On algebraic branching programs of small width

Karl Bringmann¹  Christian Ikenmeyer¹  Jeroen Zuiddam²

February 17, 2017

Abstract

In 1979 Valiant showed that the complexity class $\textbf{VP}_e$ of families with polynomially bounded formula size is contained in the class $\textbf{VP}_s$ of families that have algebraic branching programs (ABPs) of polynomially bounded size. Motivated by the problem of separating these classes we study the topological closure $\textbf{VP}_e$, i.e. the class of polynomials that can be approximated arbitrarily closely by polynomials in $\textbf{VP}_e$. We describe $\textbf{VP}_e$ with a strikingly simple complete polynomial (in characteristic different from 2) whose recursive definition is similar to the Fibonacci numbers. Further understanding this polynomial seems to be a promising route to new formula lower bounds.

Our methods are rooted in the study of ABPs of small constant width. In 1992 Ben-Or and Cleve showed that formula size is polynomially equivalent to width-3 ABP size. We extend their result (in characteristic different from 2) by showing that approximate formula size is polynomially equivalent to approximate width-2 ABP size. This is surprising because in 2011 Allender and Wang gave explicit polynomials that cannot be computed by width-2 ABPs at all! The details of our construction lead to the aforementioned characterization of $\textbf{VP}_e$.

As a natural continuation of this work we prove that the class $\textbf{VNP}$ can be described as the class of families that admit a hypercube summation of polynomially bounded dimension over a product of polynomially many affine linear forms. This gives the first separations of algebraic complexity classes from their nondeterministic analogs.

¹Max-Planck-Institut für Informatik, Saarland Informatics Campus, Germany. Email: kbringma@mpi-inf.mpg.de,cikenmey@mpi-inf.mpg.de

²Centrum Wiskunde & Informatica, Amsterdam, Netherlands. Email: j.zuiddam@cwi.nl. Supported by NWO through the research programme 617.023.116.
1. Introduction

Let $\text{VP}_e$ denote the class of families of polynomials with polynomially bounded formula size and let $\text{VP}_s$ denote the class of families of polynomials that can be written as determinants of matrices of polynomially bounded size whose entries are affine linear forms. In 1979 Valiant [Val79] proved his famous result $\text{VP}_e \subseteq \text{VP}_s$. The question whether this inclusion is strict is a long-standing open question in algebraic complexity theory: Can the determinant polynomial $\text{det}_n := \sum_{\sigma \in S_n} \text{sgn}(\sigma) \prod_{i=1}^n x_{i, \sigma(i)}$ be computed by formulas of polynomially bounded size? Motivated by this question we study the class $\text{VP}_e$ of families of polynomials that can be approximated arbitrarily closely by families in $\text{VP}_s$ (see Sec. 2 for a formal definition). We present a simple description of the closure $\text{VP}_e$ and of a $\text{VP}_e$-complete polynomial whose recursive definition is similar to the Fibonacci numbers, given the characteristic is not 2, see Theorem 3.10.

In algebraic complexity theory, the way of showing a complexity lower bound for a problem $f \in V$ for some $F$-vector space $V$ most often goes by (implicitly or explicitly) finding a function $F : V \to \mathbb{F}$ that is zero on all problems of low complexity while at the same time $F(f) \neq 0$. Grochow [Gro15] gives a long list (e.g., [NW97, Raz09, LO15, GKKS13a, LMR13, BI13]) of settings where complexity lower bounds are obtained in this way. Moreover, he points out that over the complex numbers these functions $F$ can be obtained as $\text{border Fibonacci complexity}$ of $f$ can be obtained as $3 \times 3$ iterated matrix multiplication polynomial [BOC92]. This immediately motivates the definition of border Fibonacci complexity $L_{\text{Fib}}(f)$ of a polynomial $f$, which is the smallest number $m$ such that $f$ can be obtained as $\lim_{j \to \infty} (F_{m}(t_{1}(j), \ldots, t_{m}(j)))$. To make the situation more geometric we allow the $t_{i}(j)$ to be arbitrary affine linear forms. Our results show that border Fibonacci complexity is polynomially equivalent to border formula size. This insight is quite striking because a result of Allender and Wang [AW16] implies that the Fibonacci complexity without allowing approximations can be infinite!

A promising path towards proving formula lower bounds, for example for the determinant or the permanent, is to apply to our setting the following standard geometric ideas. If we take our field to be the complex numbers and fix the number of variables $n$ and the degree $d$, then the set of
homogeneous degree \( d \) polynomials \( \mathbb{C}[x_1, \ldots, x_n]_d \) contains the set
\[
X_m := \{ f \in \mathbb{C}[x_1, \ldots, x_n]_d \mid L_{\text{Fib}}(f) \leq m \}
\]
as an affine subvariety. Moreover, since we allowed the \( \ell_i(j) \) to be affine linear forms, the group \( \text{GL}(\mathbb{C}^n) \) acts canonically on \( X_m \), making \( X_m \) an affine \( \text{GL}(\mathbb{C}^n) \)-variety. If we find a polynomial \( F \) that vanishes identically on \( X_m \), then a nonzero evaluation \( F(f) \neq 0 \) implies that \( L_{\text{Fib}}(f) > m \). This approach looks feasible given the very simple structure of the generalized Fibonacci polynomial. This is emphasized by the fact that the action of \( \text{GL}(\mathbb{C}^n) \) puts a lot of structure on the coordinate ring of \( X_m \), see for example [BI11, BO11, LO15, BI13, HIL13, Gua15, OS16] where the action of the general linear group on the coordinate ring of a variety is used to classify some of its defining equations.

1.1. Main Results

(I) Algebraic Branching Programs (ABPs) of width 2 Our main objects of study are the following classes of families of polynomials: the class of families of polynomials with polynomially bounded formula size \( \text{VP}_e \) (fan-in 2 arithmetic formulas that use additions and multiplications as their operations), its closure \( \overline{\text{VP}}_e \), and the nondeterministic variant \( \text{VNP} \). We do so by studying algebraic branching programs of small width. These are defined as follows. An algebraic branching program (ABP) is a directed acyclic graph with a source vertex \( s \) and a sink vertex \( t \) that has affine linear forms over the base field \( F \) as edge labels. Moreover, we require that each vertex is labeled with an integer (its layer) and that edges in the ABP only point from vertices in layer \( i \) to vertices in layer \( i + 1 \). The width of an ABP is the cardinality of its largest layer. The size of an ABP is the number of its vertices. The value of an ABP is the sum of the values of all \( s-t \)-paths, where the value of an \( s-t \)-path is the product of its edge labels. We say that an ABP computes its value. The class \( \text{VP}_s \) coincides with the class of families of polynomials that can be computed by ABPs of polynomially bounded size, see e.g. [Sap16].

For this paper we introduce the class \( \text{VP}_k \), \( k \in \mathbb{N} \), which is defined as the class of families of polynomials computable by width-\( k \) ABPs of polynomially bounded size. It is well-known that \( \text{VP}_k \subseteq \text{VP}_e \) for every \( k \geq 1 \) (see Prop. C.1). In 1992, Ben-Or and Cleve [BOC92] showed that \( \text{VP}_k = \text{VP}_e \) for all \( k \geq 3 \) (we review the proof, see Thm. B.1). In 2011 Allender and Wang [AW16] showed that width-2 ABPs cannot compute every polynomial, so in particular we have a strict inclusion \( \text{VP}_2 \subsetneq \text{VP}_3 \). Let the characteristic of the base field \( F \) be different from 2. Our first main result (Thm. 3.1 and Cor. 3.7) is that the closure of \( \text{VP}_2 \) and the closure of \( \overline{\text{VP}}_e \) are equal,
\[
\overline{\text{VP}}_2 = \overline{\text{VP}}_e.
\]

Interestingly, as a direct corollary of (1) and the result of Allender and Wang, the inclusion \( \text{VP}_2 \subsetneq \overline{\text{VP}}_2 \) is strict. It is easy to see that \( \text{VP}_1 \) equals \( \overline{\text{VP}}_1 \) (Prop. A.12), so \( \text{VP}_1 \) and \( \text{VP}_2 \) are examples of quite similar algebraic complexity classes that behave differently under closure. Most importantly, from the proof of (1) we obtain our results about the generalized Fibonacci polynomial that we mentioned before.

(II) VNP via affine linear forms We define the classes \( \text{VNP}_e \) and \( \text{VNP} \) in the natural way. In 1980, Valiant [Val80] showed that \( \text{VNP}_e = \text{VNP} \) and in this paper we will always view \( \text{VNP} \)
as the nondeterministic analog of $\mathbf{VP}_e$. To $\mathbf{VP}_1$ and $\mathbf{VP}_2$ we similarly associate nondeterministic analogs $\mathbf{VNP}_1$ and $\mathbf{VNP}_2$ (see Sec. 2). Using interpolation techniques it is possible to deduce $\mathbf{VNP}_2 = \mathbf{VNP}$ from (1), provided the field is infinite. Using more sophisticated techniques we strengthen this result to get our second main result (Thm. 4.2):

$$\mathbf{VNP}_1 = \mathbf{VNP}.$$  

That is, a family $(f_n)$ is contained in $\mathbf{VNP}$ iff $f_n$ can be written as a hypercube summation of polynomially bounded dimension over a product of polynomially many affine linear forms. Using (2) it is then easy to verify that $\mathbf{VP}_1 \subsetneq \mathbf{VNP}_1$ and using [AW16] yields $\mathbf{VP}_2 \subsetneq \mathbf{VNP}_2$, which separates complexity classes from their nondeterministic analogs. Interestingly $\mathbf{VNP}_1 \subsetneq \mathbf{VNP}$ over the field with 2 elements, see Appendix E.

(III) Restricted ABP edge labels  Several more results on small-width ABPs, approximation closures, and hypercube summations are proved throughout this paper. For example, in Appendix A we investigate the subtleties of what happens if we restrict the ABP edge labels to simple affine linear forms, or to variables and constants. The precise relations between complexity classes that we obtain are listed in Figure F.4 in Appendix F. As another example, we strengthen (2) as follows (Thm. B.3): A family $(f_n)$ is contained in $\mathbf{VNP}$ iff $f_n$ can be written as a hypercube summation of polynomially bounded dimension over a product of polynomially many affine linear forms that use at most two variables each.

1.2. Related work

In the boolean setting as well as in the algebraic setting finding lower bounds for the formula size of explicit problems is considered a major open problem. For the boolean setting we refer the reader to the line of papers [Sub61, And87, IN93, PZ93, Hås98, Tal14], which results in an explicit function with formula size $\Omega(n^3/(\log^2 n \log \log n))$.

In the algebraic setting the smallest formula for the determinant has size $O(n^{\log n})$, which can be deduced from e.g. [HT15]. The best known lower bound on the formula size of $\det_n$ is $\Omega(n^3)$ by [Kal85]. That paper also gives a quadratic lower bound for an explicit polynomial (note that the lower bound for the determinant is not quadratic in the number of variables).

[Tod92] proves that several definitions for the class $\mathbf{VP}_e$ are equivalent, see also [MP08]. In particular $\mathbf{VP}_e$ is the class of polynomials that can be written as determinants of matrices of polynomially bounded size whose entries are affine linear forms. Due to its pure mathematical formulation, lower bounds for this determinantal complexity attracted the attention of geometers [MR04, LMR13, ABV15]. Moreover, Mulmuley and Sohoni’s geometric complexity approach [MS01, MS08] is also mainly focused on lower bounds for the determinantal complexity and the symmetries of the determinant polynomial play a key role in their work. Recently [BIP16] showed that it is not possible to prove superpolynomial lower bounds on the determinantal complexity using only information about the occurrences/non-occurrences of irreducible representations in the coordinate rings of the orbit closures of the determinant and the (padded) permanent. This disproves a major conjecture in geometric complexity theory. The proof in [BIP16] is fairly general and also holds for lower bounds on the formula size. Only very recently the formula size analog to determinantal complexity, the iterated matrix multiplication complexity was studied from a geometric perspective [Ges16].
There is a large number of publications on lower bounds for constant depth circuits and formulas (with superconstant fan-in), see e.g. [AV08, Koi12, GKKS13b, Tav15], which recently led to the celebrated result [GKKS13a] that the permanent does not admit size \(2^{o(\sqrt{m})}\) homogeneous \(\Sigma\Pi\Sigma\Pi\) circuits in which the bottom fan-in is bounded by \(\sqrt{m}\). In the light of the previous depth-reduction results this seemed very close to separating \(\text{VP}\) from \(\text{VNP}\). Very recently it was shown [ELSW16] that new ideas are needed to separate \(\text{VP}\) from \(\text{VNP}\).

