gct2022 : School and Conference on Geometric Complexity Theory

17-28 Jan 2022 Chennai (India)

Algebraic Complexity: Structural results

Depth reduction, Homogeneization, Multilinearization, ...

Sébastien Tavenas

October 15th, 2021

Overview

1. Why to look at structural results
2. Homogeneization / (Set)-multilinearization
   - Homogeneization
   - Multilinearization
3. Parallelization
   - Classical depth reductions of [Brent] and [VSBR]
   - To constant depth

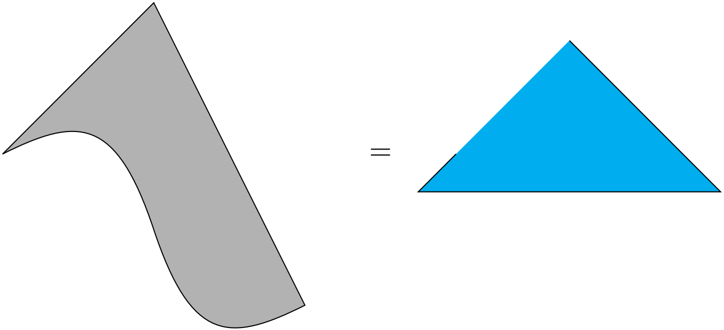
General roadmap for lower bounds

Four steps in most lower bound proofs

Step 1: Finding a "nice" form for the model

Four steps in most lower bound proofs

Step 1: Finding a "nice" form for the model



Meta Theorem 1

Every small circuit can be equivalently computed as a "nice" 

Four steps in most lower bound proofs

Step 1: Finding a “nice” form for the model

Meta Theorem 1

Every small circuit can be equivalently computed as a “nice” 

Four steps in most lower bound proofs

Step 1: Finding a “nice” form for the model

Meta Theorem 1

Every small circuit can be equivalently computed as a “nice” 

Step 2: Constructing a complexity measure

Meta Theorem 2

Find a map $\Gamma : \mathbb{F}[\mathbf{x}] \to \mathbb{Z}_{\geq 0}$ such that Γ () is small.

Four steps in most lower bound proofs

Step 1: Finding a "nice" form for the model

Meta Theorem 1

Every small circuit can be equivalently computed as a "nice" 

Step 2: Constructing a complexity measure

Meta Theorem 2

Find a map $\Gamma : \mathbb{F}[\mathbf{x}] \to \mathbb{Z}_{\geq 0}$ such that Γ () is small.

Step 3: Heuristic estimate for a random polynomial

Meta Theorem 2

Convince yourself that $\Gamma(R)$ *must be* LARGE for a random polynomial R.

Four steps in most lower bound proofs

Step 1: Finding a "nice" form for the model

Meta Theorem 1

Every small circuit can be equivalently computed as a "nice" $\triangle$

Step 2: Constructing a complexity measure

Meta Theorem 2

Find a map $\Gamma : \mathbb{F}[\mathbf{x}] \to \mathbb{Z}_{\geq 0}$ such that $\Gamma\left(\triangle\right)$ is small.

Step 3: Heuristic estimate for a random polynomial

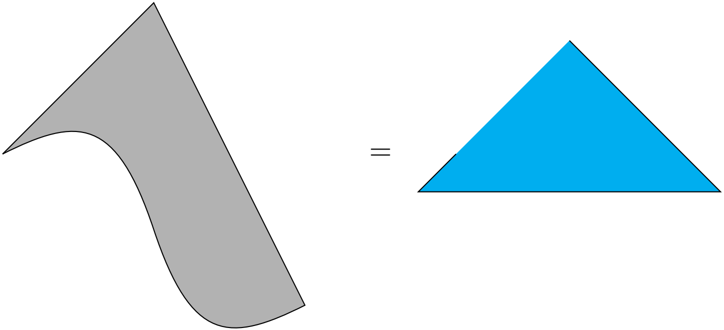
Meta Theorem 2

Convince yourself that $\Gamma(R)$ *must be* LARGE for a random polynomial R.

Step 4: Find a hay in the haystack

Four steps in most lower bound proofs

Step 1: Finding a "nice" form for the model

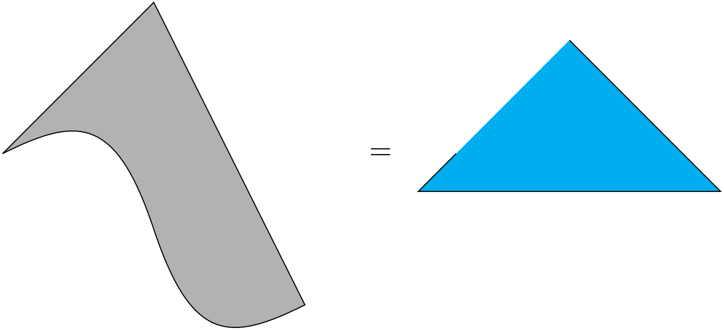


Meta Theorem 1

Every small circuit can be equivalently computed as a "nice" 

Four steps in most lower bound proofs

Step 1: Finding a "nice" form for the model



Meta Theorem 1

Every small circuit can be equivalently computed as a "nice" 

Homogeneization, (Set)-multilinearization, Depth reduction

Arithmetic models

$$\text{Formulas} \quad \subseteq \quad \text{ABP} \quad \subseteq \quad \text{Circuits}$$

Arithmetic models

$$\text{Formulas} \quad \subseteq \quad \text{ABP} \quad \subseteq \quad \text{Circuits}$$

- Reverse inclusions?

