Improved Bounds for Reduction
to Depth 4 and Depth 3

Sébastien Tavenas

LIP*, École Normale Supérieure de Lyon
sebastien.tavenas@ens-lyon.fr

Abstract. Koiran [7] showed that if an n-variate polynomial of degree d (with $d = n^{O(1)}$) is computed by a circuit of size s, then it is also computed by a homogeneous circuit of depth four and of size $2^{O(\sqrt{d} \log(d) \log(s))}$. Using this result, Gupta, Kamath, Kayal and Saptharishi [6] gave an $\exp\left(O\left(\sqrt{d \log(d) \log(n) \log(s)}\right)\right)$ upper bound for the size of the smallest depth three circuit computing an n-variate polynomial of degree $d = n^{O(1)}$ given by a circuit of size s.

We improve here Koiran's bound. Indeed, we show that if we reduce an arithmetic circuit to depth four, then the size becomes $\exp\left(O\left(\sqrt{d \log(ds) \log(n)}\right)\right)$. Mimicking the proof in [6], it also implies the same upper bound for depth three circuits.

This new bound is not far from optimal in the sense that Gupta, Kamath, Kayal and Saptharishi [5] also showed a $2^{\Omega(\sqrt{d})}$ lower bound for the size of homogeneous depth four circuits such that gates at the bottom have fan-in at most $\sqrt{d}$. Finally, we show that this last lower bound also holds if the fan-in is at least $\sqrt{d}$.

1 Introduction

Agrawal and Vinay proved [1] that if an n-variate polynomial f of degree $d = O(n)$ has a circuit of size $2^{o(d+d\log(\frac{n}{d}))}$, then f can also be computed by a depth-four circuit ($\sum \prod \sum \prod$) of size $2^{o(d+d\log(\frac{n}{d}))}$. This result shows that for proving arithmetic circuit lower bounds or black-box derandomization of identity testing, the case of depth four arithmetic circuit is the general case in a certain sense. This result arose after other ones on parallelization. Valiant, Skyum, Berkowitz and Rackoff [9] proved that if a size-s depth-d circuit computes a polynomial of degree d, then this polynomial can also be computed by a circuit of depth $O(\log(d) \log(s))$ and of size bounded by a polynomial in s. Some years later, Allender, Jiao, Mahajan and Vinay [2] showed that this parallelization could be done uniformly. Their method for parallelization is reused in [1] and will be the basis for the parallelization in this paper.

Agrawal and Vinay's result only deals with polynomials of sub-exponential complexity. But if the hypothesis is strengthened, it is possible to get a stronger

---

conclusion. Indeed, Koiran [7] showed that if the circuit at the beginning is of size s, then it can be computed by a homogeneous depth-four circuit of size $2^{O(\sqrt{d}\log(d)\log(s))}$. For example, if the permanent family is computed by a polynomial size circuit (i.e., of size n^c), then it is computed by a depth-four circuit of size $2^{O(\sqrt{n}\log^2(n))}$. These results appear as an interesting approach to lower bounds: if one finds a $2^{\omega(\sqrt{n}\log^2(n))}$ lower bound on the size of depth-4 circuits computing the permanent, then it will imply that there are no polynomial size circuits for the permanent. The interest of this approach is confirmed by Gupta, Kamath, Kayal and Saptharishi's recent result [5]. They showed that if a homogeneous $\sum\prod\sum\prod$ circuit where the bottom fan-in is bounded by t computes the permanent of a matrix of size $n \times n$, then its size is $2^{\Omega(\frac{n}{t})}$. In a recent paper [6], the same authors improve the upper bound by transforming n-variate circuits of size s and depth d (with $d = n^{O(1)}$) into depth-3 circuits of size $\exp\left(O(\sqrt{d\log s\log n\log d})\right)$, moreover if the input is a branching program (and not a circuit), the upper bound becomes $\exp\left(O(\sqrt{d\log s\log n})\right)$. In particular, this result gives a depth-3 circuit of size $2^{O(\sqrt{n}\log n)}$ computing the determinant of a matrix $n \times n$. Nevertheless, the depth-3 circuit they get is not homogeneous, and uses intermediate gates which compute polynomials of very high degree.

