Electronic Colloquium on Computational Complexity, Report No. 134 (2011)

Separating multilinear branching programs and formulas

Zeev Dvir[*] Guillaume Malod[†] Sylvain Perifel[‡] Amir Yehudayoff[§]

Abstract

This work deals with the power of linear algebra in the context of multilinear computation. By linear algebra we mean algebraic branching programs (ABPs) which are known to be computationally equivalent to two basic tools in linear algebra: iterated matrix multiplication and the determinant. We compare the computational power of multilinear ABPs to that of multilinear arithmetic formulas, and prove a tight super-polynomial separation between the two models. Specifically, we describe an explicit n-variate polynomial F that is computed by a linear-size multilinear ABP but every multilinear formula computing F must be of size $n^{\Omega(\log n)}$.

1 Introduction

Arithmetic circuits provide a model of computation that captures the complexity of computing polynomials using algebraic operations (addition, multiplication and division). Arithmetics circuits are useful when studying computations of an algebraic nature such as matrix multiplication, or over infinite fields like the real numbers. General arithmetic circuits are quite powerful, and, to this day, there are still no explicit examples of polynomials requiring super-polynomial circuit-size. By explicit we mean in the class VNP defined by Valiant [13]. The permanent is conjectured to be such a polynomial, because it is complete for VNP. For more on algebraic complexity, see [2] or the recent survey [12].

Progress has, nevertheless, been made on understanding restricted models of arithmetic computation. Of particular relevance to this work is the case of *multilinear* computation [5]. A polynomial is multilinear if it has degree at most one in each variable. Many important polynomials are multilinear, e.g., the determinant, the permanent and matrix product. A natural restricted model for computing multilinear polynomials is *multilinear computation*, in which all intermediate stages of the computation are required to be multilinear as well.

There is a large body of research devoted to multilinear computation, specifically, to proving lower bound for multilinear formulas (for which the underlying computation graph is a tree). The first result in this direction was the breakthrough paper of Raz [6] showing that multilinear formulas

[*] Department of Computer Science, Princeton University, Princeton NJ. Email: zeev.dvir@gmail.com. Research partially supported by NSF grant CCF-0832797 and by the Packard fellowship.

[†] Université Paris Diderot, Sorbonne Paris Cité, Institut de Mathématiques de Jussieu, UMR 7586 CNRS, F-75205 Paris, France. Email: malod@logique.jussieu.fr.

[‡] Université Paris Diderot, Sorbonne Paris Cité, LIAFA, UMR 7089 CNRS, F-75205 Paris, France. Email: sylvain.perifel@liafa.jussieu.fr.

[§] Department of Mathematics, Technion–IIT, Haifa, Israel. Email: amir.yehudayoff@gmail.com. Horev fellow – supported by the Taub Foundation. Research supported by grants from ISF and BSF, and by NSF Grant CCF-0832797.

ISSN 1433-8092

for both the permanent and the determinant must be of super-polynomial size. Later, in [7], Raz showed that multilinear *circuits* are super-polynomially stronger than multilinear formulas (see [9] for a simpler proof). Exponential lower bounds for *constant depth* multilinear circuits, as well as strong separations based on circuit-depth, were proved in [10]. Super-linear lower bounds for the size of arithmetic circuits were proved in [8].

In this article, we further extend this line of work by proving a super-polynomial separation between multilinear algebraic branching programs (ABPs) and multilinear formulas. As multilinear circuits can efficiently simulate multilinear ABPs, in particular, we strengthen the mentioned results of [7, 10]. Before stating our results we take a moment to formally define and to briefly motivate the two models (for more details, see the survey [12]).

An *algebraic branching program (ABP)* is a directed acyclic graph with two special nodes in it: a start-node and an end-node. The edges of the ABP are labeled by either variables or field elements. Every directed path γ from the start-node to the end-node computes the monomial f_γ which is the product of all labels on the path γ. The ABP computes the polynomial $f = \sum_\gamma f_\gamma$, where the sum is over all paths γ from start-node to end-node.

A *formula* is a rooted directed binary tree (the edges are directed toward the root). The leaves of the formula are labeled by either variables or field elements. The inner nodes which have in-degree two are labeled by either $+$ or $\times$. A formula computes a polynomial in the obvious way.

Both ABPs and formulas have natural restrictions to the multilinear world. An ABP is *multilinear* if on every directed path from start-node to end-node no variable appears more than once. A formula is *multilinear* if every sub-formula in it computes a multilinear polynomial.

ABPs capture the computational power of iterated matrix product: For every ABP of size s, there are poly(s) many matrices $A_1, A_2, \ldots$ of dimensions poly$(s) \times$ poly(s) with entries that are either variables or field elements, so that the polynomial computed by the ABP is the $(1,1)$ entry in the matrix $A_1 A_2 \cdots$. In the other direction, for every s matrices of dimensions $s \times s$, there is a (multi-start-node and multi-end-node) ABP of size poly(s) computing the product of the matrices. In fact, ABPs also capture the computational power of the determinant: For every ABP of size s, there is a matrix A of dimension poly(s) with entries that are either variables or field elements, so that the determinant of A is the polynomial the ABP computes [13, 4], and the determinant can be computed by a polynomial-size ABP [1, 11, 3]. However, the known polynomial-size ABPs for the determinant are not multilinear, so the lower bound of [6] does not yield our result (by current knowledge).

Formulas, on the other hand, capture a computational model in which every sub-computation can be used only once (as the underlying computation graph is a tree). Since formulas can be parallelized to have depth which is logarithmic in their size, they also capture the parallel time it takes to perform the computation.

It is known that ABPs can efficiently simulate formulas [13]. Similar ideas show that *multilinear* ABPs can efficiently simulate *multilinear* formulas. A natural question is thus whether the other direction holds as well. We show that in the multilinear world it does not (a similar separation is believed to hold for general algebraic computation). This is the first separation between branching programs and formulas we are aware of.

Theorem 1.1. *For every positive integer n, there is a multilinear polynomial $F = F_n$ in n variables with zero-one coefficients so that the following holds:*

(i) There is a uniform algorithm that, given n, runs in time $O(n)$ and outputs a multilinear branching program computing F.

(ii) Over any field, every multilinear formula computing F must be of size $n^{\Omega(\log n)}$.

Our lower bound of $n^{\Omega(\log n)}$ is tight since every polynomial-size multilinear ABP can be simulated by a multilinear formula of size $n^{O(\log n)}$ (see, e.g., [9]).

We mention two directions for future study. First, multilinear ABPs can be efficiently simulated by multilinear circuits. Is the other direction true? The guess would be that the answer is negative, but current techniques are not sufficient to prove strong lower bound for multilinear ABPs. Second, as mentioned, there is a polynomial-size non-multilinear ABP computing the determinant. Is there a polynomial-size multilinear ABP computing the determinant? A positive answer would yield a new type of algorithm for the determinant (and will imply our result via [6]). A negative answer would yield a strong lower bound for ABPs and emphasize the power of non-multilinear computation.