Arithmetic models

$$\text{Formulas} \quad \subseteq \quad \text{ABP} \quad \subseteq \quad \text{Circuits}$$

- Reverse inclusions?
- Circuit of size s $\rightsquigarrow$

Arithmetic models

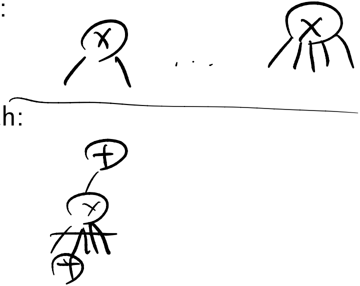
$$\text{Formulas} \quad \subseteq \quad \text{ABP} \quad \subseteq \quad \text{Circuits}$$

- Reverse inclusions?
- Circuit of size s $\rightsquigarrow$Formula of size $s^{O(\log d)}$.

Few words about fan-ins

If nothing is mentionned

- For circuits, formula of "large depth":
  - $+$-gate : unbounded
  - $*$-gate : constant
- For circuits, formula of constant depth:
  - $+$-gate : unbounded
  - $*$-gate : unbounded



Homogenization (basics)

- All gates compute *homogeneous polynomials*.

Homogenization (basics)

- All gates compute *homogeneous polynomials*.
- Hence, no gate can compute polynomials of degree larger than output.

Homogenization (basics)

- All gates compute *homogeneous polynomials*.
- Hence, no gate can compute polynomials of degree larger than output.
- For circuits and ABPs, homogeneity can be assumed without loss of generality.

Homogenization (basics)

- All gates compute *homogeneous polynomials*.
- Hence, no gate can compute polynomials of degree larger than output.
- For circuits and ABPs, homogeneity can be assumed without loss of generality.
  For formulas, probably not.

Homogenization (basics)

- All gates compute *homogeneous polynomials*.
- Hence, no gate can compute polynomials of degree larger than output.
- For circuits and ABPs, homogeneity can be assumed without loss of generality.
  For formulas, probably not.
  For constant depth formulas, certainly not.

(Det_n) in $\Sigma\Pi\Sigma\Pi$

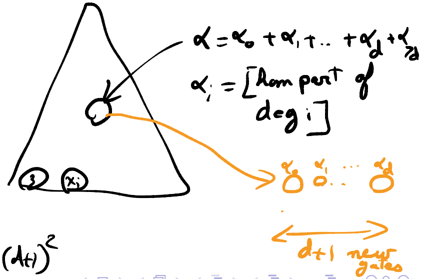
* non hom $\to n^{O(\sqrt[3]{n})}$
* hom $\to 2^{\Omega(\sqrt{n})}$

Homogenization (basics)

- All gates compute *homogeneous polynomials*.
- Hence, no gate can compute polynomials of degree larger than output.
- For circuits and ABPs, homogeneity can be assumed without loss of generality.
  For formulas, probably not.
  For constant depth formulas, certainly not.

$$g = \sum_j g_j \longrightarrow g^{(i)} = \sum_j g_j^{(i)}$$

$$g = g_1 \times g_2 \longrightarrow g^{(i)} = \sum_{j=0}^{i} g_1^{(j)} \times g_2^{(i-j)}$$



Total: $s \mapsto s(d+1) + s(d+1) \times (d+1) \le s(d+1)^2$

(Syntactic) (Set)-multilinearization

- Multilinear, Set-multilinear

(Syntactic) (Set)-multilinearization

- Multilinear, Set-multilinear
- Semantic vs. $\boxed{\text{Syntactic}}$

A circuit syntac set-multilinear

* $(c) \to \{\ \}$
* $(x_i) \to \{j\}$

$X_j \supset$

* $(+)$ with children $\alpha_1 \ldots \alpha_p$ $\longmapsto S_{\alpha_1} = S_{\alpha_2} = \ldots = S_{\alpha_p} = S_\alpha$
* $(\times)$ with children α_1, α_2 $\longmapsto S_\alpha = S_{\alpha_1} \uplus S_{\alpha_2}$.

