Electronic Colloquium on Computational Complexity, Revision 1 of Report No. 80 (2014)

Lower Bounds for Tropical Circuits and Dynamic Programs

Stasys Jukna[1]

University of Frankfurt, Institut of Computer Science, D-60054 Frankfurt, Germany

Abstract

Tropical circuits are circuits with Min and Plus, or Max and Plus operations as gates. Their importance stems from their intimate relation to dynamic programming algorithms. The power of tropical circuits lies somewhere between that of monotone boolean circuits and monotone arithmetic circuits. In this paper we survey known and present some new lower bounds arguments for tropical circuits, and hence, for dynamic programs.

Key words: Tropical circuits, Dynamic programming, Monotone arithmetic circuits, Lower bounds

1. Introduction

Understanding the power and limitations of fundamental algorithmic paradigms—such as greedy or dynamic programming—is one of the basic questions in the algorithm design and in the whole theory of computational complexity. In this paper we focus on the dynamic programming paradigm.

Our starting point is a simple observation that many dynamic programming algorithms for optimization problems are just recursively constructed *circuits* over the corresponding semirings. Each such circuit computes, in a natural way, some polynomial over the underlying semiring. Most of known dynamic programming algorithms correspond to circuits over the $(\min, +)$ or $(\max, +)$ semirings, that is, to *tropical circuits*.[2] Thus, lower bounds for tropical circuits show the limitations of dynamic programming algorithms over the corresponding semirings.

The power of tropical circuits (and hence, of dynamic programming) lies somewhere between that of monotone boolean circuits and monotone arithmetic circuits:

$$\text{monotone boolean} \;\leqslant\; \text{tropical} \;\leqslant\; \text{monotone arithmetic}$$

and the gaps may be even exponential (we will show this in Section 9).

Monotone *boolean* circuits are most powerful among these three models and, for a long time, only linear lower bounds were known for such circuits. First super-polynomial lower bounds for the k-clique function CLIQUE and the perfect matching function PER were proved by Razborov [35, 34] by inventing his method of approximations. At almost about the same time, explicit exponential

[☆]Research supported by the DFG grant SCHN 503/6-1.

Email addresses: jukna@thi.informatik.uni-frankfurt.de (Stasys Jukna)

[1]Affiliated with Vilnius University, Institute of Mathematics and Informatics, Vilnius, Lithuania.

[2]There is nothing special about the term "tropical". Simply, this term is used in honor of Imre Simon who lived in Sao Paulo (south tropic). Tropical algebra and tropical geometry are now intensively studied topics in mathematics.

ISSN 1433-8092

lower bounds were also proved by Andreev [3, 4]. Alon and Boppana [1] improved Razborov's lower bound for CLIQUE from super-polynomial until exponential. Finally, Jukna [15] gave a general and easy to apply lower bounds criterium for monotone boolean and real-valued circuits, yielding strong lower bounds for a row of explicit boolean functions. These lower bounds hold for tropical circuits as well.

On the other hand, monotone *arithmetic* circuits are much easier to analyze: such a circuit cannot produce anything else but the monomials of the computed polynomial, no "simplifications" (as $x^2 = x$ or $x + xy = x$) are allowed here. Exponential lower bounds on the monotone arithmetic circuit complexity were proved already by Schnorr [36] (for CLIQUE), and Jerrum and Snir [13] (for PER and some other polynomials). A comprehensive survey on arithmetic (not necessarily monotone) circuits can be found in the book by Shpilka and Yehudayoff [38].

In this paper we summarize our knowledge about the power of tropical circuits. As far as we know, no similar attempt was undertaken in this direction after the classical paper by Jerrum and Snir [13]. The main message of the paper is that not only methods developed for monotone *boolean* circuits, but (sometimes) even those for a much weaker model of monotone *arithmetic* circuits can be used to establish limitations of dynamic programming. Although organized as a survey, the paper contains some new results, including:

1. A short and direct proof that tropical circuits for optimization problems with *homogeneous* target polynomials are not more powerful than monotone arithmetic circuits (Theorem 9). This explains why we do not have efficient dynamic programming algorithms for optimization problems whose target sums all have the same length. In the case of Min-semirings, this was proved by Jerrum and Snir [13] using the Farkas lemma.
2. A new and simple proof of Schnorr's [36] lower bound on the size of monotone arithmetic circuits computing so-called "separated" polynomials (Theorem 12). A polynomial f is *separated* if the product of any two of its monomials contains no third monomial of f distinct from these two ones.
3. A new and simpler proof of Gashkov and Sergeev's [9, 10] lower bound on the size of monotone arithmetic circuits computing so-called "k-free" polynomials (Theorem 18). A polynomial is *k-free* if it does not contain a product of two polynomials, both with more than k monomials. This extend's Schnorr's bound, since every separated polynomial is also 1-free.
4. An easy to apply "rectangle" lower bound (Lemma 22).
5. A truly exponential lower bound for monotone arithmetic circuits using expander graphs (Theorem 27).

2. Semirings

A (commutative) semiring is a system $\mathrm{S} = (S, +, \times, \mathsf{0}, \mathsf{1})$, where S is a set, $+$ ("sum") and $\times$ ("product") are binary operations on S, and 0 and 1 are elements of S having the following three properties:

(i) in both $(S, +, \mathsf{0})$ and $(S, \times, \mathsf{1})$, operation are associative and commutative with identities $\mathsf{0}$ and $\mathsf{1}$: $a + \mathsf{0} = a$ and $a \times \mathsf{1} = a$ hold for all $a \in S$;

(ii) product distributes over sum: $a \times (b + c) = (a \times b) + (a \times c)$;

(iii) $a \times \mathsf{0} = \mathsf{0}$ for all $a \in S$ ("annihilation" axiom).

A semiring is *additively-idempotent* if $a+a=a$ holds for all $a \in S$, and is *multiplicatively-idempotent* if $a \times a = a$ holds for all $a \in S$.

We will use the common conventions to save parenthesis by writing $a \times b + c \times d$ instead of $(a \times b) + (a \times c)$, and replacing $a \times b$ by ab. Also, a^n will stand for $a \times a \times \cdots \times a$ n-times. If desired, we will also assume that the sets $\mathbb{N}$, $\mathbb{Z}$ or $\mathbb{R}$ also contain $+\infty$ and/or $-\infty$.

In this paper, we will be interested in the following semirings:

- Arithmetic semiring $\mathrm{A} = (\mathbb{N}, +, \cdot, 0, 1)$.
- Boolean semiring $\mathrm{B} = (\{0,1\}, \vee, \wedge, 0, 1)$.
- Min-semirings $\mathrm{Min} = (\mathbb{N}, \min, +, +\infty, 0)$ and $\mathrm{Min}^- = (\mathbb{Z}, \min, +, +\infty, 0)$.
- Max-semirings $\mathrm{Max} = (\mathbb{N}, \max, +, -\infty, 0)$ and $\mathrm{Max}^- = (\mathbb{Z}, \max, +, -\infty, 0)$.
- Min- and Max-semirings are called *tropical* semirings.

Note that all these semirings, but A, are additively-idempotent, and none of them, but B, is multiplicatively-idempotent. Note also that in arithmetic and in tropical semirings one usually allows rational or even real numbers, not just integers. This corresponds to considering optimization problems with real, not necessarily integral "weights". The point, however, is that lower-bound techniques, we will consider below, work already on smaller domains. In fact, they work when, besides ∞ or $-\infty$, the domain contains 0 and 1 or 0 and -1. Roughly speaking, the larger is the domain, the easier is to prove lower bounds over them. In particular, the bounds remain true in larger domains as well.

Due to their intimate relation to discrete optimization, we will be mainly interested in tropical semirings, and circuits over these semirings. Lower bounds for such circuits give lower bounds for the number of subproblems used by dynamic programming algorithm. The semirings Min^- and Max^- are isomorphic via the transformation $x \mapsto -x$, so we will not consider Max^- separately: all results holding for Min^- hold also for Max^-.

3. Polynomials

Let $\mathrm{S} = (S, +, \times, \mathsf{0}, \mathsf{1})$ be a semiring, and let $x_1, \ldots, x_n$ be variables ranging over S. A *monomial* is any product of these variables, where repetitions are allowed. By commutativity and associativity, we can sort the products and write monomials in the usual notation, with the variables raised to exponents. Thus, every monomial $x_1^{a_1} x_2^{a_2} \cdots x_n^{a_n}$ is uniquely determined by the vector of exponents $(a_1, \ldots, a_n) \in \mathbb{N}^n$, where $x_i^0 = \mathsf{1}$. (Note that in tropical semirings, monomials are linear combinations $a_1 x_1 + a_2 x_2 + \cdots + a_n x_n$, that is, sums, not products.) The *degree*, $|p|$, of a monomial is the sum $|p| = a_1 + \cdots + a_n$ of its exponents. A monomial p is *multilinear* if every exponent a_i is either 0 or 1. A monomial $p = x_1^{a_1} \cdots x_n^{a_n}$ *contains* a monomial $q = x_1^{b_1} \cdots x_n^{b_n}$ (or q is a *factor* of p) if $a_i \geqslant b_i$ for all $i = 1, \ldots, n$, that is, if $p = qq'$ for some monomial q'.

By a *polynomial*[3] we will mean a finite sum of monomials, where repetitions of monomials are allowed. That is, we only consider polynomials with nonnegative integer coefficients. A polynomial is *homogeneous* if all its monomials have the same degree, and is *multilinear* if all its monomials

[3]Usually, polynomials of more than one variable are called *multivariate*, but we will omit this for shortness.

are multilinear (no variables of degree > 1). For example, $f = x^2y + xyz$ is homogeneous but not multilinear, whereas $g = x + yz$ is multilinear but not homogeneous. The sum and product of two polynomials is defined in the standard way. For polynomials f, h and a monomial p, we will write:

- $f = h$ if f and h have the same monomials appearing not necessarily with the same coefficients;
- $f \rightleftharpoons h$ if f and h have the same monomials appearing with the same coefficients;
- $f \subseteq h$ if every monomial of f is also a monomial of h;
- $p \in f$ if p is a monomial of f;
- $|f|$ to denote the number of *distinct* monomials in f;
- X_p to denote the set of variables appearing in p with non-zero degree.

Every polynomial $f(x_1, \ldots, x_n)$ defines a function $\hat{f} : S^n \to S$, whose value $\hat{f}(s_1, \ldots, s_n)$ is obtained by substituting elements $s_i \in S$ for x_i in f. Polynomials f and g are *equivalent* (or *represent the same function*) over a given semiring, if $\hat{f}(s) = \hat{h}(s)$ holds for all $s \in S^n$. It is important to note that the same polynomial $f(x) \rightleftharpoons \sum_{I \in \mathcal{I}} c_I \prod_{i \in I} x_i^{a_i}$ represents different functions over different semirings:

$$\hat{f}(x) = \sum_{I \in \mathcal{I}} c_I \prod_{i \in I} x_i^{a_i} \qquad \text{over A (counting)}$$

$$\hat{f}(x) = \bigvee_{I \in \mathcal{I}} \bigwedge_{i \in I} x_i \qquad \text{over B (existence)}$$

$$\hat{f}(x) = \min_{I \in \mathcal{I}} \sum_{i \in I} a_i x_i \qquad \text{over Min and Min}^- \text{ (minimization)}$$

$$\hat{f}(x) = \max_{I \in \mathcal{I}} \sum_{i \in I} a_i x_i \qquad \text{over Max and Max}^- \text{ (maximization)}$$

Note that in the boolean semiring as well as in all four tropical semirings, the coefficients c_I do not influence the computed value $\hat{f}(x)$, and we can assume that $c_I = 1$ for all $I \in \mathcal{I}$; this is because, say, $\min\{x, x, y\} = \min\{x, y\}$. The degrees, however, are important: say, $\min\{2x, y\} \neq \min\{x, y\}$.

4. Structure of Equivalent Polynomials

Let f and h be any two polynomials on the same set of variables. In general, if f and h are equivalent (i.e. if $\hat{f} = \hat{h}$ holds) over some semiring, then neither $f \rightleftharpoons h$ nor even $f = h$ need to hold. The arithmetic semiring is here an exception.

Lemma 1. *If $\hat{f} = \hat{h}$ holds over the arithmetic semiring* A*, then $f \rightleftharpoons h$.*

Proof. There are several ways to prove this fact. We follow the argument suggested by Sergey Gashkov (personal communication). Suppose that $\hat{f} = \hat{h}$ but $f \not\rightleftharpoons h$. Since $f \not\rightleftharpoons h$, the polynomial $g = f - h$ contains at least one monomial. Let p be a monomial of g of maximum degree. Take all partial derivatives of g with respect to the variables of p until all they disappear. Since p has maximum degree, we obtain some constant $\neq 0$. But since $\hat{g} = \hat{f} - \hat{h}$ is a zero function, the derivative should be zero, a contradiction. □

In tropical semirings, we only have weaker structural properties. For a polynomial f, let $f_{\min} \subseteq f$ denote the set of all monomials of f not containing any other monomial of f, and $f_{\max} \subseteq f$ denote the set of all monomials of f not contained in any other monomial of f. For example, if $f = \{x, x^2y, yz\}$, then $f_{\min} = \{x, yz\}$ and $f_{\max} = \{x^2y, yz\}$. Note that every monomial of f contains (properly or not) at least one monomial of $f_{\min}$, and is contained in at least one monomial of $f_{\max}$. Note also that $\hat{f}_{\min} = \hat{f}$ holds in Min semirings, and $\hat{f}_{\max} = \hat{f}$ holds in Max semirings.

Lemma 2. *If $\hat{f} = \hat{h}$ holds over* Min*, and if h is multilinear, then $f_{\min} = h_{\min}$.*

Proof. Let us first show that every monomial of f must contain at least one monomial of h, and hence, of $h_{\min}$. To see this, assume that there is a monomial $p \in f$ which contains no monomial of h. Since h is multilinear, this means that every monomial of h must contain a variable not in X_p. So, on the assignment a_p which sets to 1 all variables in X_p, and sets to ∞ all the remaining variables, we have that $\hat{h}(a_p) = \infty$. But $\hat{f}(a_p) \leqslant \hat{p}(a_p) = |X_p| < \infty$, a contradiction with $\hat{f} = \hat{h}$.

