Demystifying the border of depth-3 circuits

Joint works with Pranjal Dutta & Prateek Dwivedi. [CCC'21, FOCS'21, FOCS'22]

Nitin Saxena
CSE, IIT Kanpur

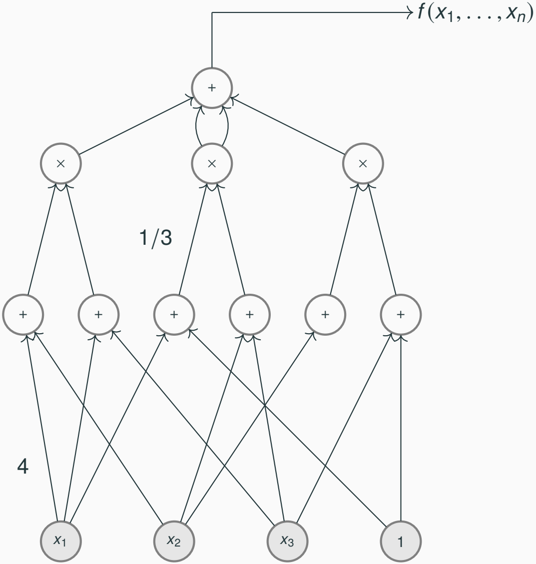
September the 13^{th}, 2022

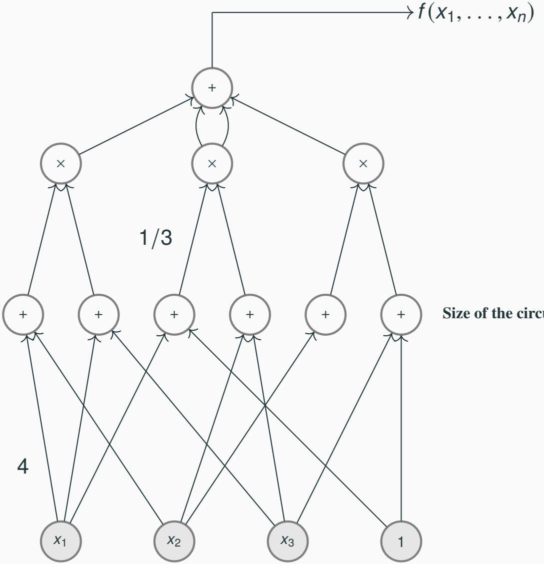
Schloss Dagstuhl, Leibniz-Zentrum

Table of Contents

Basic Definitions and Terminologies

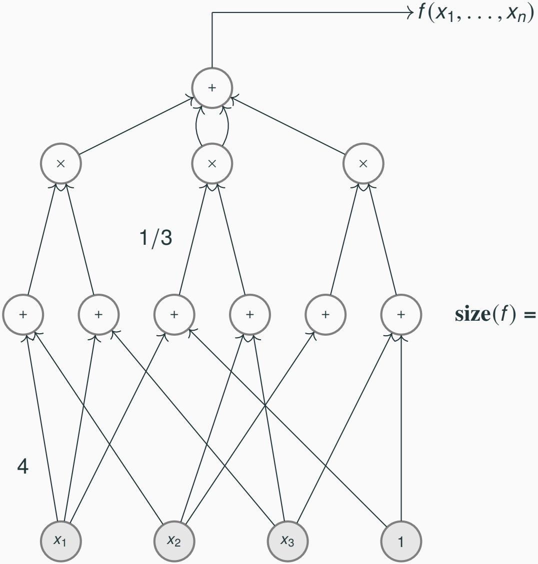
Algebraic circuits– VP





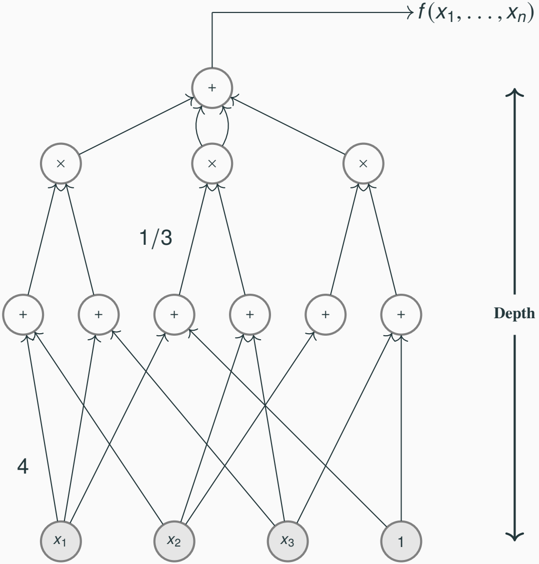
Size of the circuit = number of nodes + edges

Algebraic circuits– VP



size(f) = min size of the circuit computing f

Algebraic circuits– VP



The determinant polynomial– VBP

The determinant polynomial– VBP

- Let $X_n = [x_{i,j}]_{1 \leq i,j \leq n}$ be a $n \times n$ matrix of distinct variables $x_{i,j}$. Let $S_n := \{\pi \mid \pi : \{1, \ldots, n\} \longrightarrow \{1, \ldots, n\}$ such that π is bijective $\}$. Define

$$f_n \; := \; \det(X_n) \; = \; \sum_{\pi \in S_n} \operatorname{sgn}(\pi) \prod_{i=1}^{n} x_{i,\pi(i)} \; .$$

The determinant polynomial– VBP

- Let $X_n = [x_{i,j}]_{1 \le i,j \le n}$ be a $n \times n$ matrix of distinct variables $x_{i,j}$. Let $S_n := \{\pi \mid \pi : \{1, \ldots, n\} \longrightarrow \{1, \ldots, n\}$ such that π is bijective $\}$. Define

$$f_n := \det(X_n) = \sum_{\pi \in S_n} \mathrm{sgn}(\pi) \prod_{i=1}^{n} x_{i,\pi(i)} .$$

- $\det$ is *universal*, i.e. any polynomial $f(\boldsymbol{x})$ can be computed as a determinant of a square matrix whose entries are polynomials of degree ≤ 1.

The determinant polynomial– VBP

- Let $X_n = [x_{i,j}]_{1 \leq i,j \leq n}$ be a $n \times n$ matrix of distinct variables $x_{i,j}$. Let $S_n := \{\pi \mid \pi : \{1, \ldots, n\} \longrightarrow \{1, \ldots, n\}$ such that π is bijective $\}$. Define

$$f_n := \det(X_n) = \sum_{\pi \in S_n} \mathrm{sgn}(\pi) \prod_{i=1}^{n} x_{i,\pi(i)} .$$

- $\det$ is *universal*, i.e. any polynomial $f(\boldsymbol{x})$ can be computed as a determinant of a square matrix whose entries are polynomials of degree ≤ 1.
- The minimum dimension of the matrix to compute f, is called the **determinantal complexity** $\mathrm{dc}(f)$.

The determinant polynomial– VBP

- Let $X_n = [x_{i,j}]_{1 \le i,j \le n}$ be a $n \times n$ matrix of distinct variables $x_{i,j}$. Let $S_n := \{\pi \mid \pi : \{1, \ldots, n\} \longrightarrow \{1, \ldots, n\}$ such that π is bijective $\}$. Define

$$f_n := \det(X_n) = \sum_{\pi \in S_n} \operatorname{sgn}(\pi) \prod_{i=1}^{n} x_{i,\pi(i)} .$$

- det is *universal*, i.e. any polynomial $f(\boldsymbol{x})$ can be computed as a determinant of a square matrix whose entries are polynomials of degree ≤ 1.
- The minimum dimension of the matrix to compute f, is called the **determinantal complexity** $\mathrm{dc}(f)$.
- E.g. $\mathrm{dc}(x_1 \cdots x_n) = n$, since

$$x_1 \cdots x_n = \det \begin{pmatrix} x_1 & 0 & \ldots & 0 \\ 0 & x_2 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & x_n \end{pmatrix} .$$

The determinant polynomial– VBP

- Let $X_n = [x_{i,j}]_{1 \leq i,j \leq n}$ be a $n \times n$ matrix of distinct variables $x_{i,j}$. Let $S_n := \{\pi \mid \pi : \{1, \ldots, n\} \longrightarrow \{1, \ldots, n\}$ such that π is bijective $\}$. Define

$$f_n := \det(X_n) = \sum_{\pi \in S_n} \operatorname{sgn}(\pi) \prod_{i=1}^{n} x_{i,\pi(i)} .$$

- det is *universal*, i.e. any polynomial $f(\boldsymbol{x})$ can be computed as a determinant of a square matrix whose entries are polynomials of degree ≤ 1.
- The minimum dimension of the matrix to compute f, is called the **determinantal complexity** $\mathrm{dc}(f)$.
- E.g. $\mathrm{dc}(x_1 \cdots x_n) = n$, since

$$x_1 \cdots x_n = \det \begin{pmatrix} x_1 & 0 & \ldots & 0 \\ 0 & x_2 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & x_n \end{pmatrix} .$$

- VBP: The class VBP is defined as the set of all sequences of polynomials $(f_n)_n$ with polynomially bounded $\mathrm{dc}(f_n)$.

‘Hard’ polynomials?

‘Hard’ polynomials?

- Are there hard polynomial families $(f_n)_n$ such that it cannot be computed by an n^c-size circuit, for *every* constant c? i.e. $\text{size}(f_n) = n^{\omega(1)}$?

'Hard' polynomials?

- Are there hard polynomial families $(f_n)_n$ such that it cannot be computed by an n^c-size circuit, for *every* constant c? i.e. $\text{size}(f_n) = n^{\omega(1)}$?
- A *random* polynomial with 0-1 coefficient is **hard** [Hrubeš-Yehudayoff ToC'11].

‘Hard’ polynomials?

- Are there hard polynomial families $(f_n)_n$ such that it cannot be computed by an n^c-size circuit, for *every* constant c? i.e. $\text{size}(f_n) = n^{\omega(1)}$?
- A *random* polynomial with 0-1 coefficient is **hard** [Hrubeš-Yehudayoff ToC’11]. Challenge: Find an **explicit** one!

‘Hard’ polynomials?

- Are there hard polynomial families $(f_n)_n$ such that it cannot be computed by an n^c-size circuit, for *every* constant c? i.e. $\text{size}(f_n) = n^{\omega(1)}$?
- A *random* polynomial with $0\text{-}1$ coefficient is **hard** [Hrubeš-Yehudayoff ToC’11]. Challenge: Find an **explicit** one!
- Candidate hard polynomial:

$$\text{perm}(X_n) = \sum_{\pi \in S_n} \prod_{i=1}^{n} x_{i,\pi(i)} \,.$$

‘Hard’ polynomials?

- Are there hard polynomial families $(f_n)_n$ such that it cannot be computed by an n^c-size circuit, for *every* constant c? i.e. $\text{size}(f_n) = n^{\omega(1)}$?

- A *random* polynomial with 0-1 coefficient is **hard** [Hrubeš-Yehudayoff ToC’11]. Challenge: Find an **explicit** one!

- Candidate hard polynomial:

$$\text{perm}(X_n) \;=\; \sum_{\pi \in S_n} \prod_{i=1}^{n} x_{i,\pi(i)} \;.$$

- perm is *universal*, i.e. any polynomial $f(\boldsymbol{x})$ can be computed as a permanent of a square matrix whose entries are polynomials of degree ≤ 1.

‘Hard’ polynomials?

- Are there hard polynomial families $(f_n)_n$ such that it cannot be computed by an n^c-size circuit, for *every* constant c? i.e. $\mathrm{size}(f_n) = n^{\omega(1)}$?

- A *random* polynomial with 0-1 coefficient is **hard** [Hrubeš-Yehudayoff ToC’11]. Challenge: Find an **explicit** one!

- Candidate hard polynomial:

$$\mathrm{perm}(X_n) \;=\; \sum_{\pi \in S_n} \prod_{i=1}^{n} x_{i,\pi(i)}\;.$$

- perm is *universal*, i.e. any polynomial $f(\boldsymbol{x})$ can be computed as a permanent of a square matrix whose entries are polynomials of degree ≤ 1.

- The minimum dimension of the matrix to compute f, is called the **permanental complexity** $\mathrm{pc}(f)$.

Valiant's Conjecture– VNP

VNP = "hard to compute?" [Valiant 1979]

The class VNP is defined as the set of all sequences of polynomials $(f_n(x_1, \ldots, x_n))_{n \geq 1}$ such that $\mathrm{pc}(f_n)$is bounded by n^c for some constant c.

Valiant's Conjecture– VNP

VNP = "hard to compute?" [Valiant 1979]

The class VNP is defined as the set of all sequences of polynomials $(f_n(x_1, \ldots, x_n))_{n \geq 1}$ such that $\mathrm{pc}(f_n)$is bounded by n^c for some constant c.

- $\mathrm{VBP} \subseteq \mathrm{VP} \subseteq \mathrm{VNP}$.

Valiant's Conjecture– VNP

VNP = "hard to compute?" [Valiant 1979]

The class VNP is defined as the set of all sequences of polynomials $(f_n(x_1, \ldots, x_n))_{n \geq 1}$ such that $\mathrm{pc}(f_n)$is bounded by n^c for some constant c.