   In this paper we improve Koiran's bound. We show that a circuit of size s can be parallelized homogeneously in depth 4 and in size $\exp\left(O\left(\sqrt{d\log(ds)\log(n)}\right)\right)$ such that the fan-in of each multiplication gate is bounded by $O\left(\sqrt{d\frac{\log ds}{\log n}}\right)$. We can notice that as $n \leq s$, the result implies Koiran's bound and is generally better (in the case where $d, s = n^{\Theta(1)}$, Koiran's bound is $2^{O(\sqrt{n}\log^2 n)}$ while the new bound is $2^{O(\sqrt{n}\log n)}$). It implies that a $2^{\omega(\sqrt{n}\log(n))}$ lower bound for depth-4 circuits computing the permanent gives a super-polynomial lower bound for general circuits computing the permanent. Moreover, using this result in Gupta, Kamath, Kayal and Saptharishi's proof instead of Koiran's result slightly improves the depth-3 upper bound. An n-variate circuit of size s and depth d is computed by a depth-3 circuit of size $\exp\left(O(\sqrt{d\log(ds)\log n})\right)$. So, we get the same bound for the reduction at depth 3 starting from an arithmetic circuit as from an arithmetic branching program. Finally in Section 6, we show, by a counting argument, that if a homogeneous $\sum\prod\sum\prod$ circuit where the bottom fan-in is lower-bounded by t computes the permanent (or the determinant) of a matrix of size $n \times n$, then its size is $2^{\Omega(t\log n)}$.

2   Arithmetic Circuits

We give here a brief introduction to arithmetic circuits theory. The reader can find more detailed information in [10,3,8,4]. In this theory, we measure the complexity of polynomial functions using arithmetic circuits.

Definition 1. *An arithmetic circuit is a finite acyclic directed graph with vertices of in-degree 0 or more and exactly one vertex of out-degree 0. Vertices of*

in-degree 0 *are called inputs and labeled by a constant or a variable. The other vertices are labeled by* × *or* + *(or sometimes by* ⊙ *in this paper) and called computation gates (the in-degree of these gates will be also called the fan-in). The vertex of out-degree* 0 *is called the output. The vertices of a circuit are commonly called gates and its edges arrows. Finally, we call a formula, an arithmetic circuit such that the underlying graph is a tree.*

Each gate of a circuit computes a polynomial (defined by induction). The polynomial computed by a circuit corresponds to the polynomial computed by the output of this circuit. For a gate α, we denote $[\alpha]$ the polynomial computed by this gate. A circuit is called *homogeneous* is all its gates compute homogeneous polynomials. In fact, for some proofs, we will use circuits with several outputs (each one corresponds to an out-degree 0 gate). A ⊙-gate corresponds to a multiplication-by-a-scalar gate. The fan-in of such a gate will be always 2 and at least one of its inputs corresponds to a constant (We will give a syntactic restriction just after the next definition).

Definition 2. *The size of a circuit is its number of gates. The depth is the maximal length of a directed path from an input to an output. The degree of a gate is defined recursively: any variable input is of degree 1, constant inputs are of degree 0, the degree of a* + *or* ⊙*-gate is the maximum of the incoming degrees and the degree of a* ×*-gate is the sum of the incoming degrees.*

We can now put a restriction for the ⊙-gates. For each one of these gates, one of its child has to be of degree 0.

For a given circuit we will consider graphs called parse trees. A parse tree corresponds, in the spirit, to the computation of one particular monomial.

Definition 3. *The set of parse trees of a circuit C is defined by induction on its size:*

- *If C is of size 1 it has only one parse tree, itself.*
- *If the output gate of C is a +-gate whose arguments are the gates $\alpha_1, \ldots, \alpha_k$, then the parse trees of C are obtained by taking, for an arbitrary $i \leq k$, a parse tree of the sub-circuit rooted in α_i and the arrow from α_i to the output.*
- *If the output gate of C is a ×-gate or an ⊙-gate whose arguments are the gates $\alpha_1, \ldots, \alpha_k$, the parse trees of C are obtained by taking disjoint copies of parse tree of the sub-circuits rooted in α_i for all $i \leq k$ and the arrows from all α_i to the output.*

The polynomial computed by a circuit C becomes the sum of the monomials computed by the parse trees of C.

We will use some convenient notations which are defined in [6]. A depth-4 circuit such that gates are multiplication gates at level one and three and addition gates at levels two and four are denoted $\sum \prod \sum \prod$ circuits. Furthermore, a $\sum \prod^{[\alpha]} \sum \prod^{[\beta]}$ circuit is a $\sum \prod \sum \prod$ circuit such that the fan-in of the multiplication gates at level 3 is bounded by α, and the fan-in of the multiplication

gates at level 1 is bounded by β. For example, a $\sum \prod^{[\alpha]} \sum \prod^{[\beta]}$ circuit computes a polynomial of the form:

$$\sum_{i=1}^{t} \prod_{j=1}^{a_i} \sum_{k=1}^{u_{i,j}} \prod_{l=1}^{b_{i,j,k}} x_{i,j,k,l}$$

where $a_i \leq \alpha$, $b_{i,j,k} \leq \beta$.

3   Upper Bounds

Here, we state the main theorem in this paper.