A circuit is called synt multilinear

* $(c) \to \{\ \}$
* $(x_i) \to \{x_i\}$
* $(+)$ with children (α), (β); S_α, S_β $\longmapsto S_\alpha \cup S_\beta$
* $(\times)$ with children (α), (β) $\longmapsto S_\alpha \uplus S_\beta$

(Syntactic) (Set)-multilinearization

- Multilinear, Set-multilinear
- Semantic vs. Syntactic
- Expensive!

	Syn. Multilinear	Syn. Set-multilinear
Ciruits	???	$s \cdot 2^{O(d)}$
Formulas	???	$2^{O(d \log\log s)}$
Hom. formulas of cst depth	???	$s \cdot d^{O(d)}$

$\rightsquigarrow s^{\log d} 2^{O(d \log d)}$

Trivial $\mapsto 2^n$

A short history of depth reduction

Class	Depth	Size	
Formulas	$O(\log s)$	$\mathrm{poly}(s)$	[Brent]

A short history of depth reduction

Class	Depth	Size	
Formulas	$O(\log s)$	$\mathrm{poly}(s)$	[Brent]
Circuits	$O(\log d)$	$s^{\log s}$	[Hyafil]

A short history of depth reduction

Class	Depth	Size	
Formulas	$O(\log s)$	$\text{poly}(s)$	[Brent]
Circuits	$O(\log d)$	$s^{\log s}$	[Hyafil]
Circuits	$O(\log d)$	$\text{poly}(s)$	[Valiant-Skyum-Berkowitz-Rackoff]

A short history of depth reduction

Class	Depth	Size	
Formulas	$O(\log s)$	$\mathrm{poly}(s)$	[Brent]
Circuits	$O(\log d)$	$s^{\log s}$	[Hyafil]
Circuits	$O(\log d)$	$\mathrm{poly}(s)$	[Valiant-Skyum-Berkowitz-Rackoff]
Circuits	4	$2^{o(n)}$	[Agrawal-Vinay]

A short history of depth reduction

Class	Depth	Size	
Formulas	$O(\log s)$	$\text{poly}(s)$	[Brent]
Circuits	$O(\log d)$	$s^{\log s}$	[Hyafil]
Circuits	$O(\log d)$	$\text{poly}(s)$	[Valiant-Skyum-Berkowitz-Rackoff]
Circuits	4	$\cancel{2^{o(N)}}$ $s^{O(\sqrt{d}\log d)}$	[Agrawal-Vinay] [Koiran]

A short history of depth reduction

Class	Depth	Size	
Formulas	$O(\log s)$	$\text{poly}(s)$	[Brent]
Circuits	$O(\log d)$	$s^{\log s}$	[Hyafil]
Circuits	$O(\log d)$	$\text{poly}(s)$	[Valiant-Skyum-Berkowitz-Rackoff]
Circuits	4	$\cancel{2^{o(N)}}$	[Agrawal-Vinay]
		$\cancel{s^{O(\sqrt{d}\log d)}}$	[Koiran]
		$s^{O(\sqrt{d})}$	[T.]

A short history of depth reduction

Class	Depth	Size	
Formulas	$O(\log s)$	$\text{poly}(s)$	[Brent]
Circuits	$O(\log d)$	$s^{\log s}$	[Hyafil]
Circuits	$O(\log d)$	$\text{poly}(s)$	[Valiant-Skyum-Berkowitz-Rackoff]
Circuits	4	$\cancel{2^{o(N)}}$	[Agrawal-Vinay]
		$\cancel{s^{O(\sqrt{d}\log d)}}$	[Koiran]
		$s^{O(\sqrt{d})}$	[T.]
Circuits	3^*	$s^{O(\sqrt{d})}$	[Gupta-Kamath-Kayal-Saptharishi]

A short history of depth reduction

Class	Depth	Size	
Formulas	$O(\log s)$	$\text{poly}(s)$	[Brent]
Circuits	$O(\log d)$	$s^{\log s}$	[Hyafil]
Circuits	$O(\log d)$	$\text{poly}(s)$	[Valiant-Skyum-Berkowitz-Rackoff]
Circuits	4	$\cancel{2^{o(N)}}$	[Agrawal-Vinay]
		$\cancel{s^{O(\sqrt{d}\log d)}}$	[Koiran]
		$s^{O(\sqrt{d})}$	[T.]
	4^*	$s^{O(d^{1/3})}$	[Gupta-Kamath-Kayal-Saptharishi]
Circuits	3^*	$s^{O(\sqrt{d})}$	[Gupta-Kamath-Kayal-Saptharishi]

Other depth reductions in lower bounds

Multilinear formulas [Raz, Raz-Yehudayoff]

$$f \quad = \quad \sum_{i=1}^{s} g_{i1} \cdot g_{i2} \dots g_{i\ell} \quad , \quad (1/3)^j \cdot n \ \leq \ \mathrm{Var}(g_{ij}) \ \leq \ (2/3)^j \cdot n$$

