq |\Phi_v|$.

Case 4: v is a plus gate.
Denote the sons of v by v_1, v_2. W.l.o.g. assume that v is not k-unbalanced (otherwise we can apply Case 2). Recall that $b(v) \leq b(v_1) + b(v_2)$ and as before assume w.l.o.g. that $b(v) \leq 2b(v_1)$. As before, it follows that v_1 is k-weak. We will separate the proof into two subcases, according to whether or not $b(v) \leq 2b(v_2)$.

If $b(v) \leq 2b(v_2)$ it follows in the same way that v_2 is also k-weak. Since $b(v) \geq b(v_1)$ and $b(v) \geq b(v_2)$, we have by induction

$$\mathbf{Rank}(M_{v_1}) \leq |\Phi_{v_1}| \cdot 2^{b(v)-k/2},$$

and

$$\mathbf{Rank}(M_{v_2}) \leq |\Phi_{v_2}| \cdot 2^{b(v)-k/2}.$$

Therefore by Proposition 3.2,

$$\mathbf{Rank}(M_v) \leq (|\Phi_{v_1}| + |\Phi_{v_2}|) \cdot 2^{b(v)-k/2},$$

and the proof follows since $|\Phi_{v_1}| + |\Phi_{v_2}| \leq |\Phi_v|$.

If $b(v) > 2b(v_2)$ then $a(v_2) \leq b(v_2) \leq b(v) - k/2$. Hence by Proposition 3.1,

$$\mathbf{Rank}(M_{v_2}) \leq 2^{b(v)-k/2}.$$

Since v_1 is k-weak, we have by induction

$$\mathbf{Rank}(M_{v_1}) \leq |\Phi_{v_1}| \cdot 2^{b(v)-k/2}.$$

Therefore by Proposition 3.2,

$$\mathbf{Rank}(M_v) \leq (|\Phi_{v_1}| + 1) \cdot 2^{b(v)-k/2},$$

and the proof follows since $|\Phi_{v_1}| + 1 \leq |\Phi_v|$. $\square$

5 Random Partition

For any integer n, denote $[n] = \{1, ..., n\}$. In all that comes below, assume that $n \geq 10$. Let Φ be a syntactic multilinear arithmetic formula over the set of variables $X = \{x_{i,j}\}_{i,j \in [n]}$. We think of X as a matrix of variables, with n rows and n columns. Let $m = \lceil n^{1/3} \rceil$.

We will define a random assignment A to the variables in X. Formally, A is a (randomly chosen) function from the set of variables X to the set $\{0,1\} \cup \{y_1, ..., y_m\} \cup \{z_1, ..., z_m\}$. That is, for each variable in X, we assign either a value in $\{0,1\}$ or a variable in $\{y_1, ..., y_m\} \cup \{z_1, ..., z_m\}$. The assignment will have the property that for each variable in $\{y_1, ..., y_m\} \cup \{z_1, ..., z_m\}$ there is exactly one variable in X that is assigned that value.

The assignment A is defined as follows. First choose, uniformly at random, for every $i \in [m]$ two values $q_i, r_i \in [n]$, such that all the $2m$ chosen values are different. (We think of q_i, r_i as the indices of rows). Then choose, uniformly at random, for every $i \in [m]$ two additional values $s_i, t_i \in [n]$, such that all the $2m$ chosen values are different. (We think of s_i, t_i as the indices of columns). For every i, we consider the four variables $x_{q_i,s_i}, x_{q_i,t_i}, x_{r_i,s_i}, x_{r_i,t_i}$. With probability half, we assign

$$\begin{pmatrix} x_{q_i,s_i} & x_{q_i,t_i} \\ x_{r_i,s_i} & x_{r_i,t_i} \end{pmatrix} \longrightarrow \begin{pmatrix} y_i & z_i \\ 1 & 1 \end{pmatrix}$$

that is,

$$A(x_{q_i,s_i}) = y_i \quad , \quad A(x_{q_i,t_i}) = z_i \quad , \quad A(x_{r_i,s_i}) = 1 \quad , \quad A(x_{r_i,t_i}) = 1$$

and with probability half, we assign

$$\begin{pmatrix} x_{q_i,s_i} & x_{q_i,t_i} \\ x_{r_i,s_i} & x_{r_i,t_i} \end{pmatrix} \longrightarrow \begin{pmatrix} y_i & 1 \\ z_i & 1 \end{pmatrix}$$

that is,

$$A(x_{q_i,s_i}) = y_i \quad , \quad A(x_{q_i,t_i}) = 1 \quad , \quad A(x_{r_i,s_i}) = z_i \quad , \quad A(x_{r_i,t_i}) = 1$$