1.1 Our techniques

The proof of Theorem 1.1 consists of two parts: (i) constructing a small multilinear ABP computing some polynomial F and (ii) showing that any multilinear formula computing F is of super-polynomial size. The two parts have conflicting demands: In part (i) we wish to make the polynomial F simple enough so that a small ABP can compute it, whereas in part (ii) we will need to rely on the hardness of F to prove a lower bound. To succeed in both parts we need to take full advantage of the expressive power that ABPs grant us. Below we give a high-level description of the proof, focusing on part (ii), which is considerably more complicated. Along the way we will highlight ideas from previous works that are used in the proof.

The lower bound part of the proof uses several ideas introduced in previous works [6, 7, 9]. Of particular importance is the notion of a *full-rank* polynomial. A given polynomial f can be used to define a family of matrices $\{M(f_\Pi)\}_\Pi$, where Π ranges over all partitions of the variables X to two sets of variables Y, Z of equal size (these are the so-called *partial derivative* matrices). The polynomial f is said to have *full-rank* if the rank of $M(f_\Pi)$ is full for every such Π. This property turns out to be useful in showing complexity lower bounds for f. Indeed, Raz showed that every full-rank polynomial f cannot have polynomial-size multilinear formulas [6, 7].

To the best of our knowledge, full-rank polynomials may also require super-polynomial-size ABPs. Thus, in order to prove our separation we will look for a property which is *weaker* than being full-rank and is still useful for proving lower bounds. One of the main new ideas in our proof is a construction of a special *subset* of partitions, called *arc-partitions*, which is sufficiently powerful to carry through the lower bound proof and, at the same time, simple enough to carry part (i) of the proof. The number of arc-partitions is much smaller than the total number of partitions. Nevertheless, we are still able to show that every *arc-full-rank* polynomial f (i.e., $M(f_\Pi)$ has full rank for all arc-partitions Π) does not have polynomial-size multilinear formulas.

We now go into more details as to how this family of partitions is defined and what makes it useful. We will start by describing a *distribution* over partitions. The partitions that will have positive probability of being obtained in this distribution will be called arc-partitions. The distribution is defined according to the following (iterative) sampling algorithm. Position the n variables on a cycle with n nodes so that there is an edge between i and $i+1$ modulo n. Start with the *arc* $[L_1, R_1] = \{0, 1\}$ (an arc is a connected path on the cycle). At step $t > 1$ of the process, maintain a partition of the arc $[L_t, R_t]$. "Grow" this partition by first picking a pair uniformly at random out of the three possible pairs $(L_t - 2, L_t - 1), (L_t - 1, R_t + 1), (R_t + 1, R_t + 2)$, and then defining the partition Π on this pair to map to a random permutation of the two variables

y_{t+1}, z_{t+1}. After $n/2$ steps, we have chosen a partition of the n variables into two disjoint, equal-size sets of variables (for more details, see Section 2.2).

The arc-partitions allow us to adapt the key argument in [6]. Let us remind roughly how this argument works after the simplifications from [12]. Every multilinear formula defines a "non-redundant" K-coloring of the n-variables with $K \sim \log n$. This is simply a mapping $C : [n] \mapsto [K]$ so that the pre-image of every color $k \in [K]$ is not too small. A color k is said to be "balanced" with respect to a partition Π if the number of Y variables of color k is roughly the same as the number of Z variables of color k. Now, for a given coloring C, if we choose a random partition Π from the set of *all* partitions, simple properties of the hyper-geometric distribution imply that the probability that all colors in C are "balanced" is at most $p = n^{-\Omega(K)} = n^{-\Omega(\log n)}$. This bound, in turn, proves a roughly $1/p = n^{\Omega(\log n)}$ lower bound for the size of multilinear formulas.

Following a similar strategy, we show that for any "non-redundant" K-coloring C, for a random arc-partition, the probability that all colors in C are "balanced" is at most $n^{-\Omega(K)}$ as well. This turns out to be significantly more difficult than showing it for a random partition (from the set of all partitions). The hardest part of the proof is analyzing a random walk on a two-dimensional "distorted chessboard" where we need to prove certain anti-concentration results (see Section 5 for details).

1.2 Organization

Section 2 contains some preliminary useful definitions. Section 3 introduces the basic notion behind our proof, arc-full-rank polynomials, and describes a construction of an ABP computing an arc-full-rank polynomial. Section 4 contains the main probability estimate we require. Finally, in Section 5 we study a random walk on a "distorted chessboard" that is the key part of the main probability estimate.

2 Preliminaries

2.1 The partial derivative matrix

Let $\mathbb{F}$ be a field. Let Y, Z be two disjoint sets of variables. Given a multilinear polynomial $f \in \mathbb{F}[Y, Z]$, the *partial derivative matrix* $M(f)$ is the coefficient matrix of f, that is, the $2^{|Y|} \times 2^{|Z|}$ matrix whose (p, q) entry is the coefficient of the monomial pq in f, where p is a monic multilinear monomial in Y and q is a monic multilinear monomial in Z.

The following proposition from [6] gives some basic properties of this matrix.

Proposition 2.1. *Given two polynomials $f, g \in \mathbb{F}[Y, Z]$, the following holds:*

(i) $\mathsf{rank}(M(f + g)) \leq \mathsf{rank}(M(f)) + \mathsf{rank}(M(g))$.

(ii) If f, g are over disjoint sets of variables, $\mathsf{rank}(M(f \cdot g)) = \mathsf{rank}(M(f)) \cdot \mathsf{rank}(M(g))$.

(iii) $\mathsf{rank}(M(f)) \leq 2^{\min(Y(f), Z(f))}$*, where $Y(f)$ is the number of Y variables appearing in f and $Z(f)$ is the number of Z variables appearing in f.*

2.2 Arc-partitions

Let n be an even integer, let $X = \{x_0, x_1, \dots, x_{n-1}\}$, let $Y = \{y_1, \dots, y_{n/2}\}$ and let $Z = \{z_1, \dots, z_{n/2}\}$. A *partition* is a one-to-one map Π from X to $Y \cup Z$. Given a polynomial f in the variables X, define the polynomial f_Π as the polynomial obtained by substituting $\Pi(x_i)$ instead of x_i in f for all x_i in X.

Define *arc-partitions* as the following family of partitions. In fact, we shall define a distribution $\mathcal{D}$ on partitions, whose support is by definition the set of arc-partitions. For the purpose of the definition, we identify X with the set $\{0, 1, \dots, n-1\}$ in the natural way. Consider the n-cycle graph, i.e., the graph with nodes $\{0, 1, \dots, n-1\}$ and edges between i and $i+1$ modulo n. For two nodes $i \neq j$ in the n-cycle, denote by $[i, j]$ the *arc* between i, j, that is, the set of nodes on the path $\{i, i+1, \dots, j-1, j\}$ from i to j in n-cycle.