Since no monomial in $f_{\min}$ can contain another monomial of f, it remains therefore to show that $h_{\min} \subseteq f$. For this, assume that there is a monomial $q \in h_{\min}$ such that $q \notin f$. If we take the assignment a_q, then $\hat{h}(a_q) = \hat{q}(a_q) = |X_q|$. On the other hand, the assignment a_q sets all monomials $p \in f$ such that $X_p \not\subseteq X_q$ to ∞. Each of the remaining monomials $p \in f$ (if there is any) must satisfy $X_p \subseteq X_q$. But we already know that p must contain some monomial $q' \in h_{\min}$, that is, $X_{q'} \subseteq X_p \subseteq X_q$. Since both monomials q and q' are multilinear and belong to $h_{\min}$, this implies $q = q'$, and hence, also $X_p = X_q$. Since q is multilinear and $p \neq q$, this means that p must have strictly larger degree $|p|$ than $|X_q|$, and hence, $\hat{p}(a_q) = |p| > |X_q| = \hat{h}(a_q)$, a contradiction with $\hat{f} = \hat{h}$. □

Remark 1. Note that Lemma 2 needs not to hold, if both polynomials are not multilinear. Say, if $f = \min\{2x, x+y, 2y\}$ and $h = \min\{2x, 2y\}$, then $\hat{f} = \hat{h}$ holds (because $x + y \geqslant \min\{2x, 2y\}$), but $f_{\min} = f \neq h = h_{\min}$. In this example, monomial $x + y = \frac{1}{2}(2x) + \frac{1}{2}(2y)$ is a convex combination of the monomials $2x$ and $2y$. And in fact, using the Farkas lemma about solvability of systems of linear inequalities, Jerrum and Snir [13] have proved that, if f and h are arbitrary (not necessarily multilinear) polynomials such that $\hat{f} = \hat{h}$ holds over Min, then there is a set $h' \subseteq h$ of monomials such that $h' \subseteq f$, and every monomial of $f \cup h$ is at least some convex combination of the monomials in h'.

Lemma 3. *If $\hat{f} = \hat{h}$ holds over* Max*, and if h is multilinear, then f is also multilinear, and $f_{\max} = h_{\max}$.*

Proof. Assume that f is not multilinear. Then f contains a monomial p (sum) in which some variable x_i appears more than once. If we set this variable to 1 and the rest to 0, then $\hat{h}$ takes some value $\leqslant 1$, but $\hat{f}$ takes value $|p| \geqslant 2$, a contradiction with $\hat{f} = \hat{h}$. Thus, both polynomials f and h must be multilinear.

We claim that every monomial of f must be contained in at least one monomial of h. Indeed, if some monomial $p \in f$ is contained in none of the monomials $q \in h$, then every monomial $q \in h$ is missing at least one variable from X_p. So, on the assignment b_p which sets to 1 all variables in X_p, and sets to 0 all the remaining variables, we have that $\hat{h}(b_p) \leqslant |X_p| - 1$. But $\hat{f}(b_p) \geqslant \hat{p}(b_p) = |X_p|$, a contradiction with $\hat{f} = \hat{h}$.

It remains therefore to show that $h_{\max} \subseteq f$. For this, assume that there is a monomial $q \in h_{\max}$ such that $q \notin f$. If we take the assignment b_q, then $\hat{h}(a_q) = \hat{q}(a_q) = |X_q|$. On the other hand,

$X_p \not\supseteq X_q$ must holds for every monomial $p \in f$, implying that $\hat{f}(b_q) \leqslant |X_q| - 1$. Indeed, we already know that every monomial $p \in f$ must be contained in some monomial $q' \in h_{\max}$. So, $X_p \supseteq X_q$ implies $X_{q'} \supseteq X_p \supseteq X_q$. Since both monomials q and q' belong to $h_{\max}$, this implies $X_p = X_q$, and hence also $p = q$ because both monomials p and q are multilinear. This contradicts our assumption $q \notin f$. □

In tropical semirings Min^- and Max^-, we have an even stronger property.

Lemma 4. *If $\hat{f} = \hat{h}$ holds over a tropical semiring Min^- or Max^-, and if h is multilinear, and then $f = h$.*

Proof. We claim that the polynomial f must be also multilinear. To see this, assume that f contains a monomial p (sum) in which some variable x_i appears more than once. Then, in the semiring Min^-, we can set $x_i = -1$ and $x_j = 0$ for all $j \neq i$. Under this assignment, we have $\hat{h}(x) \geqslant -1$, because all monomials of h get value $\geqslant -1$, but $\hat{f}(x) \leqslant -2$ since already the monomial p of f gets value $\leqslant -2$, a contradiction. The the Max^- semiring (and even in Max), it is enough to set $x_i = 1$ and $x_j = 0$ for all $j \neq i$ to get the desired contradiction.

Let us now show that $f = h$ must hold over the semiring Min^-; the argument for Max^- is similar. We know that both polynomials f and h are multilinear. Hence, Lemma 2 implies that $f_{\min} = h_{\min}$ (this holds even in Min). In particular, every monomial of f must contain at least one monomial of h, and every monomial of h must contain at least one monomial of f. Thus, $h \not\subseteq f$ can only happen, if there is a monomial $p \in h$ such that, for every monomial $q \in f$, we have that either $X_p \not\supseteq X_q$ or $X_p \supset X_q$ (proper inclusion). In any case, every monomial $q \in f$ misses some variable of p. So, if we assign -1 to all variables of p, and 0 to the remaining variables, then h takes some value $\leqslant -|X_p|$. But since each monomial $q \in f$ misses at least one variable of p, the value of each of these monomials, and hence the value of f, must be $\geqslant |X_p| + 1$, a contradiction with $\hat{f} = \hat{h}$. This shows $h \subseteq f$. The proof of the converse inclusion $f \subseteq h$ is the same. □

Note that for non-multilinear polynomials f, Lemma 4 needs not to hold. For example, If $f = \min\{x, 2x, 3x\}$ and $h = \min\{x, 3x\}$, then $\hat{f} = \hat{h}$ holds over Min^-, but $f \neq h$.

5. Circuits and their Polynomials

A *circuit* F over a semiring $\mathrm{S} = (S, +, \times, \mathsf{0}, \mathsf{1})$ is a usual fanin-2 circuit whose inputs are variables $x_1, \ldots, x_n$ and constants $\mathsf{0}$ and $\mathsf{1}$. Gates are fanin-2 $+$ and $\times$. That is, we have a directed acyclic graph with $n + 2$ fanin-0 nodes labeled by $x_1, \ldots, x_n, \mathsf{0}, \mathsf{1}$. At every other node, the sum $(+)$ or the product $(\times)$ of its entering nodes is computed; nodes with assigned operations are called *gates*. The *size* of F, denoted by $\mathrm{Size}(\mathsf{F})$, is the number of gates in F. The *depth* is the largest number of edges in a path from an input gate to an output gate.

Like polynomials, circuits are also "syntactic" objects. So, we can associate with every circuit F the unique polynomial F *produced* by F inductively as follows:[4]

- If $\mathsf{F} = x_i$, then $F \rightleftharpoons x_i$.
- If $\mathsf{F} = \mathsf{G} + \mathsf{H}$, then $F \rightleftharpoons \sum_{p \in G} p + \sum_{q \in H} q$.

[4]We will always denote circuits as upright letters $\mathsf{F}, \mathsf{G}, \mathsf{H}, \ldots$, and their produced polynomials by italic versions $F, G, H, \ldots$.

- If $\mathsf{F} = \mathsf{G} \times \mathsf{H}$, then $F \rightleftharpoons \sum_{p \in G} \sum_{q \in H} pq$.

When producing the polynomial F from a circuit F we only use the generic semiring axioms (i)–(iii) to write the result as a polynomial (sum of monomials). For example, if $\mathsf{F} = x \times (1 + y)$ then $F = x + xy$, even though $\hat{F} = x$ in B and Min, and $\hat{F} = xy$ in Max. It is thus important to note that the produced by a given circuit F polynomial F is the same over *any* semiring!

Definition 1. A circuit F *computes* a polynomial f if $\hat{F} = \hat{f}$ (F and f coincide as functions). A circuit F *produces* f if $F = f$ (F and f have the same set of monomials).

A circuit F *simultaneously* computes (or produces) a given set $\mathcal{F}$ of polynomials if, for every polynomial $f \in \mathcal{F}$, there is a gate in F at which f is computed (or produced).

When analyzing circuits, the following concept of "parse graphs" is often useful. A *parse-graph* G in F is defined inductively as follows: G includes the root (output gate) of F. If u is a sum-gate, then exactly one of its inputs is included in G. If u is a product gate, then both its input gates are included in G. Note that each parse-graph produces exactly one monomial in a natural way, and that each monomial $p \in F$ is produced by at least one parse-graph. If p is multilinear, then each parse-graph for p is a tree.

- A circuit is *homogeneous*, if polynomials produced at its gates are homogeneous. It is easy to see that a circuit is homogeneous if and only if the polynomial produced by it is homogeneous.
- A circuit is *multilinear*, if for every its product gate $u = v \times w$, the sets of variables of the polynomials produced at gates v and w are disjoint. Sometimes, multilinear (in our sense) circuits are called also *syntactically multilinear*.

Note that multilinear circuits can only compute multilinear polynomials but, in general, circuits computing multilinear polynomials need not be multilinear: this happens, for example, in semirings B and Min. Still, Lemmas 3 and 4 imply that this cannot happen in the remaining three tropical semirings:

Lemma 5. *Every circuit computing a multilinear polynomial f over* A, Max, Min^- *or* Max^- *must be multilinear. Moreover, over* A, Min^- *and* Max^-*, the circuit must even produce f.*

We will be interested in the following two complexity measures of polynomials f, where the third measure is only for multilinear polynomials:

- $\mathrm{S}(f)$ = minimum size of a circuit over semiring S *computing* f.
- $\mathrm{S}[f]$ = minimum size of a circuit over semiring S *producing* f.
- $\mathrm{S}_{\mathrm{lin}}(f)$ = minimum size of a *multilinear* circuit over semiring S computing f.

What we are really interested in is the first measure $\mathrm{S}(f)$. The second measure $\mathrm{S}[f]$ is less interesting: it is the *same* for all semirings S, because the formal polynomial of a given (fixed) circuit is the same over all semirings. In particular, we have that

$$\mathrm{S}[f] = \mathrm{A}[f]$$

holds for every semiring S and every polynomial f. Still, it will be sometimes convenient *not* to focus on the arithmetic semiring A because the inequality $\mathrm{S}(f) \geqslant \mathrm{S}[f]$ is more informative: it

means that computing a given polynomial over S is not easier than to produce this polynomial. This, for example, happens in the arithmetic semiring A: Lemma 1 implies that $\mathrm{A}(f) \geqslant \mathrm{A}[f]$.

Also, Lemma 5 implies that the third measure $\mathrm{S}_{\mathrm{lin}}(f)$ may be only interesting in semirings B and Min: if $\mathrm{S} \in \{\mathrm{Max}, \mathrm{Min}^-, \mathrm{Max}^-, \mathrm{A}\}$, then for every multilinear polynomial f, we have that $\mathrm{S}_{\mathrm{lin}}(f) = \mathrm{S}(f)$.

6. Some Polynomials

For the ease of reference, here we recall some polynomials which we will use later to illustrate the lower bound arguments. Variables x_e of considered polynomials correspond to edges of K_n or $K_{n,n}$. Thus, monomials $\prod_{e \in E} x_e$ correspond to some subgraphs E of K_n or $K_{n,n}$. Here are some of the polynomials we will use later:

- Permanent polynomial PER_n = all perfect matchings in $K_{n,n}$.
- Hamiltonian cycle polynomial HC_n = all Hamiltonian cycles in K_n.
- k-clique polynomial $\mathrm{CLIQUE}_{n,k}$ = all k-cliques in K_n.
- Spanning tree polynomial ST_n = all spanning trees in K_n rooted in node 1.
- st-connectivity polynomial STCON_n = all paths from $s = 1$ to $t = n$ in K_n.
- All-pairs connectivity "polynomial" APSP_n = *set* of $\binom{n}{2}$ polynomials STCON_n corresponding to different pairs of start and target nodes s and t.
- Matrix product polynomial MP_n = special case of APSP_n when only paths of length-2 are considered.
- The connectivity polynomial CONN_n = product of all polynomials of APSP_n.

In Section 13 we will show that the first four polynomials require Min-circuits of exponential size, whereas the next result shows that the last four polynomials all have Min-circuits of polynomial size. The following result—proved independently by Moore [28], Floyd [7], and Warshall [41]—holds for every semiring with the absorption axiom $a + ab = a$, including the boolean and Min semirings.