Theorem 1. *Let f be an n-variate polynomial computed by a circuit of size s and of degree d. Then f is computed by a $\sum \prod \sum \prod$ circuit C of size $2^{O\left(\sqrt{d \log(ds) \log n}\right)}$. Furthermore, if f is homogeneous, it will be also the case for C.*

The previous theorem can be directly applied for the permanent.

Theorem 2. *If the $n \times n$ permanent is computed by a circuit of size polynomial in n, then it is also computed by a $\sum \prod \sum \prod$ circuit of size $2^{O(\sqrt{n} \log(n))}$.*

In their paper [6], Gupta, Kamath, Kayal and Saptharishi used the previous $2^{\sqrt{d} \log^2(s)}$ bound [7] for parallelizing at depth 3. They showed that:

Proposition 1 (Theorem 1.1 in [6]). *Let $f(x) \in \mathbb{Q}[x_1, \ldots, x_n]$ be an n-variate polynomial of degree $d = n^{O(1)}$ computed by an arithmetic circuit of size s. Then it can also be computed by a $\sum \prod \sum$ circuit of size $2^{O(\sqrt{d \log n \log s \log d})}$.*

In fact, their proof is divided into three parts. First they transform circuits into depth-4 circuits, then they transform depth-4 circuits into depth-5 circuits using only sum and exponentiation gates. And finally they transform these last circuits into depth-3 circuits. Using Theorem 1 instead of Theorem 4.1 in their paper improves the first part of their proof. That implies a small improvement of Theorem 1.1 in [6]:

Corollary 1. *Let $f(x) \in \mathbb{Q}[x_1, \ldots, x_n]$ be an n-variate polynomial of degree $d = n^{O(1)}$ computed by an arithmetic circuit of size s. Then it can also be computed by a $\sum \prod \sum$ circuit of size $2^{O(\sqrt{d \log n \log s})}$.*

4   Useful Propositions

For proving Theorem 1, we will need the following propositions.
    The next result is folklore. A proof can be found in [2].

Proposition 2. *If f is a degree-d polynomial computed by a $\{+, \times\}$-circuit C of size s such that the fan-in of each $+$-gate is unbounded and the fan-in of each $\times$-gate is bounded by 2, then there exists a circuit $\tilde{C}$ of size $s(d+1)^2$ with $d+1$ outputs $O_0, O_1, \ldots, O_d$ such that:*

- *the fan-in of each $+$-gate is unbounded,*
- *the fan-in of each $\times$-gate is bounded by 2,*
- *for each i, the gate O_i computes the homogeneous part of f of degree i,*
- *$\tilde{C}$ is homogeneous,*
- *the degree of each gate of $\tilde{C}$ equals the degree of the polynomial computed by this gate.*

We define $\times$-*balanced* $\{\times, +, \odot\}$-circuits.

Definition 4. *A $\{\times, +, \odot\}$-circuit C is called $\times$-balanced if and only if all the following properties are verified:*

- *the fan-in of each $\times$-gate is at most 5,*
- *the fan-in of each $+$-gate is unbounded,*
- *the fan-in of each $\odot$-gate is at most 2,*
- *for each $\times$-gate α, each one of its arguments is of degree at most half of the degree of α.*

The last condition can not be true for the multiplication by a scalar. It is the reason, we introduced the operator $\odot$.

The next proposition which is implicitly a first result of parallelization is almost the same result that we can find in Section 2 in [1] or in Theorem 2.7 in [8]. We give a proof in appendix.

Proposition 3. *Let f be a homogeneous degree-d polynomial computed by a size-s circuit $\tilde{C}$ defined as in the conclusion of Proposition 2. Then f is computed by a homogeneous $\times$-balanced $\{\times, +, \odot\}$-circuit of size $s^6 + s^4 + 1$ and of degree d.*

Agrawal and Vinay already noticed that Valiant, Skyum, Berkowitz and Rackoff's famous result [9] is a direct corollary of this proposition.

Corollary 2. *Let f be a polynomial of degree d computed by a circuit of size s. Then f is computed by a $\{+, \times\}$-circuit of size $(sd)^{O(1)}$ and of depth $O(\log(s)\log(d))$ where each $+$ and $\times$-gate is of fan-in 2.*

5   Proof of Theorem 1

For realizing the reduction to depth four, Koiran begins by transforming the circuit into an equivalent arithmetic branching program. Then, he parallelizes the branching program, and finally comes back to the circuits. The problem with this strategy is that the transformation from circuits to branching programs requires an increase in the size of our object. If the circuit is of size s, our new

branching program is of size $s^{\log(d)}$. Here, the approach is to directly parallelize the circuit without using arithmetic branching programs in intermediate steps.