First, define a distribution $\mathcal{D}_P$ on a family of *pairings* (a list of disjoint pairs of nodes in the cycle) as follows. A random pairing is constructed in $n/2$ steps. At the end of step $t \in [n/2]$, we shall have a pairing $(P_1, \dots, P_t)$ of the arc $[L_t, R_t]$. The size of $[L_t, R_t]$ is always $2t$. The first pairing contains only $P_1 = \{L_1, R_1\}$ with $L_1 = 0$ and $R_1 = 1$. Given $(P_1, \dots, P_t)$ and $[L_t, R_t]$, define the random pair P_{t+1} (independently of previous choices) by

$$P_{t+1} = \begin{cases} \{L_t - 2, L_t - 1\} & \text{with probability } 1/3, \\ \{L_t - 1, R_t + 1\} & \text{with probability } 1/3, \\ \{R_t + 1, R_t + 2\} & \text{with probability } 1/3, \end{cases}$$

where addition is modulo n. This process is illustrated in Figure 1. Define

$$[L_{t+1}, R_{t+1}] = [L_t, R_t] \cup P_{t+1}.$$

So L_{t+1} is either $L_t - 2$, $L_t - 1$ or L_t, each value is obtained with probability $1/3$, and similarly (but not independently) for R_{t+1}.

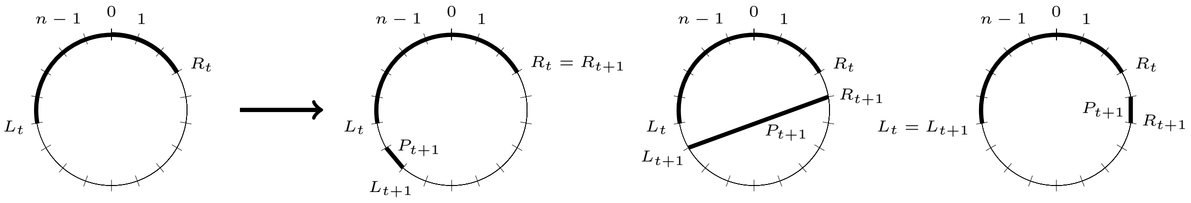

Figure 1: *Incremental definition of a pairing. On the left the arc $[L_t, R_t]$ in the n-cycle. On the right the three options for the next pair P_{t+1} and the corresponding L_{t+1}, R_{t+1}.*

The final pairing is

$$P = (P_1, P_2, \dots, P_{n/2}).$$

Denote by $P \sim \mathcal{D}_P$ a pairing distributed according to $\mathcal{D}_P$.

Secondly, given $P = (P_1, \dots, P_{n/2}) \sim \mathcal{D}_P$, define a random partition Π as follows: For every $t \in [n/2]$, if $P_t = \{i_t, j_t\}$, let with probability $1/2$, independently of all other choices,

$$\Pi(x_{i_t}) = y_t \ \text{ and } \ \Pi(x_{j_t}) = z_t,$$

and, with probability 1/2,

$$\Pi(x_{i_t}) = z_t \text{ and } \Pi(x_{j_t}) = y_t.$$

We denote by $\Pi \sim \mathcal{D}$ an arc-partition distributed as defined above.

3 Arc-full-rank polynomials

We now define the criterion by which a polynomial is difficult to compute for multilinear formulas. We say that f is *arc-full-rank* if for every arc-partition Π the partial derivative matrix $M(f_\Pi)$ has full rank.

Theorem 3.1. *If f is an arc-full-rank multilinear polynomial in n variables over a field $\mathbb{F}$, then any multilinear formula computing f over $\mathbb{F}$ has size at least $n^{\Omega(\log n)}$.*

The proof of the theorem consists of two parts, given by two lemmas. The first lemma is a well-known decomposition of multilinear formulas (see e.g. [12]). To state the lemma, we need the following definition.

Definition. *A multilinear polynomial f in variables X is called a (K,T)-*product *polynomial if there exists K disjoint sets of variables $X_1, \ldots, X_K$, each of size at least T, so that*

$$f = f_1 f_2 \cdots f_K,$$

and each f_k, $k \in [K]$, is a multilinear polynomial in X_k.

Note that, in the above definition, not all variables in X_k must occur in f_k. For example, the polynomial $x_1 x_2 \cdots x_K$ is always a (K,T)-product polynomial if it is thought of over at least KT variables.

Lemma 3.2 (see e.g. [12])**.** *Every n-variate polynomial f computed by a multilinear formula of size s can be written as a sum $f = f_1 + \ldots + f_{s+1}$, each f_i is a (K,T)-product polynomials with $K \geq (\log n)/100$ and $T \geq n^{7/8}$.*

The second lemma (whose proof is the main technical part of this paper) shows that if f is a product polynomial, then for an arc-partition $\Pi \sim \mathcal{D}$, with very high probability, the rank of $M(f_\Pi)$ is not full. Recall that the rank of $M(f_\Pi)$ cannot exceed its dimension, which is $2^{n/2}$.

Lemma 3.3. *There exists a constant $\delta > 0$ so that the following holds. Let n be a large enough even integer. Let f be a (K,T)-product polynomial in n variables with $K \geq (\log n)/100$ and $T \geq n^{7/8}$. Then*

$$\mathbb{P}\big[\mathsf{rank}(M(f_\Pi)) \geq 2^{n/2 - n^\delta}\big] \leq n^{-\delta \log n},$$

where $\Pi \sim \mathcal{D}$.

We defer the proof of Lemma 3.3 to Section 4. The two lemmas immediately imply Theorem 3.1:

Proof of Theorem 3.1. Assume toward a contradiction that Φ is a multilinear formula of size $s \leq n^{(\delta/2)\log n}$ computing a n-variate arc-full-rank polynomial f, with $\delta > 0$ from Lemma 3.3 and n large enough. Lemma 3.2 implies that $f = f_1 + \ldots + f_{s+1}$, where each f_i is a product polynomial.

Let Π be a random partition distributed according to $\mathcal{D}$. Lemma 3.3, Proposition 2.1 and the union bound imply

$$\begin{aligned}
1 &= \mathbb{P}\big[\mathsf{rank}(M(f_\Pi)) = 2^{n/2}\big] \\
&\leq \mathbb{P}\Big[\text{there exists } i \in [s+1] \text{ with } \mathsf{rank}(M((f_i)_\Pi)) \geq 2^{n/2}/(s+1)\Big] \\
&\leq \sum_{i=1}^{s+1} \mathbb{P}\big[\mathsf{rank}(M((f_i)_\Pi)) \geq 2^{n/2-n^\delta}\big] \\
&\leq (s+1)n^{-\delta \log n} < 1.
\end{aligned}$$

□

3.1 A construction of an arc-full-rank polynomial

Here we describe a simple construction of an ABP that computes an arc-full-rank polynomial. The high-level idea is to construct an ABP in which every path between start-node and end-node corresponds to a specific execution of the random process which samples arc-partitions. Each node in the ABP corresponds to an arc $[L, R]$, which sends an edge to each of the nodes $[L-2, R]$, $[L-1, R+1]$ and $[L, R+2]$. The edges have specially chosen labels that guarantee full rank w.r.t. to every arc-partition. For simplicity of presentation, we allow the edges of the program to be labeled by degree two polynomials in three variables. This assumption can be easily removed by replacing each edge with a constant-size ABP computing the degree two polynomial.

Formally, the nodes of the program are even-size arcs in the n-cycle, n an even integer. The start-node of the program is the empty arc $\emptyset$ and the end-node is the whole cycle $[n]$ (both are "special" arcs). Let $X = \{x_0, \ldots, x_n\}$ be a set of variables, and let $\Lambda = \{\lambda_e\}$ be a different set of variables of size at most $3n^2$. In the construction, the sub-script e in λ_e is an edge of the branching program (which will have at most $3n^2$ edges).