[BOC92] prove that a family of polynomials has polynomially bounded formula size iff it is computable by width-3 ABPs of polynomial size. An excellent exposition on the history of small-width computation can be found in [AW16], along with an explicit polynomial that cannot be computed by width-2 ABPs.

Bürgisser [Bür04] studies approximations in the model of general algebraic circuits, finding general upper bounds on the error degree. For most specific algebraic complexity classes \(C\) the relation between \(C\) and \(\overline{C}\) has not been an active object of study. As pointed out recently by Forbes [For16], Nisan’s result [Nis91] implies that \(C = \overline{C}\) for \(C\) being the class of size-\(k\) algebraic branching programs on noncommuting variables. Recently, a structured study of \(\overline{\text{VP}}\) and \(\overline{\text{VP}}_e\) has started, see [GMQ16]. By far the most work in lower bounds for topological approximation algorithms has been done in the area of bilinear complexity, dating back to [BCRL79, Str83, Lic84] and more recently [Lan06, LO15, HIL13, Zui15, LM16], to list a few.

Paper outline  In Sec. 2 we introduce in more detail the approximation closure and the nondeterminism closure of a complexity class. In Sec. 3 we prove the first main result: border formula size is polynomially equivalent to border width-2 ABP size and the generalized Fibonacci polynomial is \(\overline{\text{VP}}_e\)-complete under p-degenerations. In Sec. 4 we prove the second main result: a new description of \(\text{VNP}\) as the nondeterminism closure of families that have polynomial-size width-1 ABPs. The appendix contains details on how to strengthen the result from Sec. 4 and results on the power of ABPs with restricted edge labels.

2. Nondeterminism and approximation closure

In this section we introduce the approximation closure and the nondeterminism analog of a class. A family is a sequence of polynomials \((f_n)_{n \in \mathbb{N}}\). A class is a set of families and will be written in boldface, \(C\). For an introduction to the algebraic complexity classes \(\text{VP}_e, \text{VP}, \text{VNP}\) we refer the reader to [BCS97]. We denote by \(\text{poly}(n)\) the set of polynomially bounded functions \(\mathbb{N} \to \mathbb{N}\). We define the norm of a complex multivariate polynomial as the sum of the absolute values of its coefficients. This defines a topology on the polynomial ring \(\mathbb{C}[x_1, \ldots, x_m]\). Given a complexity measure \(L\), say ABP size or formula size, there is a natural notion of approximate complexity that is called border complexity. Namely, a polynomial \(f \in \mathbb{C}[x]\) has border complexity \(L^{\text{top}}\) at most \(c\) if there is a sequence of polynomials \(g_1, g_2, \ldots\) in \(\mathbb{C}[x]\) converging to \(f\) such that each \(g_i\) satisfies \(L(g_i) \leq c\). It turns out that for reasonable classes over the field of complex numbers \(\mathbb{C}\), this topological notion of approximation is equivalent to what we call algebraic approximation (see e.g. [Bür04]). Namely, a polynomial \(f \in \mathbb{C}[x]\) satisfies \(L(f)^{\text{alg}} \leq c\) iff there are polynomials \(f_1, \ldots, f_e \in \mathbb{C}[x]\) such that the polynomial

\[ h := f + \varepsilon f_1 + \varepsilon^2 f_2 + \cdots + \varepsilon^e f_e \in \mathbb{C}[\varepsilon, x] \]
has complexity $L_{\mathbb{C}(\epsilon)}(h) \leq c$, where $\epsilon$ is a formal variable and $L_{\mathbb{C}(\epsilon)}(h)$ denotes the complexity of $h$ over the field extension $\mathbb{C}(\epsilon)$. This algebraic notion of approximation makes sense over any base field and we will use it in the statements and proofs of this paper.

**Definition 2.1.** Let $\mathcal{C}(\mathbb{F})$ be a class over the field $\mathbb{F}$. We define the *approximation closure* $\overline{\mathcal{C}}(\mathbb{F})$ as follows: a family $(f_n)$ over $\mathbb{F}$ is in $\overline{\mathcal{C}}(\mathbb{F})$ if there are polynomials $f_{n;i}(x) \in \mathbb{F}[x]$ and a function $e : \mathbb{N} \to \mathbb{N}$ such that the family $(g_n)$ defined by

$$g_n(x) := f_n(x) + \epsilon f_{n;1}(x) + \epsilon^2 f_{n;2}(x) + \cdots + \epsilon^{e(n)} f_{n;e(n)}(x)$$

is in $\mathcal{C}(\mathbb{F}(\epsilon))$. We define the *poly-approximation closure* $\overline{\mathcal{C}}^{poly}(\mathbb{F})$ similarly, but with the additional requirement that $e(n) \in \text{poly}(n)$. We call $e(n)$ the *error degree*.

Interestingly, for subtle reasons, taking the approximation closure $\mathcal{C} \mapsto \overline{\mathcal{C}}$ is *not* idempotent in general and hence not a closure operator, but for reasonable classes (like $\text{VP}_k$, $\text{VP}_e$, and $\text{VP}$) it is.

One can think of $\text{VNP}$ as a “nondeterminism closure” of $\text{VP}$. We want to use the nondeterminism closure for general classes.

**Definition 2.2.** Let $\mathcal{C}$ be a class. The class $\text{N}(\mathcal{C})$ consists of families $(f_n)$ with the following property: there is a family $(g_n) \in \mathcal{C}$ and $p(n), q(n) \in \text{poly}(n)$ such that

$$f_n(x) = \sum_{b \in \{0,1\}^{p(n)}} g_q(b, x),$$

where $x$ and $b$ denote sequences of variables $x_1, x_2, \ldots$ and $b_1, b_2, \ldots, b_{p(n)}$. We will sometimes say that $f(x)$ is a hypercube sum over $g$ and that $b_1, b_2, \ldots, b_{p(n)}$ are the hypercube variables. For any $s, t$, we will use the standard notation $\text{VNP}_s^t$ to denote $\text{N}(\text{VP}_s^t)$. We remark that the map $\mathcal{C} \mapsto \text{N}(\mathcal{C})$ trivially satisfies all properties of being a closure operator.

### 3. Approximate width-2 ABPs and formula size

As mentioned in the introduction, Allender and Wang [AW16] showed that there exist polynomials that cannot be computed by any width-2 ABP. E.g., the polynomial $x_1 x_2 + x_3 x_4 + \cdots + x_{15} x_{16}$. Therefore, we have a separation $\text{VP}_2 \subset \text{VP}_3 = \text{VP}_e$. We show that allowing approximation changes the situation completely: every polynomial can be approximated by a width-2 ABP. In fact, every polynomial can be approximated by a width-2 ABP of size polynomial in the formula size, and with error degree polynomial in the formula size. That is the main result of this section.

**Theorem 3.1.** $\text{VP}_e \subset \text{VP}_2^{poly}$ when $\text{char}(\mathbb{F}) \neq 2$.

We leave as an open question what happens in characteristic 2.

The proof of Thm. 3.1 consists of several lemmas that are supported by figures that can be found in Appendix F. For a polynomial $f$ over $\mathbb{F}(\epsilon)$ define the matrix $Q(f) := \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$. A *parametrized affine linear form* is an affine linear form over the field $\mathbb{F}(\epsilon)$. A *primitive Q-matrix* is any matrix $Q(\ell)$, where $\ell$ is a parametrized linear form. For a $2 \times 2$ matrix $M$ we use the shorthand notation $M + O(\epsilon)$ for $M + \begin{pmatrix} O(\epsilon) & O(\epsilon) \\ O(\epsilon) & O(\epsilon) \end{pmatrix}$. 


In order to understand the following proofs and the corresponding figures in Appendix F it is advisable to recall that an ABP corresponds naturally to an iterated product of matrices if we number the vertices in each layer consecutively, starting with 1. Namely, consider two consecutive layers $i$ and $i+1$ and let $M_i$ be the matrix whose entry at position $(v, w)$ is the label of the edge from vertex $v$ in layer $i$ to vertex $w$ in layer $i+1$ (or 0 if there is no edge between these vertices). Then the ABP’s value equals the product $M_k \cdots M_2 M_1$.

**Lemma 3.2** (Addition). Let $k \geq 1$. Let $f, g \in \mathbb{F}[x]$ be polynomials such that some $F \in Q(f) + O(\varepsilon^k)$ and some $G \in Q(g) + O(\varepsilon^k)$ can be written as a product of $n$ and $m$ primitive $Q$-matrices, respectively. Then some matrix $H \in Q(f + g) + O(\varepsilon^k)$ can be written as the product of $n + m + 1$ primitive $Q$-matrices. Moreover, if the error degrees in $F, G$ are $e_f, e_g$, respectively, then the error degree of $H$ is at most $e_f + e_g$.

**Proof.** Note that $(Q(f) + O(\varepsilon^k)) \cdot Q(0) \cdot (Q(g) + O(\varepsilon^k)) = Q(f + g) + O(\varepsilon^k)$, so $H := F \cdot Q(0) \cdot G \in Q(f + g) + O(\varepsilon^k)$. Moreover, the largest power of $\varepsilon$ occurring in $H$ is $\varepsilon^{e_f + e_g}$. See Fig. F.1 in Appendix F for a pictorial description.

**Lemma 3.3** (Squaring). Let $f \in \mathbb{F}[x]$ be a polynomial such that some $F \in Q(f) + O(\varepsilon^3)$ can be written as the product of $n$ primitive $Q$-matrices. Then some matrix $H \in Q(f^2) + O(\varepsilon)$ and some matrix $H' \in Q(-f^2) + O(\varepsilon)$ can be written as the product of $2n + 11$ primitive $Q$-matrices. Moreover, if the error degree in $F$ is $e_f$ then the error degree of $H$ and $H'$ is at most $2 \cdot e_f + 4$.

**Proof.** We set

$$A := \begin{pmatrix} -\varepsilon^{-1} & 0 \\ 0 & \varepsilon \end{pmatrix} = Q(-\varepsilon^{-1}) \cdot Q(\varepsilon) \cdot Q(-\varepsilon^{-1}),$$

$$B := \begin{pmatrix} \varepsilon^2 & 1 \\ -1 & 0 \end{pmatrix} = Q(1) \cdot Q(-1) \cdot Q(1) \cdot Q(\varepsilon^2),$$

$$C := \begin{pmatrix} \varepsilon^{-1} & 0 \\ 0 & \varepsilon \end{pmatrix} = Q(-\varepsilon^{-1}) \cdot Q(\varepsilon - 1) \cdot Q(1) \cdot Q(\varepsilon^{-1} - 1).$$

Then one can check that

$$H := A \cdot F \cdot B \cdot F \cdot C \in A \cdot (Q(f) + O(\varepsilon^3)) \cdot B \cdot (Q(f) + O(\varepsilon^3)) \cdot C \in Q(-f^2) + O(\varepsilon).$$

To obtain $H' \in Q(f^2) + O(\varepsilon)$, we replace $B$ by

$$B' := \begin{pmatrix} -\varepsilon^{-2} & 1 \\ -1 & 0 \end{pmatrix} = Q(1) \cdot Q(-1) \cdot Q(1) \cdot Q(-\varepsilon^2).$$

One checks that the highest power of $\varepsilon$ appearing in $H$ and $H'$ is at most $2 \cdot e_f + 4$. See Fig. F.2 in Appendix F for a pictorial description.