All other variables in X are assigned values in $\{0,1\}$. This is done in the following way. Denote

$$I = [n] \setminus \{q_1, ..., q_m\} \setminus \{r_1, ..., r_m\},$$

and

$$J = [n] \setminus \{s_1, ..., s_m\} \setminus \{t_1, ..., t_m\}.$$

Let π be an arbitrary perfect matching from I to J. For every $i \in I$, we assign $x_{i,\pi(i)} \longrightarrow 1$, that is $A(x_{i,\pi(i)}) = 1$. To all other variables in X (that were not assigned values yet) we assign the value 0.

Denote by Φ_A the formula Φ after substituting in every variable of X the value assigned to it by A. Note that we don't collapse any gate in Φ. The size of Φ_A is the same as the size of Φ and there is a one to one correspondence between the nodes of Φ_A and the nodes of Φ. The only difference is that the labels of some of the leaves are changed. Since for each variable in

$\{y_1, ..., y_m\} \cup \{z_1, ..., z_m\}$ there is exactly one variable in X that is assigned that value, Φ_A is a syntactic multilinear arithmetic formula over the set of variables $\{y_1, ..., y_m\} \cup \{z_1, ..., z_m\}$.

The following lemma shows that if $|\Phi|$ is small then with high probability Φ_A is k-weak for $k = n^{1/32}$.

Lemma 5.1 *Let Φ be a syntactic multilinear arithmetic formula over the set of variables $X = \{x_{i,j}\}_{i,j\in[n]}$, such that every variable in X appears in Φ, and such that $|\Phi| \leq n^{\epsilon \log n}$, where ϵ is a small enough universal constant*[7] *(e.g., $\epsilon = 10^{-6}$). Let A be a random assignment to the variables in X, as above. Then, with probability of $1 - o(1)$ the formula Φ_A is k-weak, for $k = n^{1/32}$.*

6 Proof of Lemma 5.1

Let us first give a brief sketch of the proof. Intuitively, since the random assignment A has a lot of randomness, every node v with large enough X_v will be k-unbalanced with high probability. It can be proved that the probability that such v is not k-unbalanced is smaller than $O(n^{-\delta})$, for some constant δ. This may not be enough since the number of central paths is possibly as large as $n^{\epsilon \log n}$. Nevertheless, each central path contains $\Omega(\log n)$ nodes so we can hope to prove that the probability that none of them is k-unbalanced is as small as $n^{-\Omega(\log n)}$.

This, however, is not trivial since there are dependencies between the different nodes. We will identify $\Omega(\log n)$ nodes, $v_1, ..., v_l$, on the path (that will be "far enough" from each other). We will show that for every v_i, the probability that v_i is not k-unbalanced is smaller than $O(n^{-\delta})$, even when conditioning on the event that $v_1, ..., v_{i-1}$ are not k-unbalanced.

6.1 Notations

To simplify notations, we denote in this section the formula Φ_A by Ψ. There is a one to one correspondence between the nodes of Φ and the nodes of Ψ. For every node v in Φ, there is a corresponding node in Ψ and vice versa. For simplicity, we denote both these nodes by v, and we think about them as the same node. Hence, X_v denotes the set of variables in X that appear in the formula Φ_v, while Y_v denotes the set of variables in $\{y_1, ..., y_m\}$ that appear in the formula Ψ_v, and Z_v denotes the set of variables in $\{z_1, ..., z_m\}$ that appear in Ψ_v. Denote,

$$\alpha(v) = |X_v|/n^2.$$

For every $i \in [m]$, there is a unique variable in X that is assigned the value y_i. We denote that variable by $A^{-1}(y_i)$. In the same way, we denote by $A^{-1}(z_i)$ the unique variable in X that is assigned the value z_i. Note that the event that the formula Ψ is k-weak depends only on the values of $A^{-1}(y_1), ..., A^{-1}(y_m)$ and $A^{-1}(z_1), ..., A^{-1}(z_m)$ (and not on the values in $\{0, 1\}$ that are assigned to all other variables in X).

[7] We do not make any attempt to maximize ϵ and rather try to present the simplest proof that works for some ϵ.