Construct the branching program by connecting a node/arc of size $2t$ to three nodes/arcs of size $2t+2$. For $t = 1$, there is just one node $[0, 1]$, and the edge e from start-node to it is labeled $\lambda_e(x_0 + x_1)$. For $t > 1$, the node $[L, R] \supset [0, 1]$ of size $2t < n$ is connected to the three nodes: $[L-2, R]$, $[L-1, R+1]$, and $[L, R+2]$. (It may be the case that the three nodes are the end-node.) The edge labeling is: The edge e_1 between $[L, R]$ and $[L-2, R]$ is labeled $\lambda_{e_1} \cdot (x_{L-2} + x_{L-1})$. The edge e_2 between $[L, R]$ and $[L-1, R+1]$ is labeled $\lambda_{e_2} \cdot (x_{L-1} + x_{R+1})$. The edge e_3 between $[L, R]$ and $[L, R+2]$ is labeled $\lambda_{e_3} \cdot (x_{R+1} + x_{R+2})$.

Consider the ABP thus described, and the polynomial $F = F_n$ it computes. For every path γ from start-node to end-node in the ABP, the list of edges along γ yields a pairing P; every edge e in γ corresponds to a pair $P_e = \{i_e, j_e\}$ of nodes in n-cycle. Thus,

$$F = \sum_\gamma \prod_{e \in \gamma} \lambda_e \cdot \prod_{e \in \gamma} (x_{i_e} + x_{j_e}), \tag{3.1}$$

where the sum is over all paths γ from start-node to end-node. There is in fact a one-to-one correspondence between pairings P and such paths γ (this follows by induction on t). The sum defining F can be thought of, therefore, as over pairings P.

The following theorem summarizes the relevant properties of F.

Theorem 3.4. *Over every field $\mathbb{F}$, the polynomial $F = F_n$ defined above satisfies the following:*

1. *F is computed by a linear-size (in number of variables which is $O(n^2)$) multilinear ABP. The ABP for F can be constructed uniformly in time $O(n^2)$.*

2. *F has zero-one coefficients.*

3. *F is arc-full-rank as a polynomial in the variables X over the field $\mathbb{F}(\Lambda)$ of rational functions in Λ.*

Proof. That the branching program is multilinear is justified as follows. By induction, for every arc $[L, R]$, the X variables that occur on every path reaching the node $[L, R]$ in the ABP is a subset of $\{x_i : i \in [L, R]\}$. Every Λ variable is labeling a single edge.

The program is of linear-size since every edge is labeled by a different variable. In fact, the description above yields an algorithm that given n runs in time $O(n^2)$ and outputs the branching program.

The fact that F has zero-one coefficients follows from (3.1), since given the monomial $\prod_{e\in\gamma} \lambda_e$ one can reconstruct γ.

Finally, F is arc-full-rank over $\mathbb{F}(\Lambda)$: Let Π be an arc-partition. It remains to show that $M(F_\Pi)$ has full rank. The arc-partition Π is defined from a pairing $P = P(\Pi)$. The pairing P corresponds to a path $\gamma = \gamma(\Pi)$ from start-node to end-node. Consider the polynomial f obtained from F_Π by substituting $\lambda_e = 1$ for every e in γ, and $\lambda_e = 0$ for every e not in γ. By (3.1), and by definition of Π from P,

$$f = \prod_{t\in[n/2]} (y_t + z_t).$$

The rank over $\mathbb{F}$ of $M(y_t + z_t)$ is two. Proposition 2.1 hence implies that the rank over $\mathbb{F}$ of $M(f)$ is full. Since the rank of $M(F_\Pi)$ over $\mathbb{F}(\Lambda)$ is at least the rank of $M(f)$ over $\mathbb{F}$, the proof is complete. □

4 Arc-partitions and product polynomials

In this section (and the next one), we prove Lemma 3.3. Again, we identify the set of variables $X = \{x_0, \ldots, x_{n-1}\}$ with the n-cycle $\{0, 1, \ldots, n-1\}$, where addition is modulo n. A (K, T)-product polynomial is defined by a partition of X to K sets. It is more convenient to work with partitions of $\{0, 1, \ldots, n-1\}$ instead. Let S be a partition of the cycle to K parts, namely, $S = (S_1, \ldots, S_K)$ where $\bigcup_{k\in[K]} S_k$ is the whole cycle and $S_k \cap S_{k'} = \emptyset$ for all $k \neq k'$ in $[K]$. We also think of $[K]$ as a set of *colors*, and of S as a *coloring* of the cycle.

For a pairing P, define the number of *k-violations* by

$$V_k(P) = \{P_t \in P : |P_t \cap S_k| = 1\},$$

in words, the set of pairs in which one color is k and the other color is different. Denote

$$G(P) = \left|\{k \in [K] : |V_k(P)| \geq n^{1/1000}\}\right|.$$

We do not include S as a subscript in these two notations since S will be known from the context (and will be fixed throughout most of the discussion). The next crucial lemma shows that for every fixed non-redundant K-coloring of the cycle, a random pairing has, w.h.p., many colors with many violations.

Lemma 4.1. *There exists a constant $C > 0$ such that for all $C \leq K \leq n^{1/1000}$ the following holds: Let $S = (S_1, \ldots, S_K)$ be a partition of the n-cycle and suppose that $|S_k| \geq n^{7/8}$ for all $k \in [K]$. Then,*

$$\mathbb{P}[G(P) \leq K/1000] \leq n^{-\Omega(K)},$$

where $P \sim \mathcal{D}_P$.

We defer the proof of this lemma to Section 4.1 below and continue with the proof of Lemma 3.3. Before the formal proof, we provide some intuition. To prove Lemma 3.3, thanks to Proposition 2.1 (items (*ii*) and (*iii*)) it suffices to show that the probability that all colors are "balanced" (i.e., the number of Y variables of color k is close to that of Z variables) w.r.t. a random arc-partition is very small. Lemma 4.1 implies that, w.h.p., there are order K colors, each with many violations. For each such color k, anti-concentration implies that the probability that the color k is "balanced" is small. Numerically speaking, the colors are "independent" which implies that the probability that all colors are "balanced" is very small.

Proof of Lemma 3.3. Let f be a (K, T)-product with $K = \lceil (\log n)/100 \rceil$ and $T = \lceil n^{7/8} \rceil$. Let S be the partition of the cycle induced by the partition of the variables of f as a product polynomial. Let $P \sim \mathcal{D}_P$, and let $\Pi \sim \mathcal{D}$ be a random arc-partition obtained from P. From Lemma 4.1 we know that

$$\mathbb{P}[G(P) \leq K/1000] \leq n^{-\Omega(K)}.$$

For every P, define a graph $H(P)$ whose nodes are colors k in $[K]$ so that $|V_k(P)| \geq n^{1/1000}$, and every two nodes $k \neq k'$ in $G(P)$ are connected by an edge if the size of $V_k(P) \cap V_{k'}(P)$ is at least $n^{1/1500}$ (i.e., there are at least $n^{1/1500}$ pairs colored by both k, k'). Since $K \leq \log n$ and by definition of $G(P)$, the degree of each node in $H(P)$ is at least one.

