THEORY OF COMPUTING, Volume 6 (2010), pp. 135–177
www.theoryofcomputing.org

Elusive Functions and Lower Bounds for Arithmetic Circuits

Ran Raz*

Received: June 3, 2008; revised: May 15, 2010; published: September 11, 2010.

Abstract: It is a basic fact of linear algebra that the image of the curve

$$f(x) = (x^1, x^2, x^3, \ldots, x^m),$$

say over $\mathbb{C}$, is not contained in any $(m-1)$-dimensional affine subspace of $\mathbb{C}^m$. In other words, the image of f is not contained in the image of any polynomial mapping $\Gamma : \mathbb{C}^{m-1} \to \mathbb{C}^m$ of degree 1 (that is, an affine mapping). Can one give an explicit example of a polynomial curve $f : \mathbb{C} \to \mathbb{C}^m$ such that the image of f is not contained in the image of any polynomial mapping $\Gamma : \mathbb{C}^{m-1} \to \mathbb{C}^m$ of degree 2?

In this paper, we show that problems of this type are closely related to proving lower bounds for the size of general arithmetic circuits. For example, any explicit f as above (with the right notion of explicitness), of degree up to $2^{m^{o(1)}}$, implies super-polynomial lower bounds for computing the permanent over $\mathbb{C}$.

More generally, we say that a polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$ is (s,r)-*elusive*, if for every polynomial mapping $\Gamma : \mathbb{F}^s \to \mathbb{F}^m$ of degree r, $\mathrm{Image}(f) \not\subset \mathrm{Image}(\Gamma)$. We show that for many settings of the parameters n, m, s, r, explicit constructions of elusive polynomial mappings imply strong (up to exponential) lower bounds for general arithmetic circuits.

*Research supported by the Israel Science Foundation (ISF), the Binational Science Foundation (BSF) and the Minerva Foundation.

ACM Classification: F.2.2, F.1.3, F.1.2, G.2.0

AMS Classification: 68Q15, 68Q17, 94C05

Key words and phrases: complexity theory, lower bounds, separation of complexity classes, circuit complexity, arithmetic circuits, polynomials

DOI: 10.4086/toc.2010.v006a007

Finally, for every $r < \log n$, we give an explicit example of a polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^{n^2}$, of degree $O(r)$, that is (s,r)-elusive for $s = n^{1+\Omega(1/r)}$. We use this to construct for any r, an explicit example of an n-variate polynomial of total-degree $O(r)$, with coefficients in $\{0,1\}$, such that any depth-r arithmetic circuit for this polynomial (over any field) is of size $\geq n^{1+\Omega(1/r)}$.

In particular, for any constant r, this gives a constant degree polynomial such that any depth-r arithmetic circuit for this polynomial is of size $\geq n^{1+\Omega(1)}$. Previously, only lower bounds of the type $\Omega(n \cdot \lambda_r(n))$, where the λ_r are extremely slowly growing functions, were known for constant-depth arithmetic circuits for polynomials of constant degree (actually, for linear functions).

1 Introduction

We present a family of problems that are very simple to describe, that seem natural to study from several different points of view (such as, geometric, algebraic and combinatorial), that are seemingly unrelated to arithmetic circuit complexity, and whose solution would give strong (up to exponential) lower bounds for the size of general arithmetic circuits. We then prove lower bounds of $n^{1+\Omega(1/d)}$ for the size of arithmetic circuits of depth d for explicit polynomials of degree $O(d)$.

Let $\mathbb{F}$ be a field. A *polynomial mapping* $f : \mathbb{F}^n \to \mathbb{F}^m$ of degree r is a function such that each of its coordinates can be presented as a polynomial of total-degree at most r in the input variables. We say that a polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$ is (s,r)-*elusive*, if for every polynomial mapping $\Gamma : \mathbb{F}^s \to \mathbb{F}^m$ of degree r, $\mathrm{Image}(f) \not\subset \mathrm{Image}(\Gamma)$. (For more details about polynomial mappings and elusive polynomial mappings, see Subsection 1.4.) Can one give explicit examples of elusive polynomial mappings?

We show that for many settings of the parameters, explicit constructions of elusive polynomial mappings imply strong (up to exponential) lower bounds for general arithmetic circuits. (Here, and below, *explicit* means $\mathrm{poly}(n)$-definable, as defined in Definition 1.3. In particular, $f : \mathbb{F}^n \to \mathbb{F}^m$ is $\mathrm{poly}(n)$-definable if given a monomial q and an index i, the coefficient of the monomial q in the polynomial f_i can be computed in time $\mathrm{poly}(n)$. For more details, see Subsection 1.5.) For example, we show the following results: Let $\mathbb{F}$ be a field of characteristic not equal to 2.

1. Let $s = s(n), m = m(n)$ be such that $n^{\omega(1)} \leq m$ (i. e., m is super-polynomial in n) and $s \geq m^{0.9}$. (Think of m as relatively small, say $m = n^{\log\log n}$.)

   If there exists an explicit $(s,2)$-elusive polynomial mapping, $f : \mathbb{F}^n \to \mathbb{F}^m$ (of degree at most $\mathrm{poly}(n)$), then any arithmetic circuit for the permanent, over $\mathbb{F}$, is of super-polynomial size.

2. Let $s = s(n), m = m(n), r = r(n)$ be such that $n^{\omega(1)} \leq s < m = n^r$. (Think of r as relatively small, say $r = \log\log n$, and hence $m = n^{\log\log n}$; and think of s as significantly smaller than m, say $s = n^{\log\log\log n}$.)

   If there exists an explicit (s,r)-elusive polynomial mapping, $f : \mathbb{F}^n \to \mathbb{F}^m$ (of degree at most $\mathrm{poly}(n)$), then any arithmetic circuit for the permanent, over $\mathbb{F}$, is of super-polynomial size.

In other words, one can prove super-polynomial lower bounds for the permanent, simply by constructing elusive polynomial mappings.

We note that in the above two examples (as well as in all other cases discussed in this paper), an elusive polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$ of degree up to 2^n, (rather than $\mathrm{poly}(n)$), is also sufficient, since we can easily construct from it a *multilinear* polynomial mapping $\hat{f} : \mathbb{F}^{n^2} \to \mathbb{F}^m$ such that the image of f is contained in the image of $\hat{f}$. We prefer to state our results with the *seemingly* weaker upper bound of $\mathrm{poly}(n)$ on the degree, because there is no standard notion for explicitness of polynomials of degree larger than $\mathrm{poly}(n)$, while there is a standard and well established notion for explicitness of polynomials of degree up to $\mathrm{poly}(n)$. For more details, see Subsections 1.4 and 1.5.

In both of the above two examples, as well as in all other cases discussed in this paper, it is not hard to show the *existence* of (non-explicit) polynomial mappings $f : \mathbb{F}^n \to \mathbb{F}^m$, with the required properties. The hard problem is to construct f explicitly.

We note also that polynomial mappings $f : \mathbb{F}^n \to \mathbb{F}^m$, as above, can easily be constructed from a set H of 2^n points in $\mathbb{F}^m$ such that for every mapping $\Gamma : \mathbb{F}^s \to \mathbb{F}^m$, as above, H is not contained in the image of Γ. Once again, it is not hard to prove the existence of such a set H, and the hard problem is to construct H explicitly.

When one is interested in proving polynomial lower bounds (rather than super-polynomial lower bounds), one can even assume that the mapping Γ is given as an input. For example, we can prove the following result: Let $\mathbb{F}$ be any field. Let $s = n^{90}$, and let $m = n^{100}$. Let $\Gamma : \mathbb{F}^s \to \mathbb{F}^m$ be a polynomial mapping of degree 2, with coefficients in $\{0,1\}$. Note that Γ can be described by $\mathrm{poly}(n)$ bits. We show that if one can give a polynomial-time Turing machine that on input Γ, as above, outputs an explicit polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$ of degree at most $\mathrm{poly}(n)$, such that $\mathrm{Image}(f) \not\subset \mathrm{Image}(\Gamma)$, then one obtains an explicit lower bound of $\Omega(n^{10})$ for the size of arithmetic circuits.

Note also, that in order to obtain the above mentioned explicit lower bound of $\Omega(n^{10})$ for the size of arithmetic circuits, it is enough to give a polynomial-time Turing machine that on input Γ, as above, outputs one point outside the image of Γ. Thus, one can also obtain "win-win" results, such as: either the problem of finding a point outside the image of a polynomial mapping Γ is hard (when Γ is given as an input), in which case we have an example of a hard problem, or, otherwise, there exists an explicit lower bound of $\Omega(n^{10})$ for the size of arithmetic circuits (or both).

Finally, for every $r < \log n$, we give an explicit example of a polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^{n^2}$, of degree $O(r)$, that is (s,r)-elusive for $s = n^{1+\Omega(1/r)}$. We use this to prove lower bounds for bounded-depth arithmetic circuits, for polynomials of bounded degree. For any $r = r(n)$, we give an explicit example of an n-variate polynomial of degree $O(r)$, with coefficients in $\{0,1\}$, such that any (unbounded fanin) depth-r arithmetic circuit for this polynomial, over any field, is of size $\geq n^{1+\Omega(1/r)}$. In particular, for any constant r, this gives a constant degree polynomial such that any depth-r arithmetic circuit for this polynomial is of size $\geq n^{1+\Omega(1)}$. Previously, only slightly super-linear lower bounds were known for constant-depth arithmetic circuits, for polynomials of constant degree (actually, for linear functions).

1.1 Arithmetic circuits

Let $\mathbb{F}$ be a field, and let $\{x_1, \ldots, x_n\}$ be a set of input variables. An *arithmetic circuit* is a directed acyclic graph, as follows: Every *leaf* of the graph (i. e., a node of in-degree 0) is labelled with either an input variable or the field element 1. Every other node of the graph is labelled with either $+$ or $\times$ (in the first case the node is a *sum-gate* and in the second case a *product-gate*). Every edge in the graph is labelled with an arbitrary field element. A node of out-degree 0 is called an *output-gate* of the circuit.

Every node and every edge in an arithmetic circuit compute a polynomial in the ring $\mathbb{F}[x_1,\ldots,x_n]$ in the following way. A leaf just computes the input variable or field element that labels it. An edge (u,v), labelled by $\alpha \in \mathbb{F}$, computes the product of α and the polynomial computed by u. A sum-gate computes the sum of the polynomials computed by all edges that reach it. A product-gate computes the product of the polynomials computed by all edges that reach it. We say that a polynomial $g \in \mathbb{F}[x_1,\ldots,x_n]$ is computed by the circuit if it is computed by one of the circuit's output-gates.

The *size* of a circuit Φ is defined to be the number of edges in Φ, and is denoted by $\mathrm{Size}(\Phi)$. (We assume w.l.o.g. that the size of a circuit is larger than the number of its input variables and the number of its output-gates.) The *depth* of a circuit Φ is defined to be the length of the longest directed path in Φ, and is denoted by $\mathrm{Depth}(\Phi)$. If (u,v) is an edge in the circuit, we say that u is a *child* of v and v is a *parent* of u. The *fanin* of a circuit is defined to be the maximal in-degree of a node in the circuit, that is, the maximal number of children that a node has. Note that we do not restrict the fanin of a circuit to be 2.

1.2 Background

Arithmetic circuits is the standard computational model for computing polynomials (e. g., for computing the determinant or the permanent of a matrix, or the product of two matrices). If one considers polynomials of very high degree, it is not hard to prove high lower bounds for the size and depth of arithmetic circuits. For example, any arithmetic circuit for the polynomial x^{2^n} is obviously of depth at least n. However, interesting polynomials that we would like to study are usually of degree bounded by $\mathrm{poly}(n)$ (where n is the number of input variables). Hence, the discussion is usually restricted to polynomials of degree at most $\mathrm{poly}(n)$, and special attention is given for proving lower bounds for polynomials of a relatively low degree (e. g., constant degree).

The landmark results of Strassen [35] and Baur and Strassen [5] give lower bounds of $\Omega(n\log r)$ for the size of arithmetic circuits for explicit n-variate polynomials of degree r. In particular, when the degree r is $\mathrm{poly}(n)$, this gives explicit lower bounds of $\Omega(n\log n)$. For polynomials of constant degree, there are no lower bounds better than $\Omega(n)$.

Proving super polynomial lower bounds for arithmetic circuits (for explicit polynomials) is one of the most challenging open problems in computational complexity. Such lower bounds are only known for some restricted classes of arithmetic circuits. For example, super polynomial lower bounds were proved for non-commutative formulas [19], for multilinear formulas [25, 24], and for circuits of depth 3 over finite fields [12, 13].

For additional background on arithmetic circuit complexity, see [11, 7].

1.3 Constant-depth arithmetic circuits

Exponential lower bounds for the size of constant-depth *Boolean* circuits (for explicit functions) are well known [9, 2, 38, 14, 29, 33]. In particular, exponential lower bounds for constant-depth Boolean circuits over the basis $\{\wedge,\vee,\neg,\oplus\}$ were given by Razborov [29]. This gives exponential lower bounds for constant-depth arithmetic circuits over the field GF(2), since a product over GF(2) is just the $\wedge$ operation, and a sum over GF(2) is just the $\oplus$ operation.

However, for constant-depth arithmetic circuits over other fields, much less is known. In particular, super-polynomial lower bounds are not known, even for circuits of depth 4. For circuits of depth 3 over finite fields, exponential lower bounds were proved by Grigoriev and Karpinski [12] and Grigoriev and Razborov [13]. For circuits of depth 3 over infinite fields, only quadratic lower bounds are known, (proved by Shpilka and Wigderson [32]).

In this paper, we are interested in proving lower bounds for constant-depth arithmetic circuits, for polynomials of constant degree. Recall that Baur and Strassen proved a lower bound of $\Omega(n \log r)$ for the size of arithmetic circuits of any depth, where r is the degree of the polynomial computed. Note, however, that if one considers polynomials of constant degree, this only gives a linear lower bound. Super-linear lower bounds for constant-degree polynomials (actually, for linear functions), for arithmetic circuits of constant-depth, are well known (proved by Pudlak [21] and by [27]). These bounds, however, are extremely weak. For circuits of depth d, these bounds are of the type $\Omega(n \cdot \lambda_d(n))$, where $\lambda_d(n)$ are extremely slowly growing functions (e. g., $\lambda_5(n) = \log^* n$, and $\lambda_7(n) = \log^* \log^* n$). These bounds are based on the fact that very small graphs of very small depth cannot be *super-concentrators*, and the proofs use complicated combinatorial arguments, first used to prove lower bounds for the size of super-concentrators [8, 21].

As mentioned above, in this work we give for any d, an explicit example of a polynomial of degree $O(d)$ such that any depth-d arithmetic circuit for this polynomial is of size $\geq n^{1+\Omega(1/d)}$. In particular, for any constant d, this gives a constant degree polynomial such that any depth-d arithmetic circuit for this polynomial is of size $\geq n^{1+\Omega(1)}$.

Previous to our work, a very related approach was used by Shoup and Smolensky to show the existence of points $p_1, \ldots, p_n \in \mathbb{C}$ such that any arithmetic circuit of depth d, over $\mathbb{C}$, for polynomial evaluation (or interpolation) at these points, is of size $\Omega(dn^{1+1/d})$ [31]. This gives a lower bound of $\Omega(dn^{1+1/d})$ for depth-d arithmetic circuits, for non-explicit linear-forms, over $\mathbb{C}$. We note also, that one can view the points $p_1, \ldots, p_n$ as a part of the input to the circuit and hence view the lower bound of [31] as a lower bound for explicit polynomials of degree $O(n)$ (rather than a lower bound for non-explicit linear-forms).

The techniques used by Shoup and Smolensky are very related to ours. In particular, implicit in their work is an explicit example of a polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^{n^2}$, of degree $O(n)$, that is (s,r)-elusive for $s = n^{1+\Omega(1/r)}$; and that function is used there to prove their lower bound. This compares to an explicit construction of a polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^{n^2}$, of degree $O(r)$, that is (s,r)-elusive for $s = n^{1+\Omega(1/r)}$, that we present here (for every $r < \log n$); and that we use here to prove our lower bound. The construction of the function, and the proof that it is elusive is the main technical difference between the two proofs.

Thus, our lower bounds for bounded-depth arithmetic circuits can be viewed as a generalization and improvement of the techniques and results of Shoup and Smolensky. The main technical difference between the results is that our lower bound is for polynomials of degree $O(d)$ while their lower bound (when viewed as a lower bound for explicit polynomials) is for polynomials of degree $O(n)$. There are several other differences, as follows:

1. In [31], the points $p_1, \ldots, p_n$ were not viewed as a part of the input to the circuit. Hence, Shoup

and Smolensky do not view their result as a lower bound for explicit polynomials, and rather state their lower bound as a lower bound for non-explicit linear-forms. Here, we view (the equivalent of) the points $p_1,\dots,p_n$ as a part of the input to the circuit, and hence we obtain lower bounds for explicit polynomials.

Technically, this is just an observation.

2. Shoup and Smolensky only prove their lower bounds over $\mathbb{C}$, while here we prove lower bounds over any field $\mathbb{F}$.

   Technically, this improvement is not hard. It is obtained by working over a large enough field extension $\mathbb{G} \supset \mathbb{F}$.

3. Shoup and Smolensky's lower bound (when viewed as a lower bound for explicit polynomials) is for polynomials of degree $O(n)$, while here we prove lower bounds for polynomials of degree $O(d)$.

   This is the main advantage of our lower bounds over the ones of Shoup and Smolensky. Technically, this is the main difference between the proofs, and the hard part of our argument.

We will now try to explain the importance of proving lower bounds for polynomials of constant degree (rather than for polynomials of degree, say, $O(n)$). One reason is that strong enough lower bounds for constant-depth arithmetic circuits, for polynomials of constant degree, would imply lower bounds for general arithmetic circuits!

Consider for example the following trivial but striking fact: Any (unbounded-depth) arithmetic circuit of size s, for a polynomial of a constant degree r, can be translated into an arithmetic circuit of size $O(s^2)$ and depth $O(r)$, for the same[1] polynomial.[2] Thus, surprisingly, a lower bound of $\Omega(n^{2+\varepsilon})$ for constant-depth arithmetic circuits, for an explicit polynomial of constant degree, would imply a lower bound of $\Omega(n^{1+\varepsilon/2})$ for the size of general arithmetic circuits.

Or consider the following fact: Any fanin-2 arithmetic circuit of depth $O(\log n)$ and size $O(n^{1+\varepsilon})$, for a polynomial of a constant degree, can be translated into an (unbounded-fanin) arithmetic circuit of size $O(n^{1+\varepsilon'})$ and constant-depth, for the same polynomial, (for any $\varepsilon' > \varepsilon$).[3] Thus, a lower bound of $\Omega(n^{1+\varepsilon'})$ for constant-depth arithmetic circuits, for polynomials of constant degree, would imply a lower bound of $\Omega(n^{1+\varepsilon})$ for fanin-2 arithmetic circuits of depth $O(\log n)$, that is, a strong size-depth tradeoff for general arithmetic circuits.

Thus, our lower bounds are close to the best possible, without implying strong size-depth tradeoffs for general arithmetic circuits.

[1] This is done roughly as follows: we will have r levels in the new circuit, where the nodes in level i compute the parts of degree i of all polynomials computed by nodes in the original circuit (by a depth-2 circuit that uses nodes from previous levels).

[2] Moreover, any arithmetic circuit of size s, for a polynomial of degree r, can be translated into an arithmetic circuit of size $\mathrm{poly}(s,r)$ and depth $O(\log r)$, for the same polynomial [37].

[3] One can start from any fanin-2 arithmetic circuit of depth $d = O(\log n)$ and size s (for a polynomial of a constant degree, say, 10), and translate it into an unbounded-fanin arithmetic circuit of depth $d/(\delta \log n) = O(1)$ and size $s \cdot n^{O(\delta)}$ (for any constant $\delta > 0$). This is done roughly as follows: first, assume without loss of generality that every partial computation in the circuit computes a polynomial of constant degree. Then, replace any partial computation of depth $\delta \log n$ by a straightforward computation of the polynomial that it computes.

Finally, we note that our lower bounds match size-depth tradeoffs (of $n^{1+\Omega(1/d)}$) that were previously known for, so called, *bounded coefficient circuits*, a restricted class of arithmetic circuits over the field $\mathbb{C}$ [20, 17, 22, 23].