- VBP $\subseteq$ VP $\subseteq$ VNP.

Valiant's Conjecture [Valiant 1979]

VBP $\neq$ VNP & VP $\neq$ VNP. Equivalently, $\mathrm{dc}(\mathrm{perm}_n)$ and $\mathrm{size}(\mathrm{perm}_n)$ are both $n^{\omega(1)}$.

Connections to Boolean circuit complexity

Connections to Boolean circuit complexity

- Separating algebraic classes are "easier" than separating classes in Boolean complexity [Bürgisser 1998]:

Connections to Boolean circuit complexity

- Separating algebraic classes are "easier" than separating classes in Boolean complexity [Bürgisser 1998]:
  - $\mathsf{P/poly} \neq \mathsf{NP/poly} \implies \mathsf{VBP} \neq \mathsf{VNP}$ and $\mathsf{VP} \neq \mathsf{VNP}$ (over finite fields).

Connections to Boolean circuit complexity

- Separating algebraic classes are “easier” than separating classes in Boolean complexity [Bürgisser 1998]:
  - $\mathsf{P/poly} \neq \mathsf{NP/poly} \implies \mathsf{VBP} \neq \mathsf{VNP}$ and $\mathsf{VP} \neq \mathsf{VNP}$ (over finite fields).
  - Assumin GRH (Generalized Riemann hypothesis), the results hold over $\mathbb{C}$ as well.

Connections to Boolean circuit complexity

- Separating algebraic classes are “easier” than separating classes in Boolean complexity [Bürgisser 1998]:
  - $\mathsf{P/poly} \neq \mathsf{NP/poly} \implies \mathsf{VBP} \neq \mathsf{VNP}$ and $\mathsf{VP} \neq \mathsf{VNP}$ (over finite fields).
  - Assumin GRH (Generalized Riemann hypothesis), the results hold over $\mathbb{C}$ as well.

Border Complexity and GCT

Waring rank— Depth-3 diagonal

- Can ‘approximations’ also help in algebraic computational models?

Waring rank— Depth-3 diagonal

- Can 'approximations' also help in algebraic computational models?
- An important measure is **Waring rank**, denoted $\mathsf{WR}(\cdot)$.

Waring rank— Depth-3 diagonal

- Can 'approximations' also help in algebraic computational models?
- An important measure is **Waring rank**, denoted $\mathsf{WR}(\cdot)$.

Waring Rank

The smallest r such that a *homogeneous* degree d polynomial h can be written as a sum of d-th power of linear forms ℓ_i, i.e. $h = \sum_{i=1}^{r} \ell_i^d$.

Waring rank— Depth-3 diagonal

- Can 'approximations' also help in algebraic computational models?
- An important measure is **Waring rank**, denoted $\mathsf{WR}(\cdot)$.

Waring Rank

The smallest r such that a *homogeneous* degree d polynomial h can be written as a sum of d-th power of linear forms ℓ_i, i.e. $h = \sum_{i=1}^{r} \ell_i^d$.

- Recall: $h = \sum_{e_1,..,e_n} a_{e_1,...,e_n} x_1^{e_1} \cdots x_n^{e_n}$, is called **homogeneous** degree d polynomial if $\sum e_i = d$, for every tuple $(e_1, \ldots, e_n)$ with $a_{e_1,...,e_n} \neq 0$.

Waring rank— Depth-3 diagonal

- Can 'approximations' also help in algebraic computational models?
- An important measure is **Waring rank**, denoted $\mathsf{WR}(\cdot)$.

Waring Rank

The smallest r such that a *homogeneous* degree d polynomial h can be written as a sum of d-th power of linear forms ℓ_i, i.e. $h = \sum_{i=1}^{r} \ell_i^d$.

- Recall: $h = \sum_{e_1,..,e_n} a_{e_1,\ldots,e_n} x_1^{e_1} \cdots x_n^{e_n}$, is called **homogeneous** degree d polynomial if $\sum e_i = d$, for every tuple $(e_1, \ldots, e_n)$ with $a_{e_1,\ldots,e_n} \neq 0$.
- Recall: A linear form ℓ is of the form $a_1 x_1 + \ldots + a_n x_n$.

Waring rank— Depth-3 diagonal

- Can 'approximations' also help in algebraic computational models?
- An important measure is **Waring rank**, denoted $\mathsf{WR}(\cdot)$.

Waring Rank

The smallest r such that a *homogeneous* degree d polynomial h can be written as a sum of d-th power of linear forms ℓ_i, i.e. $h = \sum_{i=1}^{r} \ell_i^d$.

- Recall: $h = \sum_{e_1,..,e_n} a_{e_1,...,e_n} x_1^{e_1} \cdots x_n^{e_n}$, is called **homogeneous** degree d polynomial if $\sum e_i = d$, for every tuple $(e_1, \ldots, e_n)$ with $a_{e_1,...,e_n} \neq 0$.
- Recall: A linear form ℓ is of the form $a_1 x_1 + \ldots + a_n x_n$.
- For any homogeneous polynomial h, $\mathsf{WR}(h)$ is *finite*.

Waring rank— Depth-3 diagonal

- Can 'approximations' also help in algebraic computational models?
- An important measure is **Waring rank**, denoted $\mathsf{WR}(\cdot)$.

Waring Rank

The smallest r such that a *homogeneous* degree d polynomial h can be written as a sum of d-th power of linear forms ℓ_i, i.e. $h = \sum_{i=1}^{r} \ell_i^d$.

- Recall: $h = \sum_{e_1,..,e_n} a_{e_1,...,e_n} x_1^{e_1} \cdots x_n^{e_n}$, is called **homogeneous** degree d polynomial if $\sum e_i = d$, for every tuple $(e_1, \ldots, e_n)$ with $a_{e_1,...,e_n} \neq 0$.
- Recall: A linear form ℓ is of the form $a_1x_1 + \ldots + a_nx_n$.
- For any homogeneous polynomial h, $\mathsf{WR}(h)$ is *finite*.
- Often $\mathsf{WR}(h) \leq r$ is denoted as $h \in \Sigma^{[r]} \wedge \Sigma$ (homogeneous *depth-3 diagonal* circuits).

Approximation helps

- Example: $\mathsf{WR}(x^2y) \leq 3$, because

Approximation helps

- Example: $\mathsf{WR}(x^2y) \leq 3$, because

$$x^2y \;=\; \frac{1}{6}\cdot(x+y)^3 - \frac{1}{6}\cdot(x-y)^3 - \frac{1}{3}\cdot y^3\;.$$

Approximation helps

- Example: $\mathrm{WR}(x^2y) \le 3$, because

$$x^2y \;=\; \frac{1}{6}\cdot(x+y)^3 - \frac{1}{6}\cdot(x-y)^3 - \frac{1}{3}\cdot y^3 \;.$$

- Prove: $\mathrm{WR}(x^2y) = 3$.

Approximation helps

- Example: $\mathsf{WR}(x^2y) \leq 3$, because

$$x^2y \;=\; \frac{1}{6}\cdot(x+y)^3 - \frac{1}{6}\cdot(x-y)^3 - \frac{1}{3}\cdot y^3 \;.$$

- Prove: $\mathsf{WR}(x^2y) = 3$.

- Let $h_\epsilon := \frac{1}{3\epsilon}\left((x+\epsilon y)^3 - x^3\right)$

Approximation helps

- Example: $\mathsf{WR}(x^2y) \leq 3$, because

$$x^2y = \frac{1}{6} \cdot (x+y)^3 - \frac{1}{6} \cdot (x-y)^3 - \frac{1}{3} \cdot y^3 .$$

- Prove: $\mathsf{WR}(x^2y) = 3$.

- Let $h_\epsilon := \frac{1}{3\epsilon}\left((x+\epsilon y)^3 - x^3\right)$

$$= x^2y + \epsilon xy^2 + \frac{\epsilon^2}{3}y^3 \overset{\epsilon\to 0}{\longrightarrow} x^2y =: h \quad \text{(coefficient-wise).}$$

Approximation helps

- Example: $\mathrm{WR}(x^2y) \leq 3$, because

$$x^2y \;=\; \frac{1}{6}\cdot(x+y)^3 - \frac{1}{6}\cdot(x-y)^3 - \frac{1}{3}\cdot y^3 \,.$$

- Prove: $\mathrm{WR}(x^2y) = 3$.

- Let $h_\epsilon := \frac{1}{3\epsilon}\left((x+\epsilon y)^3 - x^3\right)$

  $= x^2y + \epsilon xy^2 + \frac{\epsilon^2}{3}y^3 \overset{\epsilon\to 0}{\longrightarrow} x^2y =: h$ (coefficient-wise).

- Note: $\mathrm{WR}(h_\epsilon) \leq 2$, for any fixed non-zero ϵ. But $\mathrm{WR}(h) = 3$!

Approximation helps

- Example: $\mathrm{WR}(x^2y) \leq 3$, because

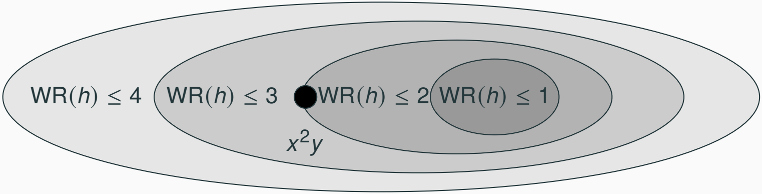
$$x^2y = \frac{1}{6}\cdot(x+y)^3 - \frac{1}{6}\cdot(x-y)^3 - \frac{1}{3}\cdot y^3 .$$

- Prove: $\mathrm{WR}(x^2y) = 3$.

- Let $h_\epsilon := \frac{1}{3\epsilon}\left((x+\epsilon y)^3 - x^3\right)$

$$= x^2y + \epsilon xy^2 + \frac{\epsilon^2}{3}y^3 \overset{\epsilon\to 0}{\longrightarrow} x^2y =: h \quad \text{(coefficient-wise).}$$

- Note: $\mathrm{WR}(h_\epsilon) \leq 2$, for any fixed non-zero ϵ. But $\mathrm{WR}(h) = 3$!



Border Waring Rank— *Approximative* depth-3 diagonal

Border Waring Rank— *Approximative* depth-3 diagonal

Border Waring rank

The border Waring rank $\overline{\mathsf{WR}}(h)$, of a d-form h is defined as the smallest s such that $h = \lim_{\epsilon \to 0} \sum_{i \in [s]} \ell_i^d$, where $\ell_i \in \mathbb{F}(\epsilon)[\mathbf{x}]$, are homogeneous linear forms.

Border Waring Rank— *Approximative* depth-3 diagonal

Border Waring rank

The border Waring rank $\overline{\mathsf{WR}}(h)$, of a d-form h is defined as the smallest s such that $h = \lim_{\epsilon \to 0} \sum_{i \in [s]} \ell_i^d$, where $\ell_i \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$, are homogeneous linear forms.

- $\overline{\mathsf{WR}}(x^2y) = 2 < \mathsf{WR}(x^2y) = 3$.

Border complexity

Border complexity

- Let Γ be any sensible measure. It can be size, dc and so on.

Border complexity

- Let Γ be any sensible measure. It can be size, dc and so on.
- For any Γ, we can define the border complexity measure $\overline{\Gamma}$ via:
  $\overline{\Gamma}(h)$ is the *smallest* n such that $h(\boldsymbol{x})$ can be approximated arbitrarily closely by polynomials $h_\epsilon(\boldsymbol{x})$ with $\Gamma(h_\epsilon) \leq n$.