For every $i \in [m]$, denote
$$W^i = \{A^{-1}(y_i), A^{-1}(z_i)\}.$$
Denote
$$W = \bigcup_i W^i.$$
For every $i \in [m]$ and every node v in Φ, denote
$$W^i_v = W^i \cap X_v.$$
For every node v in Φ, denote
$$W_v = W \cap X_v = \bigcup_i W^i_v.$$
A set W^i_v can be of size either 0 or 1 or 2. For every node v in Φ, denote by W'_v the union of all the sets W^i_v of size 1. That is,
$$W'_v = \bigcup_{\{i:|W^i_v|=1\}} W^i_v.$$

Note that the event that a path γ in Ψ is central depends only on the content of the set W. At the other hand, the event that a path γ is k-unbalanced depends also on the specification of which variables in W were originated from $A^{-1}(y_i)$ for some y_i and which ones were originated from $A^{-1}(z_i)$ for some z_i.

6.2 Generating $A^{-1}(y_1), ..., A^{-1}(z_m)$

We will now describe an equivalent way to generate the variables $A^{-1}(y_1), ..., A^{-1}(y_m)$ and $A^{-1}(z_1), ..., A^{-1}(z_m)$ (equivalent to the original way in which they were generated). This will be done in three steps. In the first two steps we generate $W^1, ..., W^m$, and in the third step we specify for every i, which variable in W^i is $A^{-1}(y_i)$ and which one is $A^{-1}(z_i)$.

1. **Step 1:** Generating $W^1, ..., W^m$.
   For every $i \in [m]$, choose w^i_1 to be a (uniformly distributed) random element of X. Then, with probability half choose w^i_2 to be a (uniformly distributed) random element of X at the same row[8] as w^i_1 (and such that w^i_1 and w^i_2 are different), and with probability half choose w^i_2 to be a (uniformly distributed) random element of X at the same column as w^i_1 (and such that w^i_1 and w^i_2 are different). Finally, set $W^i = \{w^i_1, w^i_2\}$.

2. **Step 2:** Repeating Step 1.
   If for any $i \neq i'$ and any j, j' the variables w^i_j and $w^{i'}_{j'}$ are either at the same row or at the same column (of the matrix X) repeat the entire Step 1, until all variables are in different rows and columns (except for w^i_1 and w^i_2, for every i).

[8]Recall that we think of X as a matrix of variables.

3. **Step 3:** Specifying $A^{-1}(y_i), A^{-1}(z_i)$.
   For every $i \in [m]$, with probability half fix $A^{-1}(y_i)$ to be w_1^i and with probability half fix $A^{-1}(y_i)$ to be w_2^i. Then fix $A^{-1}(z_i)$ to be the other variable.

We denote by μ the distribution on $W^1, ..., W^m$ obtained by the above first two steps. That is, we say that $W^1, ..., W^m$ are distributed according to μ if they were generated by the above first two steps.

We denote by μ^* the distribution on $W^1, ..., W^m$ obtained by applying only the first step. That is, we say that $W^1, ..., W^m$ are distributed according to μ^* if they were generated by the above first step.

Note that once the sets $W^1, ..., W^m$ were generated, we also have the sets W, W_v, W'_v (for every v), as defined above. When $W^1, ..., W^m$ are distributed according to μ^*, we think of W, W_v, W'_v as multi-sets, as they may contain variables more than once.

Since the probability for two variables in X to be at the same row or at the same column is $O(1/n)$ and since $m = \lceil n^{1/3} \rceil$, the probability that Step 1 was repeated more than once is $o(1)$. Hence, the two distributions μ and μ^* are of statistical difference $o(1)$. That is, the probability for any event according to μ is the same as its probability according to μ^*, up to an additive term of $o(1)$. This is helpful because in some cases the distribution μ^* is somewhat simpler to analyze.

Note that the event that a path γ in Ψ is central depends only on the first two steps. That is, after the first two steps we already know which paths are central. At the other hand, the event that a path γ is k-unbalanced depends on all three steps.

6.3 Technical Claims

We will now prove two claims that will be needed for our analysis. For the proof of the claims we will use the following version of Chernoff's bound (see for example [ASE] ,Corollary A.7).