**Lemma 3.4** (Multiplication). Let $f, g \in \mathbb{F}[x]$ be polynomials such that some $F \in Q(f/2) + O(\varepsilon^3)$ and some $G \in Q(g) + O(\varepsilon^3)$ can be written as the product of $n$ and $m$ primitive $Q$-matrices respectively. Then some $H \in Q(f \cdot g) + O(\varepsilon)$ can be written as the product of $4n + 4m + 37$ primitive $Q$-matrices. Moreover, if the error degrees in $F, G$ are $e_f, e_g$, respectively, then the error degree of $H$ is at most $4 \cdot e_f + 4 \cdot e_g + 12$.  


Proof. We make use of the identity $(-(f/2)^2) + (-g^2) + (f/2 + g)^2 = f \cdot g$. By the addition lemma (Lem. 3.2), $(f/2 + g) + \mathcal{O}(\varepsilon^3)$ can be written as the product of $n + m + 1$ primitive Q-matrices with error degree at most $e_f + e_g$. By the squaring lemma (Lem. 3.3), $Q(-(f/2)^2) + \mathcal{O}(\varepsilon)$, $Q(-g^2) + \mathcal{O}(\varepsilon)$, and $Q((f/2 + g)^2) + \mathcal{O}(\varepsilon)$ can be written as the product of $2n + 11$, $2m + 11$, and $2(n + m + 1) + 11$ primitive Q-matrices, respectively. The corresponding error degrees are at most $2 \cdot e_f + 4$, $2 \cdot e_g + 4$, and $2(e_f + e_g) + 4$. Finally, by the addition lemma again, $Q(f \cdot g) + \mathcal{O}(\varepsilon) = Q(-(f/2)^2) + (-g^2) + (f/2 + g)^2 + \mathcal{O}(\varepsilon)$ can be written as the product of $(2n + 11) + 1 + (2m + 11) + 1 + (2(n + m + 1) + 11) = 4n + 4m + 37$ primitive Q-matrices. The corresponding error degree is at most $(2 \cdot e_f + 4) + (2 \cdot e_g + 4) + (2(e_f + e_g) + 4) = 4 \cdot e_f + 4 \cdot e_g + 12$. See Fig. F.3 in Appendix F for a pictorial description. \hfill \square

Proposition 3.5. Let $f$ be a polynomial computed by a formula of depth $d$. For every constant $\alpha \in \mathbb{F}$, some matrix in $F \in Q(\alpha f) + \mathcal{O}(\varepsilon)$ can be written as a product of at most $45 \cdot 9^d$ primitive Q-matrices. Moreover, $F$ has error degree at most $12 \cdot 25^d$.

Proof. The proof is by induction on $d$. For $d = 0$, that is, $f$ is a constant $\beta \in \mathbb{F}$ or a variable $x$, note that $Q(f)$ can be written directly as a primitive Q-matrix (with error degree 0). Since also $Q(\alpha/2)$ can be written directly (also with error degree 0), we can use the multiplication lemma (Lem. 3.4), to write $Q(\alpha f) + \mathcal{O}(\varepsilon)$ as a product of $4 + 4 + 37 = 45$ primitive Q-matrices (with error degree at most 12).

For $d \geq 1$, fix a constant $\alpha$. We know that either $f = g + h$ or $f = g \cdot h$ with formulas $g, h$ of depth $< d$. By the induction hypothesis, for any constant $\beta, \gamma$, we can write $Q(\beta g) + \mathcal{O}(\varepsilon)$ and $Q(\gamma h) + \mathcal{O}(\varepsilon)$ as a product of $n_g, n_h \leq 45 \cdot 9^{d-1}$ primitive Q-matrices, with error degrees $e_g, e_h \leq 12 \cdot 25^{d-1}$.

Case $f = g + h$. We set $\beta = \gamma = \alpha$ and use the addition lemma (Lem. 3.2) to obtain $Q(\alpha f) + \mathcal{O}(\varepsilon) = Q(\alpha g + \alpha h) + \mathcal{O}(\varepsilon)$ as a product of $n_g + n_h + 1 \leq 2 \cdot 45 \cdot 9^{d-1} + 1 \leq 45 \cdot 9^d$ primitive Q-matrices, with error degree at most $e_g + e_h \leq 2 \cdot 12 \cdot 25^{d-1} \leq 12 \cdot 25^d$.

Case $f = g \cdot h$. By replacing $\varepsilon$ by $\varepsilon^3$ in all primitive Q-matrices, we obtain matrices in $Q(\beta g) + \mathcal{O}(\varepsilon^3)$ and $Q(\gamma h) + \mathcal{O}(\varepsilon^3)$ as a product of $n_g$ and $n_h$ primitive Q-matrices with error degree at most $3 \cdot e_g$ and $3 \cdot e_h$, respectively. Now we set $\beta = \alpha/2$ and $\gamma = 1$ and use the multiplication lemma (Lem. 3.4) to obtain $Q(\alpha f) + \mathcal{O}(\varepsilon) = Q((\alpha \cdot g \cdot h) + \mathcal{O}(\varepsilon)$ as a product of $4n_g + 4n_h + 37 \leq 8 \cdot 45 \cdot 9^{d-1} + 37 \leq 45 \cdot 9^d$ primitive Q-matrices. The error degree is at most $4(3 \cdot e_g) + 4(3 \cdot e_h) + 12 = 12(e_g + e_h + 1) \leq 24 \cdot 12 \cdot 25^{d-1} + 12 \leq 12 \cdot 25^d$. \hfill \square

Proposition 3.6. If $(f_n) \in \mathbf{VP}_e$, then for each $n$ a matrix in $F \in Q(f_n) + \mathcal{O}(\varepsilon)$ can be written as a product of $\text{poly}(n)$ many primitive Q-matrices. Moreover, $F$ has error degree at most $\text{poly}(n)$.

Proof. The construction uses the classical depth-reduction theorem for formulas by Brent [Bre74], for which a modern proof can be found in the survey of Saptharishi [Sap16, Lemma 5.5]: If a family $(f_n)$ has polynomially bounded formula size, then there are formulas computing $f_n$ that have size $\text{poly}(n)$ and depth $O(\log n)$. Applying Prop. 3.5 now yields the result. \hfill \square

Proof of Thm. 3.1. This follows directly from Prop. 3.6. Namely, let $(f_n) \in \mathbf{VP}_e$. By Prop. 3.6 there is an $F \in Q(f_n) + \mathcal{O}(\varepsilon)$ which is a product of polynomially many primitive Q-matrices such that $F$ has polynomially bounded error degree. The width-2 ABP computing $f_n + \mathcal{O}(\varepsilon)$ is given by $(10)F(\frac{1}{0})$. \hfill \square
Corollary 3.7. \( \mathbf{VP}_2 = \mathbf{VP}_e \) and \( \mathbf{VP}_2^{\text{poly}} = \mathbf{VP}_e^{\text{poly}} \) when \( \text{char} (F) \neq 2 \).

Proof. The inclusion \( \mathbf{VP}_2 \subseteq \mathbf{VP}_e \) is standard (see Prop. C.1). Taking closures on both sides, we obtain \( \mathbf{VP}_2 \subseteq \mathbf{VP}_e \) and \( \mathbf{VP}_2^{\text{poly}} \subseteq \mathbf{VP}_e^{\text{poly}} \).

On the other hand, when \( \text{char} (F) \neq 2 \), we have the inclusion \( \mathbf{VP}_e \subseteq \mathbf{VP}_2^{\text{poly}} \) (Thm. 3.1). By taking closures this implies \( \mathbf{VP}_e \subseteq \mathbf{VP}_2 \) and \( \mathbf{VP}_e^{\text{poly}} \subseteq \mathbf{VP}_2^{\text{poly}} \).

Corollary 3.8. \( \mathbf{VP}_2^{\text{poly}} = \mathbf{VP}_e \) when \( \text{char} (F) \neq 2 \) and \( F \) is infinite.

Proof sketch. By Cor. 3.7 we have \( \mathbf{VP}_2^{\text{poly}} = \mathbf{VP}_e^{\text{poly}} \). It remains to show \( \mathbf{VP}_e^{\text{poly}} = \mathbf{VP}_e \). We give a proof of this via a standard interpolation argument in Appendix D.

As a consequence of Prop. 3.5, we obtain a new description of \( \mathbf{VP}_e \) as follows. We define the generalized Fibonacci polynomial \( F_n(x_1, \ldots, x_n) \) by \( F_0 := 1, F_1 := x_1, \) and \( F_n := x_n F_{n-1} + F_{n-2} \) for all \( n \geq 2 \). The name comes from the fact that \( F_n(1,1,\ldots,1) \) is the \( n \)-th Fibonacci number and \( F_n(x,x,\ldots,x) \) is the \( n \)-th Fibonacci polynomial. Another description of the polynomial \( F_n \) is that it is the upper left entry of a product of \( Q \)-matrices \( Q(x_i) \), that is, \( F_n(x_1,\ldots,x_n) = Q(x_n)Q(x_{n-1})\cdots Q(x_1))_{1,1} \).

Definition 3.9. A polynomial \( f \) is a projection of \( F_m \) if there exist affine linear forms \( \ell_1, \ldots, \ell_m \) such that \( f = F_m(\ell_1, \ldots, \ell_m) \). The smallest \( m \) such that \( f \) is a projection of \( F_m \) we call the Fibonacci complexity of \( f \). A polynomial is a degeneration of \( F_m \) if there exist parametrized affine linear forms \( \ell_1(\varepsilon), \ldots, \ell_m(\varepsilon) \) such that \( f = F_m(\ell_1(\varepsilon), \ldots, \ell_m(\varepsilon)) \). The smallest \( m \) such that \( f + \mathcal{O}(\varepsilon) \) is a degeneration of \( F_m \), we call the border Fibonacci complexity of \( f \), and is denoted by \( L_{\text{Fib}}(f) \). A family \( (h_n) \) of polynomials is called \( \mathbf{VP}_e \)-complete under p-degnerations if \( (h_n) \in \mathbf{VP}_e \) and for every \( (f_n) \in \mathbf{VP}_e \) there exists a polynomially bounded function \( t \) such that some polynomial in \( f_n + \mathcal{O}(\varepsilon) \) is a degeneration of \( F_{t(n)} \).

The Fibonacci complexity is not always finite ([AW16]), but Prop. 3.6 shows that the border Fibonacci complexity \( L_{\text{Fib}}(f) \) is always finite and that \( \mathbf{VP}_e \) can be characterized as the class of families with polynomially bounded border Fibonacci complexity:

Theorem 3.10. \( \mathbf{VP}_e = \{ (f_n) \mid L_{\text{Fib}}(f_n) \in \text{poly}(n) \} \).

Proof. Clearly the right-hand side is contained in the left-hand side. \( \mathbf{VP}_e \) is contained in the right-hand side by Prop. 3.6. A moment’s thought reveals that the right-hand side is closed under the approximation closure in the sense of Def. 2.1. Thus taking the closure on both sides yields the result.

Theorem 3.10 says that \( (F_n) \) is \( \mathbf{VP}_e \)-complete under p-degnerations. From the proof of Prop. 3.5 it follows that also \( (F_{2n+1}) \) is \( \mathbf{VP}_e \)-complete under p-degnerations, that is, we only need the \( F_m \) with odd index \( m \) (this follows from \( \det(Q(f)) = -1 \)).