Border complexity

- Let Γ be any sensible measure. It can be size, dc and so on.
- For any Γ, we can define the border complexity measure $\overline{\Gamma}$ via:
  $\overline{\Gamma}(h)$ is the *smallest* n such that $h(\boldsymbol{x})$ can be approximated arbitrarily closely by polynomials $h_\epsilon(\boldsymbol{x})$ with $\Gamma(h_\epsilon) \leq n$. In other words,

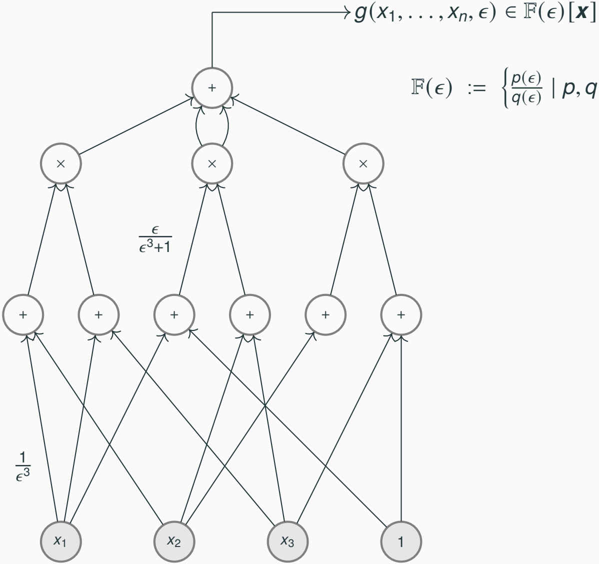
$$\lim_{\epsilon \to 0} h_\epsilon \;=\; h \;\; \text{(coefficient-wise)}\;.$$

Border complexity

- Let Γ be any sensible measure. It can be size, dc and so on.
- For any Γ, we can define the border complexity measure $\overline{\Gamma}$ via:
  $\overline{\Gamma}(h)$ is the *smallest* n such that $h(\mathbf{x})$ can be approximated arbitrarily closely by polynomials $h_\epsilon(\mathbf{x})$ with $\Gamma(h_\epsilon) \leq n$. In other words,

$$\lim_{\epsilon \to 0} h_\epsilon = h \text{ (coefficient-wise)} .$$

- We will work with *'approximative circuits'*.



$$\mathbb{F}(\epsilon) := \left\{ \frac{p(\epsilon)}{q(\epsilon)} \mid p, q \in \mathbb{F}[\epsilon], q(\epsilon) \neq 0 \right\}$$

Algebraic approximation

Algebraic approximation

- Suppose, we assume the following:
  - $g(\boldsymbol{x}, \epsilon) \in \mathbb{F}[x_1, \ldots, x_n, \epsilon]$, i.e. it is a polynomial of the form

$$g(\boldsymbol{x}, \epsilon) = \sum_{i=0}^{M} g_i(x_1, \ldots, x_n) \cdot \epsilon^i ,$$

Algebraic approximation

- Suppose, we assume the following:
  - $g(\boldsymbol{x}, \epsilon) \in \mathbb{F}[x_1, \ldots, x_n, \epsilon]$, i.e. it is a polynomial of the form

$$g(\boldsymbol{x}, \epsilon) \ = \ \sum_{i=0}^{M} g_i(x_1, \ldots, x_n) \cdot \epsilon^i \ ,$$

  - Can we say anything about the complexity of g_0?

Algebraic approximation

- Suppose, we assume the following:
  - $g(\boldsymbol{x}, \epsilon) \in \mathbb{F}[x_1, \ldots, x_n, \epsilon]$, i.e. it is a polynomial of the form

$$g(\boldsymbol{x}, \epsilon) \;=\; \sum_{i=0}^{M} g_i(x_1, \ldots, x_n) \cdot \epsilon^i \;,$$

  - Can we say anything about the complexity of g_0?
- Obvious attempt:
  - Since, $g(\boldsymbol{x}, 0) = g_0$, why not just set $\epsilon = 0$?!

Algebraic approximation

- Suppose, we assume the following:
  - $g(\boldsymbol{x}, \epsilon) \in \mathbb{F}[x_1, \ldots, x_n, \epsilon]$, i.e. it is a polynomial of the form

$$g(\boldsymbol{x}, \epsilon) \;=\; \sum_{i=0}^{M} g_i(x_1, \ldots, x_n) \cdot \epsilon^i \;,$$

  - Can we say anything about the complexity of g_0?
- Obvious attempt:
  - Since, $g(\boldsymbol{x}, 0) = g_0$, why not just set $\epsilon = 0$?! Setting $\epsilon = 0$ *may not* be 'valid' as it could be using $1/\epsilon$ in the wire. Though it is well-defined!

Algebraic approximation

- Suppose, we assume the following:
  - $g(\boldsymbol{x}, \epsilon) \in \mathbb{F}[x_1, \ldots, x_n, \epsilon]$, i.e. it is a polynomial of the form

$$g(\boldsymbol{x}, \epsilon) \;=\; \sum_{i=0}^{M} g_i(x_1, \ldots, x_n) \cdot \epsilon^i \;,$$

  - Can we say anything about the complexity of g_0?
- Obvious attempt:
  - Since, $g(\boldsymbol{x}, 0) = g_0$, why not just set $\epsilon = 0$?! Setting $\epsilon = 0$ *may not* be 'valid' as it could be using $1/\epsilon$ in the wire. Though it is well-defined!
- Summary: g_0 is really something **non-trivial** and being 'approximated' by the circuit since $\lim_{\epsilon \to 0} g(\boldsymbol{x}, \epsilon) = g_0$.

Algebraic approximation— $\overline{\mathrm{VP}}$

Algebraic approximation— $\overline{\mathrm{VP}}$

Algebraic Approximation [Bürgisser 2004]

A polynomial $h(x) \in \mathbb{F}[\boldsymbol{x}]$ approximative complexity s, if there is a $g(\boldsymbol{x}, \epsilon) \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$, of size s, over $\mathbb{F}(\epsilon)$, and a polynomial $S(\boldsymbol{x}, \epsilon) \in \mathbb{F}[\epsilon][\boldsymbol{x}]$ such that $g(\boldsymbol{x}, \epsilon) = h(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon)$. In other words, $\lim_{\epsilon \to 0} g = h$.

Algebraic Approximation [Bürgisser 2004]

A polynomial $h(x) \in \mathbb{F}[\boldsymbol{x}]$ approximative complexity s, if there is a $g(\boldsymbol{x}, \epsilon) \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$, of size s, over $\mathbb{F}(\epsilon)$, and a polynomial $S(\boldsymbol{x}, \epsilon) \in \mathbb{F}[\epsilon][\boldsymbol{x}]$ such that $g(\boldsymbol{x}, \epsilon) = h(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon)$. In other words, $\lim_{\epsilon \to 0} g = h$.

- $\overline{\mathrm{size}}(h) \leq \mathrm{size}(h)$.

Algebraic approximation— $\overline{\mathrm{VP}}$

Algebraic Approximation [Bürgisser 2004]

A polynomial $h(x) \in \mathbb{F}[\boldsymbol{x}]$ approximative complexity s, if there is a $g(\boldsymbol{x}, \epsilon) \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$, of size s, over $\mathbb{F}(\epsilon)$, and a polynomial $S(\boldsymbol{x}, \epsilon) \in \mathbb{F}[\epsilon][\boldsymbol{x}]$ such that $g(\boldsymbol{x}, \epsilon) = h(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon)$. In other words, $\lim_{\epsilon \to 0} g = h$.

- $\overline{\text{size}}(h) \leq \text{size}(h)$. $[h = h + \epsilon \cdot 0.]$

Algebraic approximation— $\overline{\mathrm{VP}}$

Algebraic Approximation [Bürgisser 2004]

A polynomial $h(x) \in \mathbb{F}[\boldsymbol{x}]$ approximative complexity s, if there is a $g(\boldsymbol{x}, \epsilon) \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$, of size s, over $\mathbb{F}(\epsilon)$, and a polynomial $S(\boldsymbol{x}, \epsilon) \in \mathbb{F}[\epsilon][\boldsymbol{x}]$ such that $g(\boldsymbol{x}, \epsilon) = h(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon)$. In other words, $\lim_{\epsilon \to 0} g = h$.

- $\overline{\text{size}}(h) \leq \text{size}(h)$. $[h = h + \epsilon \cdot 0.]$
- If g has circuit of size s over $\mathbb{F}(\epsilon)$, then one can assume that the highest degree of ϵ in g can be *exponentially large* 2^{s^2} [Bürgisser 2004, 2020].

Algebraic Approximation [Bürgisser 2004]

A polynomial $h(x) \in \mathbb{F}[\boldsymbol{x}]$ approximative complexity s, if there is a $g(\boldsymbol{x}, \epsilon) \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$, of size s, over $\mathbb{F}(\epsilon)$, and a polynomial $S(\boldsymbol{x}, \epsilon) \in \mathbb{F}[\epsilon][\boldsymbol{x}]$ such that $g(\boldsymbol{x}, \epsilon) = h(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon)$. In other words, $\lim_{\epsilon \to 0} g = h$.

- $\overline{\text{size}}(h) \leq \text{size}(h)$. [$h = h + \epsilon \cdot 0$.]
- If g has circuit of size s over $\mathbb{F}(\epsilon)$, then one can assume that the highest degree of ϵ in g can be *exponentially large* 2^{s^2} [Bürgisser 2004, 2020]. [Bezout's degree theorem.]

Algebraic Approximation [Bürgisser 2004]

A polynomial $h(x) \in \mathbb{F}[\boldsymbol{x}]$ approximative complexity s, if there is a $g(\boldsymbol{x}, \epsilon) \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$, of size s, over $\mathbb{F}(\epsilon)$, and a polynomial $S(\boldsymbol{x}, \epsilon) \in \mathbb{F}[\epsilon][\boldsymbol{x}]$ such that $g(\boldsymbol{x}, \epsilon) = h(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon)$. In other words, $\lim_{\epsilon \to 0} g = h$.

- $\overline{\text{size}}(h) \leq \text{size}(h)$. $[h = h + \epsilon \cdot 0.]$
- If g has circuit of size s over $\mathbb{F}(\epsilon)$, then one can assume that the highest degree of ϵ in g can be *exponentially large* 2^{s^2} [Bürgisser 2004, 2020]. [Bezout's degree theorem.]
- Let us assume that $g(\boldsymbol{x}, \epsilon) = \sum_{i=0}^{M} g_i \epsilon^i$, where $M = 2^{s^2}$. Note: $g_0 = h$.

Algebraic Approximation [Bürgisser 2004]

A polynomial $h(x) \in \mathbb{F}[\boldsymbol{x}]$ approximative complexity s, if there is a $g(\boldsymbol{x}, \epsilon) \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$, of size s, over $\mathbb{F}(\epsilon)$, and a polynomial $S(\boldsymbol{x}, \epsilon) \in \mathbb{F}[\epsilon][\boldsymbol{x}]$ such that $g(\boldsymbol{x}, \epsilon) = h(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon)$. In other words, $\lim_{\epsilon \to 0} g = h$.

- $\overline{\text{size}}(h) \leq \text{size}(h)$. [$h = h + \epsilon \cdot 0$.]
- If g has circuit of size s over $\mathbb{F}(\epsilon)$, then one can assume that the highest degree of ϵ in g can be *exponentially large* 2^{s^2} [Bürgisser 2004, 2020]. [Bezout's degree theorem.]
- Let us assume that $g(\boldsymbol{x}, \epsilon) = \sum_{i=0}^{M} g_i \epsilon^i$, where $M = 2^{s^2}$. Note: $g_0 = h$.
    - ➢ Pick $M + 1$ many distinct values from $\mathbb{F}$ *randomly* and interpolate;

Algebraic approximation— $\overline{\text{VP}}$

Algebraic Approximation [Bürgisser 2004]

A polynomial $h(x) \in \mathbb{F}[\boldsymbol{x}]$ approximative complexity s, if there is a $g(\boldsymbol{x}, \epsilon) \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$, of size s, over $\mathbb{F}(\epsilon)$, and a polynomial $S(\boldsymbol{x}, \epsilon) \in \mathbb{F}[\epsilon][\boldsymbol{x}]$ such that $g(\boldsymbol{x}, \epsilon) = h(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon)$. In other words, $\lim_{\epsilon \to 0} g = h$.

- $\overline{\text{size}}(h) \leq \text{size}(h)$. [$h = h + \epsilon \cdot 0$.]
- If g has circuit of size s over $\mathbb{F}(\epsilon)$, then one can assume that the highest degree of ϵ in g can be *exponentially large* 2^{s^2} [Bürgisser 2004, 2020]. [Bezout's degree theorem.]
- Let us assume that $g(\boldsymbol{x}, \epsilon) = \sum_{i=0}^{M} g_i \epsilon^i$, where $M = 2^{s^2}$. Note: $g_0 = h$.
  - Pick $M + 1$ many distinct values from $\mathbb{F}$ *randomly* and interpolate;
  - $\text{size}(h) \leq \exp(\overline{\text{size}}(h))$.

Algebraic Approximation [Bürgisser 2004]

A polynomial $h(x) \in \mathbb{F}[\boldsymbol{x}]$ approximative complexity s, if there is a $g(\boldsymbol{x}, \epsilon) \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$, of size s, over $\mathbb{F}(\epsilon)$, and a polynomial $S(\boldsymbol{x}, \epsilon) \in \mathbb{F}[\epsilon][\boldsymbol{x}]$ such that $g(\boldsymbol{x}, \epsilon) = h(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon)$. In other words, $\lim_{\epsilon \to 0} g = h$.

- $\overline{\text{size}}(h) \leq \text{size}(h)$. [$h = h + \epsilon \cdot 0$.]
- If g has circuit of size s over $\mathbb{F}(\epsilon)$, then one can assume that the highest degree of ϵ in g can be *exponentially large* 2^{s^2} [Bürgisser 2004, 2020]. [Bezout's degree theorem.]
- Let us assume that $g(\boldsymbol{x}, \epsilon) = \sum_{i=0}^{M} g_i \epsilon^i$, where $M = 2^{s^2}$. Note: $g_0 = h$.
  - ➢ Pick $M + 1$ many distinct values from $\mathbb{F}$ *randomly* and interpolate;
  - ➢ $\text{size}(h) \leq \exp(\overline{\text{size}}(h))$.
- $\overline{\text{size}}(h) \leq \text{size}(h) \leq \exp(\overline{\text{size}}(h))$.