Lemma 6.1 (Chernoff Bound) *Let $\chi_1, ..., \chi_l$ be mutually independent random variables, such that, $\mathbf{Pr}[\chi_i = 1] = p$ and $\mathbf{Pr}[\chi_i = 0] = 1 - p$ (for every i). Then for any $c > 0$,*

$$\mathbf{Pr}\left[\left|\sum_{i=1}^{l} \chi_i - pl\right| > cpl\right] < 2e^{-2(cp)^2 l}.$$

Claim 6.2 *With probability of $1 - o(1)$, the following is satisfied for every node v in Φ.*

1. *If $\alpha(v) \geq n^{-1/8}$ then*
$$|W_v| \leq 1.5 \cdot \alpha(v) \cdot 2m,$$
$$|W_v| \geq 0.5 \cdot \alpha(v) \cdot 2m$$

2. *If $\alpha(v) < n^{-1/8}$ then*
$$|W_v| \leq 1.5 \cdot n^{-1/8} \cdot 2m$$

Proof:
For the proof of the claim, we assume that $W^1, ..., W^m$ are distributed according to μ^*. The claim is about the distribution μ. Nevertheless, as mentioned above, the statistical difference between the distributions μ^* and μ is $o(1)$. That is, the probability for any event according to μ is the same as its probability according to μ^*, up to an additive term of $o(1)$. Hence, the claim follows for μ as well.

Recall that we think of $|W_v|$ as a multi-set. Denote,

$$\sigma_1(v) = |X_v \cap \{w_1^1, ..., w_1^m\}|,$$

and

$$\sigma_2(v) = |X_v \cap \{w_2^1, ..., w_2^m\}|$$

(where we think of $\{w_1^1, ..., w_1^m\}$ and $\{w_2^1, ..., w_2^m\}$ as multi-sets). Then

$$|W_v| = \sigma_1(v) + \sigma_2(v).$$

$w_1^1, ..., w_1^m$ are m independent random variables uniformly distributed over X. The probability for each one of these variables to be in X_v is exactly $\alpha(v)$. Thus, $\sigma_1(v)$ is the sum of $m = \lceil n^{1/3} \rceil$ indicator random variables and the probability for each of these variables to be 1 is $\alpha(v)$. Hence, by Chenoff's bound, for every node v with $\alpha(v) \geq n^{-1/8}$

$$\mathbf{Pr}[|\sigma_1(v) - \alpha(v)m| > 0.5 \cdot \alpha(v) \cdot m] < 2e^{-n^{1/12}/2},$$

and in the same way

$$\mathbf{Pr}[|\sigma_2(v) - \alpha(v)m| > 0.5 \cdot \alpha(v) \cdot m] < 2e^{-n^{1/12}/2}.$$

Hence, for every node v with $\alpha(v) \geq n^{-1/8}$,

$$\mathbf{Pr}[||W_v| - \alpha(v)2m| > 0.5 \cdot \alpha(v) \cdot 2m] < 4e^{-n^{1/12}/2}.$$

In the same way, by Chenoff's bound, for every node v with $\alpha(v) < n^{-1/8}$,

$$\mathbf{Pr}[\sigma_1(v) > 1.5 \cdot n^{-1/8} \cdot m] < 2e^{-n^{1/12}/2},$$

and

$$\mathbf{Pr}[\sigma_2(v) > 1.5 \cdot n^{-1/8} \cdot m] < 2e^{-n^{1/12}/2}.$$

Hence, for every node v with $\alpha(v) < n^{-1/8}$,

$$\mathbf{Pr}[|W_v| > 1.5 \cdot n^{-1/8} \cdot 2m] < 4e^{-n^{1/12}/2}.$$

Since we assumed that the number of nodes in Φ is at most $n^{\epsilon \log n}$, the proof of the claim follows by the union bound. $\square$

Claim 6.3 *With probability of $1-o(1)$, the following is satisfied for every node v in Φ. If $1/8 \geq \alpha(v) \geq n^{-1/8}$ then*

$$|W'_v| \geq (1/16) \cdot \alpha(v) \cdot m.$$

Proof:
As before, we assume that $W^1, ..., W^m$ are distributed according to μ^*, and we think of W'_v as a multi-set. Since the probability for any event according to μ is the same as its probability according to μ^*, up to an additive term of $o(1)$, the claim follows for μ as well.