Other depth reductions in lower bounds

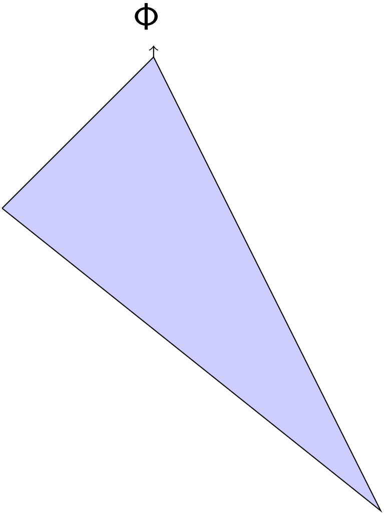
Multilinear formulas [Raz, Raz-Yehudayoff]

$$f \quad = \quad \sum_{i=1}^{s} g_{i1} \cdot g_{i2} \ldots g_{i\ell} \quad , \quad (1/3)^j \cdot n \;\leq\; \mathrm{Var}(g_{ij}) \;\leq\; (2/3)^j \cdot n$$

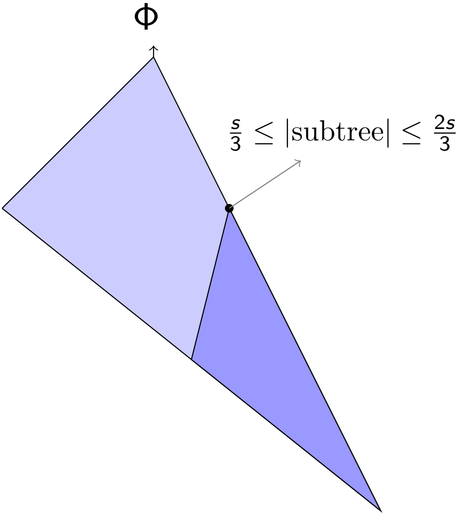
Homogeneous formulas [Hrubes-Yehudayoff]

$$f \quad = \quad \sum_{i=1}^{s} g_{i1} \cdot g_{i2} \ldots g_{i\ell} \quad , \quad (1/3)^j \cdot d \;\leq\; \deg(g_{ij}) \;\leq\; (2/3)^j \cdot d$$

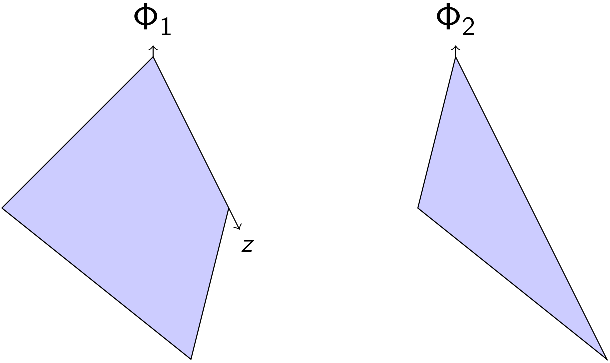
Depth reducing formulas



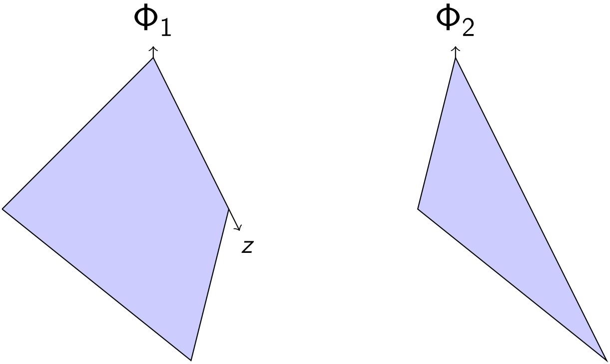
Depth reducing formulas



Depth reducing formulas

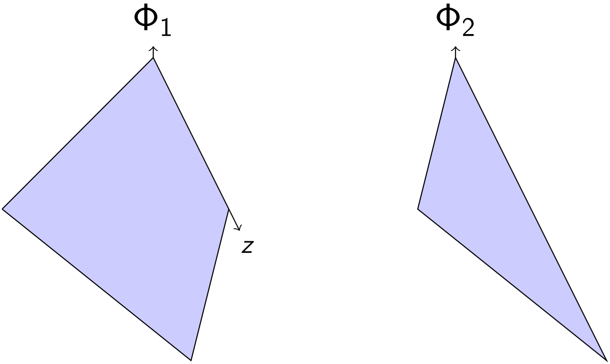


Depth reducing formulas



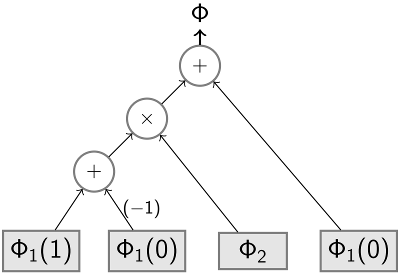
$$\begin{array}{lllll} \Phi_1(z) & = & A \cdot z & + & B \\ \Phi & = & A \cdot \Phi_2 & + & B \end{array}$$