Theorem 6 ([28, 7, 41])**.** *Over semirings* Min *and* B*, the polynomials of* APSP_n *can all be simultaneously computed by a circuit of size* $O(n^3)$*.*

Proof. Inputs for APSP_n over the Min semiring are non-negative weights x_{ij} of the edges of K_n. For every pair $i < j$ of distinct nodes of K_n, the goal is to compute the weight of the lightest path between i and j; the weight of a path is the sum of weights of its edges. The idea is to recursively compute the polynomials $f_{i,j}^{[k]}$ for $k = 0, 1, \ldots, n$, whose value is the weight of the lightest walk between i and j whose all inner nodes lie in $[k] = \{1, \ldots, k\}$. Then $f_{i,j}^{[0]} = x_{ij}$, and the recursion is: $f_{i,j}^{[k]} = \min\left\{f_{i,j}^{[k-1]},\ f_{i,k}^{[k-1]} + f_{k,j}^{[k-1]}\right\}$. The output gates are $f_{i,j}^{[n]}$ for all $i < j$. The total number of gates is $O(n^3)$. Even though the circuit actually searches for weights of lightest *walks*, it correctly computes APSP because every walk between two nodes i and j also contains a simple path (with

Polynomial f	**Bound**	**Reference**
ST_n	$\mathrm{B}(f) = O(n^3)$, $\mathrm{S}(f) = 2^{\Omega(n)}$	Rem. 3, Thm. 23
CONN_n, STCON_n	$\mathrm{Min}(f) = O(n^3)$, $\mathrm{A}[f] \geqslant \mathrm{Max}(f) = 2^{\Omega(n)}$	Rem. 3
APSP_n, MP_n	$\mathrm{Min}(f) = \Theta(n^3)$	Cor. 15
PER_n, HC_n	$\mathrm{S}(f) = 2^{\Omega(n)}$	Thm. 23
$\mathrm{CLIQUE}_{n,k}$	$\mathrm{S}(f) \geqslant \binom{n}{k} - 1$	Cor. 14

Table 1: Summary of specific bounds; $\mathrm{S}(f)$ stands for any of $\mathrm{Min}(f)$, $\mathrm{Max}(f)$ and $\mathrm{B}_{\mathrm{lin}}(f)$.

no repeated nodes) between these nodes. Since the weights are non-negative, the minimum must be achieved on a simple path. If we replace min-gates by OR-gates, and sum-gates by AND-gates, then the resulting circuit will compute APSP_n over the boolean semiring B. □

Remark 2. Earlier dynamic programming algorithm of Bellman [6] and Ford [8] gives a (structurally) simpler Min-circuit for STCON_n. It tries to compute the polynomials $f_j^{[k]}$ whose value is the weight of the lightest walk between 1 and j with at most k edges. Then $f_j^{[1]} = x_{1j}$, and the recursion is: $f_j^{[k]}$ = the minimum of $f_j^{[k-1]}$ and of $f_i^{[k-1]} + x_{i,j}$ over all nodes $i \neq j$. The output gate is $f_n^{[n-1]}$. The circuit also has $O(n^3)$ fanin-2 gates.

Remark 3. Theorem 6 immediately implies that the polynomials MP_n, CONN_n, and STCON_n can also be computed by Min-circuits of size $O(n^3)$. Moreover, over the boolean semiring, the spanning tree polynomial ST represents the same boolean function as CONN. Thus, Theorem 6 also gives $\mathrm{B}(\mathrm{ST}_n) = O(n^3)$.

In the rest of the paper, we will present various lower bound argument for tropical circuits. Table 1 summarizes the resulting specific bounds obtained by these arguments for the polynomials listed above.

7. Reduction to the Boolean Semiring

A semiring $\mathrm{S} = (S, +, \times, \mathsf{0}, \mathsf{1})$ is of *zero-characteristic*, if $\mathsf{1} + \mathsf{1} + \cdots + \mathsf{1} \neq \mathsf{0}$ holds for any finite sum of the unity $\mathsf{1}$. Note that all semirings we consider are of zero-characteristic. The following seems to be a "folklore" observation.

Lemma 7. *If a semiring* S *is of zero-characteristic, then* $\mathrm{S}(f) \geqslant \mathrm{B}(f)$ *holds for every polynomial* f.

Proof. Let F be a circuit over S computing a given polynomial f. The circuit must correctly compute f on any subset of the domain S. We choose the subset $S_+ = \{\mathsf{0}, \overline{\mathsf{1}}, \overline{\mathsf{2}}, \ldots\}$, where $\overline{n} = \mathsf{1} + \cdots + \mathsf{1}$ is the n-fold sum of the multiplicative unit element $\mathsf{1}$. Note that $\overline{n} \neq \mathsf{0}$ holds for all $n \geqslant 1$, because S has zero-characteristic.

Since $\overline{n} + \overline{m} = \overline{n+m}$ and $\overline{n} \times \overline{m} = \overline{n \cdot m}$, $\mathrm{S}_+ = (S_+, +, \times, \mathsf{0}, \mathsf{1})$ is a semiring. Since $S_+ \subseteq S$, the circuit must correctly compute f over this semiring as well. But the mapping $h : S_+ \to \{0, 1\}$ given by $h(\mathsf{0}) = 0$ and $h(\overline{n}) = 1$ for all $n \geqslant 1$, is a homomorphism from S_+ into the boolean semiring B with $h(x + y) = h(x) \vee h(y)$ and $h(x \times y) = h(x) \wedge h(y)$. So, if we replace each $+$-gate by a logical OR, and each $\times$-gate by a logical AND, then the resulting monotone boolean circuit computes the polynomial f over B. □

Remark 4. One can easily show that, if the input variables can only take boolean values 0 and 1, then $\mathrm{Min}(f) \leqslant 2 \cdot \mathrm{B}(f)$ holds for every multilinear polynomial. Indeed, having a (boolean) circuit F for f, just replace each AND gate $u \wedge v$ by a Min gate $\min(u, v)$, and each OR gate $u \vee v$ by $\min(1, u + v)$. The point however is that tropical circuits must work correctly on much larger domain than $\{0, 1\}$. This is why lower bounds for tropical circuits do not translate to lower bounds for monotone boolean circuits. And indeed, there are explicit polynomials f, as the spanning tree polynomial $f = \mathrm{ST}_n$, such that $\mathrm{B}(f) = O(n^3)$ but $\mathrm{Min}(f) = 2^{\Omega(n)}$; the upper bound is shown in Remark 3, and the lower bound will be shown in Theorem 23.

To prove lower bounds in the boolean semiring—and hence, by Lemma 7, also in tropical semirings—one can try to use the following general lower bounds criterion proved in [15] (see also [17, Sect. 9.4] for a simplified proof).

For $a \in \{0, 1\}$, an *a-term* of a monotone boolean function is a subset of its variables such that, when all these variables are fixed to the constant a, the function outputs value a, independent of the values of other variables. It is easy to see that every 0-term must intersect every 1-term, and vice versa. Say that a family of sets A *covers* a family of sets B if every set in B contains at least one set of A.

Definition 2. A monotone boolean function $f(X)$ of $|X| = n$ variables is *t-simple* if for all integers integers $2 \leqslant r, s \leqslant n$, such that

(i) either the set of all 0-terms of f can be covered by $t(r-1)^s$ s-element subsets of X,

(ii) or the set of all 1-terms of f can be covered by at most $t(s-1)^r$ r-element subsets of X plus $s-1$ single variables.

Note that this "asymmetry" between (i) and (ii) (allowing additional $s-1$ single variables in a cover) is important: say, condition (i) is trivially violated, if f contains a 0-term $T = \{x_1, \ldots, x_k\}$ with $k < s$. But then (ii) is fatisfied, because T must intersect all 1-terms, implying that the single variables $x_1, \ldots, x_k$ cover all of them.

Theorem 8 ([15])**.** *If f is not t-simple, then* $\mathrm{B}(f) > t$.

8. Reduction to the Arithmetic Semiring

As we already mentioned in the introduction, circuits over the arithmetic semiring A are no more powerful than circuits over boolean or tropical semirings. The weakness of circuits computing a given polynomial f over A lies in the fact (following from Lemma 1) that they cannot produce any "redundant" monomials, those not in f. That is, here we have $\mathrm{A}(f) \geqslant \mathrm{A}[f]$. On the other hand, if the semiring S is additively-idempotent, then

$$\mathrm{S}(f) \leqslant \mathrm{S}[f] = \mathrm{A}[f]\,. \tag{1}$$

This holds because in an additively-idempotent semiring S (where $x + x = x$ holds), the multiplicities of monomials have no effect on the represented function. But, in general, we have no converse inequality $\mathrm{S}(f) \geqslant \mathrm{A}[f]$: for some polynomials f, $\mathrm{A}[f]$ may be even exponentially larger than $\mathrm{S}(f)$. Such is, for example, the *st*-connectivity polynomial $f = \mathrm{STCON}_n$. For this polynomial, we have $\mathrm{Min}(f) = O(n^3)$ (see Remark 3), but it is relatively easy to show that $\mathrm{Min}[f] = \mathrm{A}[f] = 2^{\Omega(n)}$ (see

Theorem 24 below). We will now show that the reason for such a large gap is the non-homogeneity of STCON.

Following Jerrum and Snir [13], define the *lower envelope* of a polynomial f to be the polynomial $f_{\rm le}$ consisting of all monomials of f of smallest degree. Similarly, the *higher envelope*, $f_{\rm he}$, of f consists of all monomials of f of largest degree. Note that both polynomials $f_{\rm le}$ and $f_{\rm he}$ are homogeneous, and $f_{\rm le} = f_{\rm he} = f$, if f itself is homogeneous.

Observation 1. If a polynomial f can be produced by a circuit of size s, then both $f_{\rm le}$ and $f_{\rm he}$ can be produced by homogeneous circuits of size s.

Proof. Take a circuit producing f. The desired homogeneous sub-circuits producing the lower or the higher envelope can be obtain by starting with input gates, and removing (if necessary) one of the wires of every sum-gate, at inputs of which polynomials of different degrees are produced. □

Theorem 9. *For every multilinear polynomial f, we have*

$$\mathrm{A}[f] \geqslant \mathrm{B}_{\rm lin}(f) \geqslant \mathrm{Min}(f) \geqslant \mathrm{A}[f_{\rm le}] \quad \textit{and} \quad \mathrm{A}[f] \geqslant \mathrm{Max}(f) \geqslant \mathrm{A}[f_{\rm he}]\,. \tag{2}$$

If f is also homogeneous, then

$$\mathrm{B}_{\rm lin}(f) = \mathrm{Min}(f) = \mathrm{Max}(f) = \mathrm{A}[f]\,.$$

Proof. By (1), we only have to prove the lower bounds (2). To prove that $\mathrm{B}_{\rm lin}(f) \geqslant \mathrm{Min}(f)$, let F be a multilinear monotone boolean circuit computing f. Since the circuit is multilinear, its produced polynomial F is also multilinear. Since every monotone boolean function has a *unique* shortest monotone DNF, this implies that $F_{\rm min} = f_{\rm min}$. Since f and $f_{\rm min}$ represent the same function over Min, the circuit F with OR gates replaced by Min gates, and AND gates by Sum gates will compute f over Min.

To prove the inequality $\mathrm{Min}(f) \geqslant \mathrm{A}[f_{\rm le}]$, take a minimal circuit F over Min computing f. Observation 1 implies that the lower envelope $F_{\rm le}$ of the polynomial F produced by F can be also produced by a (homogeneous) circuit of size at most $\mathrm{Size}(\mathsf{F})$. Hence, $\mathrm{A}[F_{\rm le}] \leqslant \mathrm{Size}(\mathsf{F}) = \mathrm{Min}(f)$. On the other hand, Lemma 2 implies that $f_{\rm le} = F_{\rm le}$, and we are done.

The proof of $\mathrm{Max}(f) \geqslant \mathrm{A}[f_{\rm he}]$ is the same by using Lemma 3. □

The second claim of Theorem 9 has an important implication concerning the power of dynamic programs, which can be roughly stated as follows:

> For optimization problems whose target polynomials are *homogeneous*, dynamic programming is no more powerful than monotone arithmetic circuits!

9. Relative Power of Semirings

The reductions to the boolean and to the arithmetic semirings (Lemma 7 and Theorem 9) give us the following relations for every multilinear polynomial f:

$$\mathrm{B}(f) \leqslant \mathrm{Min}(f) \leqslant \mathrm{B}_{\rm lin}(f) \leqslant \mathrm{Min}^{-}(f) = \mathrm{A}[f]$$

and

$$\mathrm{B}(f) \leqslant \mathrm{Max}(f) \leqslant \mathrm{Max}^{-}(f) = \mathrm{A}[f]\,.$$

If, additionally, f is also homogeneous, then

$$\mathrm{B}(f) \leqslant \mathrm{B}_{\mathrm{lin}}(f) = \mathrm{Min}(f) = \mathrm{Max}(f) = \mathrm{Min}^{-}(f) = \mathrm{Max}^{-}(f) = \mathrm{A}[f]\,.$$

Moreover, all inequalities are strict: for some polynomials f, one side can be even exponentially smaller than the other. Moreover, the Max/Min and Min/Max gaps can be also exponential.

To show that circuits over the tropical semirings can be exponentially weaker than those over the boolean semiring, consider the the spanning tree polynomial $f = \mathrm{ST}_n$ and the graph connectivity polynomial $g = \mathrm{CONN}_n$. Over the boolean semiring B, these polynomials represent the same boolean function: a graph is connected if and only if it has a spanning tree. This gives $\mathrm{B}(f) = \mathrm{B}(g)$ and $\mathrm{B}_{\mathrm{lin}}(f) = \mathrm{B}_{\mathrm{lin}}(g)$. Moreover, we already know (see Remark 3) that $\mathrm{B}(g) = O(n^3)$ and $\mathrm{Min}(g) = O(n^3)$. On the other hand, a relatively simple argument (the "rectangle bound") yields $\mathrm{A}[f] = 2^{\Omega(n)}$ (see Theorem 23 below). Since the polynomial f is homogeneous, Theorem 9 implies that $\mathrm{Min}(f)$, $\mathrm{Max}(f)$ and $\mathrm{B}_{\mathrm{lin}}(f)$ coincide with $\mathrm{A}[f]$, and hence, are also exponential in n. We thus have gaps:

$$\begin{aligned} \mathrm{Min}(f)/\mathrm{B}(f),\ \mathrm{Max}(f)/\mathrm{B}(f) &= 2^{\Omega(n)} \quad \text{for } f = \mathrm{ST}_n; \\ \mathrm{B}_{\mathrm{lin}}(g)/\mathrm{Min}(g),\ \mathrm{B}_{\mathrm{lin}}(g)/\mathrm{B}(g) &= 2^{\Omega(n)} \quad \text{for } g = \mathrm{CONN}_n. \end{aligned}$$

The latter gap $\mathrm{B}_{\mathrm{lin}}(g)/\mathrm{B}(g) = 2^{\Omega(n)}$ also shows that there is no "multilinear version" of the Floyd–Warshall algorithm, even in the boolean semiring.