1.4 Polynomial mappings

Let $\mathbb{F}$ be a field. A *polynomial mapping* $f : \mathbb{F}^n \to \mathbb{F}^m$ of degree r is a tuple $f = (f_1, \ldots, f_m)$, where for every $i \in \{1, \ldots, m\}$, $f_i(x_1, \ldots, x_n) \in \mathbb{F}[x_1, \ldots, x_n]$ is a polynomial of total-degree at most r.[4] The mapping f is *multilinear*, if $f_1, \ldots, f_m$ are multilinear polynomials (i. e., the degree of every input variable in every f_i is at most 1). The mapping f is *homogenous*, if $f_1, \ldots, f_m$ are homogenous polynomials of the same total-degree (i. e., the total-degree of every monomial in every f_i is the same). We denote the image of a polynomial mapping f by $\mathrm{Image}(f)$.

Note that given polynomials $f_1, \ldots, f_m \in \mathbb{F}[x_1, \ldots, x_n]$, for any field extension $\mathbb{G} \supset \mathbb{F}$, we can think of $f = (f_1, \ldots, f_m)$ as a polynomial mapping $f : \mathbb{G}^n \to \mathbb{G}^m$ (since $\mathbb{F}[x_1, \ldots, x_n] \subset \mathbb{G}[x_1, \ldots, x_n]$).

Definition 1.1 (Eludes, Elusive)**.** We say that a polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$ eludes a polynomial mapping $\Gamma : \mathbb{F}^s \to \mathbb{F}^m$ if $\mathrm{Image}(f) \not\subset \mathrm{Image}(\Gamma)$. We say that $f : \mathbb{F}^n \to \mathbb{F}^m$ is (s,r)-elusive, if it eludes every polynomial mapping $\Gamma : \mathbb{F}^s \to \mathbb{F}^m$ of degree at most r.

For every r and every polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$ of degree less than 2^r in each variable, we can construct a multilinear polynomial mapping $\hat{f} : \mathbb{F}^{n \cdot r} \to \mathbb{F}^m$ such that the image of f is contained in the image of $\hat{f}$. This is done as follows. For every input variable x_i, we introduce r new input variables $x_{i,1}, \ldots, x_{i,r}$. We replace every occurrence of x_i^k (in each of the polynomials $f_1, \ldots, f_m$) by the product $\prod_{j=1}^r x_{i,j}^{k_j}$, where $(k_r, \ldots, k_1)$ is the binary representation of k.

Proposition 1.2. *Let $f : \mathbb{F}^n \to \mathbb{F}^m$ be a polynomial mapping of degree less than 2^r in each variable, and let $\hat{f} : \mathbb{F}^{n \cdot r} \to \mathbb{F}^m$ be the multilinear polynomial mapping as above. Then* $\mathrm{Image}(f) \subset \mathrm{Image}(\hat{f})$.

Proof. For any $a_1, \ldots, a_n \in \mathbb{F}$,

$$f(a_1, \ldots, a_n) = \hat{f}(a_1^1, a_1^2, \ldots, a_1^{2^{r-1}}, \cdots, a_n^1, a_n^2, \ldots, a_n^{2^{r-1}}).$$

□

By Proposition 1.2, if f eludes a mapping Γ, then so does $\hat{f}$. In particular, if f is (s,r)-elusive, then so is $\hat{f}$. For that reason, it is enough for us to limit the discussion to polynomial mappings f that are multilinear, and in particular, are of degree at most $poly(n)$. (Note, however, that the polynomial mappings Γ that we consider are not necessarily multilinear.)

[4] Note that given a function $f : \mathbb{F}^n \to \mathbb{F}^m$, the representation of $f_1, \ldots, f_m$ as polynomials in $\mathbb{F}[x_1, \ldots, x_n]$, if exists, is not necessarily unique (if $\mathbb{F}$ is finite). Hence, we assume that a polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$, is already given as a tuple $(f_1, \ldots, f_m)$ of polynomials in $\mathbb{F}[x_1, \ldots, x_n]$.

1.5 Explicit polynomial mappings

The standard notion of explicitness of a polynomial $f \in \mathbb{F}[x_1,\ldots,x_n]$ is that f is *explicit* if it is (uniformly) poly(n)-definable, that is, it belongs to the (uniform version of the) class VNP, Valiant's algebraic version of the class NP [36] (see also [10, 6]).

Formally, $f \in \mathbb{F}[x_1,\ldots,x_n]$ is poly(n)-definable, iff for some $\ell = \text{poly}(n)$, there exists a polynomial $g \in \mathbb{F}[x_1,\ldots,x_n,e_1,\ldots,e_\ell]$ of degree poly(n), that can be computed by an arithmetic circuit of size poly(n), and such that

$$f(x_1,\ldots,x_n) = \sum_{e_1,\ldots,e_\ell \in \{0,1\}} g(x_1,\ldots,x_n,e_1,\ldots,e_\ell)$$

(see [6], Definition 2.5, or [10], Theorem 4.2). Many equivalent definitions of poly(n)-definability can be found in [10, 6].

In particular, if f is multilinear with coefficients in $\{0,1\}$, and there exists a deterministic polynomial-time Turing machine that on inputs $e_1,\ldots,e_n \in \{0,1\}$ outputs the coefficient of the monomial $x_1^{e_1}\cdots x_n^{e_n}$ in f, then f is poly(n)-definable (see [10], Proposition 4.4)

Here, we extend the notion of poly(n)-definability to polynomial mappings $f : \mathbb{F}^n \to \mathbb{F}^m$, by the following definition.

Definition 1.3 (poly(n)-Definable)**.** A polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$ is poly(n)-definable if for some $\ell = \text{poly}(n)$, and for $k \doteq \lceil \log_2 m \rceil$, there exists a polynomial $g \in \mathbb{F}[x_1,\ldots,x_n,e_1,\ldots,e_\ell,w_1,\ldots,w_k]$ of degree poly(n), that can be computed by an arithmetic circuit of size poly(n), and such that for every $i \in \{1,\ldots,m\}$,

$$f_i(x_1,\ldots,x_n) = \sum_{e_1,\ldots,e_\ell \in \{0,1\}} g(x_1,\ldots,x_n,e_1,\ldots,e_\ell,i_1,\ldots,i_k)\,,$$

(where $(i_k,\ldots,i_1)$ is the binary representation of $i-1$).

Note that we allow the size of the arithmetic circuit for g to depend polynomially on n, but we do not allow it to depend polynomially on m. This is important because we will consider cases where m is super-polynomial in n. Intuitively, this means that it is not enough that for every i the function f_i can be defined by a different polynomial-size arithmetic circuit g_i. We require that $f_1,\ldots,f_m$ can all be defined by the same polynomial-size arithmetic circuit g.

Finally, we note that if $f : \mathbb{F}^n \to \mathbb{F}^m$ is a multilinear polynomial mapping, and the coefficient of every monomial in every f_i is in $\{0,1\}$, and there exists a deterministic polynomial-time Turing machine that on inputs i and $e_1,\ldots,e_n \in \{0,1\}$ outputs the coefficient of the monomial $x_1^{e_1}\cdots x_n^{e_n}$ in f_i, then, the polynomial mapping f is poly(n)-definable.

1.6 Techniques

Denote by $m = \binom{n+r-1}{r}$ the number of monomials of total-degree r in n variables. Consider polynomials $g \in \mathbb{F}[z_1,\ldots,z_n]$ of total-degree r. Every such polynomial g can be viewed as a vector of m coefficients, that is, a vector in $\mathbb{F}^m$.

We take a universal arithmetic circuit, with s edges, and consider the polynomial $g \in \mathbb{F}^m$ computed by the circuit, as a function of the s labels of the edges of the circuit. This defines a polynomial mapping

$\Gamma : \mathbb{F}^s \to \mathbb{F}^m$. We show that if one takes an arithmetic circuit in the right form, the mapping Γ is of a relatively small degree.

Roughly speaking, we can show, for example, that for every n, r, s', there is $s = \mathrm{poly}(s', n, r)$, and a polynomial mapping $\Gamma : \mathbb{F}^s \to \mathbb{F}^m$, of degree $O(r)$, such that if g is computable by an arithmetic circuit of size s', then g is in the image of Γ, (and if g is in the image of Γ then g is computable by an arithmetic circuit of size s). Moreover, the mapping Γ can be efficiently constructed in time $\mathrm{poly}(s^r)$.

Thus, the image of the polynomial mapping Γ captures the set of polynomials of low complexity. A polynomial is of low complexity only if it is in the image of Γ. Thus, our goal in proving lower bounds is just to find polynomials that are not in the image of Γ.

There are several possible ways to approach this problem. First, one can try to take the explicit description of Γ and find (say, in polynomial time) a point outside its image. Since the explicit description of Γ is of size $\mathrm{poly}(s^r)$, this approach is limited to s, r such that $s^r = \mathrm{poly}(n)$, and hence is limited to proving polynomial lower bounds. Note, however, that one can also try to use this approach for finding polynomials of super-polynomial complexity, in super-polynomial time.

A more promising approach (at least if one is interested in proving unconditional super-polynomial lower bounds), is to try to find a set of points that is not contained in the image of Γ, for any polynomial mapping $\Gamma : \mathbb{F}^s \to \mathbb{F}^m$ of degree $O(r)$.

Consider for example a polynomial $f \in \mathbb{F}[x_1, \dots, x_n, z_1, \dots, z_n]$, of total-degree r in the set of variables $\{z_1, \dots, z_n\}$. For every $a_1, \dots, a_n \in \mathbb{F}$, we can substitute $x_1 = a_1, \dots, x_n = a_n$ and obtain a polynomial $f_{a_1,\dots,a_n} \in \mathbb{F}[z_1, \dots, z_n]$ of total-degree r, that is, a point in $\mathbb{F}^m$. Thus, we obtain a polynomial mapping $f' : \mathbb{F}^n \to \mathbb{F}^m$. If $\mathrm{Image}(f') \not\subset \mathrm{Image}(\Gamma)$ then one of the polynomials $f_{a_1,\dots,a_n}$ cannot be computed by a circuit of size s', and hence f cannot be computed by a circuit of size s'. If, in addition, f is explicit, we obtain an explicit lower bound for arithmetic circuits.

How to prove that $\mathrm{Image}(f') \not\subset \mathrm{Image}(\Gamma)$? One idea that comes to mind is to prove the existence of a polynomial $\Lambda : \mathbb{F}^m \to \mathbb{F}$, such that, $\Lambda \circ \Gamma \equiv 0$, while $\Lambda \circ f' \not\equiv 0$. We use this idea to obtain our lower bounds for bounded-depth arithmetic circuits in Section 4.

We note that many variants of these ideas can also be considered. For example, if the model of computation is restricted, one can capture the polynomials of low complexity by a mapping Γ that may have some additional helpful properties. Another idea that comes to mind is to try to prove the existence of a polynomial $g \in \mathbb{F}^m$ such that there is a short proof for the statement $g \notin \mathrm{Image}(\Gamma)$.

Finally we note that in several places in this work we will use the result of Baur and Strassen that if a polynomial can be computed by an arithmetic circuit of size s and depth d, then all partial derivatives of that polynomial can be computed by one arithmetic circuit of size $5s$ and depth $3d$ [5] (for an improvement of the constants see [18]). For that reason, we will consider in Section 3 and Section 4 a *tuple* of polynomials, rather than a single polynomial g. This will improve the parameters in some of our results, by applying the result of Baur and Strassen. In Section 5, we work with a single polynomial g (which is somewhat simpler).

1.7 Related work

The idea to consider a polynomial computed by a circuit, as a function of the labels of the edges of the circuit, goes back to the works of Strassen [34] and Lipton [16], in the context of arithmetic circuits with a single input variable. Strassen and Lipton used this idea to prove non-explicit lower bounds for arithmetic circuits with a single input variable. Years after, the same idea was used by Shoup and Smolensky [31], in the context of bounded-depth linear arithmetic circuits (i. e., bounded-depth arithmetic circuits without product-gates). To the best of our knowledge, previous to our work, the idea was not used for general arithmetic circuits. As mentioned above, the work of Shoup and Smolensky [31] is very related to ours also in the way in which we prove lower bounds for bounded-depth arithmetic circuits. Moreover, an explicit example of an elusive function (with certain parameters) is implicit in their work, and is used there to prove their lower bound. (For more details, see the detailed discussion in Subsection 1.3.) As far as we know, [31] is the first and only previous work that uses elusive functions to prove lower bounds. Our results suggest that these ideas can possibly be extended to the more general setting of general arithmetic circuits.

Another related paper is the work of Impagliazzo and Kabanets [15]. Impagliazzo and Kabanets proved that if one can test in deterministic polynomial time (or even in nondeterministic subexponential time), whether a given arithmetic circuit over the integers computes the identically-zero polynomial, then, either $\mathsf{NEXP} \not\subset \mathsf{P}/\mathsf{poly}$, or the permanent is not computable by polynomial-size arithmetic circuits. This result is related to ours, since constructing an elusive function can also be viewed as a derandomization problem.

Another idea, related to ours, in the area of propositional proof complexity, was to study the length of propositional proofs for tautologies of the form $b \notin \mathrm{Image}(G)$, for pseudorandom generators $G : \{0,1\}^n \to \{0,1\}^m$ [3]. It was proved in [3], (as well as in subsequent works, e. g., [4]), that for some functions $G : \{0,1\}^n \to \{0,1\}^m$, tautologies of this form are hard to prove in several well-studied propositional proof systems. We refer the reader to [3] for the many motivations (given there) for studying such tautologies. We note only that one of the original motivations for studying these tautologies was that one can consider a function G that maps a description of a Boolean circuit to the truth-table of the function computed by it (see also [30]). For this particular function G, proving that tautologies of the form $b \notin \mathrm{Image}(G)$ are hard for a propositional proof system P, can be interpreted as: proving circuit complexity lower bounds are hard in the proof system P. We find these ideas very related to ours.

Finally we note that in this work we also prove a (folklore) result that strong enough exponential lower bounds for depth-4 circuits imply exponential lower bounds for general circuits (see Proposition 2.7). A beautiful recent work of Agrawal and Vinay [1] (using [37]) shows the stronger result that *any* exponential lower bound for depth-4 circuits implies exponential lower bounds for general circuits. A very recent work [26] proves that extremely strong lower bounds for depth-3 formulas imply super-polynomial lower bounds for general formulas.

1.8 Our results

We partition our results about the connections between polynomial mappings and lower bounds for arithmetic circuits into two groups: Results for polynomial mappings f that elude polynomial mappings Γ of degree 2, and results for polynomial mappings f that elude polynomial mappings Γ of degree larger than 2.

We will first prove our results for polynomial mappings f that elude polynomial mappings Γ of degree larger than 2. Then, using these results, we will prove our lower bounds for bounded-depth arithmetic circuits. Finally, we will prove our results for polynomial mappings f that elude polynomial mappings Γ of degree 2.

Recall that given a polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$, and given a field extension $\mathbb{G} \supset \mathbb{F}$, we can think of f as a polynomial mapping $f : \mathbb{G}^n \to \mathbb{G}^m$. This is because we assume that a polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$ is given as a tuple $(f_1, \ldots, f_m)$ of polynomials in $\mathbb{F}[x_1, \ldots, x_n] \subset \mathbb{G}[x_1, \ldots, x_n]$ (see the discussion in Subsection 1.4).

1.8.1 Arithmetic circuits and polynomial mappings: Part I

We can now present our main results for polynomial mappings f that elude polynomial mappings Γ of degree larger than 2. The results are given by five propositions and corollaries. The full results are restated and proved in Subsection 3.4. For more details, see Section 3.

Let $\mathbb{F}$ be a field, and let n be an integer. By r, s, s', m, we denote integers, and we think of all these parameters as functions of the basic parameter n.

Proposition 1.4. *For all integers $2 \le r \le n \le s'$, and $m = n \cdot \binom{n+r-1}{r}$, there exists a polynomial mapping (described in Proposition 3.3), $\Gamma : \mathbb{F}^s \to \mathbb{F}^m$, (where $s = O((s')^2 \cdot r^8)$), of degree $2r-1$, such that the following holds: Let $f : \mathbb{F}^n \to \mathbb{F}^m$ be a polynomial mapping. If over some field extension $\mathbb{G} \supseteq \mathbb{F}$,*

$$\mathrm{Image}(f : \mathbb{G}^n \to \mathbb{G}^m) \not\subset \mathrm{Image}(\Gamma : \mathbb{G}^s \to \mathbb{G}^m)\,,$$

then any arithmetic circuit (over $\mathbb{F}$) for a polynomial $\tilde{f} : \mathbb{F}^{3n} \to \mathbb{F}$ (explicitly defined from f in Subsection 3.3), is of size $> s'/5$.

Moreover, one can construct Γ in time $\mathrm{poly}(s^r)$ in the following sense. There exists a $\mathrm{poly}(s^r)$-time Turing machine, that on input n, r, s', outputs (all the coefficients of) the m polynomials

$$(\Gamma_G)_1, \ldots, (\Gamma_G)_m \in \mathbb{F}[y_1, \ldots, y_s]\,,$$

and such that all the coefficients in these polynomials are integers[5] *in $\{0, \ldots, (3r)!\}$.*

Corollary 1.5. *Let $2 \le r \le n \le s$, and $m = n \cdot \binom{n+r-1}{r}$ be integers. Let $f : \mathbb{F}^n \to \mathbb{F}^m$ be a polynomial mapping. If over some field extension $\mathbb{G} \supseteq \mathbb{F}$, f is $(s, (2r-1))$-elusive (see Definition 1.1), then any arithmetic circuit (over $\mathbb{F}$) for a polynomial $\tilde{f} : \mathbb{F}^{3n} \to \mathbb{F}$ (explicitly defined from f in Subsection 3.3), is of size $\ge \Omega(\sqrt{s}/r^4)$.*

Corollary 1.6. *Let $\mathbb{F}$ be a field of characteristic not equal to 2. Let $2 \le r \le n \le s$, and $m = n \cdot \binom{n+r-1}{r}$ be integers such that $s = n^{\omega(1)}$. If there exists a $\mathrm{poly}(n)$-definable polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$ such that, over some field extension $\mathbb{G} \supseteq \mathbb{F}$, f is $(s, (2r-1))$-elusive (see Definition 1.3 and Definition 1.1), then any arithmetic circuit for the permanent over $\mathbb{F}$ is of size $\ge s^{\Omega(1)}$.*

Corollary 1.7. *Let $2 \le r \le n \le s$, and $m = n \cdot \binom{n+r-1}{r}$ be integers (and recall that we think of r, s, m as functions of n). Assume that there exists a $\mathrm{poly}(s^r)$-time Turing machine T such that:*

- *The inputs for T are r, n, s, m and a polynomial mapping $\Gamma : \mathbb{F}^s \to \mathbb{F}^m$ of degree $2r-1$, given by all coefficients of the polynomials $\Gamma_1, \ldots, \Gamma_m$ (that are assumed to be integers in, say, $\{0, \ldots, (3r)!\}$).*

[5] We think of the integers as members of every field, by the inductive definition $n = (n-1)+1$.

- *The output of T is a* $\mathrm{poly}(n)$*-definable polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$ (described, e. g., by an arithmetic circuit for the polynomial g that defines it in Definition 1.3) such that, over some field extension $\mathbb{G} \supseteq \mathbb{F}$,*
$$\mathrm{Image}(f : \mathbb{G}^n \to \mathbb{G}^m) \not\subset \mathrm{Image}(\Gamma : \mathbb{G}^s \to \mathbb{G}^m)\,.$$

Then, there exists a $\mathrm{poly}(s^r)$*-time Turing machine that on input n outputs a $3n$-variables,* $\mathrm{poly}(n)$*-definable, polynomial $\tilde{f}$ (explicitly defined from f in Subsection 3.3, and described, e. g., by an arithmetic circuit for the polynomial g that defines it in Definition 1.3), such that any arithmetic circuit for $\tilde{f}$ is of size $\geq \Omega(\sqrt{s}/r^4)$.*

Proposition 1.8 gives the connection for arithmetic circuits of depth d, and is the one used to prove our lower bounds for bounded-depth arithmetic circuits.

Proposition 1.8. *Let $n, d \leq s$, and $m = n^2$ be integers. Let $f : \mathbb{F}^n \to \mathbb{F}^m$ be a polynomial mapping. If over some field extension $\mathbb{G} \supseteq \mathbb{F}$, f is (s,d)-elusive (see Definition 1.1), then any depth-d arithmetic circuit (over $\mathbb{F}$) for the n polynomials $\tilde{f}_1, \ldots, \tilde{f}_n : \mathbb{F}^{2n} \to \mathbb{F}$ (explicitly defined from f in Subsection 3.3), is of size $> s$. (Moreover, the degree of each $\tilde{f}_i$ is at most the degree of f plus 1.)*

1.8.2 Lower bounds for bounded-depth circuits

We can now state our lower bounds for bounded-depth arithmetic circuits. We give an explicit construction for an (s,d)-elusive polynomial mapping, with certain parameters s, d. We then use Proposition 1.8 to obtain lower bounds for the size of arithmetic circuits of depth d. The full results are restated and proved in Section 4.