The idea is to split the circuit into two parts: gates of degree lower than $\sqrt{d}$ and gates of larger degree. Furthermore, a circuit such that the degree of each gate is bounded by $\sqrt{d}$ computes a degree-$\sqrt{d}$ polynomial and so can be written as a sum of at most $s^{O(\sqrt{d})}$ monomials. Then, if each part of our circuit computes polynomials of degrees bounded by $\sqrt{d}$, we just have to get the two depth-2 circuits and connect them together. The main difficulty comes from the fact it is not always true that the sub-circuit obtained by the gates of degree larger than $\sqrt{d}$ is of degree smaller than $\sqrt{d}$. For example, for the comb graph with $n-1$ ×-gates and n variable inputs:

$$x_1 \cdot (x_2 \cdot (x_3 \cdot (\ldots)))$$

the degree of the first part is $\sqrt{n}$, but the degree of the second one is $n - \sqrt{n}$.

In fact, following ideas from [6], we are going to cut not exactly at level $\sqrt{d}$. It will give a sharper result.

Lemma 1. *Let f be a homogeneous n-variate polynomial of degree d computed by a homogeneous ×-balanced $\{\times, +, \odot\}$-circuit C of size σ. Then f is computed by a homogeneous $\sum \prod^{[15a]} \sum \prod^{\left[\frac{d}{a}\right]}$ circuit of size $1 + \binom{\sigma + 15a}{15a} + \sigma + \sigma \binom{n + \frac{d}{a}}{\frac{d}{a}} + n$ for any positive constant a smaller than d.*

To get nicer expressions, we will use the following consequence of Stirling's formula: (A proof appears in [1])

Lemma 2.

$$\binom{k+l}{l} = 2^{O\left(l + l \log \frac{k}{l}\right)}$$

First, let us see how Lemma 1 implies Theorem 1.

Proof (Proof of Theorem 1). Let f be an n-variate polynomial computing by a circuit of size s and degree d. Let $\tilde{C}$ be the homogeneous circuit for the polynomial that we get by Proposition 2. The circuit $\tilde{C}$ is of size $t = s(d+1)^2$ and computes all polynomials $f_0, \ldots, f_d$ where f_i is the homogeneous part of f of degree i. Then for each $i \leq d$, there exists a homogeneous ×-balanced circuit C of size $\sigma = t^6 + t^4 + 1$ computing f_i. We apply Lemma 1 for the circuit C with $a = \sqrt{d \frac{\log n}{\log \sigma}}$. Using Lemma 2 we get a homogeneous $\sum \prod \sum \prod$ circuit of size $1 + \binom{\sigma + 15a}{15a} + \sigma + \sigma \binom{n + \frac{d}{a}}{\frac{d}{a}} + n = 2^{O\left(\sqrt{d \log \sigma \log n}\right)}$. At the end, we just have to add together homogeneous parts f_i. As $\sigma = O(s^6 d^{12})$, it gives a $2^{O\left(\sqrt{d \log(ds) \log n}\right)}$ upper bound for the size.

Remark 1. Choosing the easier assignment $a = \sqrt{d}$ gives a $2^{O\left(\sqrt{d} \log(ds)\right)}$ upper bound.

Proving Lemma 1 will complete the proof.

Proof (Proof of Lemma 1). We define circuits C_1 and C_2 as follows. C_1 is the circuit we get by keeping only gates of C of degree $< \frac{d}{a}$. Circuit C_2 is made up of the remaining gates (i.e., those of degree $\geq \frac{d}{a}$) and of the inputs of these gates. These inputs are the only gates which belong both in C_1 and in C_2.

Each gate α of C_1 has degree at most $\frac{d}{a}$, so computes a polynomial of degree at most $\frac{d}{a}$. By homogeneity of C, the polynomial computed in α is homogeneous. Consequently, α is a homogeneous sum of at most $\binom{n+\frac{d}{a}}{\frac{d}{a}}$ monomials, and so, can be computed by a homogeneous depth-2 circuit of size $1 + \binom{n+\frac{d}{a}}{\frac{d}{a}} + n$ (The "1" encodes the +-gate, the "n" encodes the input gates, and the remainder encodes the ×-gates).

We are going to show now that the degree of C_2 is bounded by $15a$.

Let δ be the degree of C_2. There exists a degree-δ monomial m in C_2. Let T be a parse tree computing m.

We partition the set of ×-gates of T into 3 sets:

- $\mathcal{G}_0 = \{\alpha \in T | \alpha$ is a ×-gate and all children of α are leaves of $T\}$
- $\mathcal{G}_1 = \{\alpha \in T | \alpha$ is a ×-gate and exactly one child of α is not a leaf$\}$
- $\mathcal{G}_2 = \{\alpha \in T | \alpha$ is a ×-gate and at least two children of α are not leaves$\}$.