Remark 3.11 (Symmetry). Define the polynomial \( C_n(x_1, \ldots, x_n) := \text{trace}(Q(x_n)Q(x_{n-1})\cdots Q(x_1)) \). Since the trace of a matrix product is invariant under cyclic shifts of the matrices, the polynomial \( C_n(x_1, \ldots, x_n) \) is invariant under cyclic shifts of the variables \( x_1, \ldots, x_n \). Thus \( C_n \) can be viewed as a cyclically symmetric version of \( F_n \). (Note that \( C_n \) and \( F_n \) are also both invariant under reversing the order of the variables \( x_1, \ldots, x_n \), that is, mapping \( (x_1, \ldots, x_n) \) to \( (x_n, \ldots, x_1) \).)

Define the border cyclic Fibonacci complexity analogously to the border Fibonacci complexity by replacing \( F_n \) by \( C_n \) in Def. 3.9. Analogously to Thm. 3.10 we now see that the families \( (C_n) \) and \( (C_{2n+1}) \) are both \( \mathbf{VP}_e \)-complete under p-degnerations.
4. VNP via products of affine linear forms

Valiant proved the following characterization of VNP [Val80] (see also [BCS97, Thm. 21.26], [Bür00, Thm. 2.13] and [MP08, Thm. 2]).

**Theorem 4.1** (Valiant [Val80]). VNP = VP.

We strengthen Valiant’s characterization of VNP from VNP to VNP1.

**Theorem 4.2.** VNP1 = VNP when char(\(F\)) \(\neq 2\).

We give two proofs. The idea of the first proof is to show that the VNP-complete permanent family \(\text{per}_n := \sum_{i\in[n]} \prod_{j\in[n]} x_{i,j} \) is in VNP1. The idea of the second proof is to simulate in VNP1 the primitives that are used in the proof of VP1 = VP3 by [BOC92]. We present the second proof in Appendix B. The advantage of the second proof is that we can restrict the ABP edge labels to affine linear forms that have at most 2 variables, see Thm. B.3. Both proofs use the following lemma to write expressions of the form \(1 + xy\) as a hypercube sum of a product of affine linear forms.

**Lemma 4.3.** \(\frac{1}{2} \sum_{b \in \{0,1\}} (x + 1 - 2b)(y + 1 - 2b) = 1 + xy\) when char(\(F\)) \(\neq 2\).

**Proof.** Expanding the left side gives the right side. \(\square\)

**Proof of Thm. 4.2.** The permanent family \((\text{per}_n)\) is well-known to be VNP-complete under p-projections, see for example [Bür00, Thm. 2.10]. Therefore, to show that VNP \(\subseteq\) VNP1, it suffices to show that \((\text{per}_n) \in\) VNP1. We begin by writing \(\text{per}_n\) as an inclusion-exclusion-type expression due to Ryser [Rys63, Thm. 4.1],

\[
\text{per}_n = (-1)^n \sum_{S \subseteq [n]} (-1)^{|S|} \prod_{j \in [n]} \sum_{i \in S} x_{i,j}.
\]

Encoding every subset \(S \subseteq [n]\) by a bit string \(b = (b[1], \ldots, b[n]) \in \{0,1\}^n\), we can rewrite the above as

\[
\text{per}_n = (-1)^n \sum_{b \in \{0,1\}^n} \left( \prod_{k \in [n]} (1 - 2b[k]) \right) \prod_{j \in [n]} \sum_{i \in [n]} b[i] x_{i,j}
\]

\[
= (-1)^n \sum_{b \in \{0,1\}^n} \left( \prod_{k \in [n]} (1 - 2b[k]) \right) \sum_{i_1, \ldots, i_n \in [n]} \prod_{j \in [n]} b[i_j] x_{i_j,j}
\]

For notational convenience we use square brackets not only to refer to sets ([n] := \{1, \ldots, n\}), but also to entries in a list \((b[k] := b_k)\). We now introduce new Boolean variables \(a[i,j], 1 \leq i \leq n - 1, 1 \leq j \leq n\), and fix the values \(a[0,j] = 1, a[n,j] = 0\). (This gives an \((n + 1) \times n\) matrix of variables and constants in which the first row consists of all 1s and the last row contains only 0s.) We claim that the above expression equals

\[
\text{per}_n = (-1)^n \sum_{b \in \{0,1\}^n} \left( \prod_{k \in [n]} (1 - 2b[k]) \right) \sum_{a} \prod_{i,j \in [n]} \left( 1 + (x_{i,j} - 1)(a[i-1,j] - a[i,j]) \right)
\]

\[
\cdot (1 + (b[i] - 1)(a[i-1,j] - a[i,j]) \cdot (1 + (a[i-1,j] - 1)a[i,j])
\]

\(9\)
where the second sum is over all Boolean assignments of \( a[i, j] \). The idea is to encode the indices \( i_1, \ldots, i_n \) in the boolean variables \( a[i, j] \) in unary. For example for \( n = 4 \), if \( i_1 = 4, i_2 = 3, i_3 = 1, i_4 = 4 \), then the corresponding matrix \( a \) is

\[
\begin{pmatrix}
1 & 1 & 1 & 1 \\
1 & 1 & 0 & 1 \\
1 & 0 & 0 & 1 \\
0 & 0 & 0 & 0
\end{pmatrix}.
\]

We prove the claim (3) in three steps. Fix \( j \).

- If \( a[i-1, j] = 0 \) and \( a[i, j] = 1 \), then \( 1 + (a[i-1, j] - 1)a[i, j] = 0 \). Thus if in the sequence \( a[0, j], \ldots, a[n, j] \) a 0 is followed by a 1, then \( \prod_{i \in [n]} (1 + (a[i-1, j] - 1)a[i, j]) = 0 \). Conversely, if \( (a[0, j], \ldots, a[n, j]) = (1, \ldots, 1, 0, \ldots, 0) \), then \( \prod_{i \in [n]} (1 + (a[i-1, j] - 1)a[i, j]) = 1 \). The nontrivial assignments of \( (a[0, j], \ldots, a[n, j]) \) are thus exactly of the form \( (1, \ldots, 1, 0, \ldots, 0) \) where the first 0 occurs at some index \( 1 \leq z \leq n \) (since we have set \( a[0, j] = 1 \) and \( a[n, j] = 0 \)). Fix such an assignment with first 0 occurring at index \( z \).

- If \( i = z \), then \( 1 + (x_{i,j} - 1)(a[i-1, j] - a[i, j]) \) equals \( x_{i,j} \). If \( i \neq z \), it equals 1.

This proves (3).

Next we apply Lem. 4.3, introducing fresh hypercube variables \( c_1[i, j], c_2[i, j], \) and \( c_3[i, j], \) with \( 1 \leq i, j \leq n \), to obtain

\[
\text{per}_n = (-1)^n \left( \frac{1}{2} \right)^{3n^2} \sum_b \left( \prod_{k \in [n]} (1 - 2b[k]) \right) \cdot \sum_a \left( \prod_{i,j \in [n]} \sum_{c_1[i,j]} \left( x_{i,j} - 2c_1 \right) \cdot (a[i-1, j] - a[i, j] + 1 - 2c_1) \cdot \sum_{c_2[i,j]} \left( b[i] - 2c_2 \right) \cdot (a[i-1, j] - a[i, j] + 1 - 2c_2) \right) \cdot \sum_{c_3[i,j]} \left( (a[i-1, j] - 2c_3) \cdot (a[i, j] + 1 - 2c_3) \right),
\]

where the sum goes over all Boolean assignments of \( b[i], a[i, j], c_1[i, j], c_2[i, j], c_3[i, j], 1 \leq i, j \leq n \), except for \( a[n, j] := 0 \), and \( a[0, j] := 1 \). After a rearrangement we obtain the expression

\[
\text{per}_n = \sum_{a,b,c_1,c_2,c_3} \left( (-1)^n \left( \frac{1}{2} \right)^{3n^2} \left( \prod_{k \in [n]} (1 - 2b[k]) \right) \cdot \prod_{i,j \in [n]} \left( x_{i,j} - 2c_1[i,j] \right) \cdot (a[i-1, j] - a[i, j] + 1 - 2c_1[i,j]) \cdot \left( b[i] - 2c_2[i,j] \right) \cdot (a[i-1, j] - a[i, j] + 1 - 2c_2[i,j]) \right) \cdot \left( a[i-1, j] - 2c_3[i,j] \right) \cdot \left( a[i, j] + 1 - 2c_3[i,j] \right),
\]

where the sum goes over all Boolean assignments of \( a[i, j], b[i], c_1[i, j], c_2[i, j], c_3[i, j], 1 \leq i, j \leq n \), again except for \( a[n, j] := 0 \), and \( a[0, j] := 1 \). This shows that \( \text{per}_n \in \text{VNP}_1 \). \( \square \)

Thm. 4.2 does not hold over \( \mathbb{F}_2 \), as the following proposition shows. We give a proof in Appendix E. We leave the situation over other fields of characteristic 2 as an open problem.

**Proposition 4.4.** \( \text{VNP}_1 \subsetneq \text{VNP} \) when \( F = \mathbb{F}_2 \).
References


Appendix

A. ABPs with restricted edge labels

So far the edge labels of our ABPs were allowed to be arbitrary affine linear forms. This section is about ABPs in which the edge labels are restricted to be simple affine linear forms ("weak ABPs"), or variables and constants ("weakest ABPs"). These edge label types were also studied in [AW16].

**Definition A.1.** A wst-ABP (weakest ABP) is an ABP with edges labeled by variables or constants. A w-ABP (weak ABP) is an ABP with edges labeled by simple affine linear forms $\alpha x_i + \beta$, $\alpha, \beta \in \mathbb{F}$. A g-ABP (general ABP) is an ABP with edges labeled by general affine linear forms $\sum_i \alpha_i x_i + \beta$, $\alpha_i, \beta \in \mathbb{F}$. For $*$ equal to wst, w or g, the class $\text{VP}_k^*$ consists of all families of polynomials over polynomially many variables that are computed by polynomial-size width-$k$ $*$-ABPs. In the rest of this paper the star will act as a variable from $\{\text{wst}, \text{w}, \text{g}\}$. We write $\text{VP}_k$ if we mean $\text{VP}_k^g$.

From the above definition it follows that $\text{VP}_k^{\text{wst}} \subseteq \text{VP}_k^{\text{w}} \subseteq \text{VP}_k^{\text{g}}$.

**Remark A.2.** One checks that the construction in the proof of Thm. 3.1 actually proves the inclusion $\text{VP}_c \subseteq \text{VP}_2^{\text{wst poly}}$ when $\text{char}(\mathbb{F}) \neq 2$. The inclusion $\text{VP}_c \subseteq \text{VP}_2^{\text{wst poly}}$ implies the equalities $\text{VP}_2^{\text{wst}} = \text{VP}_2^c$ and $\text{VP}_2^{\text{wst poly}} = \text{VP}_2^c$.

In the following sections we will prove all inclusions and separations that are listed in Figure F.4.
A.1. Comparing different types of edge labels in width-2 ABPs

The aim of this subsection is to prove the following separation.

**Theorem A.3.** $\text{VP}^w_2 \subsetneq \text{VP}^g_2$.

In fact, we will show the following stronger statement.

**Theorem A.4.** The polynomial

$$p(x) = (x_{11} + x_{12} + \cdots + x_{17})(x_{21} + x_{22} + \cdots + x_{27})$$
$$+ (x_{31} + x_{32} + \cdots + x_{37})(x_{41} + x_{42} + \cdots + x_{47})$$

is computable by a width-2 $g$-ABP, but not computable by any width-2 $w$-ABP.

We leave it as an open problem whether the inclusion $\text{VP}^\text{wst}_2 \subseteq \text{VP}^w_2$ is strict.

To prove Thm. A.4 we will review and reuse the arguments used by Allender and Wang [AW16] to show that the polynomial $x_1x_2 + \cdots + x_{15}x_{16}$ cannot be computed by any width-2 $g$-ABP.