For an integer k, denote by $[k]$ the set $\{1, \ldots, k\}$. Let n be a prime. Let $m = n^2$. Let $1 \leq d \leq (\log_2 n)/100$ be an integer. Let $d' = 5d$. Let $\{x_{i,j}\}_{i \in [d'], j \in [n]}$ be a set of $n \cdot d'$ input variables. For every $(a,b) \in [n] \times [n]$, define
$$f_{(a,b)}(x_{1,1}, \ldots, x_{d',n}) = \prod_{i \in [d']} x_{i, a+i\cdot b}$$
(where the sum $a + i \cdot b$ is taken modulo n).

Let $f = (f_{(1,1)}, f_{(1,2)}, \ldots, f_{(n,n)})$. Note that for every field $\mathbb{G}$, we can view f as a polynomial mapping $f : \mathbb{G}^{n \cdot d'} \to \mathbb{G}^m$.

Lemma 1.9. *Let n be a prime, and let $m = n^2$. Let d be an integer such that $1 \leq d \leq (\log_2 n)/100$. Let $d' = 5d$. Let $\mathbb{G}$ be a field of size larger than m (e. g., an infinite field). Then, the polynomial mapping $f : \mathbb{G}^{n \cdot d'} \to \mathbb{G}^m$ (as defined above) is (s,d)-elusive (see Definition 1.1), where $s = \lfloor n^{1+1/(2d)} \rfloor$.*

Let $\{z_1, \ldots, z_n\}$, $\{w_1, \ldots, w_n\}$, be two sets of input variables. Define, for every $a \in [n]$,
$$\tilde{f}_a = \sum_{b \in [n]} z_b \cdot f_{(a,b)}\,.$$

Define
$$\tilde{f} = \sum_{a \in [n]} w_a \cdot \tilde{f}_a\,.$$

Note that every $\tilde{f}_a$ is a polynomial in $n \cdot (d'+1)$ variables, and is of total-degree $d'+1$, and $\tilde{f}$ is a polynomial in $n \cdot (d'+2)$ variables, and is of total-degree $d'+2$.

Corollary 1.10. *Let n be a prime, and let $1 \le d \le (\log_2 n)/100$ be an integer. Any depth-d arithmetic circuit, over any field $\mathbb{F}$, for the n polynomials (of total-degree $5d+1$ each) $\tilde{f}_1, \ldots, \tilde{f}_n : \mathbb{F}^{n\cdot(5d+1)} \to \mathbb{F}$, (as defined above), is of size $\ge n^{1+1/(2d)}$.*

Corollary 1.11. *Let n be a prime, and let $1 \le d \le (\log_2 n)/100$ be an integer. Any depth-$\lfloor d/3 \rfloor$ arithmetic circuit, over any field $\mathbb{F}$, for the polynomial (of total-degree $5d+2$) $\tilde{f} : \mathbb{F}^{n\cdot(5d+2)} \to \mathbb{F}$, (as defined above), is of size $\ge n^{1+1/(2d)}/5$.*

1.8.3 Arithmetic circuits and polynomial mappings: Part II

We will now present our main results for polynomial mappings f that elude polynomial mappings Γ of degree 2. The results are given by four propositions and corollaries. The full results are restated and proved in Subsection 5.4. For more details, see Section 5.

Let $\mathbb{F}$ be a field, and let n be an integer. By r, s, s', m, we denote integers, and we think of all these parameters as functions of the basic parameter n.

Proposition 1.12. *For all integers $3 \le r \le n \le s'$, and $m = \binom{n+r-1}{r}$, and $r' = \lfloor 2r/3 \rfloor$, there exists a polynomial mapping (described in Proposition 5.3), $\Gamma : \mathbb{F}^s \to \mathbb{F}^m$ (where $s = O(s' \cdot \binom{n+r'-1}{r'} \cdot r^3)$), of degree 2, such that the following holds: Let $f : \mathbb{F}^n \to \mathbb{F}^m$ be a polynomial mapping. If over some field extension $\mathbb{G} \supseteq \mathbb{F}$,*

$$\mathrm{Image}(f : \mathbb{G}^n \to \mathbb{G}^m) \not\subseteq \mathrm{Image}(\Gamma : \mathbb{G}^s \to \mathbb{G}^m),$$

then any arithmetic circuit (over $\mathbb{F}$) for the polynomial $\tilde{f} : \mathbb{F}^{2n} \to \mathbb{F}$ (explicitly defined from f in Subsection 5.3), is of size $> s'$.

Moreover, one can construct Γ in time $\mathrm{poly}(s,m)$ in the following sense. There exists a $\mathrm{poly}(s,m)$-time Turing machine, that on input n, r, s', outputs (all the coefficients of) the m polynomials

$$(\Gamma_G)_1, \ldots, (\Gamma_G)_m \in \mathbb{F}[y_1, \ldots, y_s],$$

and such that all the coefficients in these polynomials are in $\{0,1\}$.

Corollary 1.13. *Let $3 \le r \le n \le s$, and $m = \binom{n+r-1}{r}$ be integers. Let $r' = \lfloor 2r/3 \rfloor$. Let $f : \mathbb{F}^n \to \mathbb{F}^m$ be a polynomial mapping. If over some field extension $\mathbb{G} \supseteq \mathbb{F}$, f is $(s,2)$-elusive (see Definition 1.1), then any arithmetic circuit (over $\mathbb{F}$) for the polynomial $\tilde{f} : \mathbb{F}^{2n} \to \mathbb{F}$ (explicitly defined from f in Subsection 5.3), is of size $\ge$*

$$\Omega\left(\frac{s}{\binom{n+r'-1}{r'} \cdot r^3}\right).$$

Corollary 1.14. *Let $\mathbb{F}$ be a field of characteristic not equal to 2. Let $3 \le r \le n \le s$, and $m = \binom{n+r-1}{r}$ be integers. Let $r' = \lfloor 2r/3 \rfloor$. Assume that $s/\binom{n+r'-1}{r'} \ge n^{\omega(1)}$. If there exists a $\mathrm{poly}(n)$-definable polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$ such that, over some field extension $\mathbb{G} \supseteq \mathbb{F}$, f is $(s,2)$-elusive (see Definition 1.3 and Definition 1.1), then any arithmetic circuit for the permanent over $\mathbb{F}$ is of size $\ge$*

$$\left(\frac{s}{\binom{n+r'-1}{r'} \cdot r^3}\right)^{\Omega(1)}.$$

Corollary 1.15. *Let $3 \le r \le n \le s$, and $m = \binom{n+r-1}{r}$ be integers (and recall that we think of r,s,m as functions of n). Let $r' = \lfloor 2r/3 \rfloor$. Assume that there exists a $\mathrm{poly}(s,m)$-time Turing machine T such that:*

- *The inputs for T are r,n,s,m and a polynomial mapping $\Gamma : \mathbb{F}^s \to \mathbb{F}^m$ of degree 2, given by all coefficients of the polynomials $\Gamma_1,\dots,\Gamma_m$ (that are assumed to be in $\{0,1\}$).*
- *The output of T is a $\mathrm{poly}(n)$-definable polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$ (described, e. g., by an arithmetic circuit for the polynomial g that defines it in Definition 1.3) such that, over some field extension $\mathbb{G} \supseteq \mathbb{F}$,*

$$\mathrm{Image}(f : \mathbb{G}^n \to \mathbb{G}^m) \not\subseteq \mathrm{Image}(\Gamma : \mathbb{G}^s \to \mathbb{G}^m)\,.$$

Then, there exists a $\mathrm{poly}(s,m)$-time Turing machine that on input n outputs a $2n$-variables, $\mathrm{poly}(n)$-definable, polynomial $\tilde{f}$ (explicitly defined from f in Subsection 5.3, and described, e. g., by an arithmetic circuit for the polynomial g that defines it in Definition 1.3), such that any arithmetic circuit for $\tilde{f}$ is of size $\ge$

$$\Omega\left(\frac{s}{\binom{n+r'-1}{r'}\cdot r^3}\right).$$

2 Arithmetic circuits in normal forms

Definition 2.1 (Circuit-Graph)**.** Let Φ be an arithmetic circuit. We denote by G_Φ the underlying graph of Φ, together with the labels of all nodes. That is, the entire circuit, except for the labels of the edges. We call G_Φ, the circuit-graph of Φ.

We use for a circuit-graph G the same terminology as we use for circuits. For example, the size of G is the number of edges in G, and is denoted by $\mathrm{Size}(G)$, and the depth of G is the length of the longest directed path in G, and is denoted by $\mathrm{Depth}(G)$.

Note that different arithmetic circuits, over different fields, can have the same circuit-graph.

For a circuit-graph G, we define the *syntactic-degree* of a node in G, inductively, as follows. The syntactic-degree of a leaf is 0 if the leaf is labelled by the field element 1, and 1 if the leaf is labelled by an input variable. The syntactic-degree of a sum-gate is the maximum of the syntactic-degrees of its children. The syntactic-degree of a product-gate is the sum of the syntactic-degrees of its children.

For an arithmetic circuit Φ and a node v in Φ, we define the syntactic-degree of v to be its syntactic-degree in the circuit-graph G_Φ. The degree of a circuit is the maximal syntactic-degree of a node in the circuit.

2.1 Homogenization

A polynomial g is called *homogenous*, if all the monomials that occur in g (with coefficients not equal to 0) have the same total-degree.

We say that a circuit-graph G is *homogenous* if for every sum-gate v in G, the syntactic-degree of every child of v is the same. We say that G is homogenous of degree r if it is homogenous, and all output-gates in G are of syntactic-degree exactly r. We say that an arithmetic circuit Φ is *homogenous* if the circuit-graph G_Φ is homogenous.

Note that a circuit-graph G is homogenous iff for every arithmetic circuit Φ (over any field) such that $G = G_\Phi$, and every gate v in Φ, the polynomial computed by the gate v is homogenous.

Definition 2.2 (Normal-Homogenous-Form)**.** Let G be a homogenous circuit-graph. We say that G is in normal-homogenous-form if it satisfies:

1. All leaves are labelled by input variables (i. e., no leaf is labelled by 1).
2. All edges from the leaves are to sum-gates.
3. All output-gates are sum-gates.
4. The gates of G are alternating. That is, if v is a product-gate and (u,v) is an edge then u is a sum-gate, and if v is a sum-gate and (u,v) is an edge then u is either a leaf or a product-gate.
5. The in-degree of every product-gate is exactly 2.
6. The out-degree of every sum-gate is at most 1.

We say that an arithmetic circuit Φ is in normal-homogenous-form if the circuit-graph G_Φ is in normal-homogenous-form.

Proposition 2.3. *Let $\mathbb{F}$ be a field. Let Φ be an arithmetic circuit of size s, for n homogenous polynomials $g_1,\ldots,g_n \in \mathbb{F}[x_1,\ldots,x_n]$ of total-degree $r \geq 1$ each. Then, there exists an arithmetic circuit Ψ, for the polynomials $g_1,\ldots,g_n$, such that Ψ is in normal-homogenous-form, and the number of nodes in Ψ is $O(s \cdot r^2)$. Moreover, given Φ (as an input), Ψ can be efficiently constructed.*

Proof. Note that the number of nodes in Φ is $O(s)$. We assume without loss of generality that $n < s$. We will describe an algorithm that changes Φ into the required Ψ. For simplicity, we describe all steps of the algorithm without dealing with the labels of the edges. It is straightforward to verify that in all steps the labels of the edges can be fixed so that the functionality of the circuit is preserved.

We can assume without loss of generality that no (sum or product) gate in Φ is of in-degree 1 (otherwise, we just remove such a gate and connect its only child directly to all its parents). For convenience, we make sure that the in-degree of every (sum or product) gate is exactly 2. This is done by replacing any product-gate of in-degree larger than 2 by a tree of product-gates of in-degree 2, and any sum-gate of in-degree larger than 2 by a tree of sum-gates of in-degree 2. This increases the number of nodes in the circuit by at most s.

Next, we homogenize the circuit. For a polynomial $g \in \mathbb{F}[x_1,\ldots,x_n]$ and an integer i, we define the *homogenous part of degree i* of g to be the restriction of g to the set of monomials of total-degree exactly i. For every node v and every $i \in \{0,\ldots,r\}$, we "split" the node v into $r+1$ nodes $v_0,\ldots,v_r$, where the node v_i computes the homogenous part of degree i of the polynomial computed by the node v. (Note that we just ignore monomials of degree larger than r everywhere in the circuit, as they do not contribute to the functionality of the circuit). Formally, this is done inductively on the circuit. For every node v with children u,w, we add a (homogenous) arithmetic circuit with at most $O(r^2)$ nodes, that computes $v_0,\ldots,v_r$ from $u_0,\ldots,u_r$ and $w_0,\ldots,w_r$. We make sure that the in-degree of every product-gate is still exactly 2. More precisely, if v is a sum-gate, for every $i \in \{0,\ldots,r\}$ the circuit that we add computes $v_i = u_i + w_i$, and if v is a product-gate, for every $i \in \{0,\ldots,r\}$ the circuit that we add computes $v_i = \sum_{j=0}^{i} u_j \times w_{i-j}$.

Altogether, by the end of this step, we obtained a homogenous arithmetic circuit, with product-gates of in-degree exactly 2, with at most $O(s \cdot r^2)$ nodes.

Next, we remove every node of syntactic-degree 0. This is done as follows. Let u be a node of syntactic-degree 0. If u is of out-degree 0, we can just remove it as it cannot contribute to the functionality of the circuit. Otherwise, there is an edge (u,v). If v is a sum-gate then, since the circuit is now homogenous, v is of syntactic-degree 0. Thus v just computes a field element α. So we can just replace v by a leaf labelled by 1 and multiply the labels of all the edges from v by α. If v is a product-gate, with other child w, we can remove the gate v and connect w directly to all parents of v. By repeating this process as many times as needed, we remove all nodes of syntactic-degree 0, and in particular, all leaves labelled by 1.

Altogether, by the end of this step, we obtained a homogenous arithmetic circuit, with product-gates of in-degree exactly 2, with no leaves labelled by 1, and with at most $O(s \cdot r^2)$ nodes.

Next, we ensure that the gates are alternating. This is done as follows. For any edge (u,v) such that u,v are both product-gates, we add a dummy sum-gate in between them. Note that since the in-degree of every product-gate is 2, this at most triples the number of nodes in the circuit. For any edge (u,v) such that u,v are both sum-gates, we connect all children of u directly to v and remove the edge (u,v). This doesn't increase the number of nodes. By repeating this as many times as needed, we obtain a circuit with alternating gates.

Altogether, by the end of this step, we obtained a homogenous arithmetic circuit, with product-gates of in-degree exactly 2, with no leaves labelled by 1, with alternating gates, and with at most $O(s \cdot r^2)$ nodes.

Next, we connect any product output-gate to a (different) dummy sum-gate. This at most doubles the number of nodes in the circuit.

Next, for any edge from a leaf to a product-gate, we add a dummy sum-gate in between them. Note that since the in-degree of every product-gate is 2, this at most triples the number of nodes in the circuit.

Next, we ensure that the out-degree of every sum-gate is at most 1. This is done by duplicating q times any sum-gate of out-degree $q > 1$. Note that since every edge from a sum-gate reaches a product-gate, and since the in-degree of every product-gate is 2, this at most triples the number of nodes in the circuit.

Finally, we can remove all nodes of out-degree 0 that do not output one of the polynomials $g_1,\ldots,g_n$. We repeat this as many times as needed.

Altogether, we obtain a circuit in normal-homogenous-form, with at most $O(s \cdot r^2)$ nodes. □

2.2 Linearization

A polynomial g is called *linear*, if it is homogenous of degree 1, that is, if all the monomials that occur in g (with coefficients not equal to 0) are of total-degree *exactly* 1.

Definition 2.4 (Normal-Linear-Form)**.** Let G be a homogenous circuit-graph. We say that G is in normal-linear-form if it satisfies:

1. All nodes in G are either leaves or sum-gates (i. e., there are no product-gates).

2. All leaves are labelled by input variables (i. e., no leaf is labelled by 1).

Note that this implies that the syntactic-degree of every node v in G is exactly 1. We say that an arithmetic circuit Φ is in normal-linear-form if the circuit-graph G_Φ is in normal-linear-form.

Proposition 2.5. *Let $\mathbb{F}$ be a field. Let Φ be an arithmetic circuit of size s and depth d, for n linear polynomials $g_1,\dots,g_n \in \mathbb{F}[x_1,\dots,x_n]$. Then, there exists an arithmetic circuit Ψ, of size s and depth d, for the polynomials $g_1,\dots,g_n$, such that Ψ is in normal-linear-form. Moreover, given Φ (as an input), Ψ can be efficiently constructed.*

Proof. We will describe an algorithm that changes Φ into the required Ψ.

For a polynomial $g \in \mathbb{F}[x_1,\dots,x_n]$, we define the *linear part* of g to be the restriction of g to the set of monomials of total-degree exactly 1. For every node v in Φ, we define a node v' in Ψ that computes the linear part of the polynomial computed by the node v.

Formally, this is done inductively on the circuit. For a sum-gate v with children $v_1,\dots,v_k$, we define v' to be a sum-gate with children $v'_1,\dots,v'_k$, and label an edge (v'_i,v') with the same field element that labels (v_i,v). For a product-gate v with children $v_1,\dots,v_k$, note that if the linear parts of the polynomials computed by $v_1,\dots,v_k$ are $h_1,\dots,h_k$ (respectively), then the linear part of the polynomial computed by v can be written as $\sum_{i=1}^k c_i h_i$, for some field elements $c_1,\dots,c_k$. Thus, once again, we define v' to be a sum-gate with children $v'_1,\dots,v'_k$, and we label an edge (v'_i,v') by c_i. □

2.3 Reduction to depth 4

We will now define circuit-graphs and arithmetic circuits in a *normal-depth-4-form*. This form will only be used in Section 5. Roughly speaking, the computation of an arithmetic circuit in normal-depth-4-form can be presented as a homogenous sum,[6] $\sum_i P_i Q_i$, where P_i, Q_i are homogenous polynomials of degree at most $2r/3$, where r is the syntactic-degree of the output-gate.

Definition 2.6 (Normal-Depth-4-Form)**.** Let G be a homogenous circuit-graph. We say that G is in normal-depth-4-form if it satisfies:

1. The out-degree of every gate in G is at most 1 (i. e., G is a tree).
2. There is a single output-gate. The output-gate is a sum-gate.
3. Every directed path from a leaf to the output-gate is of length exactly 4.
4. All edges from the leaves are to product-gates. These product-gates (in the level above the leaves) are of syntactic-degree at most $2r/3$, where r is the syntactic-degree of the output-gate.
5. The gates of G are alternating. That is, if v is a sum-gate and (u,v) is an edge then u is a product-gate, and if v is a product-gate and (u,v) is an edge then u is either a leaf or a sum-gate.
6. The in-degree of every product-gate, which is a child of the output-gate, is exactly 2.

[6] A *homogenous sum* is a sum of homogenous polynomials, where all the non-zero polynomials in the sum are of the same degree.

We say that an arithmetic circuit Φ is in normal-depth-4-form if the circuit-graph G_Φ is in normal-depth-4-form.

The following proposition is based on a lemma from [28]. (Other forms of it follow from several previous works.) Note that the proposition implies that strong enough exponential lower bounds for depth-4 circuits imply exponential lower bounds for general circuits. A recent work of Agrawal and Vinay [1] (using [37]) shows the stronger result that *any* exponential lower bound for depth-4 circuits implies exponential lower bounds for general circuits.

Proposition 2.7. *Let $\mathbb{F}$ be a field. Let Φ be an arithmetic circuit of size s, for a homogenous polynomial $g \in \mathbb{F}[x_1,\dots,x_n]$ of total-degree $r \geq 3$. Let $r' = \lfloor 2r/3 \rfloor$. Then, there exists an arithmetic circuit Ψ, for the polynomial g, such that Ψ is in normal-depth-4-form, and is of size $O(s \cdot \binom{n+r'-1}{r'} \cdot r^3)$ (and has $O(s \cdot \binom{n+r'-1}{r'} \cdot r^2)$ product-gates). Moreover, given Φ (as an input), Ψ can be efficiently constructed (in time polynomial in the size of Ψ).*

Proof. We will describe an algorithm that constructs Ψ from Φ.