To show that the remaining gaps can also be exponential, it is enough to take *any* multilinear and homogeneous polynomial $f(x_1, \ldots, x_n)$ such that $\mathrm{A}[f]$ is exponential in n, and to consider its two "saturated" versions $\underline{f}$ and $\overline{f}$, where $\underline{f}$ is obtained by adding to f all n monomials $x_1, x_2, \ldots, x_n$ of degree 1, and $\overline{f}$ is obtained by adding to f the monomial $x_1 x_2 \cdots x_n$ of degree n.

Lemma 10. *Let $f(x_1, \ldots, x_n)$ be a multilinear and homogeneous polynomial. Then both $\mathrm{Min}(\overline{f})$ and $\mathrm{Max}(\underline{f})$ are at least $\mathrm{A}[f]$, but all $\mathrm{Max}(\overline{f})$, $\mathrm{Min}(\underline{f})$ and $\mathrm{B}_{\mathrm{lin}}(\underline{f})$ are at most n.*

Proof. Since f is the lower envelope of $\overline{f}$, and the higher envelope of $\underline{f}$. Theorem 9 implies that $\mathrm{Min}(\overline{f}) \geqslant \mathrm{A}[f]$ and $\mathrm{Max}(\underline{f}) \geqslant \mathrm{A}[f]$. On the other hand, over the Max-semiring, the polynomial $\overline{f}$ computes $x_1 + x_2 + \cdots + x_n$, whereas over the Min-semiring, $\underline{f}$ computes $\min\{x_1, x_2, \ldots, x_n\}$, and computes $x_1 \vee x_2 \vee \cdots \vee x_n$ over the boolean semiring. Hence, all $\mathrm{Max}(\overline{f})$, $\mathrm{Min}(\underline{f})$ and $\mathrm{B}_{\mathrm{lin}}(\underline{f})$ are at most n. □

Since, there are many linear and homogeneous polynomials requiring monotone arithmetic circuits of exponential size (see, e.g. Table 1), the saturated versions of f immediately give exponential gaps.

Still, the "saturation trick" leads to somewhat artificial examples, and it would be interesting to establish exponential gaps using "natural" polynomials. For example, the Max/Min gap is achieved already on a very natural *st*-connectivity polynomial $h = \mathrm{STCON}_n$. We know that $\mathrm{Min}(h) = O(n^3)$ (Remark 3), but a simple argument (see Theorem 24) shows that $\mathrm{Max}(h) = 2^{\Omega(n)}$. Hence,

$$\mathrm{Max}(h)/\mathrm{Min}(h) = 2^{\Omega(n)} \quad \text{for } h = \mathrm{STCON}_n.$$

From now on we concentrate on the lower bound *arguments* themselves.

10. Lower Bounds for Separated Polynomials

Let $g(x_1, \ldots, x_n)$ be a polynomial in $n \geqslant 3$ variables. An *enrichment* of g is a polynomial h in $n-1$ variables obtained by taking some variable x_k and replacing it by a sum $x_i + x_j$ or by a product $x_i x_j$ of some other two (not necessarily distinct) variables, where $k \notin \{i, j\}$. A *progress measure* of polynomials is an assignment of non-negative numbers $\mu(g)$ to polynomials g such that

(i) $\mu(x_i) = 0$ for each variable x_i;

(ii) $\mu(h) \leqslant \mu(g) + 1$ for every enrichment h of g.

Lemma 11. *For every polynomial f, and every progress measure $\mu(f)$, we have $\mathrm{A}[f] \geqslant \mu(f)$.*

Proof. Take a monotone arithmetic circuit F with $s = \mathrm{A}[f]$ gates producing f. We argue by induction on s. If $s = 0$, then $\mathsf{F} = x_i$ in an input variable, and we have $\mathrm{A}[f] = 0 = \mu(f)$. For the induction step, take one gate $u = x_i * x_j$ where $* \in \{+, \cdot\}$. Let $\mathsf{F}'(x_1, \ldots, x_n, y)$ be the circuit with the gate u replaced by a new variable y. Hence, $\mathrm{Size}(\mathsf{F}') = \mathrm{Size}(\mathsf{F}) - 1$ and $F(x_1, \ldots, x_n)$ is an enrichment of $F'(x_1, \ldots, x_n, y)$. By the induction hypothesis, we have that $\mathrm{Size}(\mathsf{F}') \geqslant \mu(F')$. Together with $\mu(F) \leqslant \mu(F') + 1$, this yields $\mathrm{Size}(\mathsf{F}) = \mathrm{Size}(\mathsf{F}') + 1 \geqslant \mu(F') + 1 \geqslant \mu(F)$. □

Recall that a monomial p *contains* a monomial q (as a factor), if $p = qq'$ for some monomial q'.

Definition 3. A sub-polynomial $P \subseteq f$ is *separated* if the product pq of any two monomials p and q of P contains no monomial of f distinct from p and from q. Let

$$\mathrm{sep}(f) := \max\{|P| - 1 \colon P \subseteq f \text{ is separated}\}\,.$$

Note that we consider separateness *within* the entire set f of monomials: it is not enough that the product pq contains no third monomial of P—it must not contain any third monomial of the entire polynomial f.

Note also that a multilinear polynomial f of minimum degree m is separated, if every monomial of f is uniquely determined by any subset of $\lceil m/2 \rceil$ its variables. (Being uniquely determined means that no other monomial contains the same subset of variables.) Indeed, if $p \times q$ contains some monomial r then r and p (or r and q) must share at least $\lceil m/2 \rceil$ variables, implying that $r = p$ (or $r = q$) must hold.

Theorem 12 (Schnorr [36])**.** *For every polynomial f, we have $\mathrm{A}[f] \geqslant \mathrm{sep}(f)$, where*

$$\mathrm{sep}(f) := \max\{|P| - 1 \colon P \subseteq f \textit{ is separated}\}\,.$$

In particular, $\mathrm{A}[f] \geqslant |f| - 1$ if the polynomial f itself is separated.

Proof. It is enough to show that the measure $\mathrm{sep}(f)$ is a progress measure. The first condition (i) is clearly fulfilled, since $\mathrm{sep}(x_i) = 1 - 1 = 0$. To verify the second condition (ii), let $f(x_1, \ldots, x_n, y)$ be a polynomial, and $h(x_1, \ldots, x_n)$ be its enrichment. Our goal is to show that $\mathrm{sep}(f) \geqslant \mathrm{sep}(h) - 1$. We only consider the "hard" case when y is replaced by a sum of variables: $h(x_1, \ldots, x_n) = f(x_1, \ldots, x_n, u + v)$, where $u, v \in \{x_1, \ldots, x_n\}$.

To present the proof idea, we first consider the case when no monomial of f contains more than one occurrence of the variable y. Then every monomial yp of f turns into two monomials up and

vp of h. To visualize the situation, we may consider the bipartite graph $G \subseteq f \times h$, where every monomial $yp \in f$ is connected to two monomials $up, vp \in h$; each monomial $q \in f$ without y is connected to $q \in h$. Take now a separated subset $P \subseteq h$ such that $|P| - 1 = \mathrm{sep}(h)$, and let $Q \subseteq f$ be the set of its neighbors in G. Our goal is to show that:

(a) $|Q| \geqslant |P| - 1$, and

(b) Q is separated.

Then the desired inequality $\mathrm{sep}(f) \geqslant |Q| - 1 \geqslant |P| - 2 = \mathrm{sep}(h) - 1$ follows.

To show item (a), it is enough to show that at most one monomial in Q can have both its neighbors in P. To show this, assume that this holds for some two monomials yp and yq of Q. Then all four monomials up, vp, uq, vq belong to P. But this contradicts the separateness of P, because the product $up \times vq$ contains the third monomial uq (and vp).

To show item (b), assume that the product $p \times q$ of some two monomials $p \neq q$ of Q contains some third monomial $r \in h$. Let $p', q' \in P$ be some neighbors of p and q lying in P. Then the product $p' \times q'$ must contain one (of the two) neighbors of r. Since *both* of these neighbors of r belong to h, we obtain a contradiction with the separateness of P.

In general (if y can have any degrees in f), a monomial $y^k p$ of f has $k+1$ neighbors $u^i v^{k-i} p$, $i = 0, 1, \ldots, k$ in h. To show (a), it is again enough to show that at most one monomial in Q can have two neighbors in P. For this, assume that there are two monomials $p \neq q$ such that all four monomials $u^a v^{k-a} p, u^b v^{k-b} p, u^c v^{l-c} q, u^d v^{l-d} q$ belong to P. Assume w.l.o.g. that $a = \max\{a, b, c, d\}$. Then the product $u^a v^{k-a} p \times u^c v^{l-c} q$ contains $u^a v^{l-c} q$, and (since $c \leqslant a$) contains the monomial $u^a v^{l-a} q$ of h, contradicting the separateness of P. The proof of (b) is similar. □

Remark 5. It is not difficult to see that we have a stronger inequality $\mathrm{sep}(f) \geqslant \mathrm{sep}(h)$, if the variable y is replaced by the product uv (instead of the sum $u + v$). Thus, in fact, Theorem 12 gives a lower bound on the number of sum gates.

As a simple application of Schnorr's argument, consider the *triangle polynomial*

$$\mathrm{TR}_n(x, y, z) = \sum_{i,j,k \in [n]} x_{ik} y_{kj} z_{ij} \,.$$

This polynomial has $3n$ variables and n^3 monomials.

Corollary 13. *If* $f = \mathrm{TR}_n$, *then* $\mathrm{Min}(f) = \mathrm{Max}(f) = \mathrm{A}[f] = \Theta(n^3)$.

Proof. The equalities $\mathrm{Min}(f) = \mathrm{Max}(f) = \mathrm{A}[f]$ hold by Theorem 9, because f is multilinear and homogeneous. The upper bound $\mathrm{A}[f] = O(n^3)$ is trivial. To prove the lower bound $\mathrm{A}[f] = \Omega(n^3)$, observe that every monomial $p = x_{ik} y_{kj} z_{ij}$ of f is uniquely determined by any choice of any two of its three variables. This implies that p cannot be contained in a union of any two monomials distinct from p. Thus, the polynomial f is separated, and its Schnorr's measure is $\mathrm{sep}(f) = n^3 - 1$. Theorem 12 yields $\mathrm{A}[f] \geqslant \mathrm{sep}(f) = n^3 - 1$, as desired. □

Recall that the k-clique polynomial $\mathrm{CLIQUE}_{n,k}$ has $\binom{n}{k}$ monomials $\prod_{i<j \in S} x_{ij}$ corresponding to subsets $S \subseteq [n]$ of size $|S| = k$. This is a homogeneous multilinear polynomial of degree $\binom{k}{2}$. Note that TR_n is a sub-polynomial of $\mathrm{CLIQUE}_{3n,3}$ obtained by setting some variables to 0.

By Lemma 7, an exponential lower bound for $\mathrm{CLIQUE}_{n,s}$ over the tropical Min follows from Razborov's lower bound for this polynomial over the boolean semiring B [35]. However, the proof over B is rather involved. On the other hand, in tropical semirings such a bound comes quite easily.

Corollary 14. *For* $f = \mathrm{CLIQUE}_{n,k}$, $\mathrm{Min}(f)$, $\mathrm{Max}(f)$ *and* $\mathrm{B}_{\mathrm{lin}}(f)$ *are at least* $\binom{n}{k} - 1$.

This lower bound on $\mathrm{B}_{\mathrm{lin}}(f)$ was proved by Krieger [22] using different arguments.

Proof. Since f is multilinear and homogeneous, it is enough (by Theorems 9) to show the corresponding lower bound on $\mathrm{A}[f]$. By Theorem 12, it is enough to show that f is separated.

Assume for the sake of contradiction, that the union of two distinct k-cliques A and B contains all edges of some third clique C. Since all three cliques are distinct and have the same number of nodes, C must contain a node u which does not belong to A and a node v which does not belong to B. This already leads to a contradiction because either the node u (if $u = v$) or the edge $\{u, v\}$ (if $u \neq v$) of C would remain uncovered by the cliques A and B. □

Recall that the dynamic programming algorithm of Floyd–Warshall implies that the all-pairs shortest path polynomial APSP_n, and hence, also the matrix product polynomial MP_n, have Min-circuits of size $O(n^3)$; see Theorem 6. On the other hand, using Theorem 12 one can show that this algorithm is optimal: a cubic number of gates is also necessary.

Corollary 15. *Both* $\mathrm{Min}(\mathrm{APSP}_n)$ *and* $\mathrm{Min}(\mathrm{MP}_n)$ *are* $\Theta(n^3)$.

Proof. It is enough to show that $\mathrm{Min}(\mathrm{MP}_n) = \Omega(n^3)$. Recall that $\mathrm{MP}_n(x, y)$ is the set of all n^2 polynomials $f_{ij} = \sum_{k\in[n]} x_{ik}y_{kj}$. Since the triangle polynomial $\mathrm{TR}_n = \sum_{i,j\in[n]} z_{ij}f_{ij}$ is just a single-output version of MP_n, and its complexity is by at most an additive factor of $2n^2$ larger than that of MP_n, the desired lower bound for MP_n follows directly from Corollary 13. □

Kerr [21] earlier proved $\mathrm{Min}(\mathrm{MP}_n) = \Omega(n^3)$ using a different argument, which essentially employs the fact the Min-semiring contains more than two distinct elements. Since this "domain-dependent" argument may be of independent interest, we sketch it.

Proof. (Due to Kerr [21]) Let F be a Min-circuit computing all n^2 polynomials

$$f_{ij}(x) = \min\{x_{ik} + y_{kj} \colon k = 1, \ldots, n\}\,.$$

By Lemma 2, for each polynomial f_{ij} there must be a gate u_{ij}, the polynomial F_{ij} produced at which is of the form $F_{ij} = \min\{f_{ij}, G_{ij}\}$, where G_{ij} is some set of monomials (sums), each containing at least one monomial of f_{ij}.

Assign to every monomial $p = x_{ik} + y_{kj}$ of f_{ij} a sum gate u_p with the following two properties: (i) p is produced at u_p, and (ii) there is a path from u_p to u_{ij} containing no sum gates. Since $a + a = a$ does not hold in Min, at least one such gate must exist for each of the monomials $x_{ik} + y_{kj}$.