Then, if we consider the sub-tree S of T with only gates of C_2, then $\mathcal{G}_0$ are leaves of S, $\mathcal{G}_1$ are internal vertices of fan-in 1 and $\mathcal{G}_2$ are internal vertices of fan-in at least 2.

The proof is in two parts. First we upperbound the size of the sets $\mathcal{G}_0$, $\mathcal{G}_1$ and $\mathcal{G}_2$. Then, we upperbound the degree of m.

In C, the degree of "m" is at least the sum of the degrees of the gates of $\mathcal{G}_0$ (since two of these gates can not appear on the same path). Each one of these gates is in C_2, so is of degree at least $\frac{d}{a}$ in C. As m is of degree at most d in C, it means that the number of gates in $\mathcal{G}_0$ is at most a.

In C, the degree of "m" is at least the sum of the degrees of the leaves directly connected to a gate of $\mathcal{G}_1$. For each gate α of $\mathcal{G}_1$, exactly one of its inputs β is in C_2, hence of degree at least $\frac{d}{a}$ in C. By Proposition 3, the degree of α is at least two times the degree of β, it yields that the sum of degrees of inputs of α which are in C_1 is also at least $\frac{d}{a}$. Then, the number of vertices in $\mathcal{G}_1$ is at most a.

Finally, in a tree, the number of leaves is larger than the number of vertices of fan-in at least 2. Then in S, we get that:

$$|\mathcal{G}_2| \leq |\mathcal{G}_0| \leq a.$$

In C_2, the degree of the monomial m is the number of leaves labelled by a non-constant leaf in T. We match each leaf with the first ×-gate which is connected to it. As in T, the fan-in of the ×-gates is bounded by 5, the fan-in of the +-gates is bounded by 1 and each ⊙-gates add only one constant input, then the number of variable leaves connected to a particular ×-gate is at most 5. So the number of leaves in T is at most:

$$5 \times (|\mathcal{G}_0| + |\mathcal{G}_1| + |\mathcal{G}_2|) \leq 15a.$$

This proves that the degree of C_2 is at most $15a$. Then, the number of inputs of C_2 is bounded by the number of gates in C_1 and so in C (which is σ). So, there exists a depth-2 circuit which compute C_2, of size $1 + \binom{\sigma+15a}{15a} + \sigma$ with as inputs the gates of C_1.

Consequently, each polynomial f_i can be computed by a homogeneous $\sum \prod^{[a]} \sum \prod^{\left[\frac{d}{a}\right]}$ circuit of size at most $1 + \binom{\sigma+15a}{15a} + \sigma + \sigma \binom{n+\frac{d}{a}}{\frac{d}{a}} + n$.

6   A Lower Bound

In [5], it was proved that if a homogeneous depth-four circuit computing PERM_n has its bottom fan-in bounded by t, then the size of the circuit is at least $2^{\Omega\left(\frac{n}{t}\right)}$. But what happens if bottom multiplication gates all have a large fan-in? We show that this implies a similar lower bound for the size of the circuit:

Theorem 3. *If C is a homogeneous $\sum \prod \sum \prod$ circuit which computes PERM_n (or DET_n) such that the fan-in of each bottom multiplication gate is at least t, then the size of C is at least $2^{\Omega(t \log(n))}$.*

Our approach is only based on counting the number of monomials. We begin by some definitions.

Definition 5. *For a multivariate polynomial $f(\mathbf{x}) = \sum_{i=1}^{m_f} a_i \mathbf{x_i}$, we will denote $\mathcal{M}_f$ the set $\{\, \mathbf{x_i} \mid \mathbf{x_i} \text{ is a monomial of } f \,\}$. If E is a set of polynomials, we also define $\mathcal{M}_E = \bigcup_{f \in E} \mathcal{M}_f$.*

We can notice $\mathcal{M}_{\mathrm{PERM}_n} = \{\, x_{1,\sigma(1)} \ldots x_{n,\sigma(n)} \mid \sigma \in \mathfrak{S}_n \,\}$. So, $|\mathcal{M}_{\mathrm{PERM}_n}| = n!$.

Definition 6. *Let E be a set of polynomials. Let us denote*

$$E^+ = \{\, f_1 + \ldots + f_m \mid m \in \mathbb{N} \text{ and } \forall i \le m, \ f_i \in E \,\}$$

$$\text{and } \ E^{\times k} = \{\, f_1 \times \ldots \times f_m \mid m \le k \text{ and } \forall i \le m, \ f_i \in E \,\}$$

Lemma 3. *Let E be a set of polynomials. Then,*

$$\mathcal{M}_{E^+} = \mathcal{M}_E \text{ and } |\mathcal{M}_{E^{\times s}}| \le (|\mathcal{M}_E| + 1)^s.$$

Proof. If $\mathbf{x}$ is a monomial in $\mathcal{M}_{E^+}$, it means there exist polynomials $f_1, \ldots, f_m$ in E such that $\mathbf{x}$ is a monomial of $f_1 + \ldots + f_m$. Then there exists $i \le m$ such that $\mathbf{x}$ is a monomial of f_i and so $\mathbf{x}$ is an element of $\mathcal{M}_E$. Hence $\mathcal{M}_{E^+} \subseteq \mathcal{M}_E$. Moreover, as $E \subseteq E^+$, we get $\mathcal{M}_E \subseteq \mathcal{M}_{E^+}$.