First, we make sure that the in-degree of every gate in Φ is at most 2. This is done by replacing any gate of in-degree larger than 2 by a tree of gates of in-degree 2. This increases the number of edges in the circuit by a factor of at most 2.

Following this, we homogenize the circuit (as in the proof of Proposition 2.3). For a polynomial $h \in \mathbb{F}[x_1,\dots,x_n]$ and an integer i, we define the *homogenous part of degree i of h* to be the restriction of h to the set of monomials of total-degree exactly i. For every node v and every $i \in \{0,\dots,r\}$, we "split" the node v into $r+1$ nodes $v_0,\dots,v_r$, where the node v_i computes the homogenous part of degree i of the polynomial computed by the node v. (Note that we just ignore monomials of degree larger than r everywhere in the circuit, as they do not contribute to the functionality of the circuit.) Formally, this is done inductively on the circuit. For every node v with children u,w, we add a (homogenous) arithmetic circuit with at most $O(r^2)$ edges, that computes $v_0,\dots,v_r$ from $u_0,\dots,u_r$ and $w_0,\dots,w_r$. We make sure that the in-degree of every gate is still at most 2. More precisely, if v is a sum-gate, for every $i \in \{0,\dots,r\}$ the circuit that we add computes $v_i = u_i + w_i$, and if v is a product-gate, for every $i \in \{0,\dots,r\}$ the circuit that we add computes $v_i = \sum_{j=0}^{i} u_j \times w_{i-j}$.

Altogether, by the end of this step, we obtained a homogenous arithmetic circuit, with gates of in-degree at most 2, and with at most $O(s \cdot r^2)$ edges. Denote this circuit by Φ', and its size by $s' = O(s \cdot r^2)$. Note that since the circuit is homogenous, and its output-gate computes the polynomial g of degree r, the syntactic-degree of the output-gate is r. (Formally, we can prove by induction on the nodes of a circuit, that any node u in a homogenous circuit computes either the 0 polynomial or a homogenous polynomial of degree exactly equal to the syntactic-degree of u.)

We will now show how to present the polynomial g as a homogenous sum, $g = \sum_i P_i Q_i$, where P_i, Q_i are homogenous polynomials of degree at most $2r/3$, and the sum is over at most s' elements. This is done by induction on s'.

First note that in the circuit Φ' there is at least one node u of syntactic-degree larger than $r/3$ and smaller or equal to $2r/3$. This is true because one can start from the output-gate (which is of syntactic-degree r) and move in each step from a node v to its child u of larger syntactic-degree. Since the in-degree of every gate v is at most 2, the syntactic-degree of u is at least a half of the syntactic-degree of v. Hence, at some point along the way, we reach a node u of syntactic-degree larger than $r/3$ and smaller or equal to $2r/3$. Let u be such a node.

Denote by $P \in \mathbb{F}[x_1,\dots,x_n]$ the polynomial computed by the node u of Φ'. Assume without loss of generality that P is not the 0 polynomial (otherwise, we can remove the node u and obtain a smaller circuit, with the same properties, that still computes the polynomial g, and we can continue by induction). Thus, P is a homogenous polynomial of degree larger than $r/3$ and smaller or equal to $2r/3$.

Let y be an additional input variable. Denote by $\Phi'_{u=y}$, the circuit Φ', after replacing the node u by the input variable y (i. e., we change the label of u to be y and we remove every edge from a child of u to u). Denote by $\Phi'_{u=0}$, the circuit Φ', after fixing the node u to 0 (i. e., we remove u and all edges connected to it, and we fix to 0, inductively, all the product-gate parents of u and all the sum-gates that are left without children).

Note that $\Phi'_{u=0}$ is a homogenous arithmetic circuit (formally, this is proved by induction on the nodes of the circuit, by showing, inductively, that every node in $\Phi'_{u=0}$ is of the same syntactic-degree as the corresponding node in Φ'), with gates of in-degree at most 2, and with less than s' edges.

The circuit $\Phi'_{u=y}$ computes a polynomial $g' \in \mathbb{F}[x_1,\dots,x_n,y]$. Since the syntactic-degree of u in Φ' is larger than $r/3$, and since the output of Φ' is of syntactic-degree r, the degree of y in the polynomial g' is at most 2. Hence, we can present g' as a sum,

$$g' = g_0 + g_1 \cdot y + g_2 \cdot y^2\,,$$

where $g_0, g_1, g_2 \in \mathbb{F}[x_1,\dots,x_n]$.

By the definition of $\Phi'_{u=0}$, the polynomial computed by $\Phi'_{u=0}$ is g_0. Hence, g_0 is either the 0 polynomial or a homogenous polynomial of degree r. (Formally, we can show by induction on the nodes of $\Phi'_{u=0}$ that every node in $\Phi'_{u=0}$ computes either the 0 polynomial or a homogenous polynomial of degree equals to the syntactic-degree of the corresponding node in Φ'.)

By the definition of $\Phi'_{u=y}$ and by the definition of P, we know that

$$g = g_0 + g_1 \cdot P + g_2 \cdot P^2 = g_0 + P \cdot Q\,,$$

where $Q = g_1 + g_2 \cdot P$. Note that since both g, g_0 are either the 0 polynomial or homogenous polynomials of degree r, the polynomial $P \cdot Q = g - g_0$ is also either the 0 polynomial or a homogenous polynomial of degree r. Thus, the sum $g = g_0 + P \cdot Q$ is a homogenous sum. Also, since P and $P \cdot Q$ are both homogenous, Q is also homogenous, and since $P \cdot Q$ is of degree r (unless it is the 0 polynomial), and P is of degree larger than $r/3$ and smaller or equal to $2r/3$, we conclude that Q is either the 0 polynomial or is of degree larger or equal to $r/3$ and smaller than $2r/3$. Thus, both P, Q are homogenous polynomials of degree at most $2r/3$.

Since g_0 is the polynomial computed by the circuit $\Phi'_{u=0}$, by induction, g_0 can be presented as a homogenous sum, $g_0 = \sum_i P_i Q_i$, where P_i, Q_i are homogenous polynomials of degree at most $2r/3$, and the sum is over at most $s'-1$ elements. Hence, g can be presented as a homogenous sum, $g = \sum_i P_i Q_i$, where P_i, Q_i are homogenous polynomials of degree at most $2r/3$, and the sum is over at most s' elements.

The presentation of g as a homogenous sum, $g = \sum_i P_i Q_i$, where P_i, Q_i are homogenous polynomials of degree at most $2r/3$, and the sum is over at most s' elements, gives a homogenous circuit Ψ, for the polynomial g, such that Ψ is in normal-depth-4-form, and is of size $O(s \cdot \binom{n+r'-1}{r'} \cdot r^3)$ (and has $O(s \cdot \binom{n+r'-1}{r'} \cdot r^2)$ product-gates). To obtain Ψ, we just have to present every polynomial P_i, Q_i, as a sum of monomials. (Note that since the degree of the polynomials P_i, Q_i is at most r', their presentations as sums of monomials are by arithmetic circuits of size $O(\binom{n+r'-1}{r'} \cdot r')$, and recall that $s' = O(s \cdot r^2)$.) $\square$

2.4 Universal circuit-graphs

Proposition 2.8. *For any integers $n, s, r \geq 1$ such that $s \geq n$, there is a circuit-graph G, in normal-homogenous-form and with at most $O(s \cdot r^4)$ nodes, that is universal for n-inputs and n-outputs circuits of size s that compute homogenous polynomials of degree r, in the following sense:*

Let $\mathbb{F}$ be a field. Let Φ be an arithmetic circuit of size s, for n homogenous polynomials $g_1, \ldots, g_n \in \mathbb{F}[x_1, \ldots, x_n]$ of total-degree r each. Then, there exists an arithmetic circuit Ψ, for the polynomials $g_1, \ldots, g_n$, such that $G_\Psi = G$.

Moreover, given n, s, r, the circuit-graph G can be constructed in time $\mathrm{poly}(s, r)$.

Proof. Let $\mathbb{F}$ be a field. Let Φ be an arithmetic circuit of size s, for n homogenous polynomials $g_1, \ldots, g_n \in \mathbb{F}[x_1, \ldots, x_n]$ of total-degree r. By Proposition 2.3, there exists an arithmetic circuit Ψ', for the polynomials $g_1, \ldots, g_n$, such that Ψ' is in normal-homogenous-form, and the number of nodes in Ψ' is $O(s \cdot r^2)$.

Since Ψ' is in normal-homogenous-form, the nodes of Ψ' are partitioned into $2r$ levels, according to the type of node (i. e., a leaf, a sum-gate, or a product-gate) and its syntactic-degree, as follows:

- Level-1 contains the leaves, and recall that all the leaves are labelled by variables. Without loss of generality, we can assume that every variable labels exactly one leaf.
- Level-2 contains the sum-gates of syntactic-degree 1.
- For every $i \in \{2, \ldots, r\}$, Level-$(2i-1)$ contains the product-gates of syntactic-degree i.
- For every $i \in \{2, \ldots, r\}$, Level-$(2i)$ contains the sum-gates of syntactic-degree i.
- The nodes in Level-$(2r)$ are the output-gates.

The children of every sum-gate in Level-$(2i)$ are nodes from Level-$(2i-1)$. Without loss of generality, we can assume that the children of every sum-gate in Level-$(2i)$ are all the nodes in Level-$(2i-1)$. That is, there is an edge between every node in Level-$(2i-1)$ and every node in Level-$(2i)$. Note that this doesn't increase the number of nodes.

Recall also that the out-degree of every sum-gate is at most 1, and the only sum-gates of out-degree 0 are the gates in Level-$(2r)$.

Every product-gate in Level-$(2i-1)$ has exactly two children, one is a sum-gate in Level-$(2j)$ (for some $0 < j < i$) and the other is a sum-gate in Level-$(2i-2j)$. Thus, we further partition the product-gates in Level-$(2i-1)$ into $i-1$ types (Type-1,...,Type-$(i-1)$), according to the identity of that j.

If we knew the number of sum-gates in each (even) level, and the number of product-gates of each type in each (odd) level, we could have constructed the circuit-graph $G_{\Psi'}$, as follows:

- Level-1 contains n leaves, labelled by the n input variables.

- The children of a sum-gate in Level-$(2i)$ are all the nodes in Level-$(2i-1)$.
- The two children of a product-gate of Type-j in Level-$(2i-1)$ are a sum-gate in Level-$(2j)$ and a sum-gate in Level-$(2i-2j)$. The exact identity of these two sum-gates is not important. Just pick arbitrary gates (in the right levels) that were still not used. They are all the same because they all have the exact same children and they are all of out-degree 1.

Thus, in the circuit-graph G, we just have to make sure that we have enough nodes of each type in each level. We can ensure that by having $O(s\cdot r^2)$ nodes of each type in each level, a total number of $O(s\cdot r^4)$ nodes. Since we have enough nodes of each type in each level, we can embed the circuit graph $G_{\Psi'}$ in G.

To construct the circuit Ψ, we use the circuit-graph G, and we just label by 0 every edge to or from a node that is not in $G_{\Psi'}$, and we label all other edges by their label in $G_{\Psi'}$. $\square$

The following proposition will only be used in Section 5.

Proposition 2.9. *For any integers $n,s,r\geq 3$ such that $s\geq n$, there is a circuit-graph G, in normal-depth-4-form and of size at most $O(s\cdot\binom{n+r'-1}{r'}\cdot r^4)$ (and with $O(s\cdot\binom{n+r'-1}{r'}\cdot r^3)$ product-gates), where $r'=\lfloor 2r/3\rfloor$, that is universal for n-inputs and one-output circuits of size s that compute homogenous polynomials of degree r, in the following sense:*

Let $\mathbb{F}$ be a field. Let Φ be an arithmetic circuit of size s, for a homogenous polynomial $g\in\mathbb{F}[x_1,\dots,x_n]$ of total-degree r. Then, there exists an arithmetic circuit Ψ, for the polynomial g, such that $G_\Psi=G$.

Moreover, given n,s,r, the circuit-graph G can be constructed in time $\mathrm{poly}(s,\binom{n+r'-1}{r'})$.

Proof. Let $\mathbb{F}$ be a field. Let Φ be an arithmetic circuit of size s, for a homogenous polynomial $g\in\mathbb{F}[x_1,\dots,x_n]$ of total-degree r. By Proposition 2.7, there exists an arithmetic circuit Ψ', for the polynomial g, such that Ψ' is in normal-depth-4-form, and is of size $O(s\cdot\binom{n+r'-1}{r'}\cdot r^3)$ (and has $O(s\cdot\binom{n+r'-1}{r'}\cdot r^2)$ product-gates).

The circuit Ψ' gives a presentation of g as a homogenous sum, $g=\sum_i P_iQ_i$, where P_i,Q_i are homogenous polynomials of degree at most r', and the sum is over at most $O(s\cdot r^2)$ elements (see the proof of Proposition 2.7).

Note that for every i, the sum of the degree of P_i (denoted, $\deg(P_i)$) and the degree of Q_i (denoted, $\deg(Q_i)$) is exactly r. We can hence partition the pairs (P_i,Q_i) into $O(r)$ types, acording to the degree of P_i. As in the proof of Proposition 2.8, if we knew the number of pairs (P_i,Q_i) of each type, we could have constructed the circuit-graph $G_{\Psi'}$. This is true because in Ψ' the polynomials P_i,Q_i are computed by a sum of all their monomials, and given $\deg(P_i),\deg(Q_i)$ this can be done by the same circuit-graph.

Thus, in the circuit-graph G, we just have to make sure that we have enough nodes that compute P_iQ_i of each of the $O(r)$ types. We can ensure that by having $O(s\cdot r^2)$ nodes of each type. Since we have enough nodes of each type, we can embed the circuit graph $G_{\Psi'}$ in the circuit-graph G.

To construct the circuit Ψ, we use the circuit-graph G, and we just label by 0 every edge to or from a node that is not in $G_{\Psi'}$, and we label all other edges by their label in $G_{\Psi'}$.

Note that the size of G is at most $O(r)\cdot O(s\cdot\binom{n+r'-1}{r'}\cdot r^3)=O(s\cdot\binom{n+r'-1}{r'}\cdot r^4)$ (and it has $O(s\cdot\binom{n+r'-1}{r'}\cdot r^3)$ product-gates). $\square$

3 Arithmetic circuits and polynomial mappings: Part I

In this section, we describe and prove our main results for polynomial mappings f that elude polynomial-mappings Γ of degree larger than 2. The main results appear in Subsection 3.4.

3.1 Notation

Let $\mathbb{F}$ be a field. Let n,r be integers. We fix m' to be the number of monomials of total-degree exactly r in n variables, that is, $m' = \binom{n+r-1}{r}$, and we fix $m = m' \cdot n$. Note that r is not necessarily a constant, and may be a function of n. In general, we think of all parameters as functions of the basic parameter n. We assume that $1 \leq r \leq n$, and we assume for simplicity that n is a power of 2.

For an integer k, denote by $[k]$ the set $\{1,\dots,k\}$, and denote by $\bar{k}$ the binary representation of $k-1$.

Let $Z = \{z_1,\dots,z_n\}$ be a set of n input variables. Let M be the set of all monomials of total-degree exactly r in the variables $\{z_1,\dots,z_n\}$. Note that $|M| = m'$. We can identify the set M with the set $[m']$, by the lexicographic order of monomials. Formally, let $h : M \to [m']$ be the lexicographic order of monomials. We can now identify the set $M \times [n]$ with the set $[m' \cdot n] = [m]$, by the bijection $(q,i) \to (h(q),i)$, where (here and later on) we think of $(h(q),i) \in [m'] \times [n]$ as an element of $[m' \cdot n] = [m]$ (by the lexicographic order).

We denote by $\mathcal{M}$ the set of all homogenous polynomials in $\mathbb{F}[Z]$ of total-degree exactly r. We identify the vector space $\mathcal{M} = \mathbb{F}^M$ with the vector space $\mathbb{F}^{[m']}$ (by the bijection h between the bases). We will consider tuples $(g_1,\dots,g_n) \in \mathcal{M}^n$ of n homogenous polynomials of total-degree exactly r. We identify the vector space $\mathcal{M}^n = \mathbb{F}^{M\times[n]}$ with the vector space $\mathbb{F}^m$ (by the bijection $(q,i) \to (h(q),i)$ between the bases). Formally, we denote this homomorphism by

$$H : \mathcal{M}^n \to \mathbb{F}^m .$$

Intuitively, this means that we think of a vector in $\mathbb{F}^m$ as a tuple of n polynomials in $\mathcal{M}$, and vice versa. Each coordinate of the vector in $\mathbb{F}^m$ corresponds to the coefficient of one monomial in one of the n polynomials.[7]

Denote by $\mathcal{G}_{n,r}$, the set of homogenous circuit-graphs G (see Section 2), of syntactic-degree r, over the set of input variables $Z = \{z_1,\dots,z_n\}$, such that G has exactly n output-gates, and all output-gates in G are sum-gates. For a circuit-graph G, denote by $S(G)$, the number of edges in G that reach sum-gates.

3.2 The polynomial mapping $\Gamma_G : \mathbb{F}^s \to \mathbb{F}^m$

Let $G \in \mathcal{G}_{n,r}$. That is, G is a homogenous circuit-graph, of syntactic-degree r, over the set of input variables $Z = \{z_1,\dots,z_n\}$, such that G has exactly n output-gates, and all output-gates in G are sum-gates. Denote, $s = S(G)$, that is, the number of edges in G that reach sum-gates. (Note that $s \geq n$.)

Let Φ be an arithmetic circuit over $\mathbb{F}$, with circuit-graph $G_\Phi = G$. Without loss of generality, we assume that in the circuit Φ, all edges that reach product-gates are labelled by 1 (otherwise, if an edge

[7] We consider a *tuple* of polynomials in $\mathcal{M}$, rather than a single polynomial, because it will improve the parameters in some of our results, by applying the result of Baur and Strassen [5] (see in the introduction). In Section 5, we work with a single polynomial, (which is somewhat simpler).

that reaches a product-gate is labelled by $\alpha \neq 1$, we just change its label to 1 and multiply the labels of all edges that leave that product-gate by α). Denote the labels of the s edges that reach sum-gates by $y_1, \ldots, y_s$.

The circuit Φ computes n homogenous polynomials in $\mathbb{F}[Z]$ of total-degree exactly r, (that is, a tuple of n polynomials in $\mathcal{M}$), where the coefficients in these polynomials depend on the labels $y_1, \ldots, y_s$. Since we think of a tuple of n polynomials in $\mathcal{M}$ as a point in $\mathbb{F}^m$, we obtain for every point $(y_1, \ldots, y_s) \in \mathbb{F}^s$, a point in $\mathbb{F}^m$.

Formally, we define a mapping $\Gamma_G : \mathbb{F}^s \to \mathbb{F}^m$, as follows. Given $y_1, \ldots, y_s \in \mathbb{F}$, let Φ be an arithmetic circuit over $\mathbb{F}$, with circuit-graph $G_\Phi = G$, such that the labels of all edges that reach product-gates in Φ are 1, and the labels of the s edges that reach sum-gates in Φ are $y_1, \ldots, y_s$. Denote the n polynomials computed by Φ by $g_1, \ldots, g_n \in \mathcal{M}$ (note that these polynomials depend on the labels $y_1, \ldots, y_s$). Define

$$\Gamma_G(y_1, \ldots, y_s) = H((g_1, \ldots, g_n)) .$$

Note that the n outputs of the circuit Φ can be viewed as polynomials in both $z_1, \ldots, z_n$ and $y_1, \ldots, y_s$. That is, we can think of $g_1, \ldots, g_n$ as polynomials in the input variables $z_1, \ldots, z_n$, with coefficients that are polynomials in the input variables $y_1, \ldots, y_s$. Therefore, the functions $(\Gamma_G)_1, \ldots, (\Gamma_G)_m$ are polynomials in $\mathbb{F}[y_1, \ldots, y_s]$. That is, Γ_G is a polynomial mapping. Moreover, it is straightforward to prove (formally, by induction on the circuit) that the polynomials $(\Gamma_G)_1, \ldots, (\Gamma_G)_m$ do not depend on the field $\mathbb{F}$, but only on its characteristic (intuitively, this is obvious because all the coefficients in these polynomials are derived by a sequence of sum and product operations on the constants 0,1, and are hence members of the minimal subfield of $\mathbb{F}$ that contains 0,1).