For the proof of Thm. A.4 we may without loss of generality assume that the base field $F$ is algebraically closed, because for any field $F$, if $p$ is not computable over the algebraic closure of $F$, then it is not computable over $F$ itself. Let $H$ be the set of affine linear forms that are single variables $x_i$ or constants $F$. Let $S$ be the set of single affine linear forms. Let $L$ be the set of general affine linear forms. Let $H^{2 \times 2}, S^{2 \times 2}, L^{2 \times 2}$ be the sets of $2 \times 2$ matrices with entries in $H, S, L$ respectively. In this subsection, all ABPs have width 2, and by a wst-, w- or g-ABP $\Gamma$ we will mean a sequence $\Gamma_k, \ldots, \Gamma_1$ with $\Gamma_k \in F^{1 \times 2}, \Gamma_{k-1}, \ldots, \Gamma_2 \in X^{2 \times 2}$, and $\Gamma_1 \in F^{2 \times 1}$ with $X$ equal to $H, S$ or $L$ respectively. We call $\Gamma_{k-1}, \ldots, \Gamma_2$ the inner matrices of $\Gamma$.

**Definition A.5.** A matrix $A \in L^{2 \times 2}$ is called inherently nondegenerate (indg) if $\det(A) \in F \setminus \{0\}$.

Allender and Wang prove the following necessary condition for a polynomial to be computable by a wst-, w- or g-ABP whose inner matrices are indg. Let $H(p)$ denote the highest-degree homogeneous part of a polynomial $p$.

**Theorem A.6 ([AW16, Thm. 3.9 and Lem. 4.7]).** Let $p$ be a polynomial and $\Gamma$ a wst-, w- or g-ABP computing $p$, whose inner matrices are indg. Then $H(p)$ is a product of affine linear forms.

Our next goal is to give a necessary condition for a polynomial $p$ to be computable by a w-ABP. We begin with a simple lemma, which can essentially be found in [AW16].

**Lemma A.7 ([AW16]).** Let $p$ be a polynomial. If $p$ is computed by a w-ABP that has an inner matrix containing 4 variables, then there is an assignment $\pi$ of 4 variables with $\pi(p) = 0$.

**Proof.** Let $M$ be such a matrix. Since the ABP is of type w, $M$ is of the form

$$M = \begin{pmatrix}
\alpha_{11}x_{11} + \beta_{11} & \alpha_{12}x_{12} + \beta_{12} \\
\alpha_{21}x_{21} + \beta_{21} & \alpha_{22}x_{22} + \beta_{22}
\end{pmatrix}$$

for some constants $\alpha_{ij} \in F \setminus \{0\}$, $\beta_{ij} \in F$. Applying the four assignments $x_{ij} \mapsto -\beta_{ij}/\alpha_{ij}$ makes $M$ zero and thus $p$ zero. \qed


We need two more ideas before we will state and prove the necessary condition we are after. Let $A \in \mathbb{L}^{2 \times 2}$ be nonzero and not-indg (that is, det$(A)$ is either 0 or a nonconstant polynomial). Then there is an assignment $\pi$ of the variables such that $\pi(A)$ has only constant entries and has rank 1. Let $p$ be a polynomial computed by an ABP $\Gamma$, that is, $p = \Gamma_1 \cdots \Gamma_k$. Suppose that $\Gamma$ contains a matrix $\Gamma_i$ with only constant entries and with rank 1. Then there is a $2 \times 1$ matrix $\Gamma_{i,2}$ and a $1 \times 2$ matrix $\Gamma_{i,1}$ such that $\Gamma_i = \Gamma_{i,2} \Gamma_{i,1}$. Then $p$ is a product $p = p_2 p_1$ of polynomials $p_1$, $p_2$, each computable by an ABP, namely

$$
p_2 = \Gamma_k \cdots \Gamma_{i+1} \Gamma_{i,2}
$$

$$
p_1 = \Gamma_{i,1} \Gamma_{i-1} \cdots \Gamma_1.
$$

We say that $\Gamma_i$ factors $p$ into $p_2 p_1$. Recall that $H(p)$ denotes the highest-degree homogeneous part of a polynomial $p$. The following is implicit in [AW16].

**Theorem A.8 ([AW16]).** Let $p$ be a polynomial computed by a w-ABP $\Gamma$. Then there is an assignment $\pi$ of at most 6 variables such that one of the following is true:

1. $\pi(p)$ is affine linear (including constant), or

2. $H(\pi(p))$ is a product of two polynomials of positive degree.

**Proof.** Let $(\Gamma_k, \ldots, \Gamma_1)$ be the matrices of $\Gamma$, so that $p = \Gamma_k \cdots \Gamma_1$. If there is a $\Gamma_i$ containing 4 variables, then there is an assignment $\pi$ of these 4 variables with $\pi(p) = 0$ (Lem. A.7), so we are in case 1. Otherwise, all $\Gamma_i$ have at most 3 variables. If the inner $\Gamma_i$ are all indg, then $H(p)$ is a product of linear forms (Thm. A.6), so we are in case 1 or 2. Otherwise, there is at least one not-indg inner matrix. Consider the nonempty subsequence $M = (M_\ell, \ldots, M_1)$ of not-indg inner matrices. For each $M_i$ there is an assignment $\pi$ of at most 3 variables such that $\pi(M_i)$ has only constant entries and rank 1. We consider four possible situations.

1. There is an $M \in M$ and an assignment $\pi$ of at most 3 variables such that $\pi(M)$ factors $\pi(p)$ into a product of two constants or a product of two polynomials with positive degree. Then we are in case 1 or 2.

2. There is an assignment $\pi$ of at most 3 variables such that $\pi(M_1)$ factors $\pi(p)$ into $p_2 p_1$ with $p_2$ a constant and $p_1$ not constant. Then $p_1$ is computed by an ABP consisting of indg inner matrices (since $M_1$ is the right-most not-indg inner matrix) and hence $H(p_1)$ is a product of linear forms (Thm. A.6), so we are in case 1 or 2.

3. There is an assignment $\pi$ of at most 3 variables such that $\pi(M_\ell)$ factors $\pi(p)$ into $p_2 p_1$ with $p_2$ not a constant and $p_1$ a constant. Then $p_2$ is computed by an ABP consisting of indg inner matrices (since $M_2$ is the left-most not-indg inner matrix) and one proceeds as in the previous situation.

4. Remaining situation. In the remaining situation we do the following. Let $M_i$ be the left-most matrix in $M$ such that there is an assignment $\pi$ of at most 3 variables such that $\pi(M_i)$ factors $\pi(p)$ into $p_2 p_1$ with $p_2$ not a constant and $p_1$ constant. Then there is an assignment $\sigma$ of at most 3 variables such that $\sigma(\pi(M_{i+1}))$ factors $p_2 = p_3 p_4$ with $p_3$ constant and $p_4$ not constant. Then $p_4$ is computed by an ABP consisting of indg matrices, and so $H(p_4)$ is a product of homogeneous linear forms. Therefore we are in case 1 or 2.

$\square$
Theorem A.4 (repeated). The polynomial
\[
p(x) = (x_1 + x_2 + \cdots + x_{17})(x_{21} + x_{22} + \cdots + x_{27})
+ (x_{31} + x_{32} + \cdots + x_{37})(x_{41} + x_{42} + \cdots + x_{47})
\]
is computable by a width-2 $g$-ABP, but not computable by any width-2 $w$-ABP.

Proof. Clearly $p(x)$ is computable by a width-2 $g$-ABP. Suppose $p(x)$ is computable by a width-2 $w$-ABP. Then by Thm. A.8 there is an assignment $\pi$ of at most 6 variables such that either $\pi(p)$ is affine linear or $H(\pi(p))$ is a product of two polynomials of positive degree. The first option is impossible, because distinct variables do not cancel. So $H(\pi(p))$ is a product of two polynomials of positive degree. With another assignment $\sigma$ we can achieve that $H(\sigma(\pi(p)))$ is of the form $x_i x_j + x_k x_\ell$ for some distinct variables $x_i, x_j, x_k, x_\ell$. This is not a product of two polynomials of positive degree, so $H(\pi(p))$ is not either. \qed

A.2. Comparing different types of edge labels in width-1 ABPs

Clearly, $VP_{1}^{wst} \subseteq VP_{1}^{w} \subseteq VP_{1}^{g}$ and $VP_{1}^{*} \subseteq VP_{2}^{w}$, but this does not give a complete description of all inclusions among these classes. The following two propositions realize a complete description among $VP_{1}^{*}$ and $VP_{2}^{wst}$.

Proposition A.10. $VP_{1}^{g} \subseteq VP_{2}^{wst}$.

Proof. Let $(p_n) \in VP_{1}^{g}$. Then each $p_n$ is a product of $\operatorname{poly}(n)$ affine linear forms in $\operatorname{poly}(n)$ variables. Let $\ell(x) = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + \cdots + \alpha_m x_m$ be such an affine linear form with $\alpha_0 \in \mathbb{F}$ and $\alpha_1, \ldots, \alpha_m \in \mathbb{F} \setminus \{0\}$. We can compute $\ell(x)$ with the width-2 $wst$-ABP

A product of affine linear forms can be computed by the width-2 $wst$-ABP that is the concatenation of the width-2 $wst$-ABPs computing the affine linear forms. For $p_n$ the resulting ABP has $\operatorname{poly}(n)$ size. Thus, $(p_n) \in VP_{2}^{wst}$. \qed
Proposition A.11. $\text{VP}^\text{wst}_1 \subseteq \text{VP}_1^w \subseteq \text{VP}^\text{g}_1 \subseteq \text{VP}_2^{\text{wst}}$.

Proof. If $(p_n) \in \text{VP}^\text{wst}_1$, then $p_n$ is a monomial. However, $(\alpha_0 + \alpha_1 x_1) \in \text{VP}_1^w$ and $\alpha_0 + \alpha_1 x_1$ is not a monomial, so $\text{VP}^\text{wst}_1 \not\subseteq \text{VP}_1^w$. If $(p_n) \in \text{VP}_1^w$ and $p_n$ is homogeneous, then $p_n$ is a monomial. However, $(x_1 + x_2)^2) \in \text{VP}^\text{g}_1$ and $(x_1 + x_2)^2$ is not a monomial, so $\text{VP}_1^w \subseteq \text{VP}^\text{g}_1$. The last inclusion is Prop. A.10. To see the strictness, if $(p_n) \in \text{VP}_1^w$, then the highest-degree homogeneous part $H(p_n)$ of $p_n$ is a product of homogeneous linear forms. However, $(x_1 x_2 + x_3 x_4) \in \text{VP}_2^{\text{wst}}$ and $x_1 x_2 + x_3 x_4$ is not a product of homogeneous linear forms, so $\text{VP}^\text{g}_1 \not\subseteq \text{VP}_2^{\text{wst}}$. \qed

A.3. Approximation in width-1 ABPs

The following proposition says that each of $\text{VP}^\text{wst}_1$, $\text{VP}_1^w$ and $\text{VP}^\text{g}_1$ is closed under approximation.

Proposition A.12. $\text{VP}_1^* = \overline{\text{VP}_1^*}$.