Use the following simple graph-theoretic claim.

Claim 4.2. *Let H be a graph with minimal degree at least one and M nodes, then there is a subset $\{h_1, \ldots, h_N\}$ of the nodes of H of size $N \geq M/2 - 1$, so that for every $j \in [N-1]$, the degree of h_{j+1} in the graph induced on the nodes not in $\{h_1, \ldots, h_j\}$ is at least one.*

Proof. The claim follows by induction. If $M \leq 2$, the claim trivially holds. For $M > 2$, argue as follows. Let h_1 be a node of least degree in H. Consider the graph H_1 induced on all nodes except h_1. By choice of h_1, the graph H_1 has at most one isolated node. If such an isolated node exists, call it h_1'. Apply the claim on the graph induced on nodes not in $\{h_1, h_1'\}$, which is of minimal degree at least one, and of size at least $M - 2$ to obtain a set of nodes $\{h_2, \ldots, h_N\}$. The set $\{h_1, h_2, \ldots, h_N\}$ satisfies the claim. □

Let $\{k_1, \ldots, k_{K'}\}$, $K' \geq G(P)/2 - 1$, be the subset of nodes in $H(P)$ given by the claim above. View the sampling of Π from P as happening in a specific order, according to the order of $k_1, k_2, \ldots, k_{K'}$: First define Π on pairs with at least one point with color k_1, then define Π on remaining pairs with at least one point with color k_2, and so forth. When finished with $k_1, \ldots, k_{K'}$, continue to define Π on all other pairs.

For every $j \in [K']$, define E_j to be the event that $\left|Y_{k_j} - |S_{k_j}|/2\right| \leq n^{1/5000}$, where Y_{k_j} is the size of $\Pi^{-1}(Y) \cap S_{k_j}$. By choice, conditioned on $E_1, \ldots, E_{j-1}$, there are at least $n^{1/1500}$ pairs P_t so that $|P_t \cap S_{k_j}| = 1$ that are not yet assigned variables in Y, Z. For every such P_t, the element in $P_t \cap S_{k_j}$ is assigned a Y variable with probability $1/2$, and is independent of any other $P_{t'}$. The probability that a binomial random variable B over a universe of size $U \geq n^{1/1500}$ and marginals $1/2$ obtains any specific value is at most $O(U^{-1/2}) = O(n^{-1/3000})$. Hence, for all $j \in [K']$, by the union bound,

$$\mathbb{P}[E_j | E_1, \ldots, E_{j-1}, P] \leq \mathbb{P}_B[U/2 - n^{1/5000} \leq B \leq U/2 + n^{1/5000}] \leq O\big(2n^{1/5000} n^{-1/3000}\big) \leq n^{-\Omega(1)}.$$

Therefore,

$$\begin{aligned} &\mathbb{P}\big[|Y_k - |S_k|/2| \leq n^{1/5000} \text{ for all } k \in [K]\big] \\ &\qquad \leq \mathbb{E}\left[n^{-\Omega(G(P))} \middle| G(P) > K/1000\right] + n^{-\Omega(K)} = n^{-\Omega(\log n)}. \end{aligned}$$

Items (ii) and (iii) in Proposition 2.1 imply that if one of the factors of f_Π is unbalanced (i.e., the number of Y variables in it is far from the number of Z variables in it) then $M(f_\Pi)$ has low rank. Formally,

$$\mathbb{P}\big[\mathsf{rank}(M(f_\Pi)) \geq 2^{n/2 - n^{1/5000}}\big] \leq \mathbb{P}\big[|Y_k - |S_k|/2| \leq n^{1/5000} \text{ for all } k \in [K]\big].$$

□

4.1 Proof of Lemma 4.1

Fix some partition/coloring $S = (S_1, \ldots, S_K)$ of the n-cycle satisfying the conditions of the lemma. Think of S as a function from the n-cycle to the set $[K]$, assigning each node its color; $S(i)$ is the color of i. Use the following definition to partition the proof into cases. For a color k, count the number of *jumps* in it (w.r.t. the partition S) to be

$$J_k = \{j \in S_k : k = S(j) \neq S(j+1)\},$$

the set of elements j of color k so that $j+1$ has a color different from k.

Case 1: Many colors with many jumps. The intuition is that each color with many jumps has many violations because a jump $j \in J_k$ gives a violation as soon as the construction of the pairing takes the pair $(j, j+1)$.

Assume that for at least $K/2$ colors k,

$$|J_k| > n^{1/100}.$$

Denote by $B \subseteq [K]$ the set of k's that satisfy the above inequality. For every k in B, there is thus a subset $Q_k \subset J_k$ of size $N = \lceil n^{1/100} \rceil$. Denote

$$Q = \bigcup_{k \in B} Q_k.$$

Think of the construction of the (random) pairing P as happening in stages, depending on Q, as follows.

For $t > 0$, define the random variable

$$Q(t) = Q \setminus [L_t - 4, R_t + 4],$$

the set Q after removing a four-neighborhood of $[L_t, R_t]$; If the distance between L_t, R_t is at most ten, define $Q(t) = \emptyset$.

Let $\tau_1 \geq \tau_0 = 1$ be the first time t after τ_0 so that the distance between $[L_t, R_t]$ and $Q(\tau_0)$ is at most two. The distance between $[L_{\tau_0}, R_{\tau_0}]$ and $Q(\tau_0)$ is at least five. The size of the arc $[L_t, R_t]$ increases by two at each time step. So, $\tau_1 \geq \tau_0 + 2$. Let q_1 be an element of $Q(\tau_0)$ that is of distance at most two from $[L_{\tau_1}, R_{\tau_1}]$; If there is more than one such q_1, choose arbitrarily. The minimality of τ_1 implies that q_1 is not in $[L_{\tau_1}, R_{\tau_1}]$.

Let $\tau_2 \geq \tau_1$ be the first time t after τ_1 so that the distance between $[L_t, R_t]$ and $Q(\tau_1)$ is at most two. Let q_2 be an element of $Q(\tau_1)$ that is of distance at most two from $[L_{\tau_2}, R_{\tau_2}]$.

Define τ_j, q_j for $j > 2$ similarly, until $Q(\tau_j)$ is empty. As long as $|Q(\tau_j)| \geq 8$, we have $|Q(\tau_{j+1})| \geq |Q(\tau_j)| - 8$. This process, therefore, has at least $KN/10$ steps.

For $1 \leq j \leq KN/10$, denote by E_j the event that during the time between τ_j and τ_{j+1} the pair $[q_j, q_j + 1]$ is added to P. The pair $[q_j, q_j + 1]$ is violating color $S(q_j)$. At time τ_j, even conditioned on all the past $P_1, \ldots, P_{\tau_j}$, in at most two steps (and before τ_{j+1}) we can add the pair $[q_j, q_j + 1]$ to P. For every j, therefore,

$$\mathbb{P}[E_j | P_1, \ldots, P_{\tau_j}] \geq (1/3)(1/3) = 1/9.$$

Subsequently,

$$\mathbb{P}\big[\text{there is } j_1, \ldots, j_{N'} \text{ so that } E_{j_1} \cap \ldots \cap E_{j_{N'}}\big] \geq 1 - c^N$$

with

$$N' = \lfloor KN/100 \rfloor$$

and $c < 1$ a universal constant.