Proposition 3.1. *Let $G \in \mathcal{G}_{n,r}$. For every $g = (g_1, \ldots, g_n) \in \mathcal{M}^n$, we have: $H(g) \in \mathrm{Image}(\Gamma_G)$ iff there exists an arithmetic circuit Φ, (over $\mathbb{F}$), with $G_\Phi = G$, for the n polynomials $g_1, \ldots, g_n$.*

Proof. If $H(g) \in \mathrm{Image}(\Gamma_G)$ then obviously, by the definition of Γ_G, there exists an arithmetic circuit Φ, (over $\mathbb{F}$), with $G_\Phi = G$, for the n polynomials $g_1, \ldots, g_n$.

If there exists an arithmetic circuit Φ, (over $\mathbb{F}$), with $G_\Phi = G$, for the n polynomials $g_1, \ldots, g_n$, without loss of generality, we assume that in the circuit Φ, all edges that reach product-gates are labelled by 1 (otherwise, if an edge that reaches a product-gate is labelled by $\alpha \neq 1$, we just change its label to 1 and multiply the labels of all edges that leave that product-gate by α). Denote the labels of the s edges in Φ that reach sum-gates by $\alpha_1, \ldots, \alpha_s$. Then, by the definition of Γ_G, we have $\Gamma_G(\alpha_1, \ldots, \alpha_s) = H(g)$. $\square$

Proposition 3.2. *If $G \in \mathcal{G}_{n,r}$ is in normal-homogenous-form (see Definition 2.2), then the mapping $\Gamma_G : \mathbb{F}^s \to \mathbb{F}^m$ (where $s = S(G)$ and $m = \binom{n+r-1}{r} \cdot n$) is a (homogenous) polynomial mapping of degree $2r-1$.*

Moreover, given G, one can construct Γ_G in time $\mathrm{poly}(s^r)$ in the following sense. There exists a $\mathrm{poly}(s^r)$-time Turing machine, that on input G outputs (all the coefficients of) the m polynomials $(\Gamma_G)_1, \ldots, (\Gamma_G)_m \in \mathbb{F}[y_1, \ldots, y_s]$, and such that all the coefficients in these polynomials are integers[8] in $\{0, \ldots, (3r)!\}$.

Proof. Let Φ be an arithmetic circuit over $\mathbb{F}$, with circuit-graph $G_\Phi = G$, such that the labels of all edges that reach product-gates are 1, and the labels of the s edges that reach sum-gates are $y_1, \ldots, y_s$.

[8]Recall that we think of the integers as members of every field $\mathbb{F}$, by the inductive definition $n = (n-1)+1$.

For a node v in Φ, denote the polynomial (in the input variables Z), computed by the node v, by $g_v \in \mathbb{F}[Z]$, and denote by r_v the syntactic-degree of v. Note that if v is a leaf, all the coefficients in g_v are in $\{0,1\}$, and hence they do not depend on $y_1,\dots,y_s$. By induction, we show that if v is a sum-gate of syntactic-degree r_v, then every coefficient in the polynomial $g_v \in \mathbb{F}[Z]$ is a (homogenous) polynomial of degree $2r_v - 1$ in the labels $y_1,\dots,y_s$, and if v is a product-gate of syntactic-degree r_v, then every coefficient in the polynomial $g_v \in \mathbb{F}[Z]$ is a (homogenous) polynomial of degree $2r_v - 2$ in the labels $y_1,\dots,y_s$.

The proof is straightforward. If v is a product-gate, with children v_1, v_2 (that are sum-gates), then, by induction, the coefficients in the polynomials g_{v_1}, g_{v_2} are (homogenous) polynomials of degrees $2r_{v_1} - 1, 2r_{v_2} - 1$, respectively (in the labels $y_1,\dots,y_s$). Since the edges (v_1,v) and (v_2,v) are labelled by 1, the coefficients in the polynomial g_v are (homogenous) polynomials of degree $2r_{v_1} - 1 + 2r_{v_2} - 1 = 2r_v - 2$ (in the labels $y_1,\dots,y_s$).

If v is a sum-gate, then, by induction, the coefficients in the polynomial g_u, for every child u of v, are (homogenous) polynomials of degree $2r_u - 2 = 2r_v - 2$ (in the labels $y_1,\dots,y_s$). Since the edge (u,v) is labelled by an element of $\{y_1,\dots,y_s\}$, the coefficients in the polynomial g_v are (homogenous) polynomials of degree $2r_v - 1$ (in the labels $y_1,\dots,y_s$).

As for the moreover part, denote $Y = \{y_1,\dots,y_s\}$ and think of Y,Z as two sets of input variables. For a node v in Φ, denote the polynomial (in the input variables Y,Z), computed by the node v, by $\tilde{g}_v \in \mathbb{F}[Y,Z]$, and note that $\tilde{g}_v$ is a homogenous polynomial of degree $\tilde{r}_v \doteq r_v + (2r_v - 1) = 3r_v - 1$, if v is a sum-gate, and $\tilde{r}_v \doteq r_v + (2r_v - 2) = 3r_v - 2$, if v is a product-gate, where r_v is the syntactic-degree of v in the circuit-graph G. Thus, each $\tilde{g}_v$ contains $\mathrm{poly}(s^r)$ monomials. Thus, we can work our way up the circuit and compute all the coefficients in all the polynomials $\tilde{g}_v$, in time $\mathrm{poly}(s^r)$. By induction, all these coefficients are (positive) integers. By induction, we can show that the coefficients in each polynomial $\tilde{g}_v$ are bounded by $(\tilde{r}_v)!$. The induction is straightforward: When we have a sum-gate, we always sum polynomials with disjoint sets of monomials, because the edges that reach the sum-gate are labelled by different variables in Y, that were not used before. Thus, a sum-gate doesn't increase the coefficients. When we have a product-gate v, it multiplies v_1 and v_2. By induction, the coefficients in the polynomials $\tilde{g}_{v_1}, \tilde{g}_{v_2}$ are bounded by $(\tilde{r}_{v_1})!, (\tilde{r}_{v_2})!$, respectively. Since each monomial in $\tilde{g}_v$ can be obtained in at most $(\tilde{r}_v)!/((\tilde{r}_{v_1})! \cdot (\tilde{r}_{v_2})!)$ different ways, from monomials in $\tilde{g}_{v_1}, \tilde{g}_{v_2}$, we obtain a bound of $(\tilde{r}_v)!$ on its coefficient. □

Proposition 3.3. *For every n,r,m,s' such that $1 \le r \le n \le s'$ and $m = n \cdot \binom{n+r-1}{r}$, there exists a circuit-graph $G \in \mathcal{G}_{n,r}$, with $S(G) \le \mathrm{Size}(G) = O((s')^2 \cdot r^8)$, such that:*

1. *G is in normal-homogenous-form. Hence, $\Gamma_G : \mathbb{F}^s \to \mathbb{F}^m$ (where $s = S(G)$) is a (homogenous) polynomial mapping of degree $2r - 1$.*

2. *For every $g = (g_1,\dots,g_n) \in \mathcal{M}^n$, if there exists an arithmetic circuit of size s' (over $\mathbb{F}$) for the n polynomials $g_1,\dots,g_n$, then $H(g) \in \mathrm{Image}(\Gamma_G)$.*

3. *For every $g = (g_1,\dots,g_n) \in \mathcal{M}^n$, if $H(g) \in \mathrm{Image}(\Gamma_G)$, then there exists an arithmetic circuit Φ, (over $\mathbb{F}$), with $G_\Phi = G$, for the n polynomials $g_1,\dots,g_n$.*

Moreover, one can construct G, Γ_G in time $\mathrm{poly}(s^r)$ in the following sense. There exists a $\mathrm{poly}(s^r)$-time Turing machine, that on input n,r,s', outputs G and (all the coefficients of) the m polynomials

$(\Gamma_G)_1,\dots,(\Gamma_G)_m \in \mathbb{F}[y_1,\dots,y_s]$, *and such that all the coefficients in these polynomials are integers in* $\{0,\dots,(3r)!\}$.

Proof. Let G be the circuit-graph from Proposition 2.8, with parameters n,s',r, that is, a universal circuit-graph for n-inputs and n-outputs circuits of size s' that compute homogenous polynomials of degree r. Denote $s = S(G)$, and note that $s \le \text{Size}(G) \le O((s')^2 \cdot r^8)$. By Proposition 2.8, G is in normal-homogenous-form, and note that G is of syntactic-degree r. Hence, by Proposition 3.2, $\Gamma_G : \mathbb{F}^s \to \mathbb{F}^m$ is a (homogenous) polynomial mapping of degree $2r-1$.

Let $g = (g_1,\dots,g_n) \in \mathcal{M}^n$. Assume that there exists an arithmetic circuit of size s' (over $\mathbb{F}$) for the n polynomials $g_1,\dots,g_n$. Then, by Proposition 2.8, there exists an arithmetic circuit Φ, for the polynomials $g_1,\dots,g_n$, such that $G_\Phi = G$. Thus, by Proposition 3.1, $H(g) \in \text{Image}(\Gamma_G)$.

The third claim is a special case of Proposition 3.1 (and is restated here for completeness). The moreover part follows immediately from the moreover parts of Proposition 2.8 and Proposition 3.2. □

Proposition 3.4. *Let $r = 1$ and $m = n^2$. If $G \in \mathcal{G}_{n,r}$ is in normal-linear-form (see Definition 2.4), then the mapping $\Gamma_G : \mathbb{F}^s \to \mathbb{F}^m$, (where $s = S(G) = \text{Size}(G)$), is a polynomial mapping of degree* $\text{Depth}(G)$.

Proof. Let Φ be an arithmetic circuit over $\mathbb{F}$, with circuit-graph $G_\Phi = G$, such that the labels of the s edges in G are $y_1,\dots,y_s$.

For a node v in Φ, denote the linear polynomial (in the input variables Z), computed by the node v, by $g_v \in \mathbb{F}[Z]$. Note that if v is a leaf, all the coefficients in g_v are in $\{0,1\}$, and hence they do not depend on $y_1,\dots,y_s$. By induction we show that if v is a gate of depth d_v (i. e., the length of the longest directed path that reaches v is d_v), then every coefficient in the polynomial $g_v \in \mathbb{F}[Z]$ is a polynomial of degree at most d_v in the labels $y_1,\dots,y_s$.

The proof is straightforward. If v is a gate of depth d_v, then all children of v are of depth at most $d_v - 1$. Then, by induction, the coefficients in the polynomial g_u, for every child u of v, are polynomials of degree at most $d_v - 1$ (in the labels $y_1,\dots,y_s$). Since the edge (u,v) is labelled by an element of $\{y_1,\dots,y_s\}$, the coefficients in the polynomial g_v are polynomials of degree at most d_v (in the labels $y_1,\dots,y_s$). □

3.3 The polynomial $\tilde{f}$

Let $X = \{x_1,\dots,x_n\}$ be an additional set of input variables. Let $f = (f_1,\dots,f_m)$, where $f_1,\dots,f_m \in \mathbb{F}[x_1,\dots,x_n]$, be a polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$. Intuitively, since we think of a point in $\mathbb{F}^m$ as a tuple of n polynomials in the set of variables Z, we can think of f as a tuple of n polynomials in the sets of variables X,Z.

Formally, given f, we define a tuple of n polynomials $\tilde{f}_1,\dots,\tilde{f}_n \in \mathbb{F}[X,Z]$, by

$$\tilde{f}_i(x_1,\dots,x_n,z_1,\dots,z_n) = \sum_{q \in M} f_{(h(q),i)}(x_1,\dots,x_n) \cdot q = \sum_{j \in [m']} f_{(j,i)}(x_1,\dots,x_n) \cdot h^{-1}(j)$$

(where, as before, we think of $(h(q),i)$ and (j,i) as elements of $[m]$). In other words, for every monomial q_x in the variables $\{x_1,\dots,x_n\}$ and monomial $q_z \in M$, the coefficient of the monomial $q_x q_z$ in $\tilde{f}_i$ is simply the coefficient of the monomial q_x in $f_{(h(q_z),i)}$. (For monomials q_x, q_z such that $q_z \notin M$, the coefficient of the monomial $q_x q_z$ in $\tilde{f}_i$ is 0.)

Finally, we define the polynomial $\tilde{f}$ as follows.[9] Let $W = \{w_1, \ldots, w_n\}$ be an additional set of input variables. Define $\tilde{f} \in \mathbb{F}[X, Z, W]$, by

$$\tilde{f}(x_1, \ldots, x_n, z_1, \ldots, z_n, w_1, \ldots, w_n) = \sum_{i=1}^{n} w_i \cdot \tilde{f}_i(x_1, \ldots, x_n, z_1, \ldots, z_n).$$

For $a = (a_1, \ldots, a_n) \in \mathbb{F}^n$, denote by $\tilde{f}_1|_a, \ldots, \tilde{f}_n|_a \in \mathbb{F}[Z]$, the n polynomials $\tilde{f}_1, \ldots, \tilde{f}_n \in \mathbb{F}[X, Z]$, after the substitution $x_1 = a_1, \ldots, x_n = a_n$.

Proposition 3.5. *$\forall a \in \mathbb{F}^n$, we have, $(\tilde{f}_1|_a, \ldots, \tilde{f}_n|_a) \in \mathcal{M}^n$, and $H((\tilde{f}_1|_a, \ldots, \tilde{f}_n|_a)) = f(a)$.*

Proof. The proof is straightforward from the definitions. For every $i \in [n]$ and $a = (a_1, \ldots, a_n) \in \mathbb{F}^n$,

$$\tilde{f}_i|_a(z_1, \ldots, z_n) = \tilde{f}_i(a_1, \ldots, a_n, z_1, \ldots, z_n) = \sum_{q \in M} f_{(h(q),i)}(a) \cdot q \in \mathcal{M}.$$

Thus,

$$\begin{aligned} H\left((\tilde{f}_1|_a, \quad \ldots \quad, \tilde{f}_n|_a)\right) &= H\left((\textstyle\sum_{q \in M} f_{(h(q),1)}(a) \cdot q, \quad \ldots \quad, \sum_{q \in M} f_{(h(q),n)}(a) \cdot q)\right) \\ &= H\left((\textstyle\sum_{j \in [m']} f_{(j,1)}(a) \cdot h^{-1}(j), \quad \ldots \quad, \sum_{j \in [m']} f_{(j,n)}(a) \cdot h^{-1}(j))\right) \\ &= \left(f_{(1,1)}(a), \quad \ldots \quad, f_{(m',n)}(a)\right) = f(a). \end{aligned}$$

□

Proposition 3.6. *If $f = (f_1, \ldots, f_m)$ is* poly(n)*-definable (see Definition 1.3), then the polynomial $\tilde{f} \in \mathbb{F}[X, Z, W]$ is* poly(n)*-definable.*

Proof. We will show that for every $i \in [n]$, the polynomial $\tilde{f}_i \in \mathbb{F}[X, Z]$ is poly(n)-definable. Obviously, this implies that the polynomial $\tilde{f} = \sum_{i=1}^{n} w_i \cdot \tilde{f}_i$ is poly(n)-definable.

First, we construct an arithmetic circuit C, of size and degree poly(n), that gets as input the variables $z_1, \ldots, z_n, u_1, \ldots, u_\ell$ (for some $\ell =$ poly(n)), and the binary representation $\bar{j}$ (of $j-1$), for an integer $j \in [m']$, and such that for every $\bar{j} \in \{0,1\}^{\log m'}$,

$$\sum_{u_1, \ldots, u_\ell \in \{0,1\}} C(z_1, \ldots, z_n, u_1, \ldots, u_\ell, \bar{j}) = h^{-1}(j).$$

(For $j \notin [m']$, we define $h^{-1}(j) = 0$.)

This is done by the following steps:

1. First construct a poly(n)-size Boolean circuit C_1 that gets as input the binary representation $\bar{j}$ and outputs r vectors $(c_{1,1}, \ldots, c_{1,n}), \cdots, (c_{r,1}, \ldots, c_{r,n}) \in \{0,1\}^n$ such that $\prod_{a=1}^{r} \sum_{b=1}^{n} c_{a,b} z_{a,b} = h^{-1}(j)$. That is, on input $\bar{j}$, the circuit C_1 generates a "description" of the monomial $h^{-1}(j) \in M$. Obviously, this can be done in poly(n) time, and hence such a circuit C_1 exists. Denote by ℓ the number of nodes in C_1.

[9] The set W is introduced, and the polynomial $\tilde{f}$ is defined in this way, in order to make use of the result of Baur and Strassen [5] (see in the introduction).

2. For every node in C_1, we introduce a variable in $\{0,1\}$ that represents the value computed at that node. Let $u_1,\dots,u_\ell \in \{0,1\}$ be these variables. We rename the variables corresponding to the $r\cdot n$ output nodes by $u_{1,1},\dots,u_{1,n},\cdots,u_{r,1},\dots,u_{r,n}$ (we think of these variables as having two names). We can now construct a poly(n)-size Boolean formula C_2 (in conjunctive-normal-form), that gets as input the binary representation $\bar{j}$ and the set of variables $\{u_1,\dots,u_\ell\}$ (including the $r\cdot n$ output variables), and outputs 1 iff $(u_1,\dots,u_\ell)$ is the correct computation of C_1 on $\bar{j}$. This is done as in Cook-Levin's proof for the NP-completeness of *SAT*. Note that if $(u_1,\dots,u_\ell)$ is the correct computation, then $u_{1,1},\dots,u_{1,n},\cdots,u_{r,1},\dots,u_{r,n}$ are such that $\prod_{a=1}^{r}\sum_{b=1}^{n} u_{a,b}z_{a,b} = h^{-1}(j)$. Hence,
$$\sum_{u_1,\dots,u_\ell\in\{0,1\}} C_2(u_1,\dots,u_\ell,\bar{j})\cdot\prod_{a=1}^{r}\sum_{b=1}^{n} u_{a,b}z_{a,b} = h^{-1}(j)\,.$$
Since any polynomial-size Boolean formula can be easily translated into an arithmetic formula of polynomial size and degree, we can think of C_2 as an arithmetic circuit of size and degree poly(n).

3. Finally, we define the arithmetic circuit C by,
$$C(z_1,\dots,z_n,u_1,\dots,u_\ell,\bar{j}) = C_2(u_1,\dots,u_\ell,\bar{j})\cdot\prod_{a=1}^{r}\sum_{b=1}^{n} u_{a,b}z_{a,b}\,.$$

Recall that for an integer k, we denote by $\bar{k}$ the binary representation of $k-1$. Note that since we assumed that n is a power of 2, the binary representation $\overline{(j,i)}$, for $(j,i)\in[m]$, is simply $(\bar{j},\bar{i})$ (where, as before, we think of $(j,i)\in[m']\times[n]$ as an element of $[m]$ by the lexicographic order).

Since the polynomial mapping $f:\mathbb{F}^n\to\mathbb{F}^m$ is poly(n)-definable, for some $\ell' = \mathrm{poly}(n)$, there is an arithmetic circuit D, of size and degree poly(n), that gets as input variables $x_1,\dots,x_n,e_1,\dots,e_{\ell'}$, and the binary representations $\bar{j},\bar{i}$ (for $j\in[m'], i\in[n]$), and such that for every $(j,i)\in[m]$,
$$f_{(j,i)}(x_1,\dots,x_n) = \sum_{e_1,\dots,e_{\ell'}\in\{0,1\}} D(x_1,\dots,x_n,e_1,\dots,e_{\ell'},\bar{j},\bar{i})\,.$$

We can now write, for every $i\in[n]$,
$$\tilde{f}_i(x_1,\dots,x_n,z_1,\dots,z_n) = \sum_{j\in[m']} f_{(j,i)}(x_1,\dots,x_n)\cdot h^{-1}(j) =$$
$$\sum_{j\in[m']}\sum_{e_1,\dots,e_{\ell'}\in\{0,1\}} D(x_1,\dots,x_n,e_1,\dots,e_{\ell'},\bar{j},\bar{i})\cdot\sum_{u_1,\dots,u_\ell\in\{0,1\}} C(z_1,\dots,z_n,u_1,\dots,u_\ell,\bar{j})\,.$$

Since we can replace the sum over $j\in[m']$ by a sum over $\bar{j}\in\{0,1\}^{\log m'}$, the polynomial $\tilde{f}_i$ is poly(n)-definable, by the definition of poly(n)-definability. □

3.4 The route to lower bounds

In this subsection, we prove our main results for polynomial mappings f that elude polynomial mappings Γ of degree larger than 2. The results are given by five propositions and corollaries. Proposition 3.7, Corollary 3.8, Corollary 3.9 and Corollary 3.10 give the connection for general arithmetic circuits. Note that these four propositions and corollaries are only interesting for $r \geq 2$, (although, to avoid confusion, they are stated for $r \geq 1$). Proposition 3.11 gives the connection for arithmetic circuits of depth d, and is the one used to prove our lower bounds for bounded-depth arithmetic circuits. All five propositions and corollaries are only interesting for $s < m$ (although this condition is not stated explicitly).