Depth reducing formulas



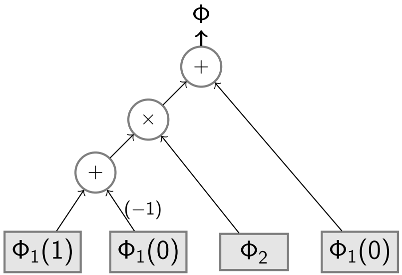
$$\begin{aligned}
\Phi_1(z) &= A \cdot z + B \\
\Phi &= A \cdot \Phi_2 + B = (\Phi_1(1) - \Phi_1(0)) \cdot \Phi_2 + \Phi_1(0)
\end{aligned}$$

Depth reducing formulas



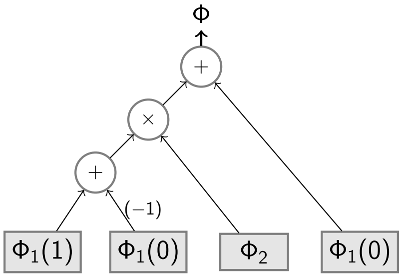
$$
\begin{aligned}
\Phi_1(z) &= A \cdot z + B \\
\Phi &= A \cdot \Phi_2 + B = (\Phi_1(1) - \Phi_1(0)) \cdot \Phi_2 + \Phi_1(0)
\end{aligned}
$$

Depth reducing formulas



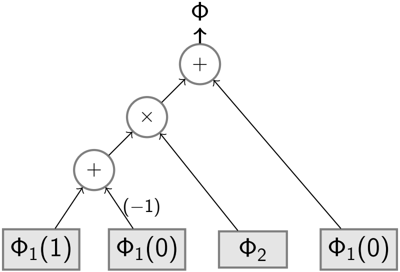
$$\mathrm{Size}(s) \leq \quad 4 \cdot \mathrm{Size}(2s/3) \quad + \quad O(1)$$
$$\mathrm{Depth}(s) \leq \quad \mathrm{Depth}(2s/3) \quad + \quad O(1)$$

Depth reducing formulas



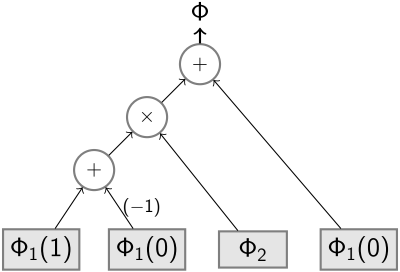
$$\begin{aligned}\mathrm{Size}(s) &\leq 4 \cdot \mathrm{Size}(2s/3) + O(1) \implies \mathrm{poly}(s)\\ \mathrm{Depth}(s) &\leq \mathrm{Depth}(2s/3) + O(1)\end{aligned}$$

Depth reducing formulas



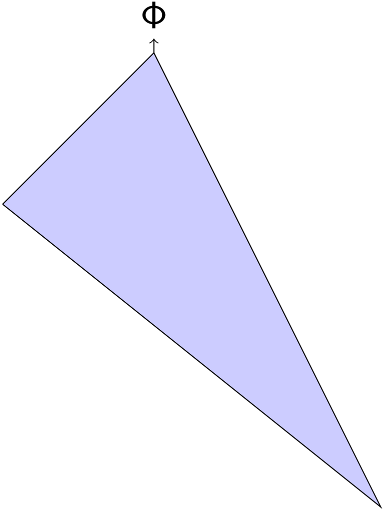
$$\mathrm{Size}(s) \leq 4 \cdot \mathrm{Size}(2s/3) + O(1) \implies \mathrm{poly}(s)$$
$$\mathrm{Depth}(s) \leq \mathrm{Depth}(2s/3) + O(1) \implies O(\log s)$$

Depth reducing formulas

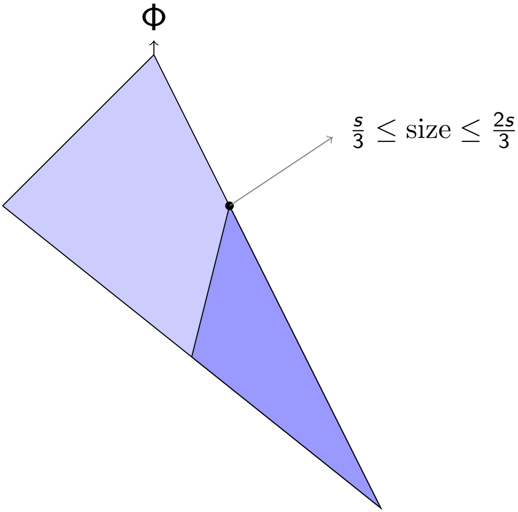


$$\begin{array}{rccccccc} \mathrm{Size}(s) \leq & 4 \cdot \mathrm{Size}(2s/3) & + & O(1) & \implies & \mathrm{poly}(s) \\ \mathrm{Depth}(s) \leq & \mathrm{Depth}(2s/3) & + & O(1) & \implies & O(\log s) \end{array}$$