Lower bounds for border depth-2 circuits

Lower bounds for border depth-2 circuits

- A few known upper bound/lower bound results on depth-2:

Lower bounds for border depth-2 circuits

- A few known upper bound/lower bound results on depth-2:
  - $\overline{\Sigma^{[s]}\Pi} = \Sigma^{[s]}\Pi$ and $\overline{\Pi\Sigma} = \Pi\Sigma$.

Lower bounds for border depth-2 circuits

- A few known upper bound/lower bound results on depth-2:
  - $\overline{\Sigma^{[s]}\Pi} = \Sigma^{[s]}\Pi$ and $\overline{\Pi\Sigma} = \Pi\Sigma$.
  - det_n is irreducible and $\mathsf{exp}(n)$-sparse. So, det_n *requires* exponential-size border depth-2 circuits!

Lower bounds for border depth-2 circuits

- A few known upper bound/lower bound results on depth-2:
  - $\overline{\Sigma^{[s]}\Pi} = \Sigma^{[s]}\Pi$ and $\overline{\Pi\Sigma} = \Pi\Sigma$.
  - det_n is irreducible and $\mathsf{exp}(n)$-sparse. So, det_n *requires* exponential-size border depth-2 circuits!
  - What about border depth-3 circuits (both upper bound and lower bound)?

Border Depth-3 Circuits

Depth-3 circuits

Depth-3 circuits

- Depth-3 circuits with top fan-in k, are denoted as $\Sigma^{[k]}\Pi\Sigma$.

Depth-3 circuits

- Depth-3 circuits with top fan-in k, are denoted as $\Sigma^{[k]}\Pi\Sigma$.
- They compute polynomials (*not necessarily* homogeneous) of the form $\sum_{i=1}^{k} \prod_{j=1}^{d_i} \ell_{ij}$, where ℓ_{ij} are linear polynomials (i.e. $a_0 + a_1x_1 + \ldots + a_nx_n$, for $a_i \in \mathbb{F}$).

Depth-3 circuits

- Depth-3 circuits with top fan-in k, are denoted as $\Sigma^{[k]}\Pi\Sigma$.
- They compute polynomials (*not necessarily* homogeneous) of the form $\sum_{i=1}^{k}\prod_{j=1}^{d_i}\ell_{ij}$, where ℓ_{ij} are linear polynomials (i.e. $a_0 + a_1x_1 + \ldots + a_nx_n$, for $a_i \in \mathbb{F}$).
- Product fan-in $= \max d_i$.

Depth-3 circuits

- Depth-3 circuits with top fan-in k, are denoted as $\Sigma^{[k]}\Pi\Sigma$.
- They compute polynomials (*not necessarily* homogeneous) of the form $\sum_{i=1}^{k} \prod_{j=1}^{d_i} \ell_{ij}$, where ℓ_{ij} are linear polynomials (i.e. $a_0 + a_1 x_1 + \ldots + a_n x_n$, for $a_i \in \mathbb{F}$).
- Product fan-in $= \max d_i$.
- How powerful are $\Sigma^{[k]}\Pi\Sigma$ circuits, for constant k? Are they *universal*?

Depth-3 circuits

- Depth-3 circuits with top fan-in k, are denoted as $\Sigma^{[k]}\Pi\Sigma$.
- They compute polynomials (*not necessarily* homogeneous) of the form $\sum_{i=1}^{k}\prod_{j=1}^{d_i} \ell_{ij}$, where ℓ_{ij} are linear polynomials (i.e. $a_0 + a_1x_1 + \ldots + a_nx_n$, for $a_i \in \mathbb{F}$).
- Product fan-in = $\max d_i$.
- How powerful are $\Sigma^{[k]}\Pi\Sigma$ circuits, for constant k? Are they *universal*?
- Impossibility result: The *Inner Product* polynomial $\langle \boldsymbol{x}, \boldsymbol{y}\rangle = x_1y_1 + \ldots + x_{k+1}y_{k+1}$ **cannot** be written as a $\Sigma^{[k]}\Pi\Sigma$ circuit, *regardless* of the product fan-in (even allowing $\exp(n)$ product fan-in!).

Depth-3 circuits

- Depth-3 circuits with top fan-in k, are denoted as $\Sigma^{[k]}\Pi\Sigma$.
- They compute polynomials (*not necessarily* homogeneous) of the form $\sum_{i=1}^{k}\prod_{j=1}^{d_i}\ell_{ij}$, where ℓ_{ij} are linear polynomials (i.e. $a_0 + a_1x_1 + \ldots + a_nx_n$, for $a_i \in \mathbb{F}$).
- Product fan-in = $\max d_i$.
- How powerful are $\Sigma^{[k]}\Pi\Sigma$ circuits, for constant k? Are they *universal*?
- Impossibility result: The *Inner Product* polynomial $\langle \boldsymbol{x}, \boldsymbol{y}\rangle = x_1y_1 + \ldots + x_{k+1}y_{k+1}$ **cannot** be written as a $\Sigma^{[k]}\Pi\Sigma$ circuit, *regardless* of the product fan-in (even allowing $\exp(n)$ product fan-in!).
- The same holds if we replace by det_n.

Lower bound for Constant fanin depth-3 circuits

- det_n *cannot* be computed by a $\Sigma^{[k]}\Pi\Sigma$-circuit, *regardless* of the product fan-in.

Lower bound for Constant fanin depth-3 circuits

- det_n *cannot* be computed by a $\Sigma^{[k]}\Pi\Sigma$-circuit, *regardless* of the product fan-in.

  Proof sketch: Let $\sum_{i \in [k]} T_i = \mathsf{det}_n$, where $T_i \in \Pi\Sigma$, product of linear polynomials.

Lower bound for Constant fanin depth-3 circuits

- $\mathsf{det_n}$ *cannot* be computed by a $\Sigma^{[k]}\Pi\Sigma$-circuit, *regardless* of the product fan-in.

  Proof sketch: Let $\sum_{i\in[k]} T_i = \mathsf{det}_n$, where $T_i \in \Pi\Sigma$, product of linear polynomials.

  - Pick ℓ_1 from T_1 and consider $\mathcal{I}_1 := \langle \ell_1 \rangle$.

Lower bound for Constant fanin depth-3 circuits

- $\mathsf{det_n}$ *cannot* be computed by a $\Sigma^{[k]}\Pi\Sigma$-circuit, *regardless* of the product fan-in.

  Proof sketch: Let $\sum_{i\in[k]} T_i = \mathsf{det}_n$, where $T_i \in \Pi\Sigma$, product of linear polynomials.

  - Pick ℓ_1 from T_1 and consider $\mathcal{I}_1 := \langle \ell_1 \rangle$.
  - LHS $\mod \mathcal{I}_1$, becomes $\sum_{i\in[k-1]} T_i'$ while $\mathsf{det}_n \mod \mathcal{I}_1 \neq 0$.

Lower bound for Constant fanin depth-3 circuits

- det_n *cannot* be computed by a $\Sigma^{[k]}\Pi\Sigma$-circuit, *regardless* of the product fan-in.

  Proof sketch: Let $\sum_{i\in[k]} T_i = \mathsf{det}_n$, where $T_i \in \Pi\Sigma$, product of linear polynomials.

  - Pick ℓ_1 from T_1 and consider $\mathcal{I}_1 := \langle \ell_1 \rangle$.
  - LHS $\mod \mathcal{I}_1$, becomes $\sum_{i\in[k-1]} T_i'$ while $\mathsf{det}_n \mod \mathcal{I}_1 \neq 0$.
  - Keep repeating and each time include an independent ℓ_i in the ideal $\mathcal{I}_i$.

Lower bound for Constant fanin depth-3 circuits

- $\mathsf{det_n}$ *cannot* be computed by a $\Sigma^{[k]}\Pi\Sigma$-circuit, *regardless* of the product fan-in.

  Proof sketch: Let $\sum_{i\in[k]} T_i = \mathsf{det}_n$, where $T_i \in \Pi\Sigma$, product of linear polynomials.

  - Pick ℓ_1 from T_1 and consider $\mathcal{I}_1 := \langle \ell_1 \rangle$.
  - LHS $\bmod \mathcal{I}_1$, becomes $\sum_{i\in[k-1]} T_i'$ while $\mathsf{det}_n \bmod \mathcal{I}_1 \neq 0$.
  - Keep repeating and each time include an independent ℓ_i in the ideal $\mathcal{I}_i$.
  - At the end, there is some $\mathcal{I}_k$ such that RHS has become 0 while LHS is *non-zero* (because $k << n$).

Lower bound for Constant fanin depth-3 circuits

- det_n *cannot* be computed by a $\Sigma^{[k]}\Pi\Sigma$-circuit, *regardless* of the product fan-in.

  Proof sketch: Let $\sum_{i\in[k]} T_i = \mathsf{det}_n$, where $T_i \in \Pi\Sigma$, product of linear polynomials.

  - Pick ℓ_1 from T_1 and consider $\mathcal{I}_1 := \langle \ell_1 \rangle$.
  - LHS $\mod \mathcal{I}_1$, becomes $\sum_{i\in[k-1]} T_i'$ while $\mathsf{det}_n \mod \mathcal{I}_1 \neq 0$.
  - Keep repeating and each time include an independent ℓ_i in the ideal $\mathcal{I}_i$.
  - At the end, there is some $\mathcal{I}_k$ such that RHS has become 0 while LHS is *non-zero* (because $k << n$).

- Therefore, $\Sigma^{[k]}\Pi\Sigma \subsetneq \mathsf{VBP}$!

Lower bound for Constant fanin depth-3 circuits

- $\mathsf{det_n}$ *cannot* be computed by a $\Sigma^{[k]}\Pi\Sigma$-circuit, *regardless* of the product fan-in.

  Proof sketch: Let $\sum_{i\in[k]} T_i = \mathsf{det}_n$, where $T_i \in \Pi\Sigma$, product of linear polynomials.

  - Pick ℓ_1 from T_1 and consider $\mathcal{I}_1 := \langle \ell_1 \rangle$.
  - LHS $\mod \mathcal{I}_1$, becomes $\sum_{i\in[k-1]} T_i'$ while $\mathsf{det}_n \mod \mathcal{I}_1 \neq 0$.
  - Keep repeating and each time include an independent ℓ_i in the ideal $\mathcal{I}_i$.
  - At the end, there is some $\mathcal{I}_k$ such that RHS has become 0 while LHS is *non-zero* (because $k << n$).

- Therefore, $\Sigma^{[k]}\Pi\Sigma \subsetneq \mathsf{VBP}$!

- How about $\overline{\Sigma^{[k]}\Pi\Sigma}$?

Power of border depth-3 circuits

Power of border depth-3 circuits

- Recall: $h \in \overline{\Sigma^{[k]}\Pi\Sigma}$ of size s if there exists a polynomial g such that

Power of border depth-3 circuits

- Recall: $h \in \overline{\Sigma^{[k]}\Pi\Sigma}$ of size s if there exists a polynomial g such that

$$g(\boldsymbol{x}, \epsilon) = h(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon) \ ,$$

Power of border depth-3 circuits

- Recall: $h \in \overline{\Sigma^{[k]}\Pi\Sigma}$ of size s if there exists a polynomial g such that

$$g(\boldsymbol{x}, \epsilon) = h(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon) \ ,$$

where g can be computed by a $\Sigma^{[k]}\Pi\Sigma$ circuit, over $\mathbb{F}(\epsilon)$, of size s.

- Recall: $h \in \overline{\Sigma^{[k]}\Pi\Sigma}$ of size s if there exists a polynomial g such that

$$g(\boldsymbol{x}, \epsilon) = h(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon) \ ,$$

where g can be computed by a $\Sigma^{[k]}\Pi\Sigma$ circuit, over $\mathbb{F}(\epsilon)$, of size s.

Border depth-3 fan-in 2 circuits are 'universal' [Kumar 2020]

Let P be *any* n-variate degree d polynomial. Then, $P \in \overline{\Sigma^{[2]}\Pi\Sigma}$, where the multiplication gate is $\mathsf{exp}(n, d)$.

Proof of Kumar's result

Proof.

1. Let $\mathsf{WR}(P) =: m$. Then, there are linear forms ℓ_i such that

Proof of Kumar's result

Proof.

1. Let $\mathsf{WR}(P) =: m$. Then, there are linear forms ℓ_i such that

$$P = \sum_{i=1}^{m} \ell_i^d \qquad [m \text{ can be as large as } \exp(n,d)] .$$

Proof of Kumar's result

Proof.

1. Let $\mathsf{WR}(P) =: m$. Then, there are linear forms ℓ_i such that

$$P = \sum_{i=1}^{m} \ell_i^d \qquad [m \text{ can be as large as } \exp(n, d)] \ .$$

2. Consider $A(\boldsymbol{x}) := \prod_{i=1}^{m} (1 + \ell_i^d) = \prod_{i=1}^{m} \prod_{j=1}^{d} (\alpha_j + \ell_i)$, for $\alpha_j \in \mathbb{C}$.