The size of every Q_k is N. So, every color k in B can contribute at most N elements to $j_1, \ldots, j_{N'}$. Hence,

$$\mathbb{P}\big[G(P) \geq K/1000\big] \geq \mathbb{P}\big[\text{there is } j_1, \ldots, j_{N'} \text{ so that } E_{j_1} \cap \ldots \cap E_{j_{N'}}\big].$$

The proof is hence complete in this case.

Case 2: Many colors with few jumps. The intuition is that many violations will come from pairs of the form $(L_t - 1, R_t + 1)$ in the construction of the pairing.

Assume that for at least $K/2$ colors k,

$$|J_k| \leq n^{1/100}.$$

Denote again by $B \subseteq [K]$ the set of k's that satisfy the above inequality. We say that a color k is *noticeable* in the arc A if

$$n^{5/8} \leq \big|S_k \cap A\big| \leq |A| - n^{5/8}.$$

Claim 4.3. *There are $K' \geq K/2 - 1$ disjoint arcs $A_1, \ldots, A_{K'}$ so that for every $j \in [K']$,*

(i) $|A_j| = m = \lfloor n^{3/4} \rfloor$, and

(ii) there is a color k_j in B that is noticeable in A_j.

Moreover, the colors $k_1, \ldots, k_{K'}$ can be chosen to be pairwise distinct.

Proof. For each color k in B, there are at least $n^{7/8}$ vertices of color k in the n-cycle and at most $n^{1/100}$ jumps in the color k. Therefore, there is at least one k-monochromatic arc of size at least $n^{7/8-1/100}$. Hence, on the n-cycle there are such monochromatic arcs $I_{k_1}, \ldots, I_{k_{|B|}}$ for the colors $k_1, \ldots, k_{|B|}$ in B, in this order $(0 < 1 < \ldots < n-1)$.

Consider an arc A of size m included in I_{k_1}. Thus $|S_{k_1} \cap A| = m$. If we "slide" the arc A until it is included in I_{k_2}, then $|S_{k_1} \cap A| = 0$. By continuity, there is an intermediate position for the arc A such that $n^{5/8} \leq |S_{k_1} \cap A| \leq m - n^{5/8}$. This provides the first arc A_1 of the claim.

Sliding an arc inside I_{k_2} to inside I_{k_3} shows that there exists an arc A_2 such that $n^{5/8} \leq |S_{k_2} \cap A_2| \leq m - n^{5/8}$. The arcs A_1 and A_2 are disjoint: The distance of the largest element of A_1 and the smallest element of S_{k_2} is at most m. The distance of the smallest element of A_2 and the largest element of S_{k_2} is at most m. The size of S_{k_2} is larger than $2m$.

Proceed in this way to define $A_3, \ldots, A_{|B|-1}$. □

Use Claim 4.3 to divide the construction of the (random) pairing into stages. Denote by $A^{(0)}$ the family of arcs given by the claim. Let τ_1 be the first time t that the arc $[L_t, R_t]$ hits one of the arcs in $A^{(0)}$. Denote by A_1 that arc that is hit at time τ_1 (break ties arbitrarily). Denote by k_1 the color that is noticeable in A_1. Let σ_1 be the first time t so that A_1 is contained in $[L_t, R_t]$. Let $A^{(1)}$ be the subset of $A^{(0)}$ of arcs that have an empty intersection with $[L_{\sigma_1}, R_{\sigma_1}]$. Similarly, let τ_2 be the first time t after σ_1 that the arc $[L_t, R_t]$ hits one of the arcs in $A^{(1)}$. If there are no arc in $A^{(1)}$, $\tau_2 = \infty$. Denote by A_2 that arc that is hit at time τ_2. Denote by k_2 the color that is noticeable in A_2. Let σ_2 be the first time t so that A_2 is contained in $[L_t, R_t]$. Let $A^{(2)}$ be the subset of $A^{(1)}$ of arcs that have an empty intersection with $[L_{\sigma_2}, R_{\sigma_2}]$. Define $\tau_j, \sigma_j, A_j, k_j, A^{(j)}$ for $j > 2$ analogously.

For every $j \geq 1$, denote by E_j the event that during the time between τ_j and τ_{j+1} the number of pairs added that violate color k_j is at most $n^{1/150}$. (If E_j does not hold, $|V_{k_j}(P)| \geq n^{1/150} \geq n^{1/1000}$.) The main part of the proof is summarized in the following proposition, whose proof is deferred to Section 5.

Proposition 4.4. *For every $j \geq 1$,*

$$\mathbb{P}\big[E_j \big| P_1, \ldots, P_{\tau_j}, |A^{(j-1)}| \geq 3\big] \leq n^{-\Omega(1)}.$$

Given the proposition, the proof is complete: Denote by T the event that the number of j's so that $|A^{(j)}| \geq 3$ is at least $K'' = 2\lfloor K'/8 - 6 \rfloor$. Using the union bound,

$$\begin{aligned} &\mathbb{P}\big[G(P) < K/1000\big] \\ &\quad \leq \mathbb{P}\big[G(P) < K/1000, T\big] + \mathbb{P}[\text{not } T] \\ &\quad \leq 2^{K''} \max_{H = \{j_1 < j_2 < \ldots < j_{K''/2}\} \subset [K'']} \mathbb{P}\Big[E_{j_1}, E_{j_2}, \ldots, E_{j_{K''/2}}, |A^{(K'')}| \geq 3\Big] + \mathbb{P}[\text{not } T]. \end{aligned}$$

For every $j \geq 1$, Chernoff's bound implies

$$\mathbb{P}\Big[|A^{(j)}| \geq |A^{(0)}| - 3 \cdot j \Big| E_1, \ldots, E_{j-1}, |A^{(j-1)}| \geq |A^{(0)}| - 3 \cdot (j-1)\Big] \geq 1 - c^{m^{1/3}}$$

with $c < 1$. Thus,

$$\mathbb{P}[\text{not } T] \leq nc^{m^{1/3}}.$$

Furthermore, for fixed H as above, by the proposition (and bounding $1-x \geq e^{-2x}$ for $0 \leq x \leq 1/2$),

$$\begin{aligned}
&\mathbb{P}\big[E_{j_1}, E_{j_2}, \ldots, E_{j_{K''/2}}, |A^{(K'')}| \geq 3\big] \\
&\leq \left(1 - c^{m^{1/3}}\right)^{-K''/2} \mathbb{P}[E_{j_1}] \frac{\mathbb{P}\big[E_{j_2}, E_{j_1}, |A^{(1)}| \geq |A^{(0)}| - 3\big]}{\mathbb{P}\big[E_{j_1}, |A^{(1)}| \geq |A^{(0)}| - 3\big]} \cdot \\
&\quad \cdot \frac{\mathbb{P}\big[E_{j_3}, E_{j_2}, E_{j_1}, |A^{(2)}| \geq |A^{(0)}| - 3 \cdot 2\big]}{\mathbb{P}\big[E_{j_2}, E_{j_1}, |A^{(2)}| \geq 3 \cdot 2\big]} \cdots \\
&\qquad \cdots \frac{\mathbb{P}\big[E_{j_1}, E_{j_2}, \ldots, E_{j_{K''/2}}, |A^{(j_{K''/2}-1)}| \geq |A^{(0)}| - 3 \cdot (K''/2 - 1)\big]}{\mathbb{P}\big[E_{j_1}, E_{j_2}, \ldots, E_{j_{K''/2}-1}, |A^{(j_{K''/2}-1)}| \geq |A^{(0)}| - 3 \cdot (K''/2 - 1)\big]} \\
&\leq e^{K'' c^{m^{1/3}}} n^{-\Omega(K'')} \leq n^{-\Omega(K)}.
\end{aligned}$$