Moreover, if $\mathbf{x}$ is a monomial in $\mathcal{M}_{E^{\times s}}$, it means there exist polynomials $f_1, \ldots, f_m$ in E such that $\mathbf{x}$ is a monomial of $f_1 \times \ldots \times f_m$ with $m \le s$. It implies that $\mathbf{x} \in \{\, \mathbf{x_1} \times \ldots \times \mathbf{x_m} \mid m \le s \text{ and } \mathbf{x_i} \in \mathcal{M}_E \,\}$. That is to say, $\mathbf{x} \in \{\, \mathbf{x_1} \times \ldots \times \mathbf{x_s} \mid \text{ and } \mathbf{x_i} \in (\mathcal{M}_E \cup \{1\}) \,\}$. It proves the lemma.

Let C be a $\sum \prod \sum \prod$ circuit. The gates of the circuit are layered into five levels. Inputs are at level 0, multiplication gates at levels 1 and 3 and addition gates at levels 2 and 4. For each level i, let us denote s_i the number of gates at this level, t_i an upper bound on the fan-in of these gates and E_i the set of polynomials computed at this level.

Lemma 4. *Any $\sum \prod \sum \prod$ circuit that computes PERM_n (or DET_n) such that the fan-in of the multiplication gates at level 3 is bounded by v must have size $\exp\left[\Omega\left(\frac{n}{v}\log(n)\right)\right]$.*

Proof. We notice that the hypothesis in the lemma about the bound of the fan-in just states that $t_3 \leq v$.

The polynomials in E_1 are just monomials. So, $|\mathcal{M}_{E_1}| \leq s_1$. We have:

$$E_4 \subseteq E_3^+, \ E_3 \subseteq E_2^{\times t_3} \text{ and } E_2 \subseteq E_1^+.$$

Then by Lemma 3,

$$|\mathcal{M}_{E_4}| \leq (s_1 + 1)^{t_3} \leq (s_1 + 1)^v.$$

However, as PERM_n is an element of E_4, we also have:

$$|\mathcal{M}_{E_4}| \geq |\mathcal{M}_{\mathrm{PERM}_n}| = n!.$$

So, $s_1 \geq (n!)^{\frac{1}{v}} - 1 = 2^{\Omega\left(\frac{n}{v}\log(n)\right)}$

The result of this lemma directly implies Theorem 3.

Proof (Proof of Theorem 3). Let C be a homogeneous $\sum \prod \sum \prod$ circuit which computes PERM_n (or DET_n) such that the fan-in of each bottom gate is at least t. It implies that the degree of each gate at level 1 and 2 is at least t. As the circuit is homogeneous, the degree of a gate at level 3 is upperbounded by n and lowerbounded by t times the number of inputs of this gate. Consequently, in C, the fan-in of the multiplication gates at level 3 is bounded by $\frac{n}{t}$. Then Lemma 4 implies the theorem.

In fact, for computing the determinant, we can also notice that the fan-in of multiplication gates in the depth-four circuits that we get either in [7] or here in Section 5, is linear in $\sqrt{n}$. It implies that in this case, the bounds are tight.

Corollary 3. *If C is a $\sum \prod \sum \prod$ circuit which computes DET_n such that the fan-in of each bottom multiplication gate is $\Omega(\sqrt{n})$ or such that the fan-in of each multiplication gate of level 3 is $O(\sqrt{n})$, then the minimal size of C is $2^{\Theta(\sqrt{n}\log(n))}$.*

Proof. Koiran's result [7] implies that there exist depth-four circuits for DET_n of size $2^{O(\sqrt{n}\log n)}$ such that all multiplication gates have fan-in bounded by $O(\sqrt{n})$. For the lowerbound, the case where the bottom fan-in is lowerbounded by $\Omega(\sqrt{n})$ is given by Theorem 3. The case where the fan-in of gates of level 3 is bounded by $O(\sqrt{n})$ is given by Lemma 4.

Consequently, it would be an interesting question to know the lower bound on the size of an homogeneous circuit computing DET_n. In [5] the authors show that if the circuit is such that the fan-in of bottom gates is bounded by $O(\sqrt{n})$, then the size is $2^{\Omega(\sqrt{n})}$. Here, we show that if all bottom fan-in are lowerbounded by $\Omega(\sqrt{n})$, then the size is $2^{\Omega(\sqrt{n}\log n)}$. What happens if in the circuit, there are some bottom gates with a large fan-in and some bottom gates with a small fan-in?