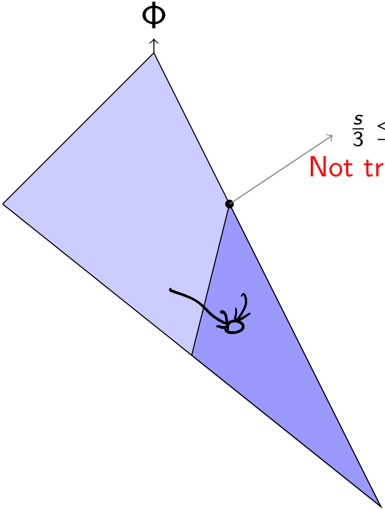
Adapting to circuits



Adapting to circuits



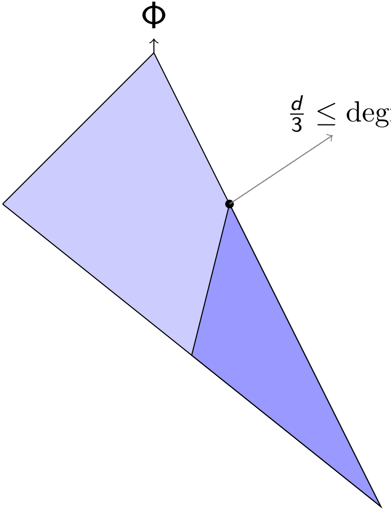
Adapting to circuits



$\frac{s}{3} \leq \text{size} \leq \frac{2s}{3}$

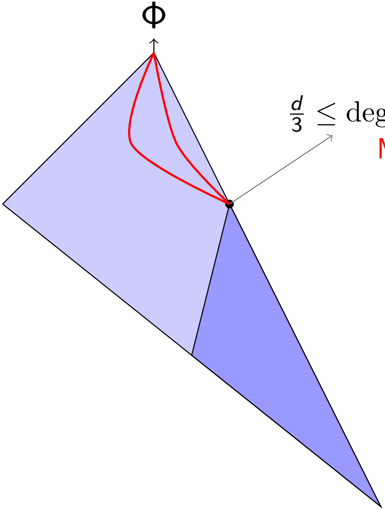
Not true for circuits!

Adapting to circuits



$\frac{d}{3} \leq \mathrm{degree} \leq \frac{2d}{3}$ (start with a homogeneous circuit)

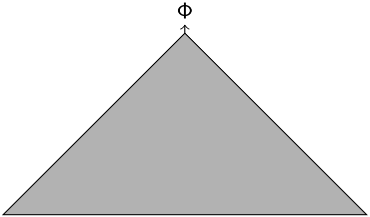
Adapting to circuits



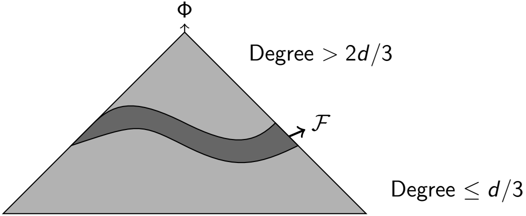
$\frac{d}{3} \leq \mathrm{degree} \leq \frac{2d}{3}$ (start with a homogeneous circuit)

Multiple paths from root!

Adapting to circuits: Attempt 1

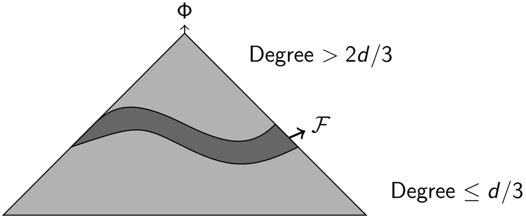


Adapting to circuits: Attempt 1



$$\mathcal{F} \quad = \quad \left\{ v \in \Phi \mid \frac{d}{3} < \deg(v) \leq \frac{2d}{3} \right\}$$

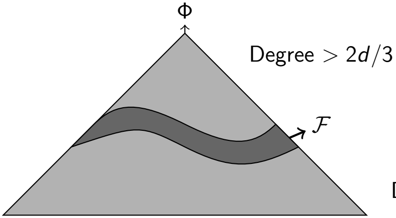
Adapting to circuits: Attempt 1



$$\mathcal{F} \quad = \quad \left\{ v \in \Phi \mid \frac{d}{3} < \deg(v) \leq \frac{2d}{3} \right\}$$

$$\Phi \quad = \quad \sum_{v_i \in \mathcal{F}} A_i \Phi_{v_i} \quad + \quad \sum_{v_i, v_j \in \mathcal{F}} A_{i,j} \Phi_{v_i} \Phi_{v_j}$$