Proof of Kumar's result

Proof.

1. Let $\mathsf{WR}(P) =: m$. Then, there are linear forms ℓ_i such that

$$P = \sum_{i=1}^{m} \ell_i^d \qquad [m \text{ can be as large as } \exp(n,d)] .$$

2. Consider $A(\boldsymbol{x}) := \prod_{i=1}^{m}(1+\ell_i^d) = \prod_{i=1}^{m}\prod_{j=1}^{d}(\alpha_j+\ell_i)$, for $\alpha_j \in \mathbb{C}$. Note that

$$A(\boldsymbol{x}) = 1 + P + B \text{ where } \deg(B) \geq 2d .$$

Proof of Kumar's result

Proof.

1. Let $\mathsf{WR}(P) =: m$. Then, there are linear forms ℓ_i such that

$$P = \sum_{i=1}^{m} \ell_i^d \qquad [m \text{ can be as large as } \exp(n,d)] \ .$$

2. Consider $A(\boldsymbol{x}) := \prod_{i=1}^{m}(1+\ell_i^d) = \prod_{i=1}^{m}\prod_{j=1}^{d}(\alpha_j + \ell_i)$, for $\alpha_j \in \mathbb{C}$. Note that

$$A(\boldsymbol{x}) = 1 + P + B \text{ where } \deg(B) \geq 2d \ .$$

3. Replace x_i by $\epsilon \cdot x_i$ to get that

Proof.

1. Let $\mathsf{WR}(P) =: m$. Then, there are linear forms ℓ_i such that

$$P = \sum_{i=1}^{m} \ell_i^d \qquad [m \text{ can be as large as } \exp(n,d)] \ .$$

2. Consider $A(\boldsymbol{x}) := \prod_{i=1}^{m}(1+\ell_i^d) = \prod_{i=1}^{m}\prod_{j=1}^{d}(\alpha_j+\ell_i)$, for $\alpha_j \in \mathbb{C}$. Note that

$$A(\boldsymbol{x}) = 1 + P + B \text{ where } \deg(B) \geq 2d \ .$$

3. Replace x_i by $\epsilon \cdot x_i$ to get that

$$\prod_{i=1}^{m}\prod_{j=1}^{d}(\alpha_j + \epsilon \cdot \ell_i) = 1 + \epsilon^d \cdot P + \epsilon^{2d} \cdot R(\boldsymbol{x}, \epsilon) \ .$$

Proof of Kumar's result

Proof.

1. Let $\mathsf{WR}(P) =: m$. Then, there are linear forms ℓ_i such that

$$P = \sum_{i=1}^{m} \ell_i^d \qquad [m \text{ can be as large as } \exp(n,d)] .$$

2. Consider $A(\boldsymbol{x}) := \prod_{i=1}^{m}(1+\ell_i^d) = \prod_{i=1}^{m}\prod_{j=1}^{d}(\alpha_j+\ell_i)$, for $\alpha_j \in \mathbb{C}$. Note that

$$A(\boldsymbol{x}) = 1 + P + B \text{ where } \deg(B) \geq 2d .$$

3. Replace x_i by $\epsilon \cdot x_i$ to get that

$$\prod_{i=1}^{m}\prod_{j=1}^{d}(\alpha_j + \epsilon\cdot\ell_i) = 1 + \epsilon^d \cdot P + \epsilon^{2d}\cdot R(\boldsymbol{x},\epsilon) .$$

4. Divide by ϵ^d and rearrange to get

$$P + \epsilon^d \cdot R(\boldsymbol{x},\epsilon) = -\epsilon^{-d} + \epsilon^{-d}\cdot\prod_{i=1}^{m}\prod_{j=1}^{d}(\alpha_j+\epsilon\cdot\ell_i) \in \Sigma^{[2]}\Pi^{[md]}\Sigma .$$

□

Proving Upper Bounds

De-bordering $\overline{\Sigma^{[2]}\Pi\Sigma}$ circuits

De-bordering $\overline{\Sigma^{[2]}\Pi\Sigma}$ circuits

- If h is approximated by a $\Sigma^{[2]}\Pi\Sigma$ circuit with product fanin, bounded by $\mathsf{poly}(n)$, what's the *exact* complexity of h?

De-bordering $\overline{\Sigma^{[2]}\Pi\Sigma}$ circuits

- If h is approximated by a $\Sigma^{[2]}\Pi\Sigma$ circuit with product fanin, bounded by $\mathsf{poly}(n)$, what's the *exact* complexity of h?

Border of polynomial-sized depth-3 top-fanin-2 circuits are 'easy' [Dutta-Dwivedi-Saxena FOCS'21].

$\overline{\Sigma^{[2]}\Pi\Sigma} \subseteq \mathsf{VBP}$, for polynomial-sized $\overline{\Sigma^{[2]}\Pi\Sigma}$-circuits.

De-bordering $\overline{\Sigma^{[2]}\Pi\Sigma}$ circuits

- If h is approximated by a $\Sigma^{[2]}\Pi\Sigma$ circuit with product fanin, bounded by $\mathsf{poly}(n)$, what's the *exact* complexity of h?

Border of polynomial-sized depth-3 top-fanin-2 circuits are 'easy' [Dutta-Dwivedi-Saxena FOCS'21].

$\overline{\Sigma^{[2]}\Pi\Sigma} \subseteq \mathsf{VBP}$, for polynomial-sized $\overline{\Sigma^{[2]}\Pi\Sigma}$-circuits.

Remark. The result holds if one replaces the top-fanin-2 by arbitrary constant k.

Proof sketch for $k = 2$

Proof sketch for $k = 2$

Grand Idea: Reduce to $k = 1$!

Proof sketch for $k = 2$

Grand Idea: Reduce to $k = 1$!

- $T_1 + T_2 = f(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon)$, where $T_i \in \Pi\Sigma \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$. Assume $\deg(f) = d$.

Proof sketch for $k = 2$

Grand Idea: Reduce to $k = 1$!

- $T_1 + T_2 = f(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon)$, where $T_i \in \Pi\Sigma \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$. Assume $\deg(f) = d$.
- Apply a map Φ, defined by $\Phi : x_i \mapsto z \cdot x_i + \alpha_i$, where $\alpha_i \in \mathbb{F}$ are *random*.

Proof sketch for $k = 2$

Grand Idea: Reduce to $k = 1$!

- $T_1 + T_2 = f(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon)$, where $T_i \in \Pi\Sigma \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$. Assume $\deg(f) = d$.
- Apply a map Φ, defined by $\Phi : x_i \mapsto z \cdot x_i + \alpha_i$, where $\alpha_i \in \mathbb{F}$ are *random*.
  - The variable z is the "degree counter",

Proof sketch for $k = 2$

Grand Idea: Reduce to $k = 1$!

- ❑ $T_1 + T_2 = f(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon)$, where $T_i \in \Pi\Sigma \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$. Assume $\deg(f) = d$.
- ❑ Apply a map Φ, defined by $\Phi : x_i \mapsto z \cdot x_i + \alpha_i$, where $\alpha_i \in \mathbb{F}$ are *random*.
  - ➢ The variable z is the "degree counter",
  - ➢ α_i to make sure: If $\ell \mid T_i$, then $\Phi(\ell)|_{z=0} = \ell(\alpha_1, \ldots, \alpha_n) \in \mathbb{F}(\epsilon)^*$.

Proof sketch for $k = 2$

Grand Idea: Reduce to $k = 1$!

- ❑ $T_1 + T_2 = f(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon)$, where $T_i \in \Pi\Sigma \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$. Assume $\deg(f) = d$.
- ❑ Apply a map Φ, defined by $\Phi : x_i \mapsto z \cdot x_i + \alpha_i$, where $\alpha_i \in \mathbb{F}$ are *random*.
  - ➢ The variable z is the "degree counter",
  - ➢ α_i to make sure: If $\ell \mid T_i$, then $\Phi(\ell)|_{z=0} = \ell(\alpha_1, \ldots, \alpha_n) \in \mathbb{F}(\epsilon)^*$.
- ❑ It suffices to find $\Phi(f) \mod z^{d+1} = \Phi(f)$.

Proof sketch for $k = 2$

Grand Idea: Reduce to $k = 1$!

- $T_1 + T_2 = f(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon)$, where $T_i \in \Pi\Sigma \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$. Assume $\deg(f) = d$.
- Apply a map Φ, defined by $\Phi : x_i \mapsto z \cdot x_i + \alpha_i$, where $\alpha_i \in \mathbb{F}$ are *random*.
  - The variable z is the "degree counter",
  - α_i to make sure: If $\ell \mid T_i$, then $\Phi(\ell)|_{z=0} = \ell(\alpha_1, \ldots, \alpha_n) \in \mathbb{F}(\epsilon)^*$.
- It suffices to find $\Phi(f) \mod z^{d+1} = \Phi(f)$. [Truncation by degree.]

Proof sketch for $k = 2$

Grand Idea: Reduce to $k = 1$!

- $T_1 + T_2 = f(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon)$, where $T_i \in \Pi\Sigma \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$. Assume $\deg(f) = d$.
- Apply a map Φ, defined by $\Phi : x_i \mapsto z \cdot x_i + \alpha_i$, where $\alpha_i \in \mathbb{F}$ are *random*.
  - The variable z is the "degree counter",
  - α_i to make sure: If $\ell \mid T_i$, then $\Phi(\ell)|_{z=0} = \ell(\alpha_1, \ldots, \alpha_n) \in \mathbb{F}(\epsilon)^*$.
- It suffices to find $\Phi(f) \mod z^{d+1} = \Phi(f)$. [Truncation by degree.]
- We devise a technique called DiDIL - **Di**vide, **D**erive, **I**nterpolate with **L**imit.

$k = 2$ proof continued: Divide and Derive

$k = 2$ proof continued: Divide and Derive

- $\mathrm{val}_\epsilon(\cdot)$ denotes the highest power of ϵ dividing it.
  E.g., $h = \epsilon^{-2}x_1 + \epsilon^{-1}x_2 + \epsilon x_3$. Then, $\mathrm{val}_\epsilon(h) = -2$.

$k = 2$ proof continued: Divide and Derive

- $\mathsf{val}_\epsilon(\cdot)$ denotes the highest power of ϵ dividing it.
  E.g., $h = \epsilon^{-2}x_1 + \epsilon^{-1}x_2 + \epsilon x_3$. Then, $\mathsf{val}_\epsilon(h) = -2$.

- Let $\Phi(T_i) =: \epsilon^{a_i} \cdot \tilde{T}_i$, for $i \in [2]$, where $a_i := \mathsf{val}_\epsilon(\Phi(T_i))$.

$k = 2$ proof continued: Divide and Derive

- $\mathsf{val}_\epsilon(\cdot)$ denotes the highest power of ϵ dividing it.
  E.g., $h = \epsilon^{-2}x_1 + \epsilon^{-1}x_2 + \epsilon x_3$. Then, $\mathsf{val}_\epsilon(h) = -2$.

- Let $\Phi(T_i) =: \epsilon^{a_i} \cdot \tilde{T}_i$, for $i \in [2]$, where $a_i := \mathsf{val}_\epsilon(\Phi(T_i))$.
  Then, (i) $\tilde{T}_i \in \mathbb{F}[\epsilon, \boldsymbol{x}, z]$, and (ii) $\lim_{\epsilon \to 0} \tilde{T}_2 = t_2 \in \mathbb{F}[\boldsymbol{x}, z] \setminus \{0\}$, exists.

$k = 2$ proof continued: Divide and Derive

- $\mathsf{val}_\epsilon(\cdot)$ denotes the highest power of ϵ dividing it.
  E.g., $h = \epsilon^{-2}x_1 + \epsilon^{-1}x_2 + \epsilon x_3$. Then, $\mathsf{val}_\epsilon(h) = -2$.

- Let $\Phi(T_i) =: \epsilon^{a_i} \cdot \tilde{T}_i$, for $i \in [2]$, where $a_i := \mathsf{val}_\epsilon(\Phi(T_i))$.
  Then, (i) $\tilde{T}_i \in \mathbb{F}[\epsilon, \boldsymbol{x}, z]$, and (ii) $\lim_{\epsilon \to 0} \tilde{T}_2 = t_2 \in \mathbb{F}[\boldsymbol{x}, z] \setminus \{0\}$, exists.