Let v be a node with $1/8 \geq \alpha(v) \geq n^{-1/8}$. Say that a row, in the matrix of variables X, is *dense* if at least half of the variables in the row are contained in X_v. Say that a column, in the matrix of variables X, is *dense* if at least half of the variables in the column are contained in X_v. Since $|X_v| = \alpha(v)n^2$, at most $2\alpha(v)n$ rows are dense and at most $2\alpha(v)n$ columns are dense. Hence, at most $4\alpha(v)^2n^2$ variables in X are in both, dense row and dense column. Since $\alpha(v) \leq 1/8$, this is at most $\alpha(v)n^2/2$. Hence, at least half of the variables in X_v are either not in a dense row or not in a dense column. We call these variables *good* variables.

For every $i \in [m]$, the probability for $(w_1^i \in X_v) \cap (w_2^i \notin X_v)$ can be bounded as follows. The probability for $w_1^i \in X_v$ is $\alpha(v)$. Given that $w_1^i \in X_v$, with probability at least half w_1^i is good. If w_1^i is good then w.l.o.g. it is not in a dense row. Then, with probability half w_2^i is chosen to be at the same row as w_1^i. If w_2^i is chosen to be at the same row as w_1^i and that row is not dense then with probability of at least half $w_2^i \notin X_v$. Altogether, the probability for $(w_1^i \in X_v) \cap (w_2^i \notin X_v)$ is at least $\alpha(v)/(2 \cdot 2 \cdot 2) = \alpha(v)/8$.

Denote by σ the number of indices $i \in [m]$, such that $(w_1^i \in X_v) \cap (w_2^i \notin X_v)$. Thus, σ is the sum of m indicator random variables and the probability for each of these random variables to be 1 is at least $\alpha(v)/8$. Since $\alpha(v) \geq n^{-1/8}$, by Chernoff's bound

$$\mathbf{Pr}[\sigma < (1/16) \cdot \alpha(v) \cdot m] < 2e^{-n^{1/12}/128}.$$

Since $|W'_v| \geq \sigma$,

$$\mathbf{Pr}[|W'_v| < (1/16) \cdot \alpha(v) \cdot m] < 2e^{-n^{1/12}/128}.$$

Since we assumed that the number of nodes in Φ is at most $n^{\epsilon \log n}$, the proof of the claim follows by the union bound. $\square$

6.4 Central Paths are Unbalanced

Once $\vec{W} \equiv (W^1, ..., W^m)$ was generated (by the first two steps of Subsection 6.2), it is already determined whether or not the statements of Claim 6.2 and Claim 6.3 are satisfied. If both statements are satisfied (for every node v) we say that $\vec{W}$ is *good*. Note that $\vec{W}$ is good with probability $1-o(1)$.

Once $\vec{W}$ was generated, it is also already determined whether or not each path γ (from a leaf to a node in Φ) will be central in Ψ. If $\vec{W}$ determines γ to be central in Ψ we say that

γ is central for $\vec{W}$. More formally, γ is central for $\vec{W}$ if for every u, u' on γ, such that u' is a direct son of u, we have $|W_u| \leq 2|W_{u'}|$.

We denote probabilities conditioned on the event that some specific $\vec{W}$ was chosen by $\mathbf{Pr}[\ \cdot\ |\vec{W}]$. We will show that if γ is central for $\vec{W}$ (after the first two steps) then with high probability the third step makes γ unbalanced in the formula Ψ.

Claim 6.4 *Assume that $\vec{W}$ is good and γ is a path from a leaf to the root of Φ, such that γ is central for $\vec{W}$. Then,*

$$\mathbf{Pr}[\gamma \text{ is not } k\text{-unbalanced in } \Psi\ |\vec{W}] \leq n^{-\Omega(\log n)}.$$

Proof:
Assume that $\vec{W}$ is good and that γ is a path from a leaf to the root of Φ such that γ is central for $\vec{W}$. All probabilities in the proof of the claim are conditioned on the event that that specific $\vec{W}$ was chosen. For simplicity, we omit the conditioning on $\vec{W}$ from the discussion and the notations.

Recall that the first node of γ is a leaf and hence $\alpha(v)$ for that node is at most $1/n^2$, and the last node of γ is the root and hence $\alpha(v)$ for that node is 1. Note that $\alpha(v)$ is monotonously increasing along γ. Let $v_1, ..., v_l$ be nodes on γ, chosen by the following process: Let v_1 be the first node on γ, such that $\alpha(v_1) \geq 100 \cdot n^{-1/8}$. For every i, let v_{i+1} be the first node on γ, such that $\alpha(v_{i+1}) \geq 100 \cdot \alpha(v_i)$. Stop when such v_{i+1} doesn't exist or $\alpha(v_{i+1}) > 1/8$. Denote by l the index i of the last v_i in this process.