Recall that we think of all the parameters (r, s, m, etc.) as functions of the basic parameter n.

Recall that given a polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$, and given a field extension $\mathbb{G} \supset \mathbb{F}$, we can think of f as a polynomial mapping $f : \mathbb{G}^n \to \mathbb{G}^m$. This is because we assume that a polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$ is given as a tuple $(f_1, \ldots, f_m)$ of polynomials in $\mathbb{F}[x_1, \ldots, x_n] \subset \mathbb{G}[x_1, \ldots, x_n]$ (see the discussion in Subsection 1.4).

Proposition 3.7. *For integers $1 \leq r \leq n \leq s'$, and $m = n \cdot \binom{n+r-1}{r}$, let $\Gamma_G : \mathbb{F}^s \to \mathbb{F}^m$ (where $s = O((s')^2 \cdot r^8)$) be the (homogenous) polynomial mapping of degree $2r-1$ from Proposition 3.3. Let $f : \mathbb{F}^n \to \mathbb{F}^m$ be a polynomial mapping.*

If over some field extension $\mathbb{G} \supseteq \mathbb{F}$,

$$\operatorname{Image}(f : \mathbb{G}^n \to \mathbb{G}^m) \not\subset \operatorname{Image}(\Gamma_G : \mathbb{G}^s \to \mathbb{G}^m),$$

then any arithmetic circuit (over $\mathbb{F}$) for the polynomial $\tilde{f} : \mathbb{F}^{3n} \to \mathbb{F}$ (explicitly defined from f in Subsection 3.3), is of size $> s'/5$.

Proof. Let us first prove the proposition for $\mathbb{G} = \mathbb{F}$.

By Proposition 3.3, for every $(g_1, \ldots, g_n) \in \mathcal{M}^n$, if there exists an arithmetic circuit of size s' (over $\mathbb{F}$) for the n polynomials $g_1, \ldots, g_n$, then $H((g_1, \ldots, g_n)) \in \operatorname{Image}(\Gamma_G)$.

Assume for a contradiction that there exists an arithmetic circuit (over $\mathbb{F}$) of size $s'/5$, for the polynomial $\tilde{f} : \mathbb{F}^{3n} \to \mathbb{F}$. Baur and Strassen proved that if a polynomial ($\tilde{f}$) can be computed by an arithmetic circuit of size s'', then all partial derivatives of that polynomial can be computed by one arithmetic circuit of size $5s''$ [5] (for an improvement of the constant see [18]). By the result of Baur and Strassen, there is an arithmetic circuit of size s' for the n polynomials $\tilde{f}_1, \ldots, \tilde{f}_n : \mathbb{F}^{2n} \to \mathbb{F}$. By substituting in this circuit $x_1 = a_1, \ldots, x_n = a_n$, where $a = (a_1, \ldots, a_n) \in \mathbb{F}^n$, we obtain an arithmetic circuit of size s' for the n polynomials $\tilde{f}_1|_a, \ldots, \tilde{f}_n|_a$, and note that by Proposition 3.5, $\tilde{f}_1|_a, \ldots, \tilde{f}_n|_a \in \mathcal{M}$. Hence, by Proposition 3.3, (for every $a_1, \ldots, a_n \in \mathbb{F}$), $H((\tilde{f}_1|_a, \ldots, \tilde{f}_n|_a)) \in \operatorname{Image}(\Gamma_G)$. Thus, by Proposition 3.5, for every $a_1, \ldots, a_n \in \mathbb{F}$,

$$f(a_1, \ldots, a_n) = H((\tilde{f}_1|_a, \ldots, \tilde{f}_n|_a)) \in \operatorname{Image}(\Gamma_G).$$

That is,

$$\operatorname{Image}(f) \subset \operatorname{Image}(\Gamma_G).$$

Assume now that $\mathbb{G}$ is a general field, extending $\mathbb{F}$, and assume that

$$\operatorname{Image}(f : \mathbb{G}^n \to \mathbb{G}^m) \not\subset \operatorname{Image}(\Gamma_G : \mathbb{G}^s \to \mathbb{G}^m).$$

By the part that we already proved (i. e., the case $\mathbb{G} = \mathbb{F}$), we know that any arithmetic circuit over $\mathbb{G}$, for the polynomial $\tilde{f}$, is of size $> s'/5$. But any arithmetic circuit over $\mathbb{F}$ is in particular an arithmetic circuit over $\mathbb{G}$. Thus, any arithmetic circuit over $\mathbb{F}$ for the polynomial $\tilde{f}$ is of size $> s'/5$. (Formally, we need to verify that the polynomials $\tilde{f}$ and $(\Gamma_G)_1, \ldots, (\Gamma_G)_m$ remain the same polynomials when we work over $\mathbb{G}$, rather than over $\mathbb{F}$. This can be easily verified. For $(\Gamma_G)_1, \ldots, (\Gamma_G)_m$, it was noted after the definition of Γ_G that they do not depend on the field at all, but only on its characteristic. As for $\tilde{f}$, by its definition, its coefficients are just corresponding coefficients from the polynomials $f_1, \ldots, f_m$, and hence they do not depend on the field $\mathbb{G}$.) $\square$

Corollary 3.8. *Let $1 \le r \le n \le s$, and $m = n \cdot \binom{n+r-1}{r}$ be integers. Let $f : \mathbb{F}^n \to \mathbb{F}^m$ be a polynomial mapping. If over some field extension $\mathbb{G} \supseteq \mathbb{F}$, f is $(s,(2r-1))$-elusive (see Definition 1.1), then any arithmetic circuit (over $\mathbb{F}$) for the polynomial $\tilde{f} : \mathbb{F}^{3n} \to \mathbb{F}$ (explicitly defined from f in Subsection 3.3), is of size $\ge \Omega(\sqrt{s}/r^4)$.*

Proof. The proof follows immediately from Proposition 3.7. Let $s' = c \cdot \sqrt{s}/r^4$, where c is a small enough constant. If f is $(s,(2r-1))$-elusive, then in particular it satisfies $\mathrm{Image}(f) \not\subset \mathrm{Image}(\Gamma_G)$, where Γ_G is the mapping from Proposition 3.7. $\square$

Corollary 3.9. *Let $\mathbb{F}$ be a field of characteristic not equal to 2. Let $1 \le r \le n \le s$, and $m = n \cdot \binom{n+r-1}{r}$ be integers such that $s = n^{\omega(1)}$. If there exists a $\mathrm{poly}(n)$-definable polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$ such that, over some field extension $\mathbb{G} \supseteq \mathbb{F}$, f is $(s,(2r-1))$-elusive (see Definition 1.3 and Definition 1.1), then any arithmetic circuit for the permanent over $\mathbb{F}$ is of size $\ge s^{\Omega(1)}$.*

Proof. Assume that such a polynomial mapping f exists. By Proposition 3.6, and Corollary 3.8, the polynomial $\tilde{f} : \mathbb{F}^{3n} \to \mathbb{F}$ (explicitly defined from f in Subsection 3.3) is $\mathrm{poly}(n)$-definable, and any arithmetic circuit (over $\mathbb{F}$) for $\tilde{f}$ is of size $\ge s^{\Omega(1)}$.

Valiant proved that over any field of characteristic not equal to 2, the permanent is a complete polynomial for the class VNP of $\mathrm{poly}(n)$-definable polynomials [36] (see also [10, 6]). Hence, any arithmetic circuit of size s' for the permanent implies an arithmetic circuit of size $\mathrm{poly}(s')$ for any other $\mathrm{poly}(n)$-definable polynomial. Hence, any arithmetic circuit for the permanent over $\mathbb{F}$ is of size $s' \ge s^{\Omega(1)}$. $\square$

Corollary 3.10. *Let $1 \le r \le n \le s$, and $m = n \cdot \binom{n+r-1}{r}$ be integers (and recall that we think of r,s,m as functions of n). Assume that there exists a $\mathrm{poly}(s^r)$-time Turing machine T such that:*

- *The inputs for T are r,n,s,m and a (homogenous) polynomial mapping $\Gamma : \mathbb{F}^s \to \mathbb{F}^m$ of degree $2r-1$, given by all coefficients of the polynomials $\Gamma_1, \ldots, \Gamma_m$ (that are assumed to be integers[10] in, say, $\{0, \ldots, (3r)!\}$).*
- *The output of T is a $\mathrm{poly}(n)$-definable polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$ (described, e. g., by an arithmetic circuit for the polynomial g that defines it in Definition 1.3) such that, over some field extension $\mathbb{G} \supseteq \mathbb{F}$,*

$$\mathrm{Image}(f : \mathbb{G}^n \to \mathbb{G}^m) \not\subset \mathrm{Image}(\Gamma : \mathbb{G}^s \to \mathbb{G}^m).$$

Then, there exists a $\mathrm{poly}(s^r)$-time Turing machine that on input n outputs a $3n$-variables, $\mathrm{poly}(n)$-definable, polynomial $\tilde{f}$ (explicitly defined from f in Subsection 3.3, and described, e. g., by an arithmetic circuit for the polynomial g that defines it in Definition 1.3), such that any arithmetic circuit for $\tilde{f}$ is of size $\ge \Omega(\sqrt{s}/r^4)$.

[10] Recall that we think of the integers as members of every field, by the inductive definition $n = (n-1)+1$.

Proof. The proof follows immediately from Proposition 3.7. Let $s' = c \cdot \sqrt{s}/r^4$, where c is a small enough constant. We run the Turing machine T on the polynomial mapping $\Gamma = \Gamma_G$, where Γ_G is the mapping from Proposition 3.7. Note that by Proposition 3.3, Γ_G can be constructed in time $\mathrm{poly}(s^r)$ (for details, see Proposition 3.3). □

Proposition 3.11. *Let $n, d \leq s$, and $m = n^2$ be integers. Let $f : \mathbb{F}^n \to \mathbb{F}^m$ be a polynomial mapping. If over some field extension $\mathbb{G} \supseteq \mathbb{F}$, f is (s,d)-elusive (see Definition 1.1), then any depth-d arithmetic circuit (over $\mathbb{F}$) for the n polynomials $\tilde{f}_1, \ldots, \tilde{f}_n : \mathbb{F}^{2n} \to \mathbb{F}$ (explicitly defined from f in Subsection 3.3, (using $r = 1$)), is of size $> s$. (Note that the degree of each $\tilde{f}_i$ is at most the degree of f plus 1.)*

Proof. Let us first prove the proposition for $\mathbb{G} = \mathbb{F}$.

Let $X = \{x_1, \ldots, x_n\}$ and $Z = \{z_1, \ldots, z_n\}$ be two sets of input variables.

Let $\mathbb{F}(X) = \mathbb{F}(x_1, \ldots, x_n)$ be the field of rational functions in the variables $\{x_1, \ldots, x_n\}$, over $\mathbb{F}$. Let $\tilde{f}_1, \ldots, \tilde{f}_n \in \mathbb{F}[X,Z]$ be the polynomials explicitly defined from f in Subsection 3.3, using $r = 1$. Note that the total-degree of these polynomials in the set of variables Z is 1 (by the definition of $\tilde{f}_1, \ldots, \tilde{f}_n$). We can think of the polynomials $\tilde{f}_1, \ldots, \tilde{f}_n$ as members of the ring of polynomials $\mathbb{F}(X)[z_1, \ldots, z_n]$, that is, polynomials in the set of variables Z, over the field $\mathbb{F}(X)$. Note, that $\tilde{f}_1, \ldots, \tilde{f}_n \in \mathbb{F}(X)[Z]$ are linear polynomials (as their total-degree in the set of variables Z is 1).

Assume for a contradiction that there exists an arithmetic circuit Φ, over $\mathbb{F}$, of depth d and size s, for the n polynomials $\tilde{f}_1, \ldots, \tilde{f}_n \in \mathbb{F}[X,Z]$. We can think of Φ as an arithmetic circuit for $\tilde{f}_1, \ldots, \tilde{f}_n$, over the field $\mathbb{F}(X)$ and set of input variables Z, i. e., an arithmetic circuit for $\tilde{f}_1, \ldots, \tilde{f}_n \in \mathbb{F}(X)[Z]$. By Proposition 2.5, we can assume without loss of generality that Φ is in normal-linear-form, (see Definition 2.4). That is, there is an arithmetic circuit Ψ, of depth d and size s, in normal-linear-form, that computes $\tilde{f}_1, \ldots, \tilde{f}_n$, over the field $\mathbb{F}(X)$ and set of input variables Z. Moreover, by the proof of Proposition 2.5, we can assume that the labels of the edges in Ψ are polynomials in $\mathbb{F}[X]$, rather than rational functions in $\mathbb{F}(X)$. (This is true because in the proof of Proposition 2.5, the labels of the edges in Ψ are defined from the labels of the edges in Φ without using divisions.) Denote by $G = G_\Psi$, the circuit graph of Ψ (see Definition 2.1).

By Proposition 3.4, the mapping $\Gamma_G : \mathbb{F}^s \to \mathbb{F}^m$ is a polynomial mapping of degree d. Hence, since f is (s,d)-elusive, there exists $(a_1, \ldots, a_n) \in \mathbb{F}^n$ such that $f(a_1, \ldots, a_n) \notin \mathrm{Image}(\Gamma_G)$.

By substituting in the circuit Ψ the values $x_1 = a_1, \ldots, x_n = a_n$, we obtain an arithmetic circuit of size s and depth d, over $\mathbb{F}$, with circuit-graph G, for the n polynomials $\tilde{f}_1|_a, \ldots, \tilde{f}_n|_a \in \mathbb{F}[Z]$. (Note that when we substitute $x_1 = a_1, \ldots, x_n = a_n$, we do not introduce divisions-by-zero, because the labels of the edges in Ψ are polynomials in $\mathbb{F}[X]$, rather than rational functions in $\mathbb{F}(X)$.) Note that by Proposition 3.5, $\tilde{f}_1|_a, \ldots, \tilde{f}_n|_a \in \mathcal{M}$. Hence, by Proposition 3.1, $H((\tilde{f}_1|_a, \ldots, \tilde{f}_n|_a)) \in \mathrm{Image}(\Gamma_G)$. Hence, by Proposition 3.5, $f(a_1, \ldots, a_n) = H((\tilde{f}_1|_a, \ldots, \tilde{f}_n|_a)) \in \mathrm{Image}(\Gamma_G)$, which is a contradiction.

Assume now that $\mathbb{G}$ is a general field, extending $\mathbb{F}$, and assume that f is (s,d)-elusive over $\mathbb{G}$. By the part that we already proved (i. e., the case $\mathbb{G} = \mathbb{F}$), we know that any depth-d arithmetic circuit, over $\mathbb{G}$, for the n polynomials $\tilde{f}_1, \ldots, \tilde{f}_n$ is of size $> s$. But any arithmetic circuit over $\mathbb{F}$ is in particular an arithmetic circuit over $\mathbb{G}$. Thus, any depth-d arithmetic circuit, over $\mathbb{F}$, for the n polynomials $\tilde{f}_1, \ldots, \tilde{f}_n$ is of size $> s$. (Formally, as before, we need to verify that the polynomials $\tilde{f}_1, \ldots, \tilde{f}_n$ remain the same polynomials when we work over $\mathbb{G}$, rather than over $\mathbb{F}$. This can be easily verified, as by the definition of $\tilde{f}_1,\ldots,\tilde{f}_n$, their coefficients are just the corresponding coefficients from the polynomials $f_1,\ldots,f_m$, and hence they do not depend on the field.) $\square$

4 Lower bounds for bounded-depth circuits

4.1 A construction of an elusive polynomial mapping

For an integer k, denote by $[k]$ the set $\{1,\ldots,k\}$. Let n be a prime, and let $m = n^2$. We identify the set $[m]$ with $[n]\times[n]$ (by the lexicographic order). Let $1 \le d \le (\log_2 n)/100$ be an integer. Let $d' = 5d$. Let $X = \{x_{i,j}\}_{i\in[d'],j\in[n]}$ be a set of $n\cdot d'$ input variables. For every $(a,b)\in[n]\times[n]=[m]$, define a polynomial

$$f_{(a,b)}(x_{1,1},\ldots,x_{d',n}) = \prod_{i=1}^{d'} x_{i,a+i\cdot b}$$

where the sum $a+i\cdot b$ (as well as all other sums of this sort that appear below), is taken modulo n. Let $f = (f_{(1,1)}, f_{(1,2)},\ldots,f_{(n,n)})$. Note that for every field $\mathbb{G}$, we can view f as a polynomial mapping $f:\mathbb{G}^{n\cdot d'}\to\mathbb{G}^m$.

Lemma 4.1. *Let n be a prime, and let $m = n^2$. Let d be an integer such that $1 \le d \le (\log_2 n)/100$. Let $d' = 5d$. Let $\mathbb{G}$ be a field of size larger than m (e. g., an infinite field). Then, the polynomial mapping $f:\mathbb{G}^{n\cdot d'}\to\mathbb{G}^m$ (as defined above) is (s,d)-elusive (see Definition 1.1), where $s = \lfloor n^{1+1/(2d)}\rfloor$.*

Proof. Let $r = \lfloor n^{1-1/(2d)}\rfloor$. For a set $Q\subset[n]\times[n]$, denote

$$f_Q = \prod_{(a,b)\in Q} f_{(a,b)} = \prod_{(a,b)\in Q}\prod_{i\in[d']} x_{i,a+i\cdot b}\,.$$

We say that $Q\subset[n]\times[n]$ is *retrievable*, if f_Q determines Q, that is, Q is retrievable if for every $Q'\neq Q$, we have $f_{Q'}\neq f_Q$.

Claim 4.2. *Let $Q\subset[n]\times[n]$ be a random subset of size r. Then, with probability at least a half, Q is retrievable.*

Proof. Let us first prove the following claim.

Claim 4.3. *If for every $(a,b)\in[n]\times[n]\setminus Q$, at least one of the variables in $\{x_{i,a+i\cdot b}\}_{i\in[d']}$ doesn't appear in the monomial f_Q, then the set Q is retrievable.*

Proof. Let $Q'\subset[n]\times[n]$ be such that $f_{Q'} = f_Q$. We will show that $Q' = Q$.

For every $(a,b)\in[n]\times[n]\setminus Q$, at least one of the variables in $\{x_{i,a+i\cdot b}\}_{i\in[d']}$ doesn't appear in the monomial $f_Q = f_{Q'}$, and hence $(a,b)\notin Q'$. Thus $Q'\subseteq Q$. Since the total-degrees of f_Q and $f_{Q'}$ are the same, $|Q'| = |Q|$. Hence, $Q' = Q$. $\square$

It is hence enough to show that with probability (over Q) at least $1/2$, for every $(a,b)\in[n]\times[n]\setminus Q$, at least one of the variables in $\{x_{i,a+i\cdot b}\}_{i\in[d']}$ doesn't appear in the monomial f_Q.

A variable $x_{i,j}$ appears in the monomial f_Q iff there exists $(a',b') \in Q$ such that $j = a' + i \cdot b'$. Hence, a variable $x_{i,a+i\cdot b}$ appears in the monomial f_Q iff there exists $(a',b') \in Q$ such that $a + i \cdot b = a' + i \cdot b'$, that is, $(a'-a) + i \cdot (b'-b) = 0$. Denote by $\ell_{a,b,i}$ the line $\{(a',b') \in [n] \times [n] : (a'-a) + i \cdot (b'-b) = 0\}$. Thus, a variable $x_{i,a+i\cdot b}$ appears in the monomial f_Q iff $Q \cap \ell_{a,b,i} \neq \emptyset$.

Thus, for $(a,b) \in [n] \times [n]$, the statement "at least one of the variables in $\{x_{i,a+i\cdot b}\}_{i\in[d']}$ doesn't appear in the monomial f_Q" is equivalent to the statement "there exists $i \in [d']$ such that $Q \cap \ell_{a,b,i} = \emptyset$."