It remains therefore to show that no other term $x_{ab} + y_{bc}$ gets the same gate u_p. To show this, assume the opposite. Then at the gate u_p some sum

$$\min\{x_{ik}, \alpha, \ldots\} + \min\{y_{kj}, \ldots\}$$

is computed, where $\alpha \in \{x_{ab}, y_{bc}\}$ is a single variable distinct from x_{ik} and y_{kj}. Set $\alpha := 0$, $x_{ik} = y_{kj} := 1$ and set all remaining variables to 2. Then the first minimum in the sum above

evaluates to 0, and we obtain $\hat{F}_{ij}(x) \leqslant 1$. But $\hat{f}_{ij}(x) = 2$ because the term $x_{ik} + y_{kj}$ gets value $1 + 1 = 2$, and the remaining terms of f_{ij} get values $\geqslant 2 + 0 = 2$. This gives the desired contradiction. □

Remark 6. Using more subtle arguments, Paterson [31], and Mehlhorn and Galil [27] succeeded to prove a cubic lower bound $\Omega(n^3)$ for MP_n even over the boolean semiring B.

Remark 7. The argument used by Schnorr [36] is inductive, and is currently known as the *gate-elimination method*. Having a circuit F of n variables, replace its first gate by a new variable, use induction hypothesis for the resulting circuit F' of $n+1$ variables but of smaller size to make a desired conclusion about the original circuit F. Using a similar gate-elimination reasoning, Baur and Strassen [5] proved the following surprising upper bound: if a polynomial $f(x_1, \ldots, x_n)$ can be produced by a circuit of size s, then the polynomial f and all its n partial derivatives $\partial f/\partial x_i$ $(i = 1, \ldots, n)$ can all be simultaneously produced by a circuit of size only $4n$. (Note that a trivial upper bound is about sn.) Their (relatively simple) argument uses gate-elimination together with the chain rule for partial derivatives. If the polynomial f is multilinear, then $\partial f/\partial x_i$ is a polynomial obtained from f by removing all monomials not containing x_i, and removing x_i from all remaining monomials. In particular, if a sum $f = \sum_{i=1}^{k} y_i f_i(x_1, \ldots, x_n)$ can be produced by a circuit of size s, then all polynomials $f, f_1, \ldots, f_k$ can be simultaneously produced by a circuit of size $4s$.

11. Decompositions and Cuts

Besides the gate-elimination method, most of lower bound arguments for monotone arithmetic circuits follow the following general frame: if a polynomial f can be produced by a circuit of size s, then f can be written as a sum $f = \sum_{i=1}^{t} g_i$ of $t = O(s)$ "rectangles" g_i. Usually, these "rectangles" g_i are products of two (or more) polynomials of particular degrees. Let us first explain, where these "rectangles" come from.

Let F be a circuit over some semiring $\mathrm{S} = (S, +, \times, \mathsf{0}, \mathsf{1})$. For a gate u in F, let $\mathrm{pol}(u)$ denote the polynomial produced at u, and let $\mathsf{F}_{u=\mathsf{0}}$ denote the circuit obtained from F by replacing the gate u by the additive identity $\mathsf{0}$. Recall that $a \times \mathsf{0} = \mathsf{0}$ holds for all $a \in S$. Hence, the polynomial $F_{u=\mathsf{0}}$ produced by $\mathsf{F}_{u=\mathsf{0}}$ consists of only those monomials of F which do not "use" the gate u for their production. To avoid trivialities, we will always assume that $F_{u=\mathsf{0}} \neq F$, i.e. that there are no "redundant" gates.

Lemma 16. *For every gate u in F, the polynomial F produced by F can be written as a sum $F = F_u + F_{u=\mathsf{0}}$ of two polynomials, the first of which has the form $F_u = \mathrm{pol}(u) \times \mathrm{ext}(u)$ for some polynomial $\mathrm{ext}(u)$.*

Proof. If we replace the gate u by a new variable y, the resulting circuit produces a polynomial of the form $y \times A + F_{u=\mathsf{0}}$ for some polynomial A. It remains to substitute all occurrences of the variable y with the polynomial $\mathrm{pol}(u)$ produced at the gate u. □

Remark 8. Roughly speaking, the number $|F_u|$ of monomials in the polynomial F_u is the "contribution" of the gate u to the production of the entire polynomial F. Intuitively, if this contribution is small for many gates, then there must be many gates in F. More formally, associate with each monomial $p \in F$ some of its parse-graphs F_p in F. Observe that $u \in \mathsf{F}_p$ implies $p \in F_u$. Thus,

double-counting yields

$$\mathrm{Size}(\mathsf{F}) = \sum_{u\in\mathsf{F}} 1 \geqslant \sum_{u\in\mathsf{F}} \sum_{p\in F:\ u\in\mathsf{F}_p} \frac{1}{|F_u|} = \sum_{p\in F} \sum_{u\in\mathsf{F}_p} \frac{1}{|F_u|} \geqslant |F| \cdot \min_{p\in F} \sum_{u\in\mathsf{F}_p} \frac{1}{|F_u|}\,.$$

So, in principle, one can obtain strong lower bounds on the total number of gates in F by showing that this latter minimum cannot be too small.

The polynomial $\mathrm{ext}(u)$ in Lemma 16 can be explicitly described by associating polynomials with paths in the circuit F. Let π be a path from a gate u to the output gate, $u_1,\ldots,u_m$ be all product gates along this path (excluding the first gate u, if it itself is a product gate), and $w_1,\ldots,w_m$ be input gates to these product gates *not lying* on the path π. We associate with π the polynomial $\mathrm{pol}(\pi) := \mathrm{pol}(w_1)\times\mathrm{pol}(w_2)\times\cdots\times\mathrm{pol}(w_m)$. Then

$$\mathrm{ext}(u) = \sum_{\pi} \mathrm{pol}(\pi)\,,$$

where the sum is over all paths π from u to the output gate.

Lemma 16 associates sub-polynomials $\mathrm{pol}(u)\times\mathrm{ext}(u)$ of F with *nodes* (gates) u of F. In some situations, it is more convenient to associate sub-polynomials with *edges*. For this, associate with every edge (u,v), where $v = u * w$ is some gate with $*\in\{+,\times\}$ of F, the polynomial

$$\mathrm{ext}_u(v) := A\times\mathrm{ext}(v) \quad \text{where} \quad A = \begin{cases} 1 & \text{if } * = +;\\ \mathrm{pol}(w) & \text{if } * = \times. \end{cases}$$

That is, $\mathrm{ext}_u(v) = \mathrm{ext}(v)$ if v is a sum gate, and $\mathrm{ext}_u(v) = \mathrm{pol}(w)\times\mathrm{ext}(v)$ if v is a product gate.

A *node-cut* in a circuit is a set U of its nodes (gates) such that every input-output path contains a node in U. Similarly, an *edge-cut* is a set E of edges such that every input-output path contains an edge in E. Recall that, in our notation, "$f = h$" for two polynomials f and h only means that their *sets* of monomials are the same—their multiplicities (coefficients) may differ.

Lemma 17. *If U is a node-cut and E an edge-cut in a circuit F, then*

$$F = \sum_{u\in U} \mathrm{pol}(u)\times\mathrm{ext}(u) = \sum_{(u,v)\in E} \mathrm{pol}(u)\times\mathrm{ext}_u(v)\,.$$

Proof. The fact that all monomials of the last two polynomials are also monomials of F follows from their definitions. So, it is enough to show that every monomial $p\in F$ belongs to both of these polynomials. For this, take a parse graph F_p of p. Since U forms a node-cut, the graph F_p must contain some node $u\in U$. The monomial p has a form $p = p'p''$ where p' is the monomial produced by the subgraph of F_p rooted in u. Hence, $p'\in\mathrm{pol}(u)$ and $p''\in\mathrm{ext}(u)$. Similarly, since E forms an edge-cut, the graph F_p contains some edge $(u,v)\in E$. The monomial p has the form $p = p'p''$ where p' is the monomial produced by the subgraph of F_p rooted in u. Hence, $p'\in\mathrm{pol}(u)$ and $p''\in\mathrm{ext}_u(v)$. □

12. Bounds for (k,l)-free Polynomials

A polynomial f is (k,l)-*free* $(1\leqslant k\leqslant l)$ if f does not contain a product of two polynomials, one with $>k$ monomials and the other with $>l$ monomials. A polynomial f is f-*free* if it is (k,k)-free, that is, if

$$A\times B\subseteq f \text{ implies } \min\{|A|,|B|\}\leqslant k.$$

Note that this alone gives no upper bound on the total number $|A \times B|$ of monomials in the product $A \times B$.

Theorem 18. *If a (k,l)-free polynomial f can be produced by a circuit of size s, then f can be written as a sum of at most $2s$ products $A \times B$ with $|A| \leqslant k$ and $|B| \leqslant l^2$. In particular,*

$$\mathrm{A}[f] \geqslant \frac{|f|}{2kl^2}\,.$$

Proof. Our argument is a mix of ideas of Gashkov and Sergeev [10], and of Pippenger [32]. Take a minimal circuit F producing f; hence, $F = f$ is (k,l)-free. This implies that every product gate $u = v \times w$ in F must have an input, say w, at which a "small" set $A = |\mathrm{pol}(w)|$ of only $|A| \leqslant l$ monomials is produced. We thus can remove the edge (w,u) and replace u by a unary (fanin-1) gate $u = v \times A$ of scalar multiplication by this fixed (small) polynomial A. If both inputs produce small polynomials, then we eliminate only one of them. What we achieve by doing this is that input gates remain the same as in the original circuit (variables $x_1, \ldots, x_n$ and constants $\mathsf{0}, \mathsf{1}$), each product gate has fanin 1, and for every edge (u,v) in the resulting circuit F', we have an upper bound

$$|\mathrm{ext}_u(v)| \leqslant l \cdot |\mathrm{ext}(v)|\,. \tag{3}$$

Say that an edge (u,v) in F' is *legal* if both $|\mathrm{pol}(u)| \leqslant k$ and $|\mathrm{ext}_u(v)| \leqslant l^2$ hold. Let E be the set of all legal edges; hence, $\mathrm{Size}(\mathsf{F}) \geqslant |E|/2$. By Lemma 17, it remains to show that E forms an edge-cut of F'.

To show this, take an arbitrary input-output path P in F', and let $e = (u,v)$ be the last gate of P with $|\mathrm{pol}(u)| \leqslant k$. If v is the output gate, then $\mathrm{ext}(v)$ is a trivial polynomial $\mathsf{1}$, and hence, $|\mathrm{ext}_u(v)| \leqslant l$ by (3), meaning that (u,v) is a legal edge. Suppose now that v is not the output gate. Then $|\mathrm{pol}(u)| \leqslant k$ but $|\mathrm{pol}(v)| > k$. Held also $|\mathrm{ext}_u(v)| > l^2$, then (3) would imply that $|\mathrm{ext}(v)| \geqslant |\mathrm{ext}_u(v)|/l > l$. Together with $|\mathrm{pol}(v)| > k$ and $\mathrm{pol}(v) \times \mathrm{ext}(v) \subseteq F$, this would contradict the (k,l)-freeness of F. Thus, $|\mathrm{pol}(u)| \leqslant k$ and $|\mathrm{ext}_u(v)| \leqslant l^2$, meaning that (u,v) is a legal edge. □

Together with Theorem 9, Theorem 18 yields the following lower bound over tropical semirings for polynomials, whose only lower or higher envelopes are required to be (k,l)-free.

Corollary 19. *Let f and g be polynomials such that f_{le} and g_{he} are (k,l)-free for some $1 \leqslant k \leqslant l$. Then*

$$\mathrm{Min}(f) \geqslant \frac{|f_{\mathrm{le}}|}{2kl^2} \quad \textit{and} \quad \mathrm{Max}(g) \geqslant \frac{|g_{\mathrm{he}}|}{2kl^2}\,.$$

Remark 9. Using a deeper analysis of circuit structure, Gashkov and Sergeev [9, 10] were able to even estimate the numbers of sum and product gates: every monotone arithmetic circuit computing a (k,l)-free polynomial f of n variables must have at least $|f|/K - 1$ sum gates, and at least $2\sqrt{|f|/K} - n - 2$ product gates, where $K = \max\{k^3, l^2\}$.

Remark 10. Every boolean $n \times n$ matrix $A = (a_{ij})$ defines a a set $Ay = (f_1, \ldots, f_n)$ of n linear polynomials $f_i(y) = \sum_j a_{ij} y_j$, as well as a single-output bilinear polynomial $f_A(x,y) = \sum_i x_i f_i(y) = \sum_{i,j\colon a_{ij}=1} x_i y_j$ on $2n$ variables. Call a boolean matrix A (k,l)-*free*, if it does not contain any $(k+1, l+1)$ all-1 submatrix. It is clear that the polynomial f_A is (k,l)-free if and only if the matrix A is (k,l)-free.

Results of Nechiporuk [30] (re-discovered later by Mehlhorn [26] and Pippenger [32]) imply that, if A is (k,k)-free, then $\mathrm{B}(Ax) \geqslant |A|/4k^3$, where $|A|$ is the number of 1-entries in A. This, however, does not immediately yield a similar lower bound on $\mathrm{B}(f_A)$ for the *single-output* version f_A and, in fact, no such bound is known so far in the boolean semiring. (A lower bound $\mathrm{B}(f_A) \geqslant |A|$ for $(1,1)$-free matrices is only known when restricted to circuits with gates of fanout 1; see [17, Theorem 7.2].) On the other hand, Theorem 18 gives such a bound at least for tropical and multilinear boolean circuits: if A is (k,k)-free, then

$$\mathrm{Min}(f_A) = \mathrm{Max}(f_A) = \mathrm{B}_{\mathrm{lin}}(f_A) = \mathrm{A}[f_A] \geqslant |A|/2k^3\,,$$

where the equalities follow from Theorem 9, because the polynomial f_A is homogeneous.

13. Rectangle Bound

An *m-balanced product-polynomial* is a product of two polynomials, one of which has minimum degree d satisfying $m/3 < d \leqslant 2m/3$, and is itself a product of two nonempty polynomials.