- Divide both side by $\tilde{T}_2$ and take partial derivative with respect to z, to get:

$k = 2$ proof continued: Divide and Derive

- $\text{val}_\epsilon(\cdot)$ denotes the highest power of ϵ dividing it.
  E.g., $h = \epsilon^{-2}x_1 + \epsilon^{-1}x_2 + \epsilon x_3$. Then, $\text{val}_\epsilon(h) = -2$.
- Let $\Phi(T_i) =: \epsilon^{a_i} \cdot \tilde{T}_i$, for $i \in [2]$, where $a_i := \text{val}_\epsilon(\Phi(T_i))$.
  Then, (i) $\tilde{T}_i \in \mathbb{F}[\epsilon, \boldsymbol{x}, z]$, and (ii) $\lim_{\epsilon \to 0} \tilde{T}_2 = t_2 \in \mathbb{F}[\boldsymbol{x}, z] \setminus \{0\}$, exists.
- Divide both side by $\tilde{T}_2$ and take partial derivative with respect to z, to get:

$$\begin{aligned} & \Phi(f) + \epsilon \cdot \Phi(S) = \Phi(T_1) + \Phi(T_2) \\ \implies & \Phi(f)/\tilde{T}_2 + \epsilon \cdot \Phi(S)/\tilde{T}_2 = \Phi(T_1)/\tilde{T}_2 + \epsilon^{a_2} \\ \implies & \partial_z\left(\Phi(f)/\tilde{T}_2\right) + \epsilon \cdot \partial_z\left(\Phi(S)/\tilde{T}_2\right) = \partial_z\left(\Phi(T_1)/\tilde{T}_2\right) =: g_1 \,. \end{aligned} \tag{1}$$

$k = 2$ proof continued: Divide and Derive

- $\text{val}_\epsilon(\cdot)$ denotes the highest power of ϵ dividing it.
  E.g., $h = \epsilon^{-2}x_1 + \epsilon^{-1}x_2 + \epsilon x_3$. Then, $\text{val}_\epsilon(h) = -2$.
- Let $\Phi(T_i) =: \epsilon^{a_i} \cdot \tilde{T}_i$, for $i \in [2]$, where $a_i := \text{val}_\epsilon(\Phi(T_i))$.
  Then, (i) $\tilde{T}_i \in \mathbb{F}[\epsilon, \boldsymbol{x}, z]$, and (ii) $\lim_{\epsilon \to 0} \tilde{T}_2 = t_2 \in \mathbb{F}[\boldsymbol{x}, z] \setminus \{0\}$, exists.
- Divide both side by $\tilde{T}_2$ and take partial derivative with respect to z, to get:

$$\begin{aligned} \Phi(f) + \epsilon \cdot \Phi(S) &= \Phi(T_1) + \Phi(T_2) \\ \implies \Phi(f)/\tilde{T}_2 + \epsilon \cdot \Phi(S)/\tilde{T}_2 &= \Phi(T_1)/\tilde{T}_2 + \epsilon^{a_2} \\ \implies \partial_z\left(\Phi(f)/\tilde{T}_2\right) + \epsilon \cdot \partial_z\left(\Phi(S)/\tilde{T}_2\right) &= \partial_z\left(\Phi(T_1)/\tilde{T}_2\right) =: g_1 \,. \end{aligned} \tag{1}$$

- $\lim_{\epsilon \to 0} g_1 = \lim_{\epsilon \to 0} \partial_z\left(\Phi(T_1)/\tilde{T}_2\right) = \partial_z(\Phi(f)/t_2)$.

$k = 2$ proof continued

$k = 2$ proof continued

- First target: compute $\lim_{\epsilon \to 0} g_1 = \partial_z(\Phi(f)/t_2)$.

$k = 2$ proof continued

- First target: compute $\lim_{\epsilon \to 0} g_1 = \partial_z(\Phi(f)/t_2)$.
- Logarithmic derivative: $\mathrm{dlog}_z(h) := \partial_z(h)/h$.

$k = 2$ proof continued

- First target: compute $\lim_{\epsilon \to 0} g_1 = \partial_z(\Phi(f)/t_2)$.
- Logarithmic derivative: $\mathrm{dlog}_z(h) := \partial_z(h)/h$.
- dlog *linearizes* product: $\mathrm{dlog}(h_1 h_2) = \mathrm{dlog}(h_1) + \mathrm{dlog}(h_2)$.

$k = 2$ proof continued

- First target: compute $\lim_{\epsilon \to 0} g_1 = \partial_z(\Phi(f)/t_2)$.
- Logarithmic derivative: $\mathrm{dlog}_z(h) := \partial_z(h)/h$.
- dlog *linearizes* product: $\mathrm{dlog}(h_1 h_2) = \mathrm{dlog}(h_1) + \mathrm{dlog}(h_2)$. Note:

$$\begin{aligned}\partial_z\left(\Phi(T_1)/\tilde{T}_2\right) &= \Phi(T_1)/\tilde{T}_2 \cdot \mathrm{dlog}\left(\Phi(T_1)/\tilde{T}_2\right) \\ &= \Phi(T_1)/\tilde{T}_2 \cdot \left(\mathrm{dlog}(\Phi(T_1)) - \mathrm{dlog}(\tilde{T}_2)\right) \;.\end{aligned}$$

$k = 2$ proof continued

- First target: compute $\lim_{\epsilon \to 0} g_1 = \partial_z(\Phi(f)/t_2)$.
- Logarithmic derivative: $\mathrm{dlog}_z(h) := \partial_z(h)/h$.
- dlog *linearizes* product: $\mathrm{dlog}(h_1 h_2) = \mathrm{dlog}(h_1) + \mathrm{dlog}(h_2)$. Note:

$$\begin{aligned}\partial_z\left(\Phi(T_1)/\tilde{T}_2\right) &= \Phi(T_1)/\tilde{T}_2 \cdot \mathrm{dlog}\left(\Phi(T_1)/\tilde{T}_2\right) \\ &= \Phi(T_1)/\tilde{T}_2 \cdot \left(\mathrm{dlog}(\Phi(T_1)) - \mathrm{dlog}(\tilde{T}_2)\right) .\end{aligned}$$

- Both $\Phi(T_1)$ and $\tilde{T}_2$ have $\Pi\Sigma$ circuits (they have z and ϵ).

$k = 2$ proof continued

$k = 2$ proof continued

$$\begin{aligned} g_1 &= \partial_z \left(\Phi(T_1)/\tilde{T}_2 \right) = \Phi(T_1)/\tilde{T}_2 \cdot \left(\mathrm{dlog}(\Phi(T_1)) - \mathrm{dlog}(\tilde{T}_2) \right) \\ &= \Pi\Sigma/\Pi\Sigma \cdot (\mathrm{dlog}(\Pi\Sigma) - \mathrm{dlog}(\Pi\Sigma)) \\ &= \Pi\Sigma/\Pi\Sigma \cdot \left(\sum \mathrm{dlog}(\Sigma) \right). \end{aligned}$$

$k = 2$ proof continued

$$
\begin{aligned}
g_1 &= \partial_z \left(\Phi(T_1)/\tilde{T}_2\right) = \Phi(T_1)/\tilde{T}_2 \cdot \left(\mathrm{dlog}(\Phi(T_1)) - \mathrm{dlog}(\tilde{T}_2)\right) \\
&= \Pi\Sigma/\Pi\Sigma \cdot (\mathrm{dlog}(\Pi\Sigma) - \mathrm{dlog}(\Pi\Sigma)) \\
&= \Pi\Sigma/\Pi\Sigma \cdot \left(\sum \mathrm{dlog}(\Sigma)\right).
\end{aligned}
$$

- Here Σ means just a linear polynomial ℓ (in $z, \mathbf{x}$ and *unit* wrt the former).

$k = 2$ proof continued

$$
\begin{aligned}
g_1 = \partial_z\left(\Phi(T_1)/\tilde{T}_2\right) &= \Phi(T_1)/\tilde{T}_2 \cdot \left(\mathrm{dlog}(\Phi(T_1)) - \mathrm{dlog}(\tilde{T}_2)\right) \\
&= \Pi\Sigma/\Pi\Sigma \cdot (\mathrm{dlog}(\Pi\Sigma) - \mathrm{dlog}(\Pi\Sigma)) \\
&= \Pi\Sigma/\Pi\Sigma \cdot \left(\sum \mathrm{dlog}(\Sigma)\right).
\end{aligned}
$$

- Here Σ means just a linear polynomial ℓ (in $z, \mathbf{x}$ and *unit* wrt the former).
- Recall: $\lim_{\epsilon \to 0} g_1 = \partial_z(\Phi(f)/t_2)$.

$k = 2$ proof continued

$$g_1 = \partial_z \left(\Phi(T_1)/\tilde{T}_2\right) = \Phi(T_1)/\tilde{T}_2 \cdot \left(\mathrm{dlog}(\Phi(T_1)) - \mathrm{dlog}(\tilde{T}_2)\right)$$
$$= \Pi\Sigma/\Pi\Sigma \cdot (\mathrm{dlog}(\Pi\Sigma) - \mathrm{dlog}(\Pi\Sigma))$$
$$= \Pi\Sigma/\Pi\Sigma \cdot \left(\sum \mathrm{dlog}(\Sigma)\right).$$

- Here Σ means just a linear polynomial ℓ (in $z, \mathbf{x}$ and *unit* wrt the former).
- Recall: $\lim_{\epsilon \to 0} g_1 = \partial_z(\Phi(f)/t_2)$.
- $\deg(f) = d \implies \deg_z(\Phi(f)) = d \implies \deg_z(\partial_z(\Phi(f))) = d - 1$.

$k = 2$ proof continued

$$\begin{aligned} g_1 &= \partial_z \left(\Phi(T_1)/\tilde{T}_2\right) = \Phi(T_1)/\tilde{T}_2 \cdot \left(\mathrm{dlog}(\Phi(T_1)) - \mathrm{dlog}(\tilde{T}_2)\right) \\ &= \Pi\Sigma/\Pi\Sigma \cdot (\mathrm{dlog}(\Pi\Sigma) - \mathrm{dlog}(\Pi\Sigma)) \\ &= \Pi\Sigma/\Pi\Sigma \cdot \left(\sum \mathrm{dlog}(\Sigma)\right). \end{aligned}$$

- Here Σ means just a linear polynomial ℓ (in $z, \mathbf{x}$ and *unit* wrt the former).
- Recall: $\lim_{\epsilon \to 0} g_1 = \partial_z(\Phi(f)/t_2)$.
- $\deg(f) = d \implies \deg_z(\Phi(f)) = d \implies \deg_z(\partial_z(\Phi(f))) = d - 1$.
- Suffices to compute $g_1 \mod z^d$ and take the limit!

$k = 2$ **proof:** dlog **strikes!**

- What is $\mathsf{dlog}(\ell)$?

$k = 2$ proof: dlog strikes!

- What is $\mathsf{dlog}(\ell)$? Note, $\ell = A - z \cdot B$, where $A \in \mathbb{F}(\epsilon)$, $B \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$.

$k = 2$ proof: dlog strikes!

- What is $\mathrm{dlog}(\ell)$? Note, $\ell = A - z \cdot B$, where $A \in \mathbb{F}(\epsilon)$, $B \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$.

$$\begin{aligned} \mathrm{dlog}(A - zB) &= -\frac{B}{A\,(1 - z \cdot B/A)} \\ &= -\frac{B}{A} \cdot \sum_{j=0}^{d-1} \left(\frac{z \cdot B}{A}\right)^j \\ &\in \Sigma \wedge \Sigma\,. \quad [\textit{Magic trick}] \end{aligned}$$

$k = 2$ proof: dlog strikes!

- What is $\mathrm{dlog}(\ell)$? Note, $\ell = A - z \cdot B$, where $A \in \mathbb{F}(\epsilon)$, $B \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$.

$$\begin{aligned}
\mathrm{dlog}(A - zB) &= -\frac{B}{A\,(1 - z \cdot B/A)} \\
&= -\frac{B}{A} \cdot \sum_{j=0}^{d-1} \left(\frac{z \cdot B}{A}\right)^j \\
&\in \Sigma \wedge \Sigma\,. \quad [\textit{Magic trick}]
\end{aligned}$$

- Thus,

$k = 2$ proof: dlog strikes!