Fix $(a,b) \in [n] \times [n]$. Note that for different $i_1, i_2 \in [d']$, the two lines $\ell_{a,b,i_1}, \ell_{a,b,i_2}$ intersect only at the point (a,b). Hence, the probability over Q, such that $(a,b) \notin Q$, for the event: "for every $i \in [d']$, $Q \cap \ell_{a,b,i} \neq \emptyset$," is at most

$$\left[\frac{|Q| \cdot (n-1)}{(n^2-1)}\right]^{d'} \leq n^{-d'/(2d)} \leq n^{-2}/2\,.$$

Thus, for every $(a,b) \in [n] \times [n]$, the probability, over Q, for the event: "$(a,b) \notin Q$, and for every $i \in [d']$, $Q \cap \ell_{a,b,i} \neq \emptyset$" is at most $n^{-2}/2$. By the union bound, with probability (over Q) at most $1/2$, there exists $(a,b) \in [n] \times [n]$, such that: $(a,b) \notin Q$, and for every $i \in [d']$, $Q \cap \ell_{a,b,i} \neq \emptyset$. Thus, with probability (over Q) at least $1/2$, for every $(a,b) \in [n] \times [n]$, either $(a,b) \in Q$, or there exists $i \in [d']$ such that $Q \cap \ell_{a,b,i} = \emptyset$.

Thus, with probability at least $1/2$, the set Q is retrievable. (End of proof of Claim 4.2.) $\square$

Denote by $\mathcal{L}$ the set of all multilinear homogenous polynomials $\Lambda : \mathbb{G}^m \to \mathbb{G}$ of total-degree exactly r, such that every monomial that appears in Λ (with coefficient not equal to 0) corresponds to a retrievable set $Q \subset [n] \times [n] = [m]$. Thus, by Claim 4.2, $\mathcal{L}$ is a vector space (over $\mathbb{G}$), of dimension $\geq \binom{m}{r}/2 \geq \left(\frac{m}{r}\right)^r \geq s^r$ (where the second inequality holds since m, r are large enough, by the assumptions of the lemma).

Claim 4.4. *For every $\Lambda \in \mathcal{L}$, other than the 0 polynomial, the polynomial $\Lambda \circ f : \mathbb{G}^{n\cdot d'} \to \mathbb{G}$ is not the 0 polynomial.*

Proof. $\Lambda \circ f : \mathbb{G}^{n\cdot d'} \to \mathbb{G}$ is a linear combination of monomials f_Q (in the $n \cdot d'$ variables $\{x_{i,j}\}_{i\in[d'],j\in[n]}$), where $Q \subset [n] \times [n]$ is a retrievable set of size r. Since for every two different retrievable sets Q_1, Q_2, the monomials f_{Q_1}, f_{Q_2} are different monomials (in the $n \cdot d'$ variables $\{x_{i,j}\}_{i\in[d'],j\in[n]}$), there are no cancellations of monomials, and hence the polynomial $\Lambda \circ f$ cannot be the 0 polynomial. $\square$

Fix $\Gamma : \mathbb{G}^s \to \mathbb{G}^m$ to be a polynomial mapping of degree d. Thus, for every $\Lambda \in \mathcal{L}$, the polynomial $\Lambda \circ \Gamma : \mathbb{G}^s \to \mathbb{G}$ is of total-degree at most $r \cdot d$.

Denote by $\mathcal{K}$ the set of all polynomials from $\mathbb{G}^s$ to $\mathbb{G}$ of total-degree at most $r \cdot d$. Thus, $\mathcal{K}$ is a vector space (over $\mathbb{G}$), of dimension no more than

$$\binom{s+r}{r}^d \leq \frac{(s+r)^{r\cdot d}}{(r!)^d} \leq \frac{(2s)^{r\cdot d}}{(r/e)^{r\cdot d}} \leq \left(\frac{2es}{r}\right)^{r\cdot d} \leq (6n)^r < s^r\,.$$

For a fixed $\Gamma : \mathbb{G}^s \to \mathbb{G}^m$ (of degree d), the mapping $\Lambda \to \Lambda \circ \Gamma$ (where $\Lambda \in \mathcal{L}$) is a linear mapping from $\mathcal{L}$ to $\mathcal{K}$. Hence, since $\dim(\mathcal{L}) > \dim(\mathcal{K})$, there exist $\Lambda \neq 0$ such that $\Lambda \circ \Gamma = 0$. Fix Λ_Γ to be that Λ.

By Claim 4.4, $\Lambda_\Gamma \circ f$ is not the 0 polynomial. Hence, since $|\mathbb{G}| \geq m$ and since the degree of $\Lambda_\Gamma \circ f$ is smaller than m, the function $\Lambda_\Gamma \circ f : \mathbb{G}^{n \cdot d'} \to \mathbb{G}$ is not the 0 function. Since the function $\Lambda_\Gamma \circ \Gamma$ is the 0 function, $\mathrm{Image}(f) \not\subset \mathrm{Image}(\Gamma)$.

Since this is true for every polynomial mapping $\Gamma : \mathbb{G}^s \to \mathbb{G}^m$, of degree d, the polynomial mapping f is (s,d)-elusive, by definition. (End of proof of Lemma 4.1.) □

4.2 The lower bound

Let n be a prime. Let $1 \leq d \leq (\log_2 n)/100$ be an integer. Let $d' = 5d$. Let $\{x_{i,j}\}_{i \in [d'], j \in [n]}$ be a set of $n \cdot d'$ input variables. For every $(a,b) \in [n] \times [n]$, define (as in Subsection 4.1),

$$f_{(a,b)}(x_{1,1}, \ldots, x_{d',n}) = \prod_{i \in [d']} x_{i,a+i \cdot b}$$

(where the sum $a + i \cdot b$ is taken modulo n).

Let $\{z_1, \ldots, z_n\}$, $\{w_1, \ldots, w_n\}$, be two sets of input variables. Define, for every $a \in [n]$,

$$\tilde{f}_a = \sum_{b \in [n]} z_b \cdot f_{(a,b)} .$$

Define

$$\tilde{f} = \sum_{a \in [n]} w_a \cdot \tilde{f}_a .$$

Note that these definitions are consistent with the definitions in Subsection 3.3.

Note that every $\tilde{f}_a$ is a polynomial in $n \cdot (d'+1)$ variables, and is of total-degree $d'+1$, and $\tilde{f}$ is a polynomial in $n \cdot (d'+2)$ variables, and is of total-degree $d'+2$.

Corollary 4.5. *Let n be a prime, and let $1 \leq d \leq (\log_2 n)/100$ be an integer. Any depth-d arithmetic circuit, over any field $\mathbb{F}$, for the n polynomials (of total-degree $5d+1$ each) $\tilde{f}_1, \ldots, \tilde{f}_n : \mathbb{F}^{n \cdot (5d+1)} \to \mathbb{F}$ (as defined above), is of size $\geq n^{1+1/(2d)}$.*

Proof. Let $\mathbb{G}$ be an infinite field extending $\mathbb{F}$.

The proof follows immediately by Proposition 3.11, Lemma 4.1, and the trivial observation that an (s,d)-elusive polynomial mapping stays (s,d)-elusive, when padded by zeros. □

Corollary 4.6. *Let n be a prime, and let $1 \leq d \leq (\log_2 n)/100$ be an integer. Any depth-$\lfloor d/3 \rfloor$ arithmetic circuit, over any field $\mathbb{F}$, for the polynomial (of total-degree $5d+2$) $\tilde{f} : \mathbb{F}^{n \cdot (5d+2)} \to \mathbb{F}$ (as defined above), is of size $\geq n^{1+1/(2d)}/5$.*

Proof. Baur and Strassen proved that if a polynomial ($\tilde{f}$) can be computed by an arithmetic circuit of size s' and depth d', then all partial derivatives of that polynomial can be computed by one arithmetic circuit of size $5s'$ and depth $3d'$ [5] (for an improvement of the constants see [18]).

Thus the proof follows immediately by Corollary 4.5. □

5 Arithmetic circuits and polynomial mappings: Part II

In this section, we describe and prove our main results for polynomial-mappings f that elude polynomial-mappings Γ of degree 2. The main results appear in Subsection 5.4. The description follows closely the one in Section 3 but is somewhat simpler since here we consider a single polynomial g, rather than a tuple of polynomials.

5.1 Notation

Let $\mathbb{F}$ be a field. Let n, r be integers. We fix m to be the number of monomials of total-degree exactly r in n variables, that is, $m = \binom{n+r-1}{r}$. Note that r is not necessarily a constant, and may be a function of n. In general, we think of all parameters as functions of the basic parameter n. We assume that $3 \leq r \leq n$, and we assume for simplicity that n is a power of 2. For an integer k, denote by $[k]$ the set $\{1, \ldots, k\}$.

Let $Z = \{z_1, \ldots, z_n\}$ be a set of n input variables. Let M be the set of all monomials of total-degree exactly r in the variables $\{z_1, \ldots, z_n\}$. Note that $|M| = m$. We can identify the set M with the set $[m]$, by the lexicographic order of monomials. Formally, let $h : M \to [m]$ be the lexicographic order of monomials.

We denote by $\mathcal{M}$ the set of all homogenous polynomials in $\mathbb{F}[Z]$ of total-degree exactly r. We identify the vector space $\mathcal{M} = \mathbb{F}^M$ with the vector space $\mathbb{F}^{[m]}$ (by the bijection h between the bases). Formally, we denote this homomorphism by

$$H : \mathcal{M} \to \mathbb{F}^m .$$

Intuitively, this means that we think of a vector in $\mathbb{F}^m$ as a polynomial in $\mathcal{M}$, and vice versa. Each coordinate of the vector in $\mathbb{F}^m$ corresponds to the coefficient of one monomial in the polynomial.

Denote by $\mathcal{D}_{n,r}$, the set of homogenous circuit-graphs G (see Section 2), of syntactic-degree r, over the set of input variables $Z = \{z_1, \ldots, z_n\}$, such that G has a single output-gate and is in normal-depth-4-form (see Section 2). For a circuit-graph $G \in \mathcal{D}_{n,r}$, denote by $S(G)$, the number of product-gates in G at the level above the leaves (that is, the number of product-gates that are connected by an edge to a leaf).

5.2 The polynomial mapping $\Gamma_G : \mathbb{F}^s \to \mathbb{F}^m$

Let $G \in \mathcal{D}_{n,r}$. That is, G is a homogenous circuit-graph, of syntactic-degree r, over the set of input variables $Z = \{z_1, \ldots, z_n\}$, such that G has a single output-gate, and is in normal-depth-4-form. Denote, $s = S(G)$, that is, the number of product-gates in G at the level above the leaves. (Assume without loss of generality that $s \geq n$.)

Let Φ be an arithmetic circuit over $\mathbb{F}$, with circuit-graph $G_\Phi = G$. Without loss of generality, we assume that in the circuit Φ, all edges are labelled by 1, except for the edges that leave product-gates at the level above the leaves. (Otherwise, if an edge that reaches a product-gate at the level above the leaves is labelled by $\alpha \neq 1$, we just change its label to 1 and multiply the label of the edge that leaves that product-gate by α. Also, since G is a tree, we can assume without loss of generality that edges at upper levels are labelled by 1.) Denote the labels of the s edges that leave product-gates at the level above the leaves by $y_1, \ldots, y_s$.

(Intuitively, since Φ is in normal-depth-4-form, its computation can be presented as a homogenous sum, $\sum_i P_i Q_i$, where P_i, Q_i are homogenous polynomials of degree at most $2r/3$. Intuitively, the labels $y_1, \ldots, y_s$ are just the coefficients of all the monomials in all the polynomials P_i, Q_i.)

The circuit Φ computes a homogenous polynomial in $\mathbb{F}[Z]$ of total-degree exactly r, (that is, a polynomials in $\mathcal{M}$), where the coefficients in this polynomial depend on the labels $y_1, \ldots, y_s$. Since we think of a polynomial in $\mathcal{M}$ as a point in $\mathbb{F}^m$, we obtain for every point $(y_1, \ldots, y_s) \in \mathbb{F}^s$, a point in $\mathbb{F}^m$.

Formally, we define a mapping $\Gamma_G : \mathbb{F}^s \to \mathbb{F}^m$, as follows. Given $y_1, \ldots, y_s \in \mathbb{F}$, let Φ be an arithmetic circuit over $\mathbb{F}$, with circuit-graph $G_\Phi = G$, such that the labels of all edges in Φ are 1, except for edges that leave product-gates at the level above the leaves, and the labels of the s edges that leave product-gates at the level above the leaves are $y_1, \ldots, y_s$. Denote the polynomial computed by Φ by $g \in \mathcal{M}$ (note that this polynomial depends on the labels $y_1, \ldots, y_s$). Define

$$\Gamma_G(y_1, \ldots, y_s) = H(g) .$$

Note that the output of the circuit Φ can be viewed as a polynomial in both $z_1, \ldots, z_n$ and $y_1, \ldots, y_s$. That is, we can think of g as a polynomial in the input variables $z_1, \ldots, z_n$, with coefficients that are polynomials in the input variables $y_1, \ldots, y_s$. Therefore, the functions $(\Gamma_G)_1, \ldots, (\Gamma_G)_m$ are polynomials in $\mathbb{F}[y_1, \ldots, y_s]$. That is, Γ_G is a polynomial mapping. Moreover, it is straightforward to prove (formally, by induction on the circuit) that the polynomials $(\Gamma_G)_1, \ldots, (\Gamma_G)_m$ do not depend on the field $\mathbb{F}$, but only on its characteristic (intuitively, this is obvious because all the coefficients in these polynomials are derived by a sequence of sum and product operations on the constants 0,1, and are hence members of the minimal subfield of $\mathbb{F}$ that contains 0,1).

Proposition 5.1. *Let $G \in \mathcal{D}_{n,r}$. For every $g \in \mathcal{M}$, we have: $H(g) \in \mathrm{Image}(\Gamma_G)$ iff there exists an arithmetic circuit Φ, (over $\mathbb{F}$), with $G_\Phi = G$, for the polynomial g.*

Proof. If $H(g) \in \mathrm{Image}(\Gamma_G)$ then obviously, by the definition of Γ_G, there exists an arithmetic circuit Φ, (over $\mathbb{F}$), with $G_\Phi = G$, for the polynomial g.

If there exists an arithmetic circuit Φ, (over $\mathbb{F}$), with $G_\Phi = G$, for the polynomial g, without loss of generality, we assume that in the circuit Φ, all edges are labelled by 1, except for edges that leave product-gates at the level above the leaves.

Denote the labels of the s edges in Φ that leave product-gates at the level above the leaves by $\alpha_1, \ldots, \alpha_s$. Then, by the definition of Γ_G, we have $\Gamma_G(\alpha_1, \ldots, \alpha_s) = H(g)$. □

Proposition 5.2. *Let $G \in \mathcal{D}_{n,r}$. Then, the mapping $\Gamma_G : \mathbb{F}^s \to \mathbb{F}^m$ (where $s = S(G)$ and $m = \binom{n+r-1}{r}$) is a (homogenous) polynomial mapping of degree 2.*

Moreover, given G, one can construct Γ_G in time $\mathrm{poly}(s,m)$ in the following sense. There exists a $\mathrm{poly}(s,m)$-time Turing machine, that on input G outputs (all the coefficients of) the m polynomials $(\Gamma_G)_1, \ldots, (\Gamma_G)_m \in \mathbb{F}[y_1, \ldots, y_s]$, and such that all the coefficients in these polynomials are in $\{0,1\}$.

Proof. Let Φ be an arithmetic circuit over $\mathbb{F}$, with circuit-graph $G_\Phi = G$, such that in Φ, all edges are labelled by 1, except for edges that leave product-gates at the level above the leaves, and the labels of the s edges that leave product-gates at the level above the leaves are $y_1, \ldots, y_s$.

Since Φ is in normal-depth-4-form, its computation can be presented as a homogenous sum, $g = \sum_i P_i Q_i$, where P_i, Q_i are homogenous polynomials of degree at most $2r/3$. Note that the labels $y_1, \ldots, y_s$ are just the coefficients of all the monomials in all the polynomials P_i, Q_i.

Thus, the coefficients of the monomials of g are homogenous polynomials of degree 2 in the labels $y_1, \ldots, y_s$, and are in $\{0,1\}$. Moreover, these coefficients can be computed in time polynomial in s, m. $\square$

Proposition 5.3. *For every n, r, m, s' such that $3 \le r \le n \le s'$ and $m = \binom{n+r-1}{r}$, there exists a circuit-graph $G \in \mathcal{D}_{n,r}$, with $S(G) = O(s' \cdot \binom{n+r'-1}{r'} \cdot r^3)$, and $\mathrm{Size}(G) = O(s' \cdot \binom{n+r'-1}{r'} \cdot r^4)$, where $r' = \lfloor 2r/3 \rfloor$, such that:*

1. *$\Gamma_G : \mathbb{F}^s \to \mathbb{F}^m$ (where $s = S(G)$) is a (homogenous) polynomial mapping of degree 2.*
2. *For every $g \in \mathcal{M}$, if there exists an arithmetic circuit of size s' (over $\mathbb{F}$) for the polynomial g, then $H(g) \in \mathrm{Image}(\Gamma_G)$.*
3. *For every $g \in \mathcal{M}$, if $H(g) \in \mathrm{Image}(\Gamma_G)$, then there exists an arithmetic circuit Φ, (over $\mathbb{F}$), with $G_\Phi = G$, for the polynomial g.*

Moreover, one can construct G, Γ_G in time $\mathrm{poly}(s,m)$ in the following sense. There exists a $\mathrm{poly}(s,m)$-time Turing machine, that on input n, r, s', outputs G and (all the coefficients of) the m polynomials $(\Gamma_G)_1, \ldots, (\Gamma_G)_m \in \mathbb{F}[y_1, \ldots, y_s]$, and such that all the coefficients in these polynomials are in $\{0,1\}$.

Proof. Let G be the circuit-graph from Proposition 2.9, with parameters n, s', r, that is, a universal circuit-graph for n-inputs and one-output circuits of size s' that compute homogenous polynomials of degree r. Denote $s = S(G)$, and note that $S(G) = O(s' \cdot \binom{n+r'-1}{r'} \cdot r^3)$, and $\mathrm{Size}(G) = O(s' \cdot \binom{n+r'-1}{r'} \cdot r^4)$. By Proposition 2.9, G is in normal-depth-4-form, and note that G is of syntactic-degree r. Hence, by Proposition 5.2, $\Gamma_G : \mathbb{F}^s \to \mathbb{F}^m$ is a (homogenous) polynomial mapping of degree 2.

Let $g \in \mathcal{M}$. Assume that there exists an arithmetic circuit of size s' (over $\mathbb{F}$) for the polynomial g. Then, by Proposition 2.9, there exists an arithmetic circuit Φ, for the polynomial g, such that $G_\Phi = G$. Thus, by Proposition 5.1, $H(g) \in \mathrm{Image}(\Gamma_G)$.

The third claim is a special case of Proposition 5.1 (and is restated here for completeness). The moreover part follows immediately from the moreover parts of Proposition 2.9 and Proposition 5.2. $\square$

5.3 The polynomial $\tilde{f}$

Let $X = \{x_1, \ldots, x_n\}$ be an additional set of input variables. Let $f = (f_1, \ldots, f_m)$, where $f_1, \ldots, f_m \in \mathbb{F}[x_1, \ldots, x_n]$, be a polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$. Intuitively, since we think of a point in $\mathbb{F}^m$ as a polynomial in the set of variables Z, we can think of f as a polynomial in the sets of variables X, Z.

Formally, given f, we define a polynomial $\tilde{f} \in \mathbb{F}[X,Z]$, by

$$\tilde{f}(x_1, \ldots, x_n, z_1, \ldots, z_n) = \sum_{q \in M} f_{h(q)}(x_1, \ldots, x_n) \cdot q = \sum_{j \in [m]} f_j(x_1, \ldots, x_n) \cdot h^{-1}(j).$$

In other words, for every monomial q_x in the variables $\{x_1, \ldots, x_n\}$ and monomial $q_z \in M$, the coefficient of the monomial $q_x q_z$ in $\tilde{f}$ is simply the coefficient of the monomial q_x in $f_{h(q_z)}$. (For monomials q_x, q_z such that $q_z \notin M$, the coefficient of the monomial $q_x q_z$ in $\tilde{f}$ is 0.)

For $a = (a_1, \ldots, a_n) \in \mathbb{F}^n$, denote by $\tilde{f}|_a \in \mathbb{F}[Z]$, the polynomial $\tilde{f} \in \mathbb{F}[X,Z]$, after the substitution $x_1 = a_1, \ldots, x_n = a_n$.