Overall,

$$\mathbb{P}\big[G(P) < K/1000\big] \leq n^{-\Omega(K)}.$$

This completes the proof of Lemma 4.1. □

5 Distorted chessboard random walk

This section is devoted for the proof of Proposition 4.4. To prove the proposition, we use a different point of view of the random process. We begin by describing this different view, and later describe its formal connection to the distribution on pairings.

The view uses two definitions. One is a standard definition of a two-dimensional random walk, and the other is a definition of a "chessboard" configuration in the plane. The proof of the proposition will follow by analyzing the behavior of the random walk on the "chessboard."

Let n be a large integer and $m = \lfloor n^{3/4} \rfloor$. The random walk W on $\mathbb{N}^2$ is defined as follows. It starts at the origin, $W_0 = (0, 0)$. At every step it move to one of three nodes, independently of previous choices,

$$W_{t+1} = \begin{cases} W_t + (0,2) & \text{with probability } 1/3, \\ W_t + (1,1) & \text{with probability } 1/3, \\ W_t + (2,0) & \text{with probability } 1/3. \end{cases}$$

At time t, the L_1-distance of W_t from the origin is thus $2t$.

The "chessboard" is defined as follows. Let $\alpha_1 : [m] \to \{0,1\}$ and $\alpha_2 : [2m] \to \{0,1\}$ be two Boolean functions. The functions α_1, α_2 induce a "chessboard" structure on the *board* $[m] \times [2m]$. A position in the board $\xi = (\xi_1, \xi_2)$ is colored either white or black. It is colored black if $\alpha_1(\xi_1) \neq \alpha_2(\xi_2)$ and white if $\alpha_1(\xi_1) = \alpha_2(\xi_2)$. We say that the "chessboard" is *well-behaved* if

(i) α_1 is far from constant:

$$n^{5/8} \leq \big|\{\xi_1 \in [m] : \alpha_1(\xi_1) = 1\big| \leq m - n^{5/8},$$

(ii) α_1 does not contain many jumps:

$$\big|\{\xi_1 \in [m-1] : \alpha_1(\xi_1) \neq \alpha_1(\xi_1 + 1)\}\big| \leq n^{1/100},$$

and

(iii) α_2 does not contain many jumps:

$$\left|\{\xi_2 \in [2m-1] : \alpha_2(\xi_2) \neq \alpha_2(\xi_2+1)\}\right| \leq n^{1/100}.$$

Consider a random walk W on top of the "chessboard" and stop it when reaching the boundary of the board (i.e., when it tries to make a step outside the board $[m] \times [2m]$). We define a *good* step to be a step of the form $(1,1)$ that lands in a black block. We will later relate good steps to violating edges. Our goal is, therefore, to show that a typical W makes many good steps.

Lemma 5.1. *Assume the chessboard is well-behaved. The probability that W makes less than $n^{1/100}$ good steps is at most $n^{-\Omega(1)}$.*

Using this lemma we prove Proposition 4.4. We prove the lemma in Section 5.1 below.

Proof of Proposition 4.4. Recall that A_j is an arc of size $|A_j| = m = \lfloor n^{3/4} \rfloor$ so that there is a color k_j satisfying

$$n^{5/8} \leq \left|S_{k_j} \cap A_j\right| \leq m - n^{5/8}. \tag{5.1}$$

Furthermore, condition on $P_1, \ldots, P_{\tau_j}, |A^{(j-1)}| \geq 3$. Assume w.l.o.g. that R_{τ_j} is in A_j (when L_{τ_j} is in A_j, the analysis is similar). The distance of R_{τ_j} from the smallest element of A_j is at most one (the length of "one step to the right" is between zero and two). We now grow the random interval until σ_j, i.e., as long as R_t stays in A_j. At the same time, L_t performs a movement to the left. Since $\left|A^{(j-1)}\right| \geq 3$, there are at least $2m$ steps for L_t to take to the left before hitting A_j.

There is a one-to-one correspondence between pairings P and random walks W using the correspondence

$$\begin{aligned} P_{t+1} = \{L_t - 2, L_t - 1\} \quad &\leftrightarrow \quad W_{t+1} = W_t + (0,2) \\ P_{t+1} = \{L_t - 1, R_t + 1\} \quad &\leftrightarrow \quad W_{t+1} = W_t + (1,1) \\ P_{t+1} = \{R_t + 1, R_t + 2\} \quad &\leftrightarrow \quad W_{t+1} = W_t + (2,0). \end{aligned}$$

Define the function α_1 to be 1 at positions of A_j with color k_j, and 0 at the other positions. Set the function α_2 as to describe the color k_j from L_{τ_j} leftward. The "chessboard" is well-behaved by (5.1) and since k_j is in the set B defined in case 2 of the proof Lemma 4.1 (so there are not many jumps for the color k_j).

Finally, if W makes a good step, then the corresponding pair added to P violated color k_j. So, if E_j holds for P, then the corresponding W makes less than $n^{1/100}$ good steps. Formally, by Lemma 5.1,

$$\mathbb{P}\big[E_j \big| P_1, \ldots, P_{\tau_j}, |A^{(j-1)}| \geq 3\big] \leq \mathbb{P}[W \text{ makes less than } n^{1/100} \text{ good steps}] \leq n^{-\Omega(1)}.$$

□

5.1 Proof of Lemma 5.1

Define three events E_R, E_C, E_D, all of which happen with small probability, so that every W that is not in their union makes many good steps.

Call a subset of the board of the form $I \times [2m]$ or $[m] \times I$, where I is a sub-interval, a *region*. The *width* of a region is the size of I. Let R be the set of regions of width at least $n^{1/90}$. The size of R is at most $2m^2$. For a region r in R, denote by E_r the event that the number of steps of the form $(1,1)$ that W makes in its first $n^{1/95}$ steps in r is less than $n^{1/100}$. Denote

$$E_R = \bigcup_{r \in R} E_r.$$

By the union bound and Chernoff's bound,

$$\mathbb{P}[E_R] \leq c^{n^{1/200}}$$

with $c < 1$ a universal constant.