Proof. Trivially, $\text{VP}_1^* \subseteq \overline{\text{VP}_1^*}$. To prove the opposite inclusion, let $(f_n) \in \overline{\text{VP}_1^*}$. There are polynomials $g_n(\varepsilon, x) \in \mathbb{F}[\varepsilon, x]$ such that $f_n + \varepsilon g_n(\varepsilon, x)$ can be written as a product of $\text{poly}(n)$ affine linear forms in $\mathbb{F}[\varepsilon][x]$ in $\text{poly}(n)$ variables (these affine linear forms have either wst-, w- or g-type). That is, (forgetting the subscript $n$ for the moment) $f(x) + \varepsilon g(x)$ can be written as

$$f(x) + \varepsilon g(x, \varepsilon) = \prod_{i=1}^{m} \ell_i(\varepsilon, x)$$

with

$$\ell_i(\varepsilon, x) = \sum_{j=d_i}^{e_i} \varepsilon^j k_{i,j}(x)$$

for some affine linear forms $k_{i,j} \in \mathbb{F}[x]$, such that $k_{i,d_i}(x) \neq 0$, and $d_i \leq e_i \in \mathbb{Z}$. By shifting $\varepsilon$-factors from $\ell_1, \ldots, \ell_{m-1}$ to $\ell_m$ we can assume that $d_i = 0$ for $i < m$. We claim that $d_m \geq 0$. If $d_m < 0$, then expanding $\prod_{i} \ell_i(x)$ as a Laurent series in $\varepsilon$ gives a term with a negative power of $\varepsilon$. This contradicts $f(x) + \varepsilon g(x)$ having only nonnegative powers of $\varepsilon$. Therefore, the $\ell_i(x)$ do not contain any negative powers of $\varepsilon$ and we can safely substitute $\varepsilon \mapsto 0$ in each linear form $\ell_i$ to obtain $f$ as a product of affine linear forms in $\mathbb{F}[x]$ (either of wst-, w- or g-type). Remembering our subscript $n$ again, we have thus proven $(f_n) \in \text{VP}_1^*$.

A.4. Nondeterminism in width-1 ABPs

In the following proposition we compare $\text{VP}_1^*$ to $\text{VNP}_1^*$ for all three versions $* \in \{\text{wst}, w, g\}$.

Proposition A.13.

- $\text{VP}_1^* = \text{VNP}_1^*$ for $* \text{ equal to wst or w}$.
- $\text{VP}_1^g \subseteq \text{VNP}_1^g$ when $\text{char}(\mathbb{F}) \neq 2$.

Proof. Trivially, $\text{VP}_1^* \subseteq \text{VNP}_1^*$. Let $(p_n) \in \text{VNP}_1^{\text{wst}}$. Then $p_n$ can be written as a hypercube-sum over a monomial,

$$p(x) = \sum_{b \in \{0, 1\}^{\text{poly}(n)}} m(b, x)$$

18
Theorem B.1

Proof. Prop. C.1 says that $\text{VP}^{\text{wst}}_k \subseteq \text{VP}_e$. We will prove that $\text{VP}_e \subseteq \text{VP}^{\text{wst}}_3$, from which it follows that $\text{VP}_e \subseteq \text{VP}^*_k$ and thus $\text{VP}^*_k = \text{VP}_e$. For a polynomial $h$, define the matrix

$$M(h) := \begin{pmatrix} 1 & 0 & 0 \\ h & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

with $m$ a monomial (subscripts $n$ are implied). For any $b$-variable that does not occur in $m$, we remove that $b$-variable form the summation and at the same time multiply the expression by 2, to again have an expression for $p(x)$. Assuming all $b$-variables occur in $m$, only for $b = (1,1,\ldots,1)$ can $m(b,x)$ be nonzero. So $p(x) = m((1,1,\ldots,1),x)$. Remembering the subscript $n$, we proved $(p_n) \in \text{VNP}^w_1$.

Let $(p_n) \in \text{VNP}^g_1$. Then, (forgetting the subscript $n$)

$$p(x) = \sum_{b \in \{0,1\}^{\text{poly}(n)}} \prod_i \ell_i(b) \prod_j k_j(x)$$

for some affine linear forms $\ell_i$ in the variables $b$ and some affine linear forms $k_j$ in the variables $x$. The product $\prod_j k_j(x)$ is independent of $b$, while $\sum_b \prod_i \ell_i(b)$ is a constant. We can thus write $p(x)$ as a constant times $\prod_j k_j(x)$. Therefore (remembering $n$), $p_n(x) \in \text{VP}^w_1$. This proves the first line of the proposition.

To prove the second line, recall that if $(p_n) \in \text{VP}^g_1$, then $p_n$ is a product of affine linear forms. However, let $p_n(x_1,x_2) = \sum_{b \in \{0,1\}}(x_1 + b)(x_2 + b) = 2x_1x_2 + x_1 + x_2 + 1$. Then $(p_n) \in \text{VNP}^g_1$, but $p_n(x_1,x_2)$ is a not a product of affine linear forms, as we will now verify. Suppose $2x_1x_2 + x_1 + x_2 + 1 = (\alpha_0 + \alpha_1 x_1 + \alpha_2 x_2)(\beta_0 + \beta_1 x_1 + \beta_2 x_2)$. Then $\alpha_1 \beta_1 = 0$ and $\alpha_2 \beta_2 = 0$. Since $\alpha_1 \beta_1 = 0$, we may assume without loss of generality that $\alpha_1 = 0$. Since not both $\alpha_1$ and $\alpha_2$ can be 0 (otherwise $(\alpha_0 + \alpha_1 x_1 + \alpha_2 x_2)(\beta_0 + \beta_1 x_1 + \beta_2 x_2)$ has degree 1) and since $\alpha_2 \beta_2 = 0$, we have $\alpha_2 = 0$. Hence, $2x_1x_2 + x_1 + x_2 + 1 = (\alpha_0 + \alpha_2 x_2)(\beta_0 + \beta_1 x_1)$. Then $\alpha_0 \beta_0 = 1$, $\alpha_0 \beta_1 = 1$, $\alpha_2 \beta_0 = 1$, and $\alpha_2 \beta_1 = 2$. The first two of these equations imply $\beta_0 = \beta_1$, which contradicts the last two of these equations. So $\text{VP}^g_1 \not\subseteq \text{VNP}^g_1$.

Remark A.14. It follows directly from Prop. A.13 and Prop. A.11 that we have strict inclusions $\text{VNP}^\text{wst}_1 \not\subseteq \text{VNP}^w_1 \not\subseteq \text{VNP}^g_1$, when $\text{char}(\mathbb{F}) \neq 2$.

B. Alternative proof of $\text{VNP}_1 = \text{VNP}$ via $\text{VP}_3$

Recall that in Sec. 4 we proved that

$$\text{VNP}^g_1 = \text{VNP}$$

(4)

using the completeness of the permanent (Thm. 4.2). We will present an alternative proof of (4) inspired by the proof of the following theorem by Ben-Or and Cleve. The alternative proof of (4) has the benefit that it can be extended to show a slightly stronger result, see Thm. B.3.

Theorem B.1 (Ben-Or and Cleve, [BOC92]). For $k \geq 3$, $\text{VP}_k = \text{VP}_e$.

Proof. Prop. C.1 says that $\text{VP}_k \subseteq \text{VP}_e$. We will prove that $\text{VP}_e \subseteq \text{VP}^\text{wst}_3$, from which it follows that $\text{VP}_e \subseteq \text{VP}_k^*$ and thus $\text{VP}_k^* = \text{VP}_e$. For a polynomial $h$, define the matrix

$$M(h) := \begin{pmatrix} 1 & 0 & 0 \\ h & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$
which, as part of an ABP, looks like

\[ \begin{array}{ccc} & h & \\
\downarrow & & \downarrow \\
& & \end{array} \]

We call the following matrices \textit{primitive}:

- \( M(h) \) with \( h \) any variable or any constant in \( \mathbb{F} \)
- \( \) every \( 3 \times 3 \) permutation matrix \( M_\pi \) with \( \pi \in S_3 \) any permutation
- \( \) every diagonal matrix \( M_{a,b,c} := \text{diag}(a,b,c) \) with \( a, b, c \) any constants in \( \mathbb{F} \)

The entries of the primitives are variables or constants in \( \mathbb{F} \), making them suitable to use in the construction of a width-3 wst-ABP (Def. A.1).

Let \( (f_n) \in \mathbf{VP}_e \). Then \( f_n \) can be computed by a formula of size \( s(n) \in \text{poly}(n) \). By Brent’s depth-reduction theorem for formulas ([Bre74]) \( f_n \) can then also be computed by a formula of size \( \text{poly}(n) \) and depth \( d(n) \in \mathcal{O}(\log n) \).

We will construct a sequence of primitives \( A_1, \ldots, A_{m(n)} \) such that

\[
A_1 \cdots A_{m(n)} = \begin{pmatrix}
1 & 0 & 0 \\
\frac{f_n}{1} & 1 & 0 \\
0 & 0 & 1
\end{pmatrix}
\]

with \( m(n) \in \mathcal{O}(4^{d(n)}) = \text{poly}(n) \). Then

\[
f_n(x) = (1 1 1)M_{-1,1,0}A_1 \cdots A_m \left( \frac{1}{1} \right),
\]

so \( f_n(x) \) can be computed by a width-3 wst-ABP of size \( \text{poly}(n) \), proving the theorem.

To explain the construction, let \( h \) be a polynomial and consider a formula computing \( h \) of depth \( d \). The goal is to construct (recursively on the formula structure) primitives \( A_1, \ldots, A_m \) such that

\[
A_1 \cdots A_m = \begin{pmatrix}
1 & 0 & 0 \\
\frac{h}{1} & 1 & 0 \\
0 & 0 & 1
\end{pmatrix}
\]

with \( m \in \mathcal{O}(4^d) \). (5)

Suppose \( h \) is a variable or a constant. Then \( M(h) \) is itself a primitive matrix.

Suppose \( h = f + g \) is a sum of two polynomials \( f, g \) and suppose \( M(f) \) and \( M(g) \) can be written as a product of primitives. Then \( M(f + g) \) equals a product of primitives, because \( M(f + g) = M(f)M(g) \). This can easily be verified directly, or by noting that in the corresponding partial ABPs the top-bottom paths \( (u_i - v_j \text{ paths}) \) have the same value:

\[
\begin{array}{ccc}
u_1 & u_2 & u_3 \\
\downarrow f & \downarrow g & \\
v_1 & v_2 & v_3
\end{array}
\sim
\begin{array}{ccc}
u_1 & u_2 & u_3 \\
\downarrow f + g & \\
v_1 & v_2 & v_3
\end{array}
\]
Suppose $h = fg$ is a product of two polynomials $f, g$ and suppose $M(f)$ and $M(g)$ can be written as a product of primitives. Then $M(fg)$ equals a product of primitives, because

$$M(f \cdot g) = M_{(23)}(M_{(23)}^{-1}M_{(123)}M(g)M_{(132)}M(f))^2M_{(23)}$$

(here $(23) \in S_3$ denotes the transposition $1 \mapsto 1, 2 \mapsto 3, 3 \mapsto 2$ and $(123) \in S_3$ denotes the cyclic shift $1 \mapsto 2, 2 \mapsto 3, 3 \mapsto 1$) as can be verified either directly or by checking that in the corresponding partial ABPs the top-bottom paths ($u_i$-$v_j$ paths) have the same value:

This completes the construction.

The length $m$ of the construction is $m(h) = 1$ for $h$ a variable or constant and recursively $m(f + g) = m(f) + m(g)$, $m(f \cdot g) = 2(m(f) + m(g)) + O(1)$, so $m \in O(4^d)$ where $d$ is the formula depth of $h$. The construction thus satisfies (5), proving the theorem.

We will now give an alternative proof of Thm. 4.2.

Theorem 4.2 (repeated). $\text{VNP}_1 = \text{VNP}$ when $\text{char}(F) \neq 2$.

Proof. Clearly, $\text{VNP}_1^g \subseteq \text{VNP}$ by Prop. C.1 and taking the nondeterminism closure $N$. We will prove that $\text{VNP} \subseteq \text{VNP}_1^g$.

Recall that in the proof of $\text{VP}_e \subseteq \text{VP}_3^{\text{wst}}$ (Thm. B.1), we defined for any polynomial $h$ the matrix

$$M(h) := \begin{pmatrix} 1 & 0 & 0 \\ h & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

and we called the following matrices primitive:
• $M(h)$ with $h$ any variable or any constant in $\mathbb{F}$
• every $3 \times 3$ permutation matrix $M_\pi$ with $\pi \in S_3$ any permutation
• every diagonal matrix $M_{a,b,c} := \text{diag}(a,b,c)$ with $a,b,c$ any constants.