Proposition 5.4. *$\forall a \in \mathbb{F}^n$, we have, $\tilde{f}|_a \in \mathcal{M}$, and $H(\tilde{f}|_a) = f(a)$.*

Proof. The proof is straightforward from the definitions. For every $a = (a_1, \ldots, a_n) \in \mathbb{F}^n$,

$$\tilde{f}|_a(z_1, \ldots, z_n) = \tilde{f}(a_1, \ldots, a_n, z_1, \ldots, z_n) = \sum_{j \in [m]} f_j(a) \cdot h^{-1}(j) \in \mathcal{M}.$$

Thus,

$$H\left(\tilde{f}|_a\right) = H\left(\sum_{j \in [m]} f_j(a) \cdot h^{-1}(j)\right) = \left(f_1(a), \quad \ldots \quad, f_m(a)\right) = f(a).$$

□

Proposition 5.5. *If $f = (f_1, \ldots, f_m)$ is $\operatorname{poly}(n)$-definable (see Definition 1.3), then the polynomial $\tilde{f} \in \mathbb{F}[X,Z]$ is $\operatorname{poly}(n)$-definable.*

Proof. Similar to the proof of Proposition 3.6. □

5.4 The route to lower bounds

In this subsection, we prove our main results for polynomial-mappings f that elude polynomial-mappings Γ of degree 2. The results are given by four propositions and corollaries.

All four propositions and corollaries are only interesting for $s < m$ (although this condition is not stated explicitly). Recall that we think of all the parameters (r, s, m, etc.) as functions of the basic parameter n.

Recall that given a polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$, and given a field extension $\mathbb{G} \supset \mathbb{F}$, we can think of f as a polynomial mapping $f : \mathbb{G}^n \to \mathbb{G}^m$. This is because we assume that a polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$ is given as a tuple $(f_1, \ldots, f_m)$ of polynomials in $\mathbb{F}[x_1, \ldots, x_n] \subset \mathbb{G}[x_1, \ldots, x_n]$ (see the discussion in Subsection 1.4).

Proposition 5.6. *For integers $3 \le r \le n \le s'$, and $m = \binom{n+r-1}{r}$, and $r' = \lfloor 2r/3 \rfloor$, let $\Gamma_G : \mathbb{F}^s \to \mathbb{F}^m$, where $s = O(s' \cdot \binom{n+r'-1}{r'} \cdot r^3)$, be the (homogenous) polynomial mapping of degree 2 from Proposition 5.3. Let $f : \mathbb{F}^n \to \mathbb{F}^m$ be a polynomial mapping.*

If over some field extension $\mathbb{G} \supseteq \mathbb{F}$,

$$\operatorname{Image}(f : \mathbb{G}^n \to \mathbb{G}^m) \not\subset \operatorname{Image}(\Gamma_G : \mathbb{G}^s \to \mathbb{G}^m),$$

then any arithmetic circuit (over $\mathbb{F}$) for the polynomial $\tilde{f} : \mathbb{F}^{2n} \to \mathbb{F}$ (explicitly defined from f in Subsection 5.3), is of size $> s'$.

Proof. Let us first prove the proposition for $\mathbb{G} = \mathbb{F}$.

By Proposition 5.3, for every $g \in \mathcal{M}$, if there exists an arithmetic circuit of size s' (over $\mathbb{F}$) for the polynomials g, then $H(g) \in \operatorname{Image}(\Gamma_G)$.

Assume for a contradiction that there exists an arithmetic circuit (over $\mathbb{F}$) of size s', for the polynomial $\tilde{f} : \mathbb{F}^{2n} \to \mathbb{F}$. By substituting in this circuit $x_1 = a_1, \ldots, x_n = a_n$, where $a = (a_1, \ldots, a_n) \in \mathbb{F}^n$, we obtain an arithmetic circuit of size s' for the polynomial $\tilde{f}|_a$, and note that by Proposition 5.4, $\tilde{f}|_a \in \mathcal{M}$. Hence, by Proposition 5.3, (for every $a_1,\dots,a_n \in \mathbb{F}$), $H(\tilde{f}|_a) \in \mathrm{Image}(\Gamma_G)$. Thus, by Proposition 5.4, for every $a_1,\dots,a_n \in \mathbb{F}$,

$$f(a_1,\dots,a_n) = H(\tilde{f}|_a) \in \mathrm{Image}(\Gamma_G)\,.$$

That is,

$$\mathrm{Image}(f) \subset \mathrm{Image}(\Gamma_G)\,.$$

Assume now that $\mathbb{G}$ is a general field, extending $\mathbb{F}$, and assume that

$$\mathrm{Image}(f : \mathbb{G}^n \to \mathbb{G}^m) \not\subset \mathrm{Image}(\Gamma_G : \mathbb{G}^s \to \mathbb{G}^m)\,.$$

By the part that we already proved (i. e., the case $\mathbb{G} = \mathbb{F}$), we know that any arithmetic circuit over $\mathbb{G}$, for the polynomial $\tilde{f}$, is of size $> s'$. But any arithmetic circuit over $\mathbb{F}$ is in particular an arithmetic circuit over $\mathbb{G}$. Thus, any arithmetic circuit over $\mathbb{F}$ for the polynomial $\tilde{f}$ is of size $> s'$. (Formally, we need to verify that the polynomials $\tilde{f}$ and $(\Gamma_G)_1,\dots,(\Gamma_G)_m$ remain the same polynomials when we work over $\mathbb{G}$, rather than over $\mathbb{F}$. This can be easily verified. For $(\Gamma_G)_1,\dots,(\Gamma_G)_m$, it was noted after the definition of Γ_G that they do not depend on the field at all, but only on its characteristic. As for $\tilde{f}$, by its definition, its coefficients are just corresponding coefficients from the polynomials $f_1,\dots,f_m$, and hence they do not depend on the field $\mathbb{G}$.) □

Corollary 5.7. *Let $3 \le r \le n \le s$, and $m = \binom{n+r-1}{r}$ be integers. Let $r' = \lfloor 2r/3 \rfloor$. Let $f : \mathbb{F}^n \to \mathbb{F}^m$ be a polynomial mapping. If over some field extension $\mathbb{G} \supseteq \mathbb{F}$, f is $(s,2)$-elusive (see Definition 1.1), then any arithmetic circuit (over $\mathbb{F}$) for the polynomial $\tilde{f} : \mathbb{F}^{2n} \to \mathbb{F}$ (explicitly defined from f in Subsection 5.3), is of size $\ge$*

$$\Omega\left(\frac{s}{\binom{n+r'-1}{r'} \cdot r^3}\right).$$

Proof. The proof follows immediately from Proposition 5.6. Let $s' = c \cdot s/(\binom{n+r'-1}{r'} \cdot r^3)$, where c is a small enough constant. If f is $(s,2)$-elusive, then in particular it satisfies $\mathrm{Image}(f) \not\subset \mathrm{Image}(\Gamma_G)$, where Γ_G is the mapping from Proposition 5.6. □

Corollary 5.8. *Let $\mathbb{F}$ be a field of characteristic not equal to 2. Let $3 \le r \le n \le s$, and $m = \binom{n+r-1}{r}$ be integers. Let $r' = \lfloor 2r/3 \rfloor$. Assume that $s/\binom{n+r'-1}{r'} \ge n^{\omega(1)}$. If there exists a* poly(n)*-definable polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$ such that, over some field extension $\mathbb{G} \supseteq \mathbb{F}$, f is $(s,2)$-elusive (see Definition 1.3 and Definition 1.1), then any arithmetic circuit for the permanent over $\mathbb{F}$ is of size $\ge$*

$$\left(\frac{s}{\binom{n+r'-1}{r'} \cdot r^3}\right)^{\Omega(1)}.$$

Proof. Assume that such a polynomial mapping f exists. By Proposition 5.5, and Corollary 5.7, the polynomial $\tilde{f} : \mathbb{F}^{2n} \to \mathbb{F}$ (explicitly defined from f in Subsection 5.3) is poly(n)-definable, and any arithmetic circuit (over $\mathbb{F}$) for $\tilde{f}$ is of size $\ge$

$$\Omega\left(\frac{s}{\binom{n+r'-1}{r'} \cdot r^3}\right).$$

Valiant proved that over any field of characteristic not equal to 2, the permanent is a complete polynomial for the class VNP of poly(n)-definable polynomials [36] (see also [10, 6]). Hence, any arithmetic circuit of size s' for the permanent implies an arithmetic circuit of size poly(s') for any other poly(n)-definable polynomial. Hence, any arithmetic circuit for the permanent over $\mathbb{F}$ is of size

$$s' \geq \left(\frac{s}{\binom{n+r'-1}{r'} \cdot r^3} \right)^{\Omega(1)} .$$

□

Corollary 5.9. *Let $3 \leq r \leq n \leq s$, and $m = \binom{n+r-1}{r}$ be integers (and recall that we think of r,s,m as functions of n). Let $r' = \lfloor 2r/3 \rfloor$. Assume that there exists a* poly(s,m)*-time Turing machine T such that:*

- *The inputs for T are r,n,s,m and a (homogenous) polynomial mapping $\Gamma : \mathbb{F}^s \to \mathbb{F}^m$ of degree 2, given by all coefficients of the polynomials $\Gamma_1,\dots,\Gamma_m$ (that are assumed to be in $\{0,1\}$).*
- *The output of T is a* poly(n)*-definable polynomial mapping $f : \mathbb{F}^n \to \mathbb{F}^m$ (described, e. g., by an arithmetic circuit for the polynomial g that defines it in Definition 1.3) such that, over some field extension $\mathbb{G} \supseteq \mathbb{F}$,*
$$\text{Image}(f : \mathbb{G}^n \to \mathbb{G}^m) \not\subseteq \text{Image}(\Gamma : \mathbb{G}^s \to \mathbb{G}^m) .$$

Then, there exists a poly(s,m)*-time Turing machine that on input n outputs a $2n$-variables,* poly(n)*-definable, polynomial $\tilde{f}$ (explicitly defined from f in Subsection 5.3, and described, e. g., by an arithmetic circuit for the polynomial g that defines it in Definition 1.3), such that any arithmetic circuit for $\tilde{f}$ is of size $\geq$*

$$\Omega\left(\frac{s}{\binom{n+r'-1}{r'} \cdot r^3} \right) .$$

Proof. The proof follows immediately from Proposition 5.6. Let $s' = c \cdot s/(\binom{n+r'-1}{r'} \cdot r^3)$, where c is a small enough constant. We run the Turing machine T on the polynomial mapping $\Gamma = \Gamma_G$, where Γ_G is the mapping from Proposition 5.6. Note that by Proposition 5.3, Γ_G can be constructed in time poly(s,m) (for details, see Proposition 5.3). □

Acknowledgement

I am grateful to Zeev Dvir, Yael Tauman Kalai, Toni Pitassi, Omer Reingold, Amir Shpilka, Avi Wigderson and Amir Yehudayoff, for very helpful conversations and comments at different stages of this work.

References

[1] MANINDRA AGRAWAL AND V. VINAY: Arithmetic circuits: A chasm at depth four. In *Proc. 49th FOCS*, pp. 67–75. IEEE Comp. Soc., 2008. [doi:10.1109/FOCS.2008.32]. 144, 152

[2] M. AJTAI: Σ_1^1-formulae on finite structures. *Ann. Pure Appl. Logic*, 24:1–48, 1983. [doi:10.1016/0168-0072(83)90038-6]. 138

[3] MICHAEL ALEKHNOVICH, ELI BEN-SASSON, ALEXANDER A. RAZBOROV, AND AVI WIGDERSON: Pseudorandom generators in propositional proof complexity. *SIAM J. Comput.*, 34(1):67–88, 2004. A preliminary version appeared in FOCS 2000. [doi:10.1137/S0097539701389944]. 144

[4] MICHAEL ALEKHNOVICH AND ALEXANDER A. RAZBOROV: Lower bounds for the polynomial calculus: Non-binomial case. *Proc. Steklov Inst. Math.*, 242:18–35, 2003. A preliminary version appeared in FOCS 2001. [doi:10.1109/SFCS.2001.959893]. 144

[5] WALTER BAUR AND VOLKER STRASSEN: The complexity of partial derivatives. *Theoret. Comput. Sci.*, 22(3):317–330, 1983. [doi:10.1016/0304-3975(83)90110-X]. 138, 143, 156, 160, 162, 167

[6] PETER BÜRGISSER: *Completeness and Reduction in Algebraic Complexity Theory*. Springer, Berlin, 2000. 142, 163, 173

[7] PETER BÜRGISSER, MICHEL CLAUSEN, AND MOHAMMAD A. SHOKROLLAHI: *Algebraic Complexity Theory*. Springer, Berlin, 1997. 138

[8] DANNY DOLEV, CYNTHIA DWORK, NICHOLAS PIPPENGER, AND AVI WIGDERSON: Superconcentrators, generalizers and generalized connectors with limited depth. In *Proc. 15th STOC*, pp. 42–51. ACM, 1983. [doi:10.1145/800061.808731]. 139

[9] MERRICK L. FURST, JAMES B. SAXE, AND MICHAEL SIPSER: Parity, circuits, and the polynomial-time hierarchy. *Theory Comput. Syst.*, 17(1):13–27, 1984. A preliminary version appeared in FOCS 1981. [doi:10.1007/BF01744431]. 138

[10] JOACHIM VON ZUR GATHEN: Feasible arithmetic computations: Valiant's hypothesis. *J. Symbolic Comput.*, 4(2):137–172, 1987. [doi:10.1016/S0747-7171(87)80063-9]. 142, 163, 173

[11] JOACHIM VON ZUR GATHEN: Algebraic complexity theory. *Ann. Rev. Comput. Sci.*, 3:317–347, 1988. [doi:10.1146/annurev.cs.03.060188.001533]. 138

[12] DIMA GRIGORIEV AND MAREK KARPINSKI: An exponential lower bound for depth 3 arithmetic circuits. In *Proc. 30th STOC*, pp. 577–582, 1998. [doi:10.1145/276698.276872]. 138, 139

[13] D. GRIGORIEV AND A. RAZBOROV: Exponential lower bounds for depth 3 arithmetic circuits in algebras of functions over finite fields. *Appl. Algebra Engrg. Comm. Comput.*, 10(6):465–487, 2000. A preliminary version appeared in FOCS 1998. [doi:10.1007/s002009900021]. 138, 139

[14] J. HÅSTAD: Almost optimal lower bounds for small depth circuits. *Adv. Comput. Res.*, 5:143–170, 1989. A preliminary version appeared in STOC 1986. [doi:10.1145/12130.12132]. 138

[15] VALENTINE KABANETS AND RUSSELL IMPAGLIAZZO: Derandomizing polynomial identity tests means proving circuit lower bounds. *Comput. Complexity*, 13(1–2):1–46, 2004. A preliminary version appeared in STOC 2003. [doi:10.1007/s00037-004-0182-6]. 144

[16] RICHARD J. LIPTON: Polynomials with 0-1 coefficients that are hard to evaluate. *SIAM J. Comput.*, 7(1):61–69, 1978. A preliminary version appeared in FOCS 1975. [doi:10.1137/0207004]. 143

[17] SATYANARAYANA V. LOKAM: Spectral methods for matrix rigidity with applications to size-depth tradeoffs and communication complexity. *J. Comput. System Sci.*, 63:449–473, 2001. A preliminary version appeared in FOCS 1995. [doi:10.1006/jcss.2001.1786]. 141

[18] JACQUES MORGENSTERN: How to compute fast a function and all its derivatives: A variation on the theorem of Baur-Strassen. *ACM SIGACT News*, 16:60–62, 1985. [doi:10.1145/382242.382836]. 143, 162, 167

[19] NOAM NISAN: Lower bounds for non-commutative computation. In *Proc. 13th STOC*, pp. 410–418. ACM, 1991. [doi:10.1145/103418.103462]. 138

[20] NOAM NISAN AND AVI WIGDERSON: On the complexity of bilinear forms. In *Proc. 27th STOC*, pp. 723–732. ACM, 1995. [doi:10.1145/225058.225290]. 141

[21] P. PUDLÁK: Communication in bounded depth circuits. *Combinatorica*, 14:203–216, 1994. [doi:10.1007/BF01215351]. 139

[22] PAVEL PUDLÁK: A note on the use of determinant for proving lower bounds on the size of linear circuits. *Inform. Process. Lett.*, 74:197–201, 2000. [doi:10.1016/S0020-0190(00)00058-2]. 141

[23] RAN RAZ: On the complexity of matrix product. *SIAM J. Comput.*, 32(5):1356–1369, 2003. A preliminary version appeared in STOC 2002. [doi:10.1137/S0097539702402147]. 141

[24] RAN RAZ: Separation of multilinear circuit and formula size. *Theory Comput.*, 2(6):121–135, 2006. A preliminary version appeared in FOCS 2004 with the title "Multilinear-$NC_1 \neq$ Multilinear-NC_2". [doi:10.4086/toc.2006.v002a006]. 138

[25] RAN RAZ: Multi-linear formulas for permanent and determinant are of super-polynomial size. *J. ACM*, 56, 2009. A preliminary version appeared in STOC 2004. [doi:10.1145/1502793.1502797]. 138

[26] RAN RAZ: Tensor-rank and lower bounds for arithmetic formulas. In *Proc. 42nd STOC*, pp. 659–666. ACM, 2010. [doi:10.1145/1806689.1806780]. 144

[27] RAN RAZ AND AMIR SHPILKA: Lower bounds for matrix product in bounded depth circuits with arbitrary gates. *SIAM J. Comput.*, 32(2):488–513, 2003. A preliminary version appeared in STOC 2001. [doi:10.1137/S009753970138462X]. 139

[28] RAN RAZ AND AMIR YEHUDAYOFF: Multilinear formulas, maximal-partition discrepancy and mixed-sources extractors. *J. Comput. System Sci.*, 2010. A preliminary version appeared in FOCS 2008. [doi:10.1016/j.jcss.2010.06.013]. 152

[29] A. A. RAZBOROV: Lower bounds on the size of bounded-depth networks over a complete basis with logical addition (in Russian). *Mat. Zametki*, 41(4):598–607, 1987. An English translation appeared in Mathematical Notes of the Academy of Sci. of the USSR 41(4) (1987) 333–338. 138

[30] A. A. RAZBOROV: Bounded arithmetic and lower bounds in Boolean complexity. In *Feasible Mathematics II*, volume 13 of *Progr. Comput. Sci. Appl. Logic*, pp. 344–386. Birkhäuser, 1995. 144

[31] VICTOR SHOUP AND ROMAN SMOLENSKY: Lower bounds for polynomial evaluation and interpolation problems. In *Proc. 32nd FOCS*, pp. 378–383. IEEE Comp. Soc., 1991. [doi:10.1109/SFCS.1991.185394]. 139, 144

[32] A. SHPILKA AND A. WIGDERSON: Depth-3 arithmetic circuits over fields of characteristic zero. *Comput. Complexity*, 10(1):1–27, 2001. A preliminary version appeared in Conference on Computational Complexity 1999. [doi:10.1007/PL00001609]. 139

[33] R. SMOLENSKY: Algebraic methods in the theory of lower bounds for Boolean circuit complexity. In *Proc. 19th STOC*, pp. 77–82. ACM, 1987. [doi:10.1145/28395.28404]. 138

[34] VOLKER STRASSEN: Polynomials with rational coefficients which are hard to compute. *SIAM J. Comput.*, 3(2):128–149, 1974. [doi:10.1137/0203010]. 143

[35] VOLKER STRASSEN: Die Berechnungskomplexität der symbolischen Differentiation von Interpolationspolynomen. *Theoret. Comput. Sci.*, 1(1):21–25, 1975. [doi:10.1016/0304-3975(75)90010-9]. 138

[36] L. G. VALIANT: Completeness classes in algebra. In *Proc. 11th STOC*, pp. 249–261, 1979. [doi:10.1145/800135.804419]. 142, 163, 173

[37] 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. [doi:10.1137/0212043]. 140, 144, 152

[38] ANDREW CHI-CHIH YAO: Separating the polynomial-time hierarchy by oracles. In *Proc. 26th FOCS*, pp. 1–10. IEEE Comp. Soc., 1985. [doi:10.1109/SFCS.1985.49]. 138

AUTHOR

Ran Raz
professor
Department of Computer Science and Applied Mathematics
Weizmann Institute of Science
Rehovot 76100, Israel
ran.raz@weizmann.ac.il
http://www.wisdom.weizmann.ac.il/~ranraz/

ABOUT THE AUTHOR

RAN RAZ received his Ph. D. in 1992 from the Hebrew University under the supervision of Michael Ben-Or and Avi Wigderson. Since 1994, he has been a faculty member in the Faculty of Mathematics and Computer Science at the Weizmann Institute. His main research area is complexity theory, with emphasis on proving lower bounds for computational models. More specifically, he is interested in Boolean circuit complexity, arithmetic circuit complexity, communication complexity, propositional proof theory, probabilistically checkable proofs, quantum computation and communication, and randomness and derandomization.