Since γ is central, for every u, u' on γ, such that u' is a direct son of u, we have $|W_u| \leq 2|W_{u'}|$. Since $\vec{W}$ is good, it follows from Claim 6.2 that $\alpha(v_1) < 1000 \cdot n^{-1/8}$ and that for every $i \in [l-1]$, we have $\alpha(v_{i+1}) < 1000 \cdot \alpha(v_i)$. Hence, the process above continues for $\Omega(\log n)$ steps. To summarize, we have $l = \Omega(\log n)$ and nodes $v_1, ..., v_l$ on γ , such that

1. For every $i \in [l]$,
$$1/8 \geq \alpha(v_i) \geq 100 \cdot n^{-1/8}$$
2. For every $i \in [l-1]$,
$$\alpha(v_{i+1}) \geq 100 \cdot \alpha(v_i)$$

Since $\vec{W}$ is good, it follows from Claim 6.2 and Claim 6.3 that for every $i \in [l-1]$, we can bound $|W'_{v_{i+1}}|$ by
$$|W'_{v_{i+1}}| \geq 2|W_{v_i}| \geq 200 \cdot m \cdot n^{-1/8} \geq 2 \cdot n^{1/8}.$$

Denote by $\mathcal{E}$ the event that γ is not k-unbalanced in the formula Ψ. For every $i \in [l]$, denote by $\mathcal{E}_i$ the event that the node v_i is not k-unbalanced in the formula Ψ. Since $\mathcal{E} \subset \cap_{i\in[l]}\mathcal{E}_i$,

$$\mathbf{Pr}[\mathcal{E}] \leq \mathbf{Pr}\left[\bigcap_{i\in[l]} \mathcal{E}_i\right] = \prod_{i\in[l]} \mathbf{Pr}\left[\mathcal{E}_i \;\middle|\; \bigcap_{i'\in[i-1]} \mathcal{E}_{i'}\right]$$

We will bound for every $i > 1$ the conditional probability $\mathbf{Pr}[\mathcal{E}_i \mid \cap_{i'\in[i-1]} \mathcal{E}_{i'}]$.

Fix $i \in \{2, ..., l\}$. For simplicity, denote the variables of W_{v_i} by $\{x_1, ..., x_r\}$, where $r = |W_{v_i}|$. Note that $W_{v_{i-1}} \subset W_{v_i}$. Denote,

$$S = W'_{v_i} \setminus W_{v_{i-1}}$$

and

$$T = W_{v_i} \setminus S.$$

Since $|W'_{v_i}| \geq 2|W_{v_{i-1}}| \geq 2n^{1/8}$,

$$|S| \geq |W_{v_{i-1}}| \geq n^{1/8}.$$

For every $j \in [r]$, the assignment A determines whether $A(x_j)$ is in the set $\{y_1, ..., y_m\}$ or in the set $\{z_1, ..., z_m\}$. Let χ_j be 1 if $A(x_j) \in \{y_1, ..., y_m\}$ and 0 if $A(x_j) \in \{z_1, ..., z_m\}$. By the definition of $d(v_i)$ (see Section 3),

$$d(v_i) = \left| \sum_{j \in [r]} \chi_j - \frac{r}{2} \right| = \left| \sum_{\{j : x_j \in S\}} \chi_j + \sum_{\{j : x_j \in T\}} \chi_j - \frac{r}{2} \right|.$$

Denote,

$$\sigma = \sum_{\{j : x_j \in S\}} \chi_j$$

and

$$\tau = \sum_{\{j : x_j \in T\}} \chi_j$$

Then,

$$d(v_i) = |\sigma + \tau - r/2|.$$

By the process we described for generating A (see Section 5 and Subsection 6.2), the variables $\{\chi_j : x_j \in S\}$ are mutually independent and are chosen independently from $\{\chi_j : x_j \in T\}$. That is, the variables $\{\chi_j : x_j \in S\}$ are mutually independent even when conditioning on any event in the variables $\{\chi_j : x_j \in T\}$. Hence, σ has the binomial distribution with parameters $|S|$ and $1/2$, even when conditioning on any event in the variables $\{\chi_j : x_j \in T\}$. Hence, σ does not get any specific value with probability larger than $O(|S|^{-1/2})$, which is at most $O(n^{-1/16})$, even when conditioning on any event in the variables $\{\chi_j : x_j \in T\}$. Therefore, $d(v_i)$ also does not get any specific value with probability larger than $O(n^{-1/16})$, even when conditioning on any event in the variables $\{\chi_j : x_j \in T\}$.