In the proof of $\mathbf{VP}_e \subseteq \mathbf{VP}_{3\text{st}}$ we constructed, for any family $(f_n) \in \mathbf{VP}_e$ a sequence of primitives $A_{n,1}, \ldots, A_{n,t(n)}$ with $t(n) \in \text{poly}(n)$ such that

$$f_n(x) = (1 1 1) M_{-1,1,0} A_1 \cdots A_m \left( \begin{array}{c} 1 \\ 1 \\ 1 \end{array} \right).$$

We will construct a hypercube sum over a width-1 $g$-ABP that evaluates the right-hand side, to show that $\mathbf{VP}_e \subseteq \mathbf{VNP}_g^1$. This implies $\mathbf{VNP}_e \subseteq \mathbf{VNP}_g^1$. Then by Valiant’s Thm. 4.1, $\mathbf{VNP} \subseteq \mathbf{VNP}_g^1$.

Let $f(x)$ be a polynomial and let $A_1, \ldots, A_k$ be primitives such that $f(x) = (1 1 1) A_k \cdots A_1 \left( \begin{array}{c} 1 \\ 1 \\ 1 \end{array} \right)$. View this expression as a width-3 ABP $G$, with vertex layers labeled as shown:

Assume for simplicity that all edges between layers are present, possibly with label 0. The sum of the values of every $s$-$t$ path in $G$ equals $f(x)$,

$$f(x) = \sum_{j \in [3]^k} A_k[j_k,j_{k-1}] \cdots A_1[j_2,j_1]. \quad (6)$$

We now introduce some hypercube variables. To every vertex, except $s$ and $t$, we associate a bit; the bits in the $i$th layer we call $b_1[i]$, $b_2[i]$, $b_3[i]$. To an $s$-$t$ path in $G$ we associate an assignment of the $b_j[i]$ by setting the bits of vertices visited by the path to 1 and the others to 0. For example,
below we show an $s$-$t$ path with the corresponding assignment of the bits $b_1[i], b_2[i], \text{and } b_3[i]$: 

\[
\begin{array}{c}
\text{s} \\
\text{100} \\
\text{010} \\
\text{010} \\
\text{001} \\
\text{010} \\
\text{t}
\end{array}
\]

The assignments of $b_j[i]$ corresponding to $s$-$t$ paths are the ones such that for every $i \in [k]$ exactly one of $b_1[i], b_2[i], b_3[i]$ equals 1. Let

\[
V(b_1, b_2, b_3) := \prod_{i \in [k]} (b_1[i] + b_2[i] + b_3[i]) \prod_{s, t \in [3]: s \neq t} (1 - b_s[i]b_t[i]).
\]  

(7)

The assignments of $b_j[i]$ corresponding to $s$-$t$ paths are thus the ones such that $V(b_1, b_2, b_3) = 1$. Otherwise, $V(b_1, b_2, b_3) = 0$.

We will now write $f(x)$ as a hypercube sum by replacing each $A_i[j_i, j_i-1]$ in (6) by a product of affine linear forms $S_i(A_i)$ with variables $b$ and $x$ as follows

\[
\sum_b V(b_1, b_2, b_3)S_k(A_k) \cdots S_1(A_1).
\]

Define $\text{Eq}(\alpha, \beta) : \{0,1\}^2 \rightarrow \{0,1\}$ by $(1 - \alpha - \beta)(1 - \alpha - \beta)$. This function is 1 if $\alpha = \beta$ and 0 otherwise.

- For any variable or constant $x$ define
  \[
  S_i(M(x)) := (1 + (x - 1)(b_1[i] - b_1[i-1]) \\
  \cdot (1 - (1 - b_2[i])b_2[i-1]) \\
  \cdot \text{Eq}(b_3[i-1], b_3[i]).
  \]

- For any permutation $\pi \in S_3$ define
  \[
  S_i(M_\pi) := \text{Eq}(b_1[i-1], b_\pi(1)[i]) \\
  \cdot \text{Eq}(b_2[i-1], b_\pi(2)[i]) \\
  \cdot \text{Eq}(b_3[i-1], b_\pi(3)[i]).
  \]

23
• For any constants \(a, b, c \in \mathbb{F}\) define

\[
S_i(M_{a,b,c}) := (a \cdot b_1[i-1] + b \cdot b_2[i-1] + c \cdot b_3[i-1])
\cdot \text{Eq}(b_1[i-1], b_1[i])
\cdot \text{Eq}(b_2[i-1], b_2[i])
\cdot \text{Eq}(b_3[i-1], b_3[i]).
\]

One verifies that with these definitions indeed

\[
f(x) = \sum_{b} V(b_1, b_2, b_3)S_k(A_k) \cdots S_1(A_1).
\]

Some of the factors in the \(S_i(A_i)\) are not affine linear. As a final step we apply the equation

\[
1 + xy = \frac{1}{2} \sum_{c \in \{0,1\}} (x + 1 - 2c)(y + 1 - 2c)
\]

(Lem. 4.3) to write these factors as products of affine linear forms, introducing new hypercube variables.

Combining Thm. 4.2 and Remark A.14 gives the separation \(\text{VNP}_1^w \subseteq \text{VNP}_1^{g} = \text{VNP}\). We can prove a slightly stronger separation by adjusting the construction in the above proof of Thm. 4.2. Namely, let \(S^+ := \{\alpha x_i + \beta x_j + \gamma \mid \alpha, \beta, \gamma \in \mathbb{F}\}\) be the set of affine linear forms in at most two variables and let \(\text{VP}_1^{w+}\) be the class of families that can be computed by width-1 ABPs over \(S^+\) of polynomial size. Define \(\text{VNP}_1^{w+}\) accordingly (Def. 2.2). Then we can adjust the construction in the above proof of Thm. 4.2 to show the following.

**Theorem B.3.** \(\text{VNP}_1^w \subseteq \text{VNP}_1^{w+} = \text{VNP}\) when \(\text{char}(\mathbb{F}) \neq 2\).

**Proof.** We only need to show \(\text{VNP}_1^{w+} = \text{VNP}\), as \(\text{VNP}_1^w \subseteq \text{VNP}\) was shown in Remark A.14. The adjustments we have to make to the construction in the proof of Thm. 4.2 are as follows. Most of the resulting polynomial of the construction is already of the correct form where each linear forms contains at most two variables, since the expression \(\text{Eq}(x, y) = (1 - x - y)^2\) and the expression

\[
1 + xy = \frac{1}{2} \sum_{c \in \{0,1\}} (x + 1 - 2c)(y + 1 - 2c)
\]

are of this form. Three expressions occur that are not of the correct form:

1. \(b_1[i] + b_2[i] + b_3[i]\) in \(V(b_1, b_2, b_3)\),
2. \(a \cdot b_1[i-1] + b \cdot b_2[i-1] + c \cdot b_3[i-1]\) in \(S(M_{a,b,c})\), and
3. \(1 + (x - 1)(b_1[i] - b_1[i-1])\) in \(S(M(x))\)

Expression 1 and expression 2 we can write in the correct form using the identity

\[
\frac{1}{2} \sum_{b \in \{0,1\}} (x + 1 - 2b)(y + 1 - 2b)(z + 1 - 2b) = x + y + z + xyz.
\]

Indeed, expression 1 can be replaced by

\[
\frac{1}{2} \sum_{c \in \{0,1\}} (b_1[i] + 1 - 2c)(b_2[i] + 1 - 2c)(b_3[i] + 1 - 2c)
\]

\[
= b_1[i] + b_2[i] + b_3[i] + b_1[i]b_2[i]b_3[i],
\]

expression 2
since the unwanted term \( b_1[i]b_2[i]b_3[i] \) will always vanish in our construction (because in (7) we multiply with \( 1 - b_s[i]b_t[i] \) for every \( s \neq t \)). Similarly for expression 2.

For expression 3, we first replace \( 1 + (x-1)(b_1[i] - b_1[i-1]) \) by the expression \( \frac{1}{2} \sum_{c \in \{0,1\}} (x - 1 + 1 - 2c)(b_1[i] - b_1[i-1] + 1 - 2c) \). The second factor has too many variables. We replace it, using identity (8), by

\[
\frac{1}{2} \sum_{c \in \{0,1\}} (b_1[i] + 1 - 2c') (-b_1[i-1] + 1 + 1 - 2c') (-2c + 1 - 2c') = b_1[i] - b_1[i-1] + 1 - 2c + b_1[i](1 - b_1[i-1]) (-2c).
\]

The first four summands in the right-hand side are as we want. The last summand is only nonzero if \( b_1[i] = 1 \) and \( b_1[i-1] = 0 \). However, since \( S_i(M(x)) \) contains a factor \( 1 - (1 - b_2[i])b_2[i-1] \) and a factor \( \text{Eq}(b_3[i-1], b_3[i]) \), it can be checked that this last summand will always vanish.

In the new construction thus obtained each linear form is in \( S^+ \). This completes the necessary adjustments to the construction. 

---

**C. Constant-width ABPs have small formulas**

The following well-known proposition says that the iterated product of constant-size matrices can be efficiently computed by a formula.

**Proposition C.1.** Let \( k \geq 1 \). Then \( \text{VP}_k \subseteq \text{VP}_e \).

**Proof.** Let \( (f_n) \in \text{VP}_k \), so \( f_n \) has \( v(n) \in \text{poly}(n) \) variables. There is a function \( m(n) \in \text{poly}(n) \) and there are \( k \times k \) matrices \( M_{n,1}, \ldots, M_{n,m(n)} \) with affine linear forms as entries, such that \( \text{tr}(M_{n,1} \cdots M_{n,m(n)}) = f_n \). We may assume that each affine linear form occurring in \( M_{n,i} \) has \( v(n) \) variables, since \( f_n \) has \( v(n) \) variables. We will recursively construct a multi-output formula computing the product \( M_{n,1} \cdots M_{n,m(n)} \). From this one can efficiently compute the trace. Let \( \text{size}(m, n) \) denote the size of the formula that we construct. Let \( w(n) := 2v(n) \). A single matrix \( M_{n,i} \) we can compute by a multi-output formula of size \( k^2 w(n) \) (the \( w(n) \) is needed to compute each affine linear form). So \( \text{size}(1, n) = k^2 w(n) \). Suppose that matrices \( A \) and \( B \) can be computed by a multi-output formulas \( F \) and \( G \) respectively, each of size \( s \). Then the product \( AB \) can be computed by a multi-output formula of size \( 2ks + c(k) \) with \( c(k) \in \text{O}(k^3) \), as follows: take \( k \) copies of the formula \( F \) for \( A \) and \( k \) copies of the formula \( G \) for \( B \) and appropriately add \( c(k) \times \) - and \( + \)-gates in order to perform the matrix multiplication. (The reason that we take \( k \) copies of the formulas \( F \) and \( G \) is that in the matrix multiplication, each input entry is used \( k \) times, and in formulas we cannot use intermediate results more than once.) Therefore, the recurrence relation \( \text{size}(m, n) = 2k \text{size}(m/2, n) + c(k) \) holds. Working out the recurrence relation gives \( \text{size}(m, n) = (2k)^{\log_2(m)} k^2 w(n) + [(2k)^{\log_2(m)} + (2k)^{\log_2(m)-1} + \cdots + 1] c(k) \). Since \( v(n), m(n) \in \text{poly}(n) \) and since \( k \) is constant in \( n \), we have \( \text{size}(m(n), n) \in \text{poly}(n) \). We can thus also compute the trace of \( M_{n,1} \cdots M_{n,m(n)} \) with a \( \text{poly}(n) \)-size formula. This shows that \( (f_n) \in \text{VP}_e \).

We thus have \( \text{VP}_k \subseteq \text{VP}_e \).