Lemma 20 (Sum-of-Products)**.** *If a polynomial f of minimum degree at least $m \geqslant 3$ can be produced by a circuit with s product gates, then f can be written as a sum of at most s m-balanced product-polynomials.*

Proof. Let d be the minimum degree of f, and F be a circuit with s product gates producing f. Hence, $F = f$ and $d \geqslant m$. By the degree d_u of a gate gate $u \in \mathsf{F}$ we will mean the minimum degree of the polynomial produced at u. In particular, the degree of the output gate is d.

Claim 21. *For every $\epsilon \in (1/d, 1)$, there exists a product gate u with $d_u \in (\epsilon d/2, \epsilon d]$.*

Proof. Start at the output gate of F, and traverse the circuit (in the reverse order of edges) by always choosing the input of larger degree until a gate $v = u * v$ of degree $d_v > \epsilon d$ is found such that both d_u and d_w are $\leqslant \epsilon d$. Assume w.l.o.g. that $d_u \geqslant d_w$. Since $d_v \leqslant d_u + d_w \leqslant 2d_u$, the gate u has the desired degree $\epsilon d/2 < d_u \leqslant \epsilon d$. If the gate u is a sum gate, then at least one of its inputs must have the same degree d_u. So, we can traverse the circuit further until a product gate of degree d_u is found. □

Now, we apply Claim 21 with $\epsilon := 2m/3d$ to find a product gate u of degree $m/3 = \epsilon d/2 \leqslant d_u \leqslant \epsilon d = 2m/3$. By Lemma 16, we can write F as $F = F_u + F_{u=0}$ where $F_u = A \times B$ is a product of two polynomials such that the minimum degree of A lies between $m/3$ and $2m/3$, and A itself is a product of two nonempty polynomials (since u is a product gate); hence, F_u is an m-balanced product-polynomial. The polynomial $F_{u=0}$ is obtained from F by removing some monomials. If $F_{u=0}$ is empty, then we are done. Otherwise, the polynomial $F_{u=0}$ still has minimum degree at least m, and can be produced by a circuit with one product gate fewer. So, we can repeat the same argument for it, until the empty polynomial is obtained. □

Remark 11. Lemma 20 remains true if, instead of the minimum degree measure $d(f)$ of polynomials, one takes the minimum length $l(f)$ of a monomial of f, where the *length* of a monomial p is defined as the number $|X_p|$ of distinct variables occurring in p. Hence, we always have that $d(f) \geqslant l(f)$, and $d(f) = l(f)$ holds if f is multilinear. The same argument works because $l(F_{u=0}) \geqslant l(F)$, as long as the polynomial $F_{u=0}$ is not empty.

To upper-bound the maximal possible number $|A \times B|$ of monomials in a product-polynomial $A \times B \subseteq f$, the following measure of *factor-density* naturally arises: for an integer $r \geqslant 0$, let $\#_r(f)$ be the maximum number of monomials in f containing a fixed monomial of degree r as a common factor. This measure tells us how much the monomials of f are "stretched": the faster $\#_r(f)$ decreases with increasing r, the more stretched f is. Note that, if d is the maximum degree of f, then

$$1 = \#_d(f) \leqslant \#_{d-1}(f) \leqslant \ldots \leqslant \#_1(f) \leqslant \#_0(f) = |f| \,.$$

The factor-density measure allows to upper-bound the number of monomials in product-polynomials over any semiring which is not multiplicatively-idempotent (where $a^2 = a$ holds only for $a = 1$). Such are, in particular, the arithmetic semiring as well as all four tropical semirings. The only property of such semirings we will use is that, if p is a monomial and A is a polynomial, then $|A| \leqslant |\{p\} \times A|$ holds. Note that this needs not to hold in semirings which *are* multiplicatively-idempotent: the polynomial $A = \{x, y\}$ has two monomials, but $\{xy\} \times A = \{x^2y, xy^2\} = \{xy\}$ has only one monomial.

Observation 2. Let A and B be polynomials over a not multiplicatively-idempotent semiring of maximum degrees a and b. If $A \times B \subseteq f$, then $|A \times B| \leqslant \#_a(f) \cdot \#_b(f)$.

Proof. Fix a monomial $p \in A$ of degree $|p| = a$, and a monomial $q \in B$ of degree $|q| = b$. Since $\{p\} \times B \subseteq f$, we have that $|B| \leqslant |\{p\} \times B| \leqslant \#_{|p|}(f) = \#_a(f)$. Similarly, since $A \times \{q\} \subseteq f$, we have that $|A| \leqslant |A \times \{q\}| \leqslant \#_{|q|}(f) = \#_b(f)$. □

Lemma 22 (Rectangle Bound)**.** *For every polynomial f of minimum degree at least $m \geqslant 3$, there is an integer $m/3 < r \leqslant 2m/3$ such that*

$$\mathrm{A}[f] \geqslant \frac{|f|}{\#_r(f) \cdot \#_{m-r}(f)} \,.$$

Moreover, the lower bound is on the number of product gates.

Proof. Let F be a minimal monotone arithmetic circuit representing f, and let $s = \mathrm{Size}(\mathsf{F})$. By Lemma 20, the polynomial $F = f$ can be written as a sum of at most s products $A \times B$ of polynomials, where the minimum degree $a = d(A)$ of A satisfies $m/3 \leqslant a \leqslant 2m/3$; hence, $d(B) \geqslant m - a$. Observation 2 implies that $|A \times B| \leqslant \#_{d(A)}(f) \cdot \#_{d(B)}(f) \leqslant \#_a(f) \cdot \#_{m-a}(f)$. □

The Rectangle Bound allows one to easily obtain strong lower bounds for some explicit polynomials.

Theorem 23. *If $f \in \{\mathrm{PER}_n, \mathrm{HC}_n, \mathrm{ST}_n\}$, then $\mathrm{Min}(f)$, $\mathrm{Max}(f)$ and $\mathrm{B}_{\mathrm{lin}}(f)$ are $2^{\Omega(n)}$.*

Proof. Since all these three polynomials f are multilinear and homogeneous, it is enough (by Theorem 9) to prove the corresponding lower bounds on $\mathrm{A}[f]$. We will obtain such bounds by applying Lemma 22.

The permanent polynomial $f = \mathrm{PER}_n$ has $|f| = n!$ multilinear monomials $x_{1,\pi(1)}x_{2,\pi(2)} \cdots x_{n,\pi(n)}$, one for each permutation $\pi : [n] \to [n]$. Since at most $(n-r)!$ of the permutations can take r pre-described values, we have that $\#_r(f) \leqslant (n-r)!$. (In fact, here we even have the equality $\#_r(f) = (n-r)!$.) Lemma 22 gives $\mathrm{A}[f] \geqslant n!/(n-r)!r! = \binom{n}{r}$ for some $n/3 < r \leqslant 2n/3$; so, $\mathrm{A}[f] = 2^{\Omega(n)}$.

The argument for HC_n is almost the same: the only difference is that now the monomials correspond to symmetric, not to all permutations.

The spanning tree polynomial $f = \mathrm{ST}_n$ is a homogeneous polynomial of degree $n-1$ with $|f| = n^{n-2}$ monomials $x_{2,\pi(2)}x_{3,\pi(3)}\cdots x_{n,\pi(n)}$ corresponding to the functions $\pi : \{2,3,\ldots,n\} \to [n]$ such that $\forall i\ \exists k$: $\pi^{(k)}(i) = 1$. Each spanning tree gives a function with this property by mapping sons to their father. Now, if we fix some r edges, then r values of functions π whose spanning trees contain these edges are fixed. Thus, $\#_r(f) \leqslant (n-r)^{n-r-2}$, and Lemma 22 gives $\mathrm{A}[f] = 2^{\Omega(n)}$. □

Using a tighter analysis (in the spirit of Remark 8) and more involved computations, Jerrum and Snir [13] obtained even *tight* lower bounds for PER_n and HC_n.

The three polynomials in Theorem 23 are homogeneous. To show that the rectangle bound works also for non-homogeneous polynomials, consider the *st*-connectivity polynomial STCON_n. We know that this polynomial has Min-circuits of size $O(n^3)$ (Remark 3). But Max-circuits for this polynomial must be of exponential size.

Theorem 24. *If* $f = \mathrm{STCON}_{n+2}$*, then* $\mathrm{Max}(f)$ *and* $\mathrm{Min}[f]$ *are at least* $2^{\Omega(n)}$.

Proof. Consider the higher envelope f_{he} of f. This is a homogeneous polynomial of degree n with $|f_{\mathrm{he}}| = n!$ monomials corresponding to paths in K_{n+2} from $s = 0$ to $t = n+1$ with exactly n inner nodes. Since $\#_r(f) \leqslant (n-r)!$, Lemma 22 (with $r = n/3$) gives $\mathrm{A}[f_{\mathrm{he}}] = 2^{\Omega(n)}$. By Theorem 9, the same lower bound holds for $\mathrm{Max}(f)$ and $\mathrm{Min}[f]$. □

14. Truly Exponential Lower Bounds

Note that the lower bounds above have the forms $2^{\Omega(\sqrt{n})}$, where n is the number of variables. Truly exponential lower bounds $\mathrm{A}[f] = \Omega(2^{n/2})$ on the monotone circuit size of multilinear polynomials of n variables were announced by Kasim-Zade [19, 20]. Somewhat earlier, a lower bound $\mathrm{A}[f] = 2^{\Omega(n)}$ was announced by Kuznetsov [23]. Then, Gashkov [9] proposed a general lower bounds argument for monotone arithmetic circuits and used it to prove an $\mathrm{A}[f] = \Omega(2^{2n/3})$ lower bound.

The construction of the corresponding multilinear polynomials in these works is algebraic. Say, the monomials of the polynomial $f(x,y)$ of $2n$ variables constructed in [19, 20] have the form $x_1^{a_1}\cdots x_n^{a_n}y_1^{b_1}\cdots y_n^{b_n}$ where $a \in GF(2)^n$ and $b = a^3$ (we view vector a as an element of $GF(2^n)$ when rising it to the 3rd power). That is, monomials correspond to the points of the cubic parabola $\{(a,a^3)\colon a \in GF(2^n)\}$. The monomials of the polynomial constructed in [9] are defined using triples (a,b,c) with $a,b,c \in GF(2^n)$ satisfying $a^3+b^7+c^{15} = 1$. The constructed polynomials are (k,l)-free for particular *constants* k and l, and the desired lower bounds follow from general lower bounds of Gashkov [9], and Gashkov and Sergeev [10] for (k,l)-free polynomials (see Sect. 12 for these bounds).

Without knowing these results, Raz and Yehudayoff [33] have recently used discrepancy arguments and exponential sum estimates to derive a truly exponential lower bound $\mathrm{A}[f] = 2^{\Omega(n)}$ for an explicit multilinear polynomial $f(x_1,\ldots,x_n)$. Roughly, their construction of f is as follows. Assume that n divided by a particular constant k is a prime number. View a monomial p as a 0/1 vector of its exponents. Split this vector into k blocks of length n/k, view each block as a field element, multiply these elements, and let $c_p \in \{0,1\}$ be the first bit of this product. Then include the monomial p in f if and only if $c_p = 1$.

In this section we use some ideas from [16] to show that truly exponential lower bounds can be also proved using graphs with good expansion properties. Numerically, our bounds (like those in [33]) are worse than the bounds in [19, 20, 9, 10] (have smaller constants), but the construction of polynomials is quite simple (modulo the construction of expander graphs).

Say that a partition $[n] = S \cup T$ is *balanced* if $n/3 \leqslant |S| \leqslant 2n/3$. Define the *matching number* $m(G)$ of a graph $G = ([n], E)$ as the largest number m such that, for every balanced partition of nodes of G, at least m crossing edges form an induced matching. An edge is crossing if it joins a node in one part of the partition with a node in the other part. Being an induced matching means that no two endpoints of any two edges of the matching are joined by a crossing edge.

Our construction of hard polynomials is based on the following lemma. Associate with every graph $G = ([n], E)$ the multilinear polynomial $f_G(x_1, \ldots, x_n)$ whose monomials are $\prod_{i \in S} x_i$ over all subsets $S \subseteq [n]$ such that the induced subgraph $G[S]$ has an odd number of edges of G.

Lemma 25. *For every non-empty graph G on n nodes, we have*

$$\mathrm{A}[f_G] \geqslant 2^{m(G)-2}\,.$$

We postpone the proof of this lemma and turn to its application.

The following simple claim gives us a general lower bound on the matching number $m(G)$. Say that a graph is *s-mixed* if every two disjoint s-element subsets of its nodes are joined by at least one edge.

Claim 26. *If an n-node graph G of maximum degree d is s-mixed, then $m(G) \geqslant (\lfloor n/3 \rfloor - s)/(2d+1)$.*

Proof. Fix an arbitrary balanced partition of the nodes of G into two parts. To construct the desired induced matching, formed by crossing edges, we repeatedly take a crossing edge and remove it together with all its neighbors. At each step we remove at most $2d + 1$ nodes. If the graph is s-mixed, then the procedure will run for m steps as long as $\lfloor n/3 \rfloor - (2d + 1)m$ is at least s. □

Thus, we need graphs of small degree that are still s-mixed for small s. Examples of such graphs are expander graphs. A *Ramanujan graph* is a regular graph $G_{n,q}$ of degree $q + 1$ on n nodes such that $\lambda(G) \leqslant 2\sqrt{q}$, where $\lambda(G)$ is the second largest (in absolute value) eigenvalue of the adjacency matrix of G. Explicit constructions of Ramanujan graphs on n nodes for every prime $q \equiv 1 \bmod 4$ and infinitely many values of n were given by Margulis [25], Lubotzky, Phillips and Sarnak [24]; these were later extended to the case where q is an arbitrary prime power by Morgenstern [29], and Jordan and Livné [14].