Adapting to circuits: Attempt 1

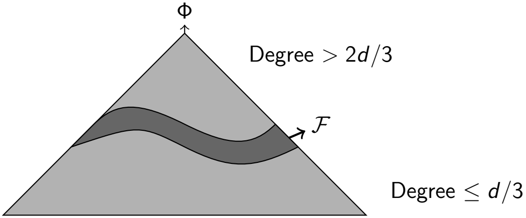


$$\mathcal{F} = \left\{ v \in \Phi \mid \frac{d}{3} < \deg(v) \leq \frac{2d}{3} \right\}$$

$$\Phi = \sum_{v_i \in \mathcal{F}} A_i \Phi_{v_i} + \sum_{v_i, v_j \in \mathcal{F}} A_{i,j} \Phi_{v_i} \Phi_{v_j}$$

each have degree at most $2d/3$

Adapting to circuits: Attempt 1



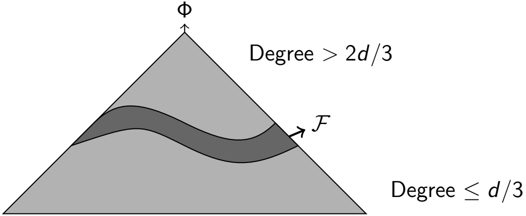
$$\mathcal{F} = \left\{ v \in \Phi \mid \frac{d}{3} < \deg(v) \leq \frac{2d}{3} \right\}$$

$$\Phi = \sum_{v_i \in \mathcal{F}} A_i \Phi_{v_i} + \sum_{v_i, v_j \in \mathcal{F}} A_{i,j} \Phi_{v_i} \Phi_{v_j}$$

each have degree at most $2d/3$

Interpolate!

Adapting to circuits: Attempt 1

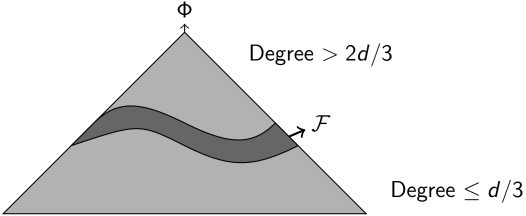


$$\mathcal{F} = \left\{ v \in \Phi \mid \frac{d}{3} < \deg(v) \leq \frac{2d}{3} \right\}$$

$$\Phi = \sum_{v_i \in \mathcal{F}} A_i \Phi_{v_i} + \sum_{v_i, v_j \in \mathcal{F}} A_{i,j} \Phi_{v_i} \Phi_{v_j}$$

$$\mathrm{Depth}(d) = \mathrm{Depth}(2d/3) + O(1)$$

Adapting to circuits: Attempt 1

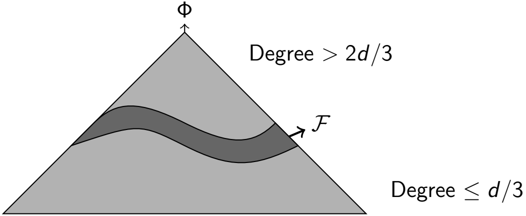


$$\mathcal{F} \quad = \quad \left\{ v \in \Phi \mid \frac{d}{3} < \deg(v) \leq \frac{2d}{3} \right\}$$

$$\Phi \quad = \quad \sum_{v_i \in \mathcal{F}} A_i \Phi_{v_i} \quad + \quad \sum_{v_i, v_j \in \mathcal{F}} A_{i,j} \Phi_{v_i} \Phi_{v_j}$$

$$\mathrm{Depth}(d) = \qquad\qquad O(\log d)$$

Adapting to circuits: Attempt 1

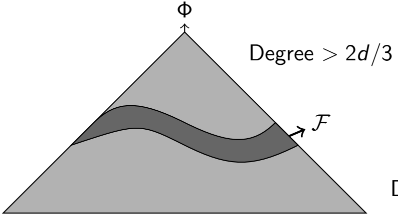


$$\mathcal{F} = \left\{ v \in \Phi \mid \frac{d}{3} < \deg(v) \leq \frac{2d}{3} \right\}$$

$$\Phi = \sum_{v_i \in \mathcal{F}} A_i \Phi_{v_i} + \sum_{v_i, v_j \in \mathcal{F}} A_{i,j} \Phi_{v_i} \Phi_{v_j}$$

$$\mathrm{Depth}(d) = O(\log d)$$
$$\mathrm{Size}(s, d) = ?$$

Adapting to circuits: Attempt 1



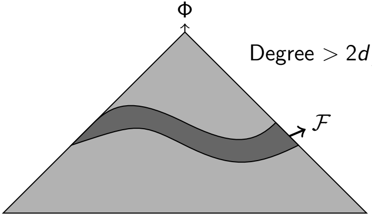
$$\mathcal{F} \quad = \quad \left\{ v \in \Phi \mid \frac{d}{3} < \deg(v) \leq \frac{2d}{3} \right\}$$