By the definition of T, we have $W_{v_{i-1}} \subset T$, and hence the event $\cap_{i' \in [i-1]} \mathcal{E}_{i'}$ depends only on the variables $\{\chi_j : x_j \in T\}$. Hence, $d(v_i)$ does not get any specific value with probability larger than $O(n^{-1/16})$, even when conditioning on the event $\cap_{i' \in [i-1]} \mathcal{E}_{i'}$. Recall that v_i is not k-unbalanced iff $d(v_i) < k$. Since $d(v_i)$ is integer, the probability for that is at most $O(k \cdot n^{-1/16}) = O(n^{-1/32})$, even when conditioning on the event $\cap_{i' \in [i-1]} \mathcal{E}_{i'}$. That is,

$$\mathbf{Pr}\left[\mathcal{E}_i \,\middle|\, \bigcap_{i' \in [i-1]} \mathcal{E}_{i'} \right] \leq O(n^{-1/32})$$

We can now bound

$$\mathbf{Pr}[\mathcal{E}] \leq \prod_{i\in[l]} \mathbf{Pr}\left[\mathcal{E}_i \;\middle|\; \bigcap_{i'\in[i-1]} \mathcal{E}_{i'}\right] = n^{-\Omega(\log n)}$$

□

We can now complete the proof of Lemma 5.1. By Claim 6.4, if $\vec{W}$ is good and γ is a path from a leaf to the root of Φ, such that γ is central for $\vec{W}$, then γ is not k-unbalanced with probability of at most $n^{-\Omega(\log n)}$. The number of paths from a leaf to the root of Φ is the same as the number of leaves in Φ, which is smaller than $n^{\epsilon \log n}$ (and we assumed that ϵ is small enough). Hence, by the union bound, if $\vec{W}$ is good then with probability $1 - o(1)$ all central paths from a leaf to the root of Ψ are k-unbalanced. Since $\vec{W}$ is good with probability $1 - o(1)$, we conclude that with probability $1 - o(1)$ the formula Ψ is k-weak. □

7 Lower Bounds for *Permanent* and *Determinant*

We will now prove our main theorem.

Theorem 1 *Any multilinear arithmetic formula for the permanent or the determinant of an $n \times n$ matrix (over any field) is of size $n^{\Omega(\log n)}$.*

Proof:
Let us start with the permanent. Let Φ be a multilinear arithmetic formula over the set of variables $X = \{x_{i,j}\}_{i,j\in[n]}$ (where $n \geq 10$), and assume that the output of Φ is the permanent of X. Assume for a contradiction that $|\Phi| \leq n^{\epsilon \log n}$, where ϵ is the universal constant from Lemma 5.1. By Proposition 2.1, we can assume w.l.o.g. that Φ is syntactic multilinear.

Let A be a random assignment to the variables in X, as defined in Section 5. Then, Φ_A is a syntactic multilinear arithmetic formula over the set of variables $\{y_1, ..., y_m\} \cup \{z_1, ..., z_m\}$. By Lemma 5.1, with probability of $1 - o(1)$ the formula Φ_A is k-weak, for $k = n^{1/32}$. Hence, by Lemma 4.1, with probability of $1 - o(1)$

$$\mathbf{Rank}(M_{\Phi_A}) \leq n^{\epsilon \log n} \cdot 2^{m-k/2} < 2^m.$$

At the other hand, since the output of Φ is the permanent of X, the output of Φ_A must be the permanent of the matrix $\{A(x_{i,j})\}_{i,j\in[n]}$. By the definition of A (see Section 5), the permanent of that matrix is always

$$f(y_1, ..., y_m, z_1, ..., z_m) \equiv \prod_{i=1}^{m} (y_i + z_i).$$

Note that M_f is a permutation matrix: For every multilinear monomial p in the set of variables $\{y_1, ..., y_m\}$ there is exactly one multilinear monomial q in the set of variables