Denote by H the set of all points in the board with L_1-norm at least $m^{5/8}$. At time T the random walk W is distributed along all points in $\mathbb{N}^2$ of L_1-norm exactly T. The distribution of W on this set is the same as that of a random walk on $\mathbb{Z}$ that is started at 0, and moves at every step to the right w.p. $1/3$, stays in place w.p. $1/3$ and moves to the left w.p. $1/3$. The probability that such a random walk on $\mathbb{Z}$ is at a specific point in $\mathbb{Z}$ at time T is at most $O(T^{-1/2})$. Hence, for every point h in H,

$$\mathbb{P}[W \text{ hits } h] \leq O(m^{-5/16}) \leq m^{-1/4}.$$

Call a point $c = (\xi_1, \xi_2)$ in the board a *corner* if both (ξ_1, ξ_2) and $(\xi_1 + 1, \xi_2 + 1)$ are of the same color $\kappa \in \{\text{black}, \text{white}\}$, but $(\xi_1 + 1, \xi_2)$ and $(\xi_1, \xi_2 + 1)$ are not of color κ. For a corner c, denote by $\Delta(c)$ the $n^{1/90}$-neighborhood of c in L_1-metric. Denote by Δ the union over all $\Delta(c)$, for corners c in H. Denote by E_C the event that W hits any point in Δ. Since the board is well-behaved, the number of jumps in each of α_1, α_2 is at most $n^{1/100}$. Therefore, the number of corners is at most $n^{1/50}$. By the union bound,

$$\mathbb{P}[E_C] \leq O\big(n^{1/50} n^{2/90} m^{-1/4}\big) = n^{-\Omega(1)}.$$

Let $m' = \lceil m^{5/8} \rceil$. Define three (vertical) lines: D_1 is the line $\{m'\} \times [2m]$, D_2 is the line $\{2m'\} \times [2m]$ and D_3 is the line $\{m - m'\} \times [2m]$. Denote by E_D the event that W does not cross the line D_3 before stopped (i.e., hitting the boundary of the board). Chernoff's bound implies

$$\mathbb{P}[E_D] \leq c^{n^{1/200}}.$$

To conclude the proof, by the union bound, it suffices to show that for every W not in $E_R \cup E_C \cup E_D$, the walk W makes at least $n^{1/100}$ good steps. Fix such a walk W. Since $W \notin E_D$, we know that W crosses the line D_2.

There are several cases to consider. A *block* is a maximal monochromatic rectangle in the board. The board is thus partitioned into black blocks and white blocks. (Hence, the name "chessboard.") Think of the board $[m] \times [2m]$ as drawn in the plane with $(1,1)$ at the bottom-left corner and $(m, 2m)$ at the upper-right corner.

Case 1: The walk W does not hit any white block after crossing D_1 and before crossing D_2. In this case, all step in the region whose left border is D_1 and right border is D_2 are in a black area. The number of such steps is at least $m^{5/8}/2$. Since $W \notin E_R$, the claim holds.

Case 2: The walk W hits a white block after crossing D_1 and before crossing D_2. There are two sub-cases to consider: Number the blocks so that every block is associated with a pair $\langle \eta_1, \eta_2 \rangle$

where η_1 is between 1 and the number of jumps in α_1 and η_2 is between 1 and the number of jumps in α_2.

Case 2.1: At some point after crossing D_1 and before crossing D_3, there are two white blocks of either the form $\langle \eta_1, \eta_2 \rangle, \langle \eta_1 + 1, \eta_2 + 1 \rangle$ or the form $\langle \eta_1, \eta_2 \rangle, \langle \eta_1 - 1, \eta_2 - 1 \rangle$ so that W intersects both blocks. Let c be the corner between these two blocks (such a corner must exist). Since $W \notin E_C$, we know that W does not visit $\Delta(c)$. Therefore, W must walk in a black area around $\Delta(c)$. Every path surrounding $\Delta(c)$ has length at least $n^{1/90}$. Since $W \notin E_R$, the claim holds.

Case 2.2: At all times after crossing D_1 and before crossing D_3, the walk never moves from a white block $\langle \eta_1, \eta_2 \rangle$ to one of the two white block $\langle \eta_1 + 1, \eta_2 + 1 \rangle$, $\langle \eta_1 - 1, \eta_2 - 1 \rangle$. Since $W \notin E_D$, this is indeed the last case. The *width* of a combinatorial rectangle in the board is the size of its "bottom side" (i.e., the corresponding subset of $[m]$). Let η be the first white block W hits after crossing D_1. Let Σ be the family of black blocks that are to the right but on the same height as η. Define γ as the maximal width of a rectangle of the form $\sigma \cap [0, m - m' - 1] \times [2m]$ over all $\sigma \in \Sigma$. Since we are in case 2, the left border of η is to the left of D_2. Since the board is well-behaved, the total width of the black area to the right of the left border of η, on the same height as η and to the left of D_3 is at least $n^{5/8} - 3m'$. Therefore, since the number of jumps is at most $n^{1/100}$,

$$\gamma \geq (n^{5/8} - 3m')/n^{1/100} \geq n^{1/90}.$$

Since we are in case 2.2, the walk W must "go through" every black block it hits: it can go from bottom side to upper side or from left side to right side (but not from left side to upper side or from bottom side to right side). Because $W \notin E_D$, for each black block in Σ, therefore, there exists a black block "in the same column" that W crosses horizontally. Focussing on one such black block of width γ, since $W \notin E_R$, the claim holds. □

References

[1] S. J. Berkowitz. On computing the determinant in small parallel time using a small number of processors. *Inf. Proc. Letters* 18, pages 147–150, 1984.

[2] P. Bürgisser. *Completeness and Reduction in Algebraic Complexity Theory*, volume 7 of *Algorithms and Computation in Mathematics.* Springer-Verlag, Berlin, 2000.

[3] M. Mahajan and V. Vinay. Determinant: Combinatorics, Algorithms, and Complexity. *Chicago J. Theor. Comput. Sci.*, 1997, 1997.

[4] G. Malod and N. Portier. Characterizing Valiant's algebraic complexity classes. *Journal of Complexity* 24 (1), pages 16–38, 2008.

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

[6] R. Raz. Multi-linear formulas for permanent and determinant are of super-polynomial size. In *Proceedings of the 36th Annual STOC*, pages 633–641, 2004.

[7] R. Raz. Multilinear $NC_1 \neq$ Multilinear NC_2. In *Proceedings of the 45th Annual FOCS*, pages 344–351, 2004.

[8] R. Raz, A. Shpilka, and A. Yehudayoff. A lower bound for the size of syntactically multilinear arithmetic circuits. *SIAM J. on Computing* 38 (4), pages 1624–1647, 2008.

[9] R. Raz and A. Yehudayoff. Balancing syntactically multilinear arithmetic circuits. *Computational Complexity* 17 (4), pages 515–535, 2008.

[10] R. Raz and A. Yehudayoff. Lower bounds and separations for constant depth multilinear circuits. *Computational Complexity* 18 (2), pages 171–207, 2009.

[11] P. A. Samuelson. A method for determining explicitly the coefficients of the characteristic equation. *Ann. Math. Statist.* 13, pages 424–429, 1942.

[12] A. Shpilka and A. Yehudayoff. Arithmetic circuits: A survey of recent results and open questions. *Found. Trends Theor. Comput. Sci.* 5, pages 207–388, 2010.

[13] L. G. Valiant. Completeness classes in algebra. In *Proceedings of the 11th Annual STOC*, pages 249–261, 1979.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de