$$\Phi \quad = \quad \sum_{v_i \in \mathcal{F}} A_i \Phi_{v_i} \quad + \quad \sum_{v_i, v_j \in \mathcal{F}} A_{i,j} \Phi_{v_i} \Phi_{v_j}$$

$$\mathrm{Depth}(d) = \qquad O(\log d)$$

$$\mathrm{Size}(s, d) = \qquad s^{O(\log d)}$$

Adapting to circuits: [Hyafil]



$$\mathcal{F} = \left\{ v \in \Phi \mid \frac{d}{3} < \deg(v) \leq \frac{2d}{3} \right\}$$

$$\Phi = \sum_{v_i \in \mathcal{F}} A_i \Phi_{v_i} + \sum_{v_i, v_j \in \mathcal{F}} A_{i,j} \Phi_{v_i} \Phi_{v_j}$$

$$\mathrm{Depth}(d) = O(\log d)$$

$$\mathrm{Size}(s, d) = s^{O(\log d)}$$

Adapting to circuits: Attempt 2

- Want an analogue of $\Phi = A \cdot \Phi_v + B$.

Adapting to circuits: Attempt 2

- Want an analogue of $\Phi = A \cdot \Phi_v + B$.
- Problem is that there are multiple paths to v.

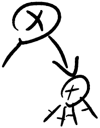
Adapting to circuits: Attempt 2

- Want an analogue of $\Phi = A \cdot \Phi_v + B$.
- Problem is that there are multiple paths to v.
  Φ isn't really an affine function in Φ_v.

Adapting to circuits: Attempt 2

- Want an analogue of $\Phi = A \cdot \Phi_v + B$.
- Problem is that there are multiple paths to v.
  Φ isn't really an affine function in Φ_v.

[VSBR]: Do not look at all paths. Only take a canonical path, like say taking the right-edge out of every $\times$-gate.

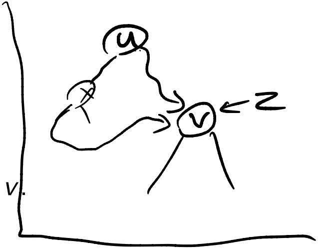


Adapting to circuits: Attempt 2

- Want an analogue of $\Phi = A \cdot \Phi_v + B$.
- Problem is that there are multiple paths to v.
  Φ isn't really an affine function in Φ_v.

[VSBR]: Do not look at all paths. Only take a canonical path, like say taking the right-edge out of every $\times$-gate. More like "suffixes"

Adapting to circuits: Attempt 2

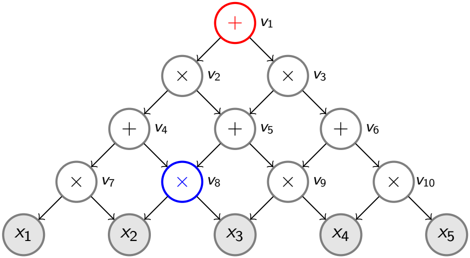


- Want an analogue of $\Phi = A \cdot \Phi_v + B$.
- Problem is that there are multiple paths to v.
  Φ isn't really an affine function in Φ_v.

[VSBR]: Do not look at all paths. Only take a canonical path, like say taking the right-edge out of every $\times$-gate. More like "suffixes"

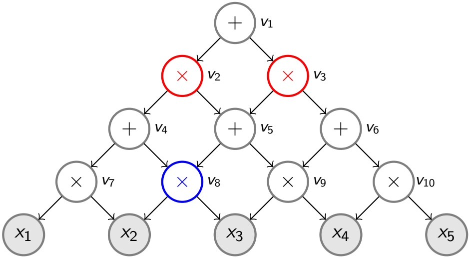
$$[u : v] = \begin{cases} 1 & \text{if } u = v \\ 0 & \text{o/w if } u \text{ is a leaf} \\ [u_1 : v] + [u_2 : v] & \text{if } u = u_1 + u_2 \\ [u_1] \cdot [u_2 : v] & \text{if } u = u_1 \times u_2 \end{cases}$$

An example



$$[v_1 : v_8] \quad =$$

An example



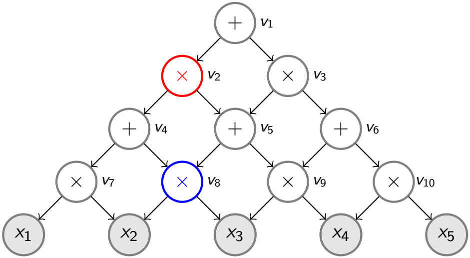
$$[v_1 : v_8] = [v_2 : v_8] + \underbrace{[v_3 : v_8]}$$

$$[v_5]\cdot\underbrace{[v_6 : v_8]}$$

$$[v_9 : v_8] + [v_{10} : v_8]$$