Question 1. Is it true that if $\mathcal{C}$ is a homogeneous depth-four circuit which computes DET_n then the size of $\mathcal{C}$ is at least $2^{\Omega(\sqrt{n})}$?

References

1. Agrawal, M., Vinay, V.: Arithmetic circuits: A chasm at depth four. In: Proceedings-Annual Symposium on Foundations of Computer Science, pp. 67–75 (2008)
2. Allender, E., Jiao, J., Mahajan, M., Vinay, V.: Non-commutative arithmetic circuits: depth reduction and size lower bounds. Theoretical Computer Science 209(1-2), 47–86 (1998)
3. Bürgisser, P.: Completeness and Reduction in Algebraic Complexity Theory. Algorithms and Computation in Mathematics, vol. 7. Springer (2000)
4. Chen, X., Kayal, N., Wigderson, A.: Partial Derivatives in Arithmetic Complexity and Beyond. Foundations and Trends in Theoretical Computer Science (2011)
5. Gupta, A., Kamath, P., Kayal, N., Saptharishi, R.: Approaching the chasm at depth four. In: Proceedings of the Conference on Computational Complexity, CCC (2013)
6. Gupta, A., Kamath, P., Kayal, N., Saptharishi, R.: Arithmetic circuits: A chasm at depth three. Electronic Colloquium on Computational Complexity (2013)
7. Koiran, P.: Arithmetic circuits: The chasm at depth four gets wider. Theoretical Computer Science 448, 56–65 (2012)
8. Shpilka, A., Yehudayoff, A.: Arithmetic circuits: A survey of recent results and open questions. Foundations and Trends in Theoretical Computer Science, vol. 5 (2010)
9. Valiant, L., Skyum, S., Berkowitz, S., Rackoff, C.: Fast parallel computation of polynomials using few processors. SIAM Journal on Computing 12(4), 641–644 (1983)
10. von zur Gathen, J.: Feasible arithmetic computations: Valiant's hypothesis. Journal of Symbolic Computation 4(2), 137–172 (1987)

Appendix: Proof of Proposition 3

Let f be a homogeneous polynomial computed by a circuit $\tilde{C}$ of size s like in the proposition. First, we can delete the "calculus with constants". To do that, we just have to replace recursively each gate such that all entries are constants by the constant value of this gate. Then, by homogeneity, constants can not be entries of a +-gate. Then, for each ×-gate such that one entry is a constant, we replace the ×-gate by a scalar ⊙-gate. We can notice that this transformation does not increase the size of the circuit. Second, we can reorder the children of the ×-gates and of the ⊙-gates so as to for each one of these gates, the degree of the rightmost child is larger or equals the degree of the other child. We get a circuit C_1 of size s.

   We define now a new circuit C_2 which satisfies the criteria of the proposition. For each pair of gates α and β in C_1, we define the gate $(\alpha; \beta)$ in C_2 as follows:

- If β is a leaf, then $[(\alpha;\beta)]$ equals the sum of the parse trees rooted in α such that β appears in the rightmost path (ie, β is the leaf of the rightmost path).
- If β is not a leaf, then $[(\alpha;\beta)]$ equals the sum of the parse trees rooted in α such that β appears in the rightmost path and where the subcircuit rooted in β is deleted. That is as if we replace the gate β by the input 1 in the rightmost path and we compute $[(\alpha;\beta)]$ with $\beta = 1$ a leaf.

We notice here that it is easy to get the polynomial computed by the gate α: $[\alpha] = \sum_{l \text{ leaf}}[(\alpha;l)]$.

Now, we show how one can compute the value of the gates $(\alpha;\beta)$.

- If β does not appear on the rightmost path of a parse tree rooted in α, then $(\alpha;\beta) = 0$.
- If α is a leaf, then $(\alpha;\alpha) = \alpha$ and else $(\alpha;\alpha) = 1$.
- Otherwise α and β are two different gates and α is not a leaf. If α is a +-gate, then $[(\alpha;\beta)]$ is simply the sum of all $[(\alpha',\beta)]$, where α' is a child of α.
- If α is a $\odot$-gate, then one child is a constant c and the other child is a gate α'. Then $(\alpha;\beta)$ is simply the scalar operation $[(\alpha;\beta)] = [(c;c)] \odot [(\alpha';\beta)]$.
- If α is a $\times$-gate. There are two cases.
  - First case: β is a leaf. Then $\deg(\alpha) > \deg(\beta) = 1$. On each rightmost path ending on β of a parse tree rooted in α, there exists exactly one $\times$-gate γ and its right child on this path γ_r such that:

  $$\deg(\gamma) > \deg(\alpha)/2 \geq \deg(\gamma_r). \tag{1}$$

  Conversely, we notice that for each gate γ satisfying (1), if $[(\alpha;\gamma)]$ and $[(\gamma_r;\beta)]$ are not zero, then γ is on a rightmost path from α to β. Then,

  $$[(\alpha;\beta)] = \sum_{l \text{ leaf, } \gamma \text{ } \times\text{-gate verifying (1)}} [(\alpha;\gamma)][(\gamma_l;l)][(\gamma_r;\beta)].$$