Theorem 27. *If $f_G(x_1, \ldots, x_n)$ is the multilinear polynomial associated with the Ramanujan graph $G = G_{n,64}$, then*

$$\mathrm{A}[f_G] \geqslant 2^{0.001n}\,.$$

Proof. The Expander Mixing Lemma ([2, Lemma 2.3]) implies that, if G is a d-regular graph on n nodes, and if $s > \lambda(G) \cdot n/d$, then G is s-mixed. Now, the graph $G = G_{n,q}$ is d-regular with $d = q + 1$ and has $\lambda(G) \leqslant 2\sqrt{q}$. Hence, the graph G is s-mixed for $s = 2n/\sqrt{q} > 2\sqrt{q}n/(q + 1)$.

Our graph $G = G_{n,64}$ is a regular graph of degree $d = 65$, and is s-mixed for $s = 2n/\sqrt{64} = n/4$. Lemma 25 gives the desired lower bound. □

It remains to prove Lemma 25.

Call polynomial $f(x_1,\ldots,x_n)$ a *product polynomial*, if f is a product of two polynomials on disjoint sets of variables, each of size at least $n/3$, that is, if $f = g(Y)\times h(Z)$ for some partition $Y\cup Z=\{x_1,\ldots,x_n\}$ of variables with $|Y|,|Z|\geqslant n/3$, and some two polynomials g and h on these variables. Note that we do not require that, say, the polynomial $g(Y)$ must depend on all variables in Y: some of them may have zero degrees in g.

Claim 28 ([33])**.** *If $\mathsf{F}(x_1,\ldots,x_n)$ is a multilinear circuit of size s with $n\geqslant 3$ input variables, then the polynomial F can be written as a sum of at most $s+1$ product polynomials.*

Proof. Induction on s. For a gate u, let X_u be the set of variables in the corresponding subcircuit of F. Let v be the output gate of F. If v is an input gate, then F itself is a product polynomial, since $n\geqslant 3$. So, assume that v is not an input gate. If $|X_v|\leqslant 2n/3$, then the polynomial F itself is a product polynomial, because $F = F\times 1$. So, assume that $|X_v|>2n/3$. Every gate u in F entered by gates u_1 and u_2 admits $|X_u|\leqslant |X_{u_1}|+|X_{u_2}|$. Thus, there exists a gate u in F such that $n/3\leqslant |X_u|\leqslant 2n/3$. By Lemma 16, we can write F as $F = F_u + F_{u=0}$ where $F_u = g_u\times h$ with $n/3\leqslant |X_u|\leqslant 2n/3$ and some polynomial h. Moreover, since the circuit is multilinear, the set X_h of variables in the polynomial h must be disjoint from X_u, implying that $|X_h|\geqslant n-|X_u|\geqslant n/3$. Thus, $g_u\times h$ is a product polynomial. Since the circuit $F_{u=0}$ has size at most $s-1$, the desired decomposition of F follows from the induction hypothesis. □

By the *characteristic function* of a multilinear polynomial $f(x_1,\ldots,x_n)$ we will mean the (unique) boolean function which accepts a binary vector $a\in\{0,1\}^n$ if and only if the polynomial f contains the monomial $x_1^{a_1}x_2^{a_2}\cdots x_n^{a_n}=\prod_{i:\,a_i=1}x_i$. (Note that this boolean function needs not to be monotone.) In particular, the characteristic function of our polynomial f_G is the quadratic boolean function

$$\phi(x)=\sum_{\{i,j\}\in E} x_ix_j \bmod 2\,.$$

That is, $\phi(a)=1$ if the subgraph $G[S]$ induced by the set of nodes $S=\{i\colon a_i=1\}$ has an odd number of edges. Since $\phi(x)$ is a non-zero polynomial of degree 2 over $GF(2)$, we have that $|f_G|=|\phi^{-1}(1)|\geqslant 2^{n-2}$.

Claim 29. *For every graph G on n nodes, every product sub-polynomial of f_G contains at most $2^{n-m(G)}$ monomials.*

Proof. Let $G\times H$ be a product polynomial contained in f_G. This polynomial gives a partition $x=(y,z)$ of the variables into two parts, each containing at least $n/3$ variables. Let $g(y)$ and $h(z)$ be the characteristic functions of G and H, and $r(x)=g(y)\wedge h(z)$. Then $|G\times H|=|r^{-1}(1)|$, and it is enough to show that $|r^{-1}(1)|\leqslant 2^{n-m(G)}$. When doing this, we will essentially use the fact that $r\leqslant\phi$, which follows from the fact that all monomials of $G\times H$ are also monomials of f_G.

By the definition of $m(G)$, some set $M=\{y_1z_1,\ldots,y_mz_m\}$ of $m=m(G)$ crossing edges y_iz_i forms an induced matching of G. Given an assignment α of constants 0 and 1 to the $n-2m$ variables outside the matching M, define vectors $a,b\in\{0,1\}^m$ and a constant $c\in\{0,1\}$ as follows:

- $a_i=1$ iff an odd number of neighbors of y_i get value 1 under α,
- $b_i=1$ iff an odd number of neighbors of z_i get value 1 under α,
- $c=1$ iff the number of edges whose both endpoints get value 1 under α is odd.

Then the subfunction ϕ_α of ϕ obtained after restriction α is

$$\begin{aligned}\phi_\alpha(y_1,\ldots,y_m,z_1,\ldots,z_m) &= \sum_{i=1}^m y_i z_i + \sum_{i=1}^m y_i a_i + \sum_{i=1}^m b_i z_i + c \mod 2\\ &= IP_m(y\oplus b, z\oplus a)\oplus IP_m(a,b)\oplus c\,,\end{aligned}$$

where $IP_n(y_1,\ldots,y_m,z_1,\ldots,z_m)=\sum_{i=1}^m y_i z_i \mod 2$ is the inner product function (scalar product). Since a,b and c are *fixed*, the corresponding $2^m\times 2^m$ ± 1 matrix H with entries $H[y,z]=(-1)^{\phi_\alpha(y,z)}$ is a Hadamard matrix (rows are orthogonal to each other). Lindsey's Lemma (see, e.g. [17, p. 479]) implies that no monochromatic submatrix of H can have more than 2^m 1-entries.

Now, the obtained subfunction $r_\alpha = g_\alpha(y_1,\ldots,y_m)\wedge h_\alpha(z_1,\ldots,z_m)$ of $r=g(y)\wedge h(z)$ also satisfies $r_\alpha(a,b)\leqslant\phi_\alpha(a,b)$ for all $a,b\in\{0,1\}^m$. Since the set of all pairs (a,b) for which $r_\alpha(a,b)=1$ forms a *submatrix* of H, this implies that r_α can accept at most 2^m such pairs. Since this holds for each of the 2^{n-2m} assignments α, the desired upper bound $|r^{-1}(1)|\leqslant 2^m\cdot 2^{n-2m}=2^{n-m}$ follows.

This completes the proof of Claim 29, and hence, the proof of Lemma 25. □

15. Depth Lower Bounds

So far, we were interested in the *size* of circuits. Another important measure is the circuit *depth*, i.e. the number of nodes in a longest input-output path. For a polynomial f, let Depth$[f]$ denote the smallest possible depth of a circuit producing f.

If a polynomial f can be produced by a circuit of *size* s, what is then the smallest *depth* of a circuit producing f? Hyafil [12] has shown that then f can be also produced by a circuit of depth proportional to $(\log d)(\log sd)$, where d is the maximum degree of f. (This can be easily shown by induction on the degree using the decomposition given in Lemma 20.) However, the size of the resulting circuit may be as large as $s^{\log d}$. A better simulation, leaving the size polynomial in s, was found by Valiant et al. [40].

Theorem 30 (Valiant et al. [40])**.** *If a polynomial f of maximum degree d can be produced by a circuit of size s, then f can be also produced by a circuit of size $O(s^3)$ and depth $O(\log s\log d)$.*

In particular, if a multilinear polynomial f of n variables can be *produced* by a circuit F of polynomial in n size, then Depth$[f]=O(\log^2 n)$. By Lemma 4, Depth$[f]=O(\log^2 n)$ also holds if f is only *computed* by a Max, Min$^-$ or Max$^-$ circuit of polynomial size. This, however, no more holds for B and Min circuits: even though $\hat{F}=\hat{f}$ holds over these semirings, the produced polynomial F may have maximum degree exponential in n.

We now turn to proving lower bounds on Depth$[f]$. In the previous section, we have shown that the factor-density measure $\#_r(f)$ can be used to lower bound the circuit size. By simplifying previous arguments of Shamir and Snir [37], Tiwari and Tompa [39] have shown that the measure $\#_r(f)$ can be also used to lower bound the circuit depth as well. The idea was demonstrated in [39] on two applications (Theorem 33 and 34 below). Here we put their idea in a general frame.

A *subadditive weighting* of a circuit F is an assignment $\mu:\mathsf{F}\to\mathbb{R}_+$ of non-negative weights to the gates of F such that the output gate gets weight $\geqslant 1$, all other gates get weight $\leqslant 1$, and and $\mu(v+w)\leqslant\mu(v)+\mu(w)$ holds for every sum gate $v+w$. Given such a weighting, define the *decrease* K_u at a product gate $u=v\times w$ as

$$K_u=\frac{\mu(v)\cdot\mu(w)}{\mu(u)}\,.$$

Note that, since $\mu(v) \leqslant 1$ holds for every non-output gate v, we have

$$\mu(u) \leqslant \frac{1}{K_u} \cdot \min\{\mu(v), \mu(w)\}\,.$$

That is, when entering u from any of its two inputs, the weight must decrease by a factor of at least K_u. This explains the use of term "decrease". Let $K_{r,s} = \min_u K_u$ be the smallest decrease at a product gate u of degree r, one of whose inputs has degree s; by the *degree* of a gate we mean the minimum degree of the polynomial produced at that gate.

Lemma 31. *Let* F *be a circuit, whose produced polynomial has minimum degree* d*, and let* $m = \log_2 d$*. Then, for every subadditive weighting, there is sequence* $d = r_0 > r_1 > \ldots > r_m = 1$ *of integers such that* $r_{i+1} \geqslant \frac{1}{2} r_i$ *for all* $i = 1, \ldots, m$*, and the circuit* F *has depth at least*

$$m + \log_2 \prod_{i=0}^{m-1} K_{r_i, r_{i+1}}\,.$$

Proof. Construct a path π from the output gate to an input gates as follows: at a sum gate choose the input of greater weight, and at a product gate choose an input of greater degree. Since the produced polynomial has minimum degree d, and since at each product gate we chose an input of greater degree, there must be at least m product gates along π. Let $d = r_1 > r_2 > \ldots > r_m > r_{m+1} = 1$ be the degrees of the product gates (and input node) on path π. Let $k_i = K_{r_i, r_{i+1}}$ be the decrease of the i-th product gate. Note by the construction of π that $r_{i+1} \geqslant \frac{1}{2} r_i$.

Let us now view the path π in the reversed order (from input to output). So, we start with some gate of weight $\leqslant 1$ (an input gate). Since the weighting is subadditive, at each edge entering a sum gate the weight can only increase by a factor of at most 2. So, if s is the number of sum gates along π, then the total increase in weight is by a factor at most 2^s. But when entering the i-th product gate, the weight decreases by a factor at least k_i. Thus, the total loss in the weight is by a factor at least $\prod_{i=0}^{m-1} k_i$. Since the last (output) gate must have weight $\geqslant 1$, this gives

$$2^s \cdot \prod_{i=0}^{m-1} \frac{1}{k_i} \geqslant 1\,, \quad \text{and hence,} \quad s \geqslant \log_2 \prod_{i=0}^{m-1} k_i\,.$$

Since $\mathrm{Depth}[f] \geqslant m + s$, we are done. □

We now give a specific weighting, based on the the factor-density measure $\#_r(f)$. Recall that $\#_r(f)$ is the maximum number of monomials in f containing a fixed monomial of degree r as a common factor. For a polynomial f of minimum degree d, and an integer $1 \leqslant s < r \leqslant d$, define

$$K_f(r, s) = \frac{\#_{d-r}(f)}{\#_{d-s}(f) \cdot \#_{d-r+s}(f)}\,.$$

Note that we have already used this measure to lower-bound the size of circuits: if f is homogeneous of degree d, then Lemma 22 yields $\mathrm{A}[f] \geqslant K_f(d, s)$ for some $d/3 \leqslant s \leqslant 2d/3$.

Lemma 32. *Let* f *be a polynomial of minimum degree* d*, and* $m = \log_2 d$*. Then there is a sequence* $d = r_0 > r_1 > \ldots > r_m = 1$ *of integers such that* $r_{i+1} \geqslant \frac{1}{2} r_i$ *for all* $i = 1, \ldots, m$*, and*

$$\mathrm{Depth}[f] \geqslant m + \log_2 \prod_{i=0}^{m-1} K_f(r_i, r_{i+1})\,.$$

Proof. Let F be a circuit producing f; hence, $F = f$. For a gate $u \in \mathsf{F}$, let d_u be the minimum degree of the polynomial produced at u. By Theorem 16, we know that F can be written as a sum $F = A_u \times B + F_{u=0}$, where A_u is the polynomial produced at gate u. Since $A_u \times B \subseteq f$, and A_u has minimum degree d_u, the polynomial B must contain a monomial p of degree $|p| \geqslant d - d_u$. Hence, by Observation 2, we have that $|A_u| \leqslant \#_{d-d_u}(f)$. This suggests the following weighting of gates:

$$\mu(u) = \frac{|A_u|}{\#_{d-d_u}(f)}\,.$$

The output gate v then gets weight $\mu(v) \geqslant |f|/\#_{d-d}(f) = 1$, whereas all other gates get weights $\leqslant 1$. Moreover, since for every product gate $u = v \times w$, we have that $|A_u| = |A_v| \cdot |A_w|$ and $d_u = d_v + d_w$, the decrease $K_{r,s}$ of this weighting coincides with $K_f(r,s)$. So, it remains to show that the weighting is subadditive.

To show this, let $u = v + w$ be a sum gate. Then $d_u = \min\{d_v, d_w\}$, and hence, $d - d_u = \max\{d - d_v, d - d_w\}$. So,

$$\mu(v+w) = \frac{|A_v| + |A_w|}{\#_{d-d_u}(f)} = \frac{|A_v| + |A_w|}{\max\{\#_{d-d_v}(f), \#_{d-d_w}(f)\}} \leqslant \mu(v) + \mu(w)\,.$$

□

Theorem 33 ([37, 39])**.** *If $f = \mathrm{PER}_n$, then* $\mathrm{Depth}[f] \geqslant n + \log_2 n - 1$.