- What is $\mathrm{dlog}(\ell)$? Note, $\ell = A - z \cdot B$, where $A \in \mathbb{F}(\epsilon)$, $B \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$.

$$\begin{aligned}\mathrm{dlog}(A - zB) &= -\frac{B}{A\,(1 - z \cdot B/A)} \\ &= -\frac{B}{A} \cdot \sum_{j=0}^{d-1} \left(\frac{z \cdot B}{A}\right)^j \\ &\in \Sigma \wedge \Sigma\,. \quad [\textit{Magic trick}]\end{aligned}$$

- Thus,

$$\begin{aligned}\lim_{\epsilon \to 0} g_1 \mod z^d &\equiv \lim_{\epsilon \to 0} \Pi\Sigma/\Pi\Sigma \cdot \left(\sum \mathrm{dlog}(\Sigma)\right) \mod z^d \\ &\equiv \lim_{\epsilon \to 0} (\Pi\Sigma/\Pi\Sigma) \cdot (\Sigma \wedge \Sigma) \mod z^d \\ &\in \overline{(\Pi\Sigma/\Pi\Sigma) \cdot (\Sigma \wedge \Sigma)} \mod z^d\,.\end{aligned}$$

Finishing the proof

Finishing the proof

- $\overline{\mathcal{C} \cdot \mathcal{D}} \subseteq \overline{\mathcal{C}} \cdot \overline{\mathcal{D}}$. Therefore,

Finishing the proof

- $\overline{\mathcal{C} \cdot \mathcal{D}} \subseteq \overline{\mathcal{C}} \cdot \overline{\mathcal{D}}$. Therefore,

$$\begin{aligned} \overline{(\Pi\Sigma/\Pi\Sigma) \cdot (\Sigma \wedge \Sigma)} &\subseteq \overline{(\Pi\Sigma/\Pi\Sigma)} \cdot \overline{\Sigma \wedge \Sigma} \\ &\subseteq (\mathsf{ABP}/\mathsf{ABP}) \cdot \mathsf{ABP} \\ &= \mathsf{ABP}/\mathsf{ABP}\,. \end{aligned}$$

Finishing the proof

- $\overline{\mathcal{C} \cdot \mathcal{D}} \subseteq \overline{\mathcal{C}} \cdot \overline{\mathcal{D}}$. Therefore,

$$\begin{aligned}\overline{(\Pi\Sigma/\Pi\Sigma) \cdot (\Sigma \wedge \Sigma)} &\subseteq \overline{(\Pi\Sigma/\Pi\Sigma)} \cdot \overline{\Sigma \wedge \Sigma} \\ &\subseteq (\mathsf{ABP}/\mathsf{ABP}) \cdot \mathsf{ABP} \\ &= \mathsf{ABP}/\mathsf{ABP} .\end{aligned}$$

- Eliminate division and integrate (interpolate) to get $\Phi(f)/t_2 = \mathsf{ABP} \implies \Phi(f) = \mathsf{ABP} \implies f = \mathsf{ABP}$.

Proving Lower Bounds

Lifting classical lower bound in the border

Lifting classical lower bound in the border

- Can we separate $\overline{\Sigma^{[k]}\Pi\Sigma}$ and VBP?

Lifting classical lower bound in the border

- Can we separate $\overline{\Sigma^{[k]}\Pi\Sigma}$ and VBP?
  - [Limaye-Srinivasan-Tavenas, FOCS 2021] showed that $\mathsf{IMM}_{n,d}$ with $d = o(\log n)$ requires $n^{\omega(1)}$-size depth-3 circuits.

Lifting classical lower bound in the border

- Can we separate $\overline{\Sigma^{[k]}\Pi\Sigma}$ and VBP?
  - [Limaye-Srinivasan-Tavenas, FOCS 2021] showed that $\mathsf{IMM}_{n,d}$ with $d = o(\log n)$ requires $n^{\omega(1)}$-size depth-3 circuits.
  - Rank-based lower bounds *can* be lifted in the border!

Lifting classical lower bound in the border

- Can we separate $\overline{\Sigma^{[k]}\Pi\Sigma}$ and VBP?
  - [Limaye-Srinivasan-Tavenas, FOCS 2021] showed that $\mathsf{IMM}_{n,d}$ with $d = o(\log n)$ requires $n^{\omega(1)}$-size depth-3 circuits.
  - Rank-based lower bounds *can* be lifted in the border!
  - Since, $\mathsf{IMM}_{n,d} \in \mathsf{VBP}$, $\overline{\Sigma^{[k]}\Pi\Sigma} \neq \mathsf{VBP}$.

Looking for finer lower bounds

Looking for finer lower bounds

- Can we show an *exponential* gap between $\overline{\Sigma^{[k]}\Pi\Sigma}$ and VBP?

Looking for finer lower bounds

- Can we show an *exponential* gap between $\overline{\Sigma^{[k]}\Pi\Sigma}$ and VBP?
- Ambitious goal: Can we separate $\overline{\Sigma^{[k]}\Pi\Sigma}$ and $\overline{\Sigma^{[k+1]}\Pi\Sigma}$?

Looking for finer lower bounds

- Can we show an *exponential* gap between $\overline{\Sigma^{[k]}\Pi\Sigma}$ and VBP?
- Ambitious goal: Can we separate $\overline{\Sigma^{[k]}\Pi\Sigma}$ and $\overline{\Sigma^{[k+1]}\Pi\Sigma}$?
- Note: This is already known (impossibility) in the classical setting!

Looking for finer lower bounds

- Can we show an *exponential* gap between $\overline{\Sigma^{[k]}\Pi\Sigma}$ and VBP?
- Ambitious goal: Can we separate $\overline{\Sigma^{[k]}\Pi\Sigma}$ and $\overline{\Sigma^{[k+1]}\Pi\Sigma}$?
- Note: This is already known (impossibility) in the classical setting!
- $x_1 \cdot y_1 + \ldots + x_{k+1} \cdot y_{k+1}$ cannot be computed by $\Sigma^{[k]}\Pi\Sigma$ circuits!

Looking for finer lower bounds

- Can we show an *exponential* gap between $\overline{\Sigma^{[k]}\Pi\Sigma}$ and VBP?
- Ambitious goal: Can we separate $\overline{\Sigma^{[k]}\Pi\Sigma}$ and $\overline{\Sigma^{[k+1]}\Pi\Sigma}$?
- Note: This is already known (impossibility) in the classical setting!
- $x_1 \cdot y_1 + \ldots + x_{k+1} \cdot y_{k+1}$ cannot be computed by $\Sigma^{[k]}\Pi\Sigma$ circuits!
- Catch: $x_1 \cdot y_1 + \ldots + x_{k+1} \cdot y_{k+1}$ does not work anymore in *border*, since, $x_1 \cdot y_1 + \ldots + x_{k+1} \cdot y_{k+1} \in \overline{\Sigma^{[2]}\Pi^{O(k)}\Sigma}$!

Looking for finer lower bounds

- Can we show an *exponential* gap between $\overline{\Sigma^{[k]}\Pi\Sigma}$ and VBP?
- Ambitious goal: Can we separate $\overline{\Sigma^{[k]}\Pi\Sigma}$ and $\overline{\Sigma^{[k+1]}\Pi\Sigma}$?
- Note: This is already known (impossibility) in the classical setting!
- $x_1 \cdot y_1 + \ldots + x_{k+1} \cdot y_{k+1}$ cannot be computed by $\Sigma^{[k]}\Pi\Sigma$ circuits!
- Catch: $x_1 \cdot y_1 + \ldots + x_{k+1} \cdot y_{k+1}$ does not work anymore in *border*, since, $x_1 \cdot y_1 + \ldots + x_{k+1} \cdot y_{k+1} \in \overline{\Sigma^{[2]}\Pi^{O(k)}\Sigma}$!
- What does work (if at all!)?

Our results

Our results

[Dutta-Saxena FOCS'22]

Fix any constant $k \geq 1$. There is an explicit n-variate and $< n$ degree polynomial f such that f can be computed by a $\overline{\Sigma^{[k+1]}\Pi\Sigma}$ circuit of size $O(n)$; but, f requires $2^{\Omega(n)}$-size $\overline{\Sigma^{[k]}\Pi\Sigma}$ circuits.

Our results

[Dutta-Saxena FOCS'22]

Fix any constant $k \geq 1$. There is an explicit n-variate and $< n$ degree polynomial f such that f can be computed by a $\overline{\Sigma^{[k+1]}\Pi\Sigma}$ circuit of size $O(n)$; but, f requires $2^{\Omega(n)}$-size $\overline{\Sigma^{[k]}\Pi\Sigma}$ circuits.

- Fix $k = 2$. Define the polynomial $P_d := x_1 \cdots x_d + y_1 \cdots y_d + z_1 \cdots z_d$, a degree-$d$ polynomial on $n = 3d$-variables.

Our results

[Dutta-Saxena FOCS'22]

Fix any constant $k \geq 1$. There is an explicit n-variate and $< n$ degree polynomial f such that f can be computed by a $\overline{\Sigma^{[k+1]}\Pi\Sigma}$ circuit of size $O(n)$; but, f requires $2^{\Omega(n)}$-size $\overline{\Sigma^{[k]}\Pi\Sigma}$ circuits.

- Fix $k = 2$. Define the polynomial $P_d := x_1 \cdots x_d + y_1 \cdots y_d + z_1 \cdots z_d$, a degree-$d$ polynomial on $n = 3d$-variables.
- P_d has trivial fanin-3 depth-3 circuit (and hence in border too!).

Our results

[Dutta-Saxena FOCS'22]

Fix any constant $k \geq 1$. There is an explicit n-variate and $< n$ degree polynomial f such that f can be computed by a $\overline{\Sigma^{[k+1]}\Pi\Sigma}$ circuit of size $O(n)$; but, f requires $2^{\Omega(n)}$-size $\overline{\Sigma^{[k]}\Pi\Sigma}$ circuits.

- Fix $k = 2$. Define the polynomial $P_d := x_1 \cdots x_d + y_1 \cdots y_d + z_1 \cdots z_d$, a degree-$d$ polynomial on $n = 3d$-variables.
- P_d has trivial fanin-3 depth-3 circuit (and hence in border too!).
- We will show that P_d requires $2^{\Omega(d)}$-size $\overline{\Sigma^{[2]}\Pi\Sigma}$ circuits.

Our results

[Dutta-Saxena FOCS'22]

Fix any constant $k \geq 1$. There is an explicit n-variate and $< n$ degree polynomial f such that f can be computed by a $\overline{\Sigma^{[k+1]}\Pi\Sigma}$ circuit of size $O(n)$; but, f requires $2^{\Omega(n)}$-size $\overline{\Sigma^{[k]}\Pi\Sigma}$ circuits.

- Fix $k = 2$. Define the polynomial $P_d := x_1 \cdots x_d + y_1 \cdots y_d + z_1 \cdots z_d$, a degree-$d$ polynomial on $n = 3d$-variables.
- P_d has trivial fanin-3 depth-3 circuit (and hence in border too!).
- We will show that P_d requires $2^{\Omega(d)}$-size $\overline{\Sigma^{[2]}\Pi\Sigma}$ circuits.
- Kumar's proof establishes that P_d has a $2^{O(d)}$-size $\overline{\Sigma^{[2]}\Pi\Sigma}$ circuits, showing *optimality*!

Our results

[Dutta-Saxena FOCS'22]

Fix any constant $k \geq 1$. There is an explicit n-variate and $< n$ degree polynomial f such that f can be computed by a $\overline{\Sigma^{[k+1]}\Pi\Sigma}$ circuit of size $O(n)$; but, f requires $2^{\Omega(n)}$-size $\overline{\Sigma^{[k]}\Pi\Sigma}$ circuits.

- Fix $k = 2$. Define the polynomial $P_d := x_1 \cdots x_d + y_1 \cdots y_d + z_1 \cdots z_d$, a degree-$d$ polynomial on $n = 3d$-variables.
- P_d has trivial fanin-3 depth-3 circuit (and hence in border too!).
- We will show that P_d requires $2^{\Omega(d)}$-size $\overline{\Sigma^{[2]}\Pi\Sigma}$ circuits.
- Kumar's proof establishes that P_d has a $2^{O(d)}$-size $\overline{\Sigma^{[2]}\Pi\Sigma}$ circuits, showing *optimality*!
- Classical is about *impossibility* while in border, it is about *optimality*.

Non-homogeneity is ‘bad’

Non-homogeneity is ‘bad’

- Recall the non-border lower bound proof, of making an ideal $\mathcal{I}_k = \langle \ell_1, \ldots, \ell_k \rangle$, such that $\mathsf{det}_n \neq 0 \bmod \mathcal{I}_k$, but $\Sigma^{[k]}\Pi\Sigma = 0 \bmod \mathcal{I}_k$.

Non-homogeneity is 'bad'

- Recall the non-border lower bound proof, of making an ideal $\mathcal{I}_k = \langle \ell_1, \ldots, \ell_k \rangle$, such that $\mathsf{det}_n \neq 0 \bmod \mathcal{I}_k$, but $\Sigma^{[k]}\Pi\Sigma = 0 \bmod \mathcal{I}_k$.
- Let $\ell_1 := 1 + \epsilon x_1$. What does taking $\bmod \ell_1$ in the 'border' ($\epsilon \to 0$) mean? Essentially we are eventually setting $x_1 = -1/\epsilon$ (and then $\epsilon \to 0$)!

Non-homogeneity is ‘bad’

- Recall the non-border lower bound proof, of making an ideal $\mathcal{I}_k = \langle \ell_1, \ldots, \ell_k \rangle$, such that $\mathsf{det}_n \neq 0 \bmod \mathcal{I}_k$, but $\Sigma^{[k]}\Pi\Sigma = 0 \bmod \mathcal{I}_k$.
- Let $\ell_1 := 1 + \epsilon x_1$. What does taking $\bmod\ \ell_1$ in the ‘border’ ($\epsilon \to 0$) mean? Essentially we are eventually setting $x_1 = -1/\epsilon$ (and then $\epsilon \to 0$)!
- In other words, work with $\mathcal{I} := \langle \ell_1, \epsilon \rangle = \langle 1 \rangle$!