  One can notice that $\deg(\alpha;\beta) = \deg(\alpha)$. Using (1):

  $$\deg(\alpha;\gamma) = \deg(\alpha) - \deg(\gamma) < \deg(\alpha)/2$$
  $$\deg(\gamma_r;\beta) = \deg(\gamma_r) \leq \deg(\alpha)/2$$
  $$\deg(\gamma_l;l) = \deg(\gamma_l) \leq \deg(\gamma_r) \leq \deg(\alpha)/2.$$

  Consequently, $[(\alpha;\beta)]$ is computed by a depth-2 circuit of size at most $s^2 + 1$: a +-gate where each child is a $\times$-gate of fan-in 3. Each child of these $\times$-gates is of degree at most the half of the degree of the $\times$-gate.
  - Second case: β is not a leaf. Then there exists on every rightmost paths rooted in α a $\times$-gate γ and its child on this path γ_r such that:

  $$\deg(\gamma) \geq (\deg(\alpha) + \deg(\beta))/2 > \deg(\gamma_r). \tag{2}$$

  Then by the same argument,

  $$[(\alpha;\beta)] = \sum_{l \text{ leaf, } \gamma \text{ } \times\text{-gate verifying (2)}} [(\alpha;\gamma)][(\gamma_l;l)][(\gamma_r;\beta)]. \tag{3}$$

We have this time with (2):

$$\deg(\alpha; \beta) = \deg(\alpha) - \deg(\beta)$$
$$\deg(\alpha; \gamma) = \deg(\alpha) - \deg(\gamma) \le (\deg(\alpha) - \deg(\beta))/2$$
$$\deg(\gamma_r; \beta) = \deg(\gamma_r) < (\deg(\alpha) - \deg(\beta))/2.$$

The problem here is that the degree of $(\gamma_l; l)$ could be larger than the average of the degrees of α and β. If γ_l is of degree at most 1 (and so exactly 1) and if the degree of $(\alpha; \beta)$ is also 1, then $\gamma = \alpha$ (they are the same gate) and $(\gamma_r; \beta)$ is of degree 0 and computes a constant c_γ. Hence,

$$[(\alpha; \beta)] = \sum_{l \text{ leaf, } \gamma \text{ ×-gate verifying (2)}} [c_\gamma] \odot [(\gamma_l; l)].$$

Now, if the degree of γ_l is again 1 but if $(\alpha; \beta)$ is of degree at least 2, then the computation of the gate $(\alpha; \beta)$ by the formula (3) works (ie., the degree of $(\gamma_l; l)$ is smaller than half of the degree of $(\alpha; \beta)$). Otherwise, the degree of γ_l is at least 2 and at most $\deg(\alpha; \beta)$. As l is a leaf, we can apply the first case (even if γ_l is not a ×-gate). There exists also on every rightmost paths rooted in γ_l a ×-gate μ and its child on this path μ_r such that:

$$\deg(\mu) > \deg(\gamma_l)/2 \ge \deg(\mu_r). \tag{4}$$

Then,

$$[(\alpha; \beta)] = \sum_{l_1, l_2, \gamma, \mu} [(\alpha; \gamma)][(\gamma_r; \beta)][(\gamma_l; \mu)][(\mu_l; l_2)][(\mu_r; l_1)] \tag{5}$$

where the sum is taken over all l_1, l_2 leaves, γ ×-gate verifying (2) and μ ×-gate verifying (4).

The degrees of the gates $(\gamma_l; \mu)$, $(\mu_l; l_2)$ and $(\mu_r; l_1)$ are bounded by half of the degree of γ_l. Hence, $[(\alpha; \beta)]$ is computed by a depth-2 size-$s^4 + 1$ circuit. The ×-gates are of fan-in bounded by 5 and the degree of their children is bounded by half their degree.

Consequently, for each gates α and β in C_1, the gate $(\alpha; \beta)$ is computed in C_2 by a sub-circuit of size at most $s^4 + 1$. At the end we get a circuit of size at most $s^6 + s^2$ which computes all gates $(\alpha; \beta)$. Finally, f is computed by a circuit of size bounded by $s^6 + s^2 + 1$.

That proves the proposition.