Proof. The permanent polynomial $f = \mathrm{PER}_n$ is a homogeneous multilinear polynomial of degree $d = n$. Moreover, $\#_l(f) = (n-l)!$ holds for any $1 \leqslant l \leqslant d$. Hence,

$$K_f(r,s) = \frac{r!}{s!(r-s)!} = \binom{r}{s}\,.$$

But $r_{i+1} \geqslant \frac{1}{2} r_i$ implies that $\binom{r_i}{r_{i+1}} \geqslant 2^{r_i - r_{i+1}}$. Hence,

$$\prod_{i=0}^{m-1} K_f(r_i, r_{i+1}) = \prod_{i=0}^{m-1} \binom{r_i}{r_{i+1}} \geqslant 2^{r_0 - r_m} = 2^{n-1}\,.$$

□

This lower bound for $f = \mathrm{PER}$ is not surprising, since $\mathrm{Depth}[f]$ is always at least logarithmic in $\mathrm{A}[f]$, and we already know (Theorem 23) that $\mathrm{A}[f]$ is exponential for this polynomial. More interesting, however, is that the argument above allows to prove super-logarithmic depth lower bounds even for polynomials that *have* circuits of polynomial size.

To demonstrate this, consider the following *layered st-connectivity polynomial* $f_{n,d}$. The monomials of this polynomial correspond to st-paths in a layered graph. We have $d+1$ disjoint layers, where the first contains only one node s, the last only one node t, and each of the remaining $d-1$ layers contains n nodes. Monomials of $f_{n,d}$ have the form $x_{s,a_1} x_{a_1,a_2} \cdots x_{a_{d-2},a_{d-1}} x_{a_{d-1},t}$ with a_i belonging to the i-th layer. In other words, this polynomial corresponds to computing the (s,t)-entry of the product of $d-1$ matrices of dimension $n \times n$. Hence, it can be produced by a circuit of depth $O((\log d)(\log n))$.

Theorem 34 ([37, 39])**.** $\mathrm{Depth}[f_{n,d}] \geqslant (\log_2 d)(1 + \log_2 n)$.

Bound	**Property of f**	**Ref.**
$\mathrm{B}(f) > t$	f is not t-simple (Def. 2)	Thm. 8
$\mathrm{S}(f) = \mathrm{A}[f]$	f is homogeneous	Thm. 9
$\mathrm{A}[f] \geqslant \lvert f\rvert$	f is separated (Def. 3)	Thm. 12
$\mathrm{A}[f] \geqslant \dfrac{\lvert f\rvert}{2kl^2}$	$A \times B \subseteq f$ implies $\lvert A\rvert \leqslant l$ or $\lvert B\rvert \leqslant k$	Thm. 18
$\mathrm{A}[f] \geqslant \dfrac{\lvert f\rvert}{\#_r(f) \cdot \#_{d-r}(f)}$	f of minimum degree d	Lem. 22

Table 2: A summary of general lower bounds. Here S is an arbitrary tropical semiring, $\#_r(f)$ is the maximum possible number of monomials of f containing a fixed monomial of degree r, and r is some integer $m/3 \leqslant r \leqslant 2m/3$.

Proof. The polynomial $f = f_{n,d}$ is a multilinear homogeneous polynomial of degree d with $|f| = n^{d-1}$ monomials. To estimate the factor-density $\#_l(f)$, let us fix a set E of $|E| = l$ edges. Every edge $e \in E$ constrains either two inner nodes (if $s, t \notin e$) or one inner node. Thus, if we fix l edges, then at least l inner nodes are constrained, implying that only $\#_l(f) \leqslant n^{d-1-l}$ paths can contain all these edges. In fact, we have an equality $\#_l(f) = n^{d-1-l}$: every monomial $x_{s,a_1}x_{a_1,a_2}\cdots x_{a_{l-1},a_l}$ consisting of initial l edges is a factor of exactly n^{d-1-l} monomials of f. Thus, the decrease in this case is

$$K_f(r,s) = \frac{\#_{d-r}(f)}{\#_{d-s}(f) \cdot \#_{d-(r-s)}(f)} = \frac{n^{r-1}}{n^{s-1} \cdot n^{r-s-1}} = n$$

for all $1 \leqslant s < r \leqslant d$. Lemma 32 yields $\mathrm{Depth}[f] \geqslant \log_2 d + \log_2 n^{\log_2 d}$, as desired. □

16. Conclusion and Open Problems

In this paper we summarized known and presented some new lower-bound arguments for tropical circuits, and hence, for the dynamic programming paradigm; Table 2 gives a short overview. We have also shown that these bounds already yield strong (even exponential) lower bounds for a full row of important polynomials (see Table 1). Still, the known arguments seem to fail for non-homogeneous polynomials like CONN or STCON.

Almost exact lower bounds on the *depth* circuits computing these polynomials are known even in the boolean semiring: $\Theta(\log^2 n)$ for STCON_n proved by Karchmer and Wigderson [18], and $\Omega(\ln^2 n/\ln\ln)$ proved by Goldmann and Håstad [11] for CONN_n; Yao [42] earlier proved $\Omega(\ln^{3/2} n/\ln\ln)$ for this latter polynomial. By Lemma 7, these bounds hold also in tropical semirings.

But the situation with estimating the *size* of circuit for these polynomial remains unclear. We know (Theorem 6) that both of them have boolean and Min-circuits of size $O(n^3)$, but no lower bound larger than a trivial quadratic is known.

Open Problem 1. *Does* $\mathrm{B}(f) = \Omega(n^3)$ *or at least* $\mathrm{Min}(f) = \Omega(n^3)$ *hold for* $f = \mathrm{STCON}_n$ *and/or* $f = \mathrm{CONN}_n$*?*

Note that the lower bound $\Omega(n^3)$ for the all-pairs shortest paths polynomial APSP, given in Corollary 15 does not automatically imply the same lower bounds for the connectivity polynomial CONN: a circuit for CONN needs *not* to compute the polynomials of APSP at *separate* gates.

One could show $\mathrm{Min}(\mathrm{CONN}) = \Omega(n^3)$ by showing that monotone arithmetic circuits for the following "multiplicative version" of the triangle polynomial TR_n require $\Omega(n^3)$ gates. Recall that $\mathrm{TR}_n(x,y,z) = \sum_{i,j\in[n]} z_{ij} \sum_{k\in[n]} x_{ik}y_{kj}$. We already know (see Corollary 15) that $\mathrm{A}[\mathrm{TR}_n] = \Theta(n^3)$, and hence also $\mathrm{Min}(\mathrm{TR}_n) = \Theta(n^3)$ since the polynomial is homogeneous. Replace now the outer sum by product, and consider the polynomial $\mathrm{TR}^*_n = \prod_{i,j\in[n]} z_{ij} \sum_{k\in[n]} x_{ik}y_{kj}$.

Open Problem 2. *Does* $\mathrm{A}[\mathrm{TR}^*_n] = \Omega(n^3)$*?*

If true, this would yield $\mathrm{Min}(\mathrm{CONN}_n) = \Omega(n^3)$, because the polynomial TR^*_n is homogeneous (of degree $3n^2$).

Acknowledgements

I am thankful to Georg Schnitger and Igor Sergeev for interesting discussions.

References

[1] N. Alon and R. Boppana. The monotone circuit complexity of boolean functions. *Combinatorica*, 7(1):1–22, 1987.

[2] N. Alon and Fan R.K. Chung. Explicit constructions of linear sized tolerant networks. *Discrete Math.*, 72:15–19, 1989.

[3] A.E. Andreev. On a method for obtaining lower bounds for the complexity of individual monotone functions. *Soviet Math. Dokl.*, 31(3):530–534, 1985.

[4] A.E. Andreev. A method for obtaining efficient lower bounds for monotone complexity. *Algebra and Logics*, 26(1):1–18, 1987.

[5] W. Baur and V. Strassen. The complexity of partial derivatives. *Theoret. Comput. Sci.*, 22:317–330, 1983.

[6] R. Bellman. On a routing problem. *Quarterly of Appl. Math.*, 16:87–90, 1958.

[7] R.W. Floyd. Algorithm 97, shortest path. *Comm. ACM*, 5:345, 1962.

[8] L.R. Ford. Network flow theory. Technical Report P-923, The Rand Corp., 1956.

[9] S.B. Gashkov. On one method of obtaining lower bounds on the monotone complexity of polynomials. *Vestnik MGU, Series 1 Mathematics, Mechanics*, 5:7–13, 1987.

[10] S.B. Gashkov and I.S. Sergeev. A method for deriving lower bounds for the complexity of monotone arithmetic circuits computing real polynomials. *Math. Sbornik*, 203(10):33–70, 2012.

[11] M. Goldmann and J. Håstad. Monotone circuits for connectivity have depth log n to the power (2-o(1)). *SIAM J. Comput.*, 27:1283–1294, 1998.

[12] L. Hyafil. On the parallel evaluation of multivariate polynomials. *SIAM J. Comput.*, 8(2):120–123, 1979.

[13] M. Jerrum and M. Snir. Some exact complexity results for straight-line computations over semirings. *J. ACM*, 29(3):874–897, 1982.

[14] J.W. Jordan and R. Livné. Ramanujan local systems on graphs. *Topology*, 36(5):1007–1–24, 1997.

[15] S. Jukna. Combinatorics of monotone computations. *Combinatorica*, 9(1):1–21, 1999. Preliminary version: ECCC Report Nr. 26, 1996.

[16] S. Jukna. Expanders and time-restricted branching programs. *Theoret. Comput. Sci.*, 409(3):471–476, 2008.

[17] S. Jukna. *Boolean Function Complexity: Advances and Frontiers.* Springer-Verlag, 2012.

[18] M. Karchmer and A. Wigderson. Monotone circuits for connectivity require super-logarithmic depth. *SIAM J. Discrete Math.*, 3:255–265, 1990.

[19] O.M. Kasim-Zade. On arithmetical complexity of monotone polynomials. In *Proc. of All-Union Conf. on Theoretical Problems in Kybernetics*, volume 1, pages 68–69, 1986. (in Russian).

[20] O.M. Kasim-Zade. On the complexity of monotone polynomials. In *Proc. of All-Union Seminar on Discrete Math. and its Appl.*, pages 136–138, 1986. (in Russian).

[21] L.R. Kerr. *The effect of algebraic structure on the computation complexity of matrix multiplications.* PhD thesis, Cornell Univ., Ithaca, N.Y., 1970.

[22] M.P. Krieger. On the incompressibility of monotone DNFs. *Theory of Comput. Syst.*, 41(2):211–231, 2007.

[23] S.E. Kuznetsov. Monotone computations of polynomials and schemes without null-chains. In *Proc. of 8-th All-Union Conf. on Theoretical Problems in Cybernetics*, volume 1, pages 108–109, 1985. (in Russian).

[24] A. Lubotzky, R. Phillips, and P. Sarnak. Ramanujan graphs. *Combinatorica*, 8(3):261–277, 1988.

[25] G.A. Margulis. Explicit constructions of concentrators. *Problems of Inf. Transm.*, pages 323–332, 1975.
[26] K. Mehlhorn. Some remarks on Boolean sums. *Acta Informatica*, 12:371–375, 1979.
[27] K. Mehlhorn and Z. Galil. Monotone switching circuits and boolean matrix product. *Computing*, 16(1-2):99–111, 1976.
[28] E.F. Moore. The shortest path through a maze. In *Proc. Internat. Sympos. Switching Theory*, volume II, pages 285–292. Harvard Univ. Press 1959, 1957.
[29] M. Morgenstern. Existence and explicit constructions of $q+1$ regular Ramanujan graphs for every prime power q. *J. Comb. Theory Ser. B*, 62(1):44–62, 1994.
[30] E.I. Nechiporuk. On the topological principles of self-correction. *Problemy Kibernetiki*, 21:5–102, 1969. English translation in: Systems Theory Res. 21 (1970), 1–99.
[31] M. Paterson. Complexity of monotone networks for boolean matrix product. *Theoret. Comput. Sci.*, 1(1):13–20, 1975.
[32] N. Pippenger. On another Boolean matrix. *Theor. Comput. Sci.*, 11:49–56, 1980.
[33] R. Raz and A. Yehudayoff. Multilinear formulas, maximal-partition discrepancy and mixed-sources extractors. *J. Comput. Syst. Sci.*, 77(1):167–190, 2011. Preliminary version in: Proc. of 49th FOCS, 2008.
[34] A.A. Razborov. A lower bound on the monotone network complexity of the logical permanent. *Math. Notes Acad. of Sci. USSR*, 37(6):485–493, 1985.
[35] A.A. Razborov. Lower bounds for the monotone complexity of some boolean functions. *Soviet Math. Dokl.*, 31:354–357, 1985.
[36] C.P. Schnorr. A lower bound on the number of additions in monotone computations. *Theor. Comput. Sci.*, 2(3):305–315, 1976.
[37] E. Shamir and M. Snir. On the depth complexity of formulas. *Math. Syst. Theory*, 13:301–322, 1980.
[38] A. Shpilka and A. Yehudayoff. Arithmetic circuits: A survey of recent results and open questions. *Foundations and Trends in Theoret. Comput. Sci.*, 5(3-4):207–388, 2009.
[39] P. Tiwari and M. Tompa. A direct version of Shamir and Snir's lower bounds on monotone circuit depth. *Inf. Process. Lett.*, 49(5):243–248, 1994.
[40] L.G. Valiant, S. Skyum, S. Berkowitz, and C. Rackoff. Fast parallel computation of polynomials using few processors. *SIAM J. Comput.*, 12(4):641–644, 1983.
[41] S. Warshall. A theorem on boolean matrices. *J. ACM*, 9:11–12, 1962.
[42] A.C. Yao. A lower bound for the monotone depth of connectivity. In *Proc. of 35th Ann. Symp. on Foundations of Comput. Sci.*, pages 302–308, 1994.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de