Non-homogeneity is 'bad'

- Recall the non-border lower bound proof, of making an ideal $\mathcal{I}_k = \langle \ell_1, \ldots, \ell_k \rangle$, such that $\mathsf{det}_n \neq 0 \bmod \mathcal{I}_k$, but $\Sigma^{[k]}\Pi\Sigma = 0 \bmod \mathcal{I}_k$.
- Let $\ell_1 := 1 + \epsilon x_1$. What does taking $\bmod \ell_1$ in the 'border' ($\epsilon \to 0$) mean? Essentially we are eventually setting $x_1 = -1/\epsilon$ (and then $\epsilon \to 0$)!
- In other words, work with $\mathcal{I} := \langle \ell_1, \epsilon \rangle = \langle 1 \rangle$!
- Lesson: Taking mod blindly fails *miserably*!

Non-homogeneity is ‘bad’

- Recall the non-border lower bound proof, of making an ideal $\mathcal{I}_k = \langle \ell_1, \ldots, \ell_k \rangle$, such that $\det_n \neq 0 \bmod \mathcal{I}_k$, but $\Sigma^{[k]}\Pi\Sigma = 0 \bmod \mathcal{I}_k$.
- Let $\ell_1 := 1 + \epsilon x_1$. What does taking $\bmod\ \ell_1$ in the ‘border’ ($\epsilon \to 0$) mean? Essentially we are eventually setting $x_1 = -1/\epsilon$ (and then $\epsilon \to 0$)!
- In other words, work with $\mathcal{I} := \langle \ell_1, \epsilon \rangle = \langle 1 \rangle$!
- Lesson: Taking mod blindly fails *miserably*!
- The worst case:

$$f + \epsilon S \;=\; T_1 + T_2\,,$$

where T_i has each linear factor of the form $1 + \epsilon\ell$!

Non-homogeneity is all we need to care

Non-homogeneity is all we need to care

- Three cases to consider:
  - Case I: Each T_1 and T_2 has one linear polynomial $\ell_i \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$ as a factor, whose ϵ-free term is a linear form. Example: $\ell = (1 + \epsilon)x_1 + \epsilon x_2$,

Non-homogeneity is all we need to care

- Three cases to consider:
  - Case I: Each T_1 and T_2 has one linear polynomial $\ell_i \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$ as a factor, whose ϵ-free term is a linear form. Example: $\ell = (1+\epsilon)x_1 + \epsilon x_2$,
  - Case II (intermediate): T_1 has one homogeneous factor (say ℓ_1) and ϵ-free part of all factors in T_2 are non-homogeneous (in $\boldsymbol{x}$). Non-homogeneous example: $(1+\epsilon) + \epsilon x_1$.

Non-homogeneity is all we need to care

- Three cases to consider:
  - Case I: Each T_1 and T_2 has one linear polynomial $\ell_i \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$ as a factor, whose ϵ-free term is a linear form. Example: $\ell = (1+\epsilon)x_1 + \epsilon x_2$,
  - Case II (intermediate): T_1 has one homogeneous factor (say ℓ_1) and ϵ-free part of all factors in T_2 are non-homogeneous (in $\boldsymbol{x}$). Non-homogeneous example: $(1+\epsilon) + \epsilon x_1$.
  - Case III (*all*-non-homogeneous): Each T_i has all the linear polynomial factors whose ϵ-free is non-homogeneous.

Non-homogeneity is all we need to care

- Three cases to consider:
  - Case I: Each T_1 and T_2 has one linear polynomial $\ell_i \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$ as a factor, whose ϵ-free term is a linear form. Example: $\ell = (1+\epsilon)x_1 + \epsilon x_2$,
  - Case II (intermediate): T_1 has one homogeneous factor (say ℓ_1) and ϵ-free part of all factors in T_2 are non-homogeneous (in $\boldsymbol{x}$). Non-homogeneous example: $(1+\epsilon) + \epsilon x_1$.
  - Case III (*all*-non-homogeneous): Each T_i has all the linear polynomial factors whose ϵ-free is non-homogeneous.
- For the first case, take $\mathcal{I} := \langle \ell_1, \ell_2, \epsilon \rangle$ ($\Rightarrow 1 \notin \mathcal{I}$) and show that $x_1 \cdots x_d + y_1 \cdots y_d + z_1 \cdots z_d = P_d \bmod \mathcal{I} \neq 0$, while RHS circuit $\equiv 0 \bmod \mathcal{I}$.

Non-homogeneity is all we need to care

- Three cases to consider:
  - Case I: Each T_1 and T_2 has one linear polynomial $\ell_i \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$ as a factor, whose ϵ-free term is a linear form. Example: $\ell = (1+\epsilon)x_1 + \epsilon x_2$,
  - Case II (intermediate): T_1 has one homogeneous factor (say ℓ_1) and ϵ-free part of all factors in T_2 are non-homogeneous (in $\boldsymbol{x}$). Non-homogeneous example: $(1+\epsilon) + \epsilon x_1$.
  - Case III (*all*-non-homogeneous): Each T_i has all the linear polynomial factors whose ϵ-free is non-homogeneous.
- For the first case, take $\mathcal{I} := \langle \ell_1, \ell_2, \epsilon \rangle$ ($\Rightarrow 1 \notin \mathcal{I}$) and show that $x_1 \cdots x_d + y_1 \cdots y_d + z_1 \cdots z_d = P_d \bmod \mathcal{I} \neq 0$, while RHS circuit $\equiv 0 \bmod \mathcal{I}$.
- For the second case, take $\mathcal{I} := \langle \ell_1, \epsilon \rangle$. Then, RHS $\bmod \mathcal{I} \in \overline{\Pi\Sigma} = \Pi\Sigma$, while $P_d \bmod \mathcal{I} \notin \Pi\Sigma$.

Non-homogeneity is all we need to care

- Three cases to consider:
  - Case I: Each T_1 and T_2 has one linear polynomial $\ell_i \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$ as a factor, whose ϵ-free term is a linear form. Example: $\ell = (1+\epsilon)x_1 + \epsilon x_2$,
  - Case II (intermediate): T_1 has one homogeneous factor (say ℓ_1) and ϵ-free part of all factors in T_2 are non-homogeneous (in $\boldsymbol{x}$). Non-homogeneous example: $(1+\epsilon) + \epsilon x_1$.
  - Case III (*all*-non-homogeneous): Each T_i has all the linear polynomial factors whose ϵ-free is non-homogeneous.
- For the first case, take $\mathcal{I} := \langle \ell_1, \ell_2, \epsilon \rangle$ ($\Rightarrow 1 \notin \mathcal{I}$) and show that $x_1 \cdots x_d + y_1 \cdots y_d + z_1 \cdots z_d = P_d \bmod \mathcal{I} \neq 0$, while RHS circuit $\equiv 0 \bmod \mathcal{I}$.
- For the second case, take $\mathcal{I} := \langle \ell_1, \epsilon \rangle$. Then, RHS $\bmod \mathcal{I} \in \overline{\Pi\Sigma} = \Pi\Sigma$, while $P_d \bmod \mathcal{I} \notin \Pi\Sigma$.
- So, all-non-homogeneous is all we have to care!

Non-homogeneity is all we need to care

- Three cases to consider:
  - Case I: Each T_1 and T_2 has one linear polynomial $\ell_i \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$ as a factor, whose ϵ-free term is a linear form. Example: $\ell = (1+\epsilon)x_1 + \epsilon x_2$,
  - Case II (intermediate): T_1 has one homogeneous factor (say ℓ_1) and ϵ-free part of all factors in T_2 are non-homogeneous (in $\boldsymbol{x}$). Non-homogeneous example: $(1+\epsilon) + \epsilon x_1$.
  - Case III (*all*-non-homogeneous): Each T_i has all the linear polynomial factors whose ϵ-free is non-homogeneous.
- For the first case, take $\mathcal{I} := \langle \ell_1, \ell_2, \epsilon \rangle$ ($\Rightarrow 1 \notin \mathcal{I}$) and show that $x_1 \cdots x_d + y_1 \cdots y_d + z_1 \cdots z_d = P_d \bmod \mathcal{I} \neq 0$, while RHS circuit $\equiv 0 \bmod \mathcal{I}$.
- For the second case, take $\mathcal{I} := \langle \ell_1, \epsilon \rangle$. Then, RHS $\bmod \mathcal{I} \in \overline{\Pi\Sigma} = \Pi\Sigma$, while $P_d \bmod \mathcal{I} \notin \Pi\Sigma$.
- So, all-non-homogeneous is all we have to care!

Lower bound for all-non-homogeneous $k = 2$

Lower bound for all-non-homogeneous $k = 2$

- $P_d(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon) = T_1 + T_2$, where $T_i \in \Pi\Sigma \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$ have all-non-homogeneous factors.

Lower bound for all-non-homogeneous $k = 2$

- $P_d(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon) = T_1 + T_2$, where $T_i \in \Pi\Sigma \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$ have all-non-homogeneous factors.
- Use DiDIL with the (different) map $\Phi : \boldsymbol{x} \mapsto z\boldsymbol{x}$.

Lower bound for all-non-homogeneous $k = 2$

- $P_d(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon) = T_1 + T_2$, where $T_i \in \Pi\Sigma \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$ have all-non-homogeneous factors.
- Use DiDIL with the (different) map $\Phi : \boldsymbol{x} \mapsto z\boldsymbol{x}$.
- DiDIL shows:

$$\partial_z(z^d P_d / t_2) = \lim_{\epsilon \to 0} g_1 \in \overline{(\Pi\Sigma/\Pi\Sigma)} \cdot \overline{(\Sigma \wedge \Sigma)} \,.$$

Lower bound for all-non-homogeneous $k = 2$

- $P_d(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon) = T_1 + T_2$, where $T_i \in \Pi\Sigma \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$ have all-non-homogeneous factors.
- Use DiDIL with the (different) map $\Phi : \boldsymbol{x} \mapsto z\boldsymbol{x}$.
- DiDIL shows:

$$\partial_z(z^d P_d / t_2) = \lim_{\epsilon \to 0} g_1 \in \overline{(\Pi\Sigma/\Pi\Sigma)} \cdot \overline{(\Sigma \wedge \Sigma)} \,.$$

- Use the minimum power of z to show that $P_d \in \overline{\Sigma \wedge \Sigma}$.

Lower bound for all-non-homogeneous $k = 2$

- $P_d(\boldsymbol{x}) + \epsilon \cdot S(\boldsymbol{x}, \epsilon) = T_1 + T_2$, where $T_i \in \Pi\Sigma \in \mathbb{F}(\epsilon)[\boldsymbol{x}]$ have all-non-homogeneous factors.
- Use DiDIL with the (different) map $\Phi : \boldsymbol{x} \mapsto z\boldsymbol{x}$.
- DiDIL shows:

$$\partial_z(z^d P_d / t_2) = \lim_{\epsilon \to 0} g_1 \in \overline{(\Pi\Sigma/\Pi\Sigma)} \cdot \overline{(\Sigma \wedge \Sigma)} .$$

- Use the minimum power of z to show that $P_d \in \overline{\Sigma \wedge \Sigma}$.
- Partial-derivative measure shows that the above implies $s \geq 2^{\Omega(d)}$!

Conclusion

Concluding remarks

Concluding remarks

Concluding remarks

- ROABP presence helps us give many PIT results (see the papers).

Concluding remarks

- ROABP presence helps us give many PIT results (see the papers).
- Can we show $\overline{\Sigma^{[k]}\Pi^{[d]}\Sigma} \subseteq \Sigma\Pi\Sigma$, for $d = \mathrm{poly}(n)$?

Concluding remarks

- ROABP presence helps us give many PIT results (see the papers).
- Can we show $\overline{\Sigma^{[k]}\Pi^{[d]}\Sigma} \subseteq \Sigma\Pi\Sigma$, for $d = \mathrm{poly}(n)$?
- Can we extend the hierarchy theorem to bounded (top & bottom fanin) depth-4 circuits? i.e., for a *fixed* constant δ, is

$$\overline{\Sigma^{[1]}\Pi\Sigma\Pi^{[\delta]}} \subsetneq \overline{\Sigma^{[2]}\Pi\Sigma\Pi^{[\delta]}} \subsetneq \overline{\Sigma^{[3]}\Pi\Sigma\Pi^{[\delta]}} \ldots,$$

  where the respective gaps are exponential? Clearly, $\delta = 1$ holds, from this work.

Concluding remarks

- ROABP presence helps us give many PIT results (see the papers).
- Can we show $\overline{\Sigma^{[k]}\Pi^{[d]}\Sigma} \subseteq \Sigma\Pi\Sigma$, for $d = \mathrm{poly}(n)$?
- Can we extend the hierarchy theorem to bounded (top & bottom fanin) depth-4 circuits? i.e., for a *fixed* constant δ, is

$$\overline{\Sigma^{[1]}\Pi\Sigma\Pi^{[\delta]}} \subsetneq \overline{\Sigma^{[2]}\Pi\Sigma\Pi^{[\delta]}} \subsetneq \overline{\Sigma^{[3]}\Pi\Sigma\Pi^{[\delta]}} \ldots,$$

where the respective gaps are exponential? Clearly, $\delta = 1$ holds, from this work.

Thank you! Questions?