$\{z_1, ..., z_m\}$ such that $M_f(p,q) = 1$ (and otherwise $M_f(p,q) = 0$), and vice-versa. Hence,

$$\mathbf{Rank}(M_{\Phi_A}) = \mathbf{Rank}(M_f) = 2^m$$

(which is a contradiction).[9]

The proof for the determinant is exactly the same, except that the determinant of the matrix $\{A(x_{i,j})\}_{i,j\in[n]}$ is, up to a sign,

$$g(y_1, ..., y_m, z_1, ..., z_m) \equiv \prod_{i=1}^{m}(y_i - z_i),$$

and as before,

$$\mathbf{Rank}(M_g) = 2^m.$$

□

Acknowledgment

I am grateful to Amir Shpilka for very helpful conversations. I would also like to thank Scott Aaronson, Toni Pitassi and Avi Wigderson.

References

[A] S. Aaronson. Multilinear Formulas and Skepticism of Quantum Computing. STOC 2004: 118-127

[ASE] N. Alon, J.H. Spencer, P.Erdos. *The Probabiliatic Method.* John Wiley and Sons, Inc., (1992)

[BCS] P. Burgisser, M. Clausen, M. A. Shokrollahi. *Algebraic Complexity Theory.* Springer-Verlag New York, Inc., (1997)

[G1] J. von zur Gathen. Feasible Arithmetic Computations: Valiant's Hypothesis. *J. Symbolic Computation* 4(2): 137-172 (1987)

[G2] J. von zur Gathen. Algebraic Complexity Theory. *Ann. Rev. Computer Science* 3: 317-347 (1988)

[GK] D. Grigoriev, M. Karpinski. An Exponential Lower Bound for Depth 3 Arithmetic Circuits. STOC 1998: 577-582

[9] Interestingly, the definition of f gives a linear size multilinear formula for f. Thus, f has very small multilinear formulas. Nevertheless, the formula Φ_A cannot output f because it is k-weak.

[GR] D. Grigoriev, A. A. Razborov. Exponential Lower Bounds for Depth 3 Arithmetic Circuits in Algebras of Functions over Finite Fields. *Applicable Algebra in Engineering, Communication and Computing* 10(6): 465-487 (2000) (preliminary version in FOCS 1998)

[IK] R. Impagliazzo, V. Kabanets. Derandomizing Polynomial Identity Tests Means Proving Circuit Lower Bounds. *Computational Complexity* 13(1-2): 1-46 (2004) (preliminary version in STOC 2003)

[K] K. Kalorkoti. A Lower Bound for the Formula Size of Rational Functions. *SIAM Journal of Computing* 14(3): 678-687 (1985)

[N] N. Nisan. Lower Bounds for Non-Commutative Computation. STOC 1991: 410-418

[NW] N. Nisan, A. Wigderson. Lower Bounds on Arithmetic Circuits Via Partial Derivatives. *Computational Complexity* 6(3): 217-234 (1996) (preliminary version in FOCS 1995)

[R1] R. Raz. Multi-Linear Formulas for Permanent and Determinant are of Super-Polynomial Size. STOC 2004: 633-641

[R2] R. Raz. Multilinear-$NC_1 \neq$ Multilinear-NC_2. FOCS 2004: 344-351

[RS] R. Raz, A. Shpilka. Deterministic Polynomial Identity Testing in Non Commutative Models. *Computational Complexity* 14(1): 1-19 (2005) (preliminary version in Conference on Computational Complexity 2004)

[S] C. P. Schnorr. A Lower Bound on the Number of Additions in Monotone Computations. *Theoretical Computer Science* 2(3): 305-315 (1976)

[SS] E. Shamir, M. Snir. On the Depth Complexity of Formulas. *Mathematical Systems Theory* 13: 301-322 (1980)

[SW] A. Shpilka, A. Wigderson. Depth-3 Arithmetic Circuits Over Fields of Characteristic Zero. *Computational Complexity* 10(1): 1-27 (2001) (preliminary version in Conference on Computational Complexity 1999)

[V1] L. G. Valiant. Negation can be Exponentially Powerful. *Theoretical Computer Science* 12: 303-314 (1980)

[V2] L. G. Valiant. Why is Boolean Complexity Theory Difficult? *In Boolean Function Complexity (M. S. Paterson, ed.) Lond. Math. Soc. Lecture Note Ser. Vol. 169, Cambridge Univ. Press* 84-94 (1992)