---

**D. Poly-approximation in width-2 ABPs**

We give the interpolation argument that completes the proof of Cor. 3.8, which says that the poly-approximation closure of \( \text{VP}_2 \) equals \( \text{VP}_e \) when \( \text{char}(\mathbb{F}) \neq 2 \) and \( \mathbb{F} \) is infinite.
Proposition D.1. $\text{VP}^\text{poly}_e = \text{VP}_e$ when $\text{char}(\mathbb{F}) \neq 2$ and $\mathbb{F}$ is infinite.

Proof. The inclusion $\text{VP}^\text{poly}_e \subseteq \text{VP}^\text{poly}_e$ is clear. For the other direction, let $(f_n) \in \text{VP}^\text{poly}_e$. Then there are polynomials $f_{n;i}(x) \in \mathbb{F}[x]$, $e(n) \in \text{poly}(n)$ such that

$$f_n(x) + \varepsilon f_{n;1}(x) + \varepsilon^2 f_{n;2}(x) + \cdots + \varepsilon^{e(n)} f_{n;\varepsilon(n)}(x)$$

is computed by a poly-size formula $\Gamma$ over $\mathbb{F}(\varepsilon)$. Let $\alpha_0, \alpha_1, \ldots, \alpha_{e(n)}$ be distinct elements in $\mathbb{F}$ such that replacing $\varepsilon$ by $\alpha_j$ in $\Gamma$ is a valid substitution (these $\alpha_j$ exist since by assumption our field is infinite). View

$$g_n(\varepsilon) := f_n(x) + \varepsilon f_{n;1}(x) + \varepsilon^2 f_{n;2}(x) + \cdots + \varepsilon^{e(n)} f_{n;\varepsilon(n)}(x)$$

as a polynomial in $\varepsilon$. The polynomial $g_n(\varepsilon)$ has degree at most $e(n)$ so we can write $g_n(\varepsilon)$ as follows (Lagrange interpolation on $e(n) + 1$ points)

$$g_n(\varepsilon) = \sum_{j=0}^{e(n)} g_n(\alpha_j) \prod_{0 \leq m \leq e(n): m \neq j} \frac{\varepsilon - \alpha_m}{\alpha_j - \alpha_m}. \quad (9)$$

Clearly, $f_n(x) = g_n(0)$. From (9) we see directly how to write $g_n(0)$ as a linear combination of the values $g_n(\alpha_j)$, namely

$$g_n(0) = \sum_{j=0}^{e(n)} g_n(\alpha_j) \prod_{0 \leq m \leq e(n): m \neq j} -\frac{\alpha_m}{\alpha_j - \alpha_m},$$

that is,

$$g_n(0) = \sum_{j=0}^{e(n)} \beta_j g_n(\alpha_j) \quad \text{with} \quad \beta_j := \prod_{0 \leq m \leq e(n): m \neq j} \frac{\alpha_m}{\alpha_m - \alpha_j}.$$

The value $g_n(\alpha_j)$ is computed by the formula $\Gamma$ with $\varepsilon$ replaced by $\alpha_j$, which we denote by $\Gamma|_{\varepsilon=\alpha_j}$. Thus $f_n(x)$ is computed by the poly-size formula $\sum_{j=0}^{e(n)} \beta_j \Gamma|_{\varepsilon=\alpha_j}$. Therefore $(f_n) \in \text{VP}_e$. \qed

Remark D.2. Prop. D.1 also holds with $\text{VP}^\text{poly}_e$ replaced by $\text{VP}^\text{poly}_s$ or $\text{VP}$ by a similar proof.

E. $\text{VNP}_1 \nsubseteq \text{VNP}$ when $\mathbb{F} = \mathbb{F}_2$

In our proofs of $\text{VNP}_1 = \text{VNP}$ (Sec. 4 and Appendix B) the assumption $\text{char}(\mathbb{F}) \neq 2$ played a crucial role. We can prove that over the finite field $\mathbb{F}_2$ the inclusion $\text{VNP}_1 \subseteq \text{VNP}$ is indeed strict.

Proposition 4.4 (repeated). $\text{VNP}_1 \nsubseteq \text{VNP}$ when $\mathbb{F} = \mathbb{F}_2$.

Proof. Let $\mathbb{F} = \mathbb{F}_2$. Clearly $(1 + xy) \in \text{VNP}$. However, we will prove that the polynomial $1 + xy$ cannot be written as a hypercube sum of affine linear forms. In fact, we will prove something stronger, namely that the function $(x, y) \mapsto 1 + xy$ cannot be written as a hypercube sum of a product of affine linear forms.
Assume the contrary: the function \((x, y) \mapsto 1 + xy\) can be written as a hypercube sum of a product of affine linear forms. We can thus write

\[
1 + xy = \sum_b L_b \quad \text{with} \quad L_b := \prod_{i=1}^n (x + A_i) \prod_{j=1}^\beta (y + B_j) \prod_{k=1}^\gamma (x + y + C_k)
\]  

(10)

for some affine linear forms \(A_i(b), B_j(b), C_k(b)\) in the hypercube variables \(b\). On \(\mathbb{F}_2\) the functions \(x, x^2, x^3, \ldots\) coincide; the functions \(y, y^2, y^3, \ldots\) coincide; and the functions \(x+y, (x+y)^2, (x+y)^3, \ldots\) coincide, so

\[
\Pi_i(x + A_i) = \Pi_i A_i + x \left( \Pi_i(1 + A_i) + \Pi_i A_i \right),
\]

\[
\Pi_j(y + B_j) = \Pi_j B_j + y \left( \Pi_j(1 + B_j) + \Pi_j B_j \right),
\]

\[
\Pi_k(x + y + C_k) = \Pi_k C_k + (x + y) \left( \Pi_k(1 + C_k) + \Pi_k C_k \right).
\]

Multiplying the three expressions and simplifying powers of \(x\) and \(y\) gives

\[
L_b = \prod_{i,j,k} A_i B_j C_k + x \left( \prod_{i,j,k} (1 + A_i) B_j (1 + C_k) + \prod_{i,j,k} A_i B_j C_k \right)
\]

\[
+ y \left( \prod_{i,j,k} A_i + B_j) (1 + C_k) + \prod_{i,j,k} A_i B_j C_k \right)
\]

\[
+ xy \left( \prod_{i,j,k} A_i + B_j) (1 + C_k) + \prod_{i,j,k} (1 + A_i) B_j (1 + C_k) + \prod_{i,j,k} (1 + A_i) B_j C_k \right).
\]

Plugging in the four possible assignments \((x, y) \in \mathbb{F}_2 \times \mathbb{F}_2\) into \(1 + xy = \sum_b L_b\), we see that

\[
\sum_b \prod_{i,j,k} A_i B_j C_k = 1, \tag{11}
\]

\[
\sum_b \prod_{i,j,k} (1 + A_i) B_j (1 + C_k) = 1, \tag{12}
\]

\[
\sum_b \prod_{i,j,k} A_i + B_j) (1 + C_k) = 1, \tag{13}
\]

\[
\sum_b \prod_{i,j,k} (1 + A_i) B_j C_k = 0. \tag{14}
\]

Note that (11) asserts that an odd number of vectors \(b\) satisfy the system of equations

\[
A_i = 1 \quad \forall i,
\]

\[
B_j = 1 \quad \forall j,
\]

\[
C_k = 1 \quad \forall k.
\]

We will show that the above system of equations is inconsistent. Recall that we defined \(\alpha, \beta, \gamma\) as the number of factors \(x + A_i, y + B_j, x + y + C_k\) in (10), respectively. Let \(m := \alpha + \beta + \gamma\). Recall that we defined \(n\) as the number of hypercube variables \(b_i\). As we work over \(\mathbb{F}_2\), any affine linear form in \(b\) can be written as \(a_0 + \sum_{\ell=1}^n a_\ell b_\ell\) with \(a_\ell \in \{0,1\}\). Write the \(\ell\)th linear form in \((A_1, \ldots, A_\alpha, B_1, \ldots, B_\beta, C_1, \ldots, C_\gamma)\) as \(v_0 + \sum_{\ell=1}^n b_\ell v_{\ell,i}\), and let \(v_\ell = (v_{\ell,1}, \ldots, v_{\ell,m})\) for \(0 \leq \ell \leq n\). We define the linear map \(M : \mathbb{F}_2^n \to \mathbb{F}_2^n\) by \(M(b) = \sum_{\ell=1}^n b_\ell v_\ell\). We call a bit vector \(b \in \mathbb{F}_2^n\) a solution of (11) if \(M(b) = v_0 + 1^n 1^\beta 1^\gamma\), where \(1^n 1^\beta 1^\gamma\) is the all-ones vector. Observe that (11) says that there is an odd number of solutions of (11). Since the set of solutions of (11) forms an affine
linear subspace of \((\mathbb{F}_2)^n\), its cardinality is a power of two. The only odd power of two is \(1\), so there is exactly one solution of (11). Let \(b^{(1)}\) be this unique solution: \(M(b^{(1)}) = v_0 + 1^\alpha 1^\beta 1^\gamma\). We do the same for (12) and (13) and find unique solutions \(M(b^{(2)}) = v_0 + 0^\alpha 1^\beta 0^\gamma\) and \(M(b^{(3)}) = v_0 + 1^\alpha 0^\beta 0^\gamma\). Equation (14) asserts that the number of solutions of (14) is even. One solution of (14) is given by \(M(b^{(1)} + b^{(2)} + b^{(3)}) = 3v_0 + 1^\alpha 1^\beta 1^\gamma + 0^\alpha 1^\beta 0^\gamma + 1^\alpha 0^\beta 0^\gamma = v_0 + 0^\alpha 0^\beta 1^\gamma\). Let \(b^{(4)}\) and \(b^{(4')}\) be two distinct solutions of (14) with \(M(b^{(4)}) = M(b^{(4')}) = v_0 + 0^\alpha 0^\beta 1^\gamma\). Then \(M(b^{(2)} + b^{(3)} + b^{(4)}) = v_0 + 1^\alpha 1^\beta 1^\gamma = M(b^{(2)} + b^{(3)} + b^{(4')}),\) which contradicts the uniqueness of \(b^{(1)}\).

**Remark E.2.** Our proof of Prop. 4.4 does not generalize to all fields \(\mathbb{F}\) of characteristic 2, because the polynomial \(1 + xy\) is in fact computable by a hypercube sum of a product of affine linear forms when \(\mathbb{F} = \mathbb{F}_4\) (and thus when \(\mathbb{F} = \mathbb{F}_{2^k}, k \in \mathbb{N}\)). Indeed, \(\mathbb{F}_4 \cong \mathbb{F}_2[Z]/(Z^2 + Z + 1)\), so the element \(Z \in \mathbb{F}_4\) is a third root of unity \((Z^3 = 1)\) and satisfies \(Z^2 + Z + 1 = 0\). It can be checked that therefore \(\sum_{b=0}^1(x + Z^2 y + Z b) \cdot (x + Z y + Z^2 b) \cdot (x + y + b) = 1 + xy.\)

**F. Figures**

The following three figures support the proof of Thm. 3.1, while the last one gives an overview over inclusions and separations of complexity classes.

![Addition Diagram](image)

**Figure F.1: Addition**
Figure F.2: Squaring
\[ f \cdot g + \mathcal{O}(\varepsilon) \sim -\left(\frac{f}{2}\right)^2 + \mathcal{O}(\varepsilon) \]

\[ -g^2 + \mathcal{O}(\varepsilon) \]

\[ (f/2 + g)^2 + \mathcal{O}(\varepsilon) \]

Figure F.3: Multiplication
Figure F.4: Overview of inclusions and separations among $\overline{\text{VP}}^*_k$, $\text{VP}_e$, $\text{VP}_s$, $\text{VP}_e$ and their closures when $\text{char}(\mathbb{F}) \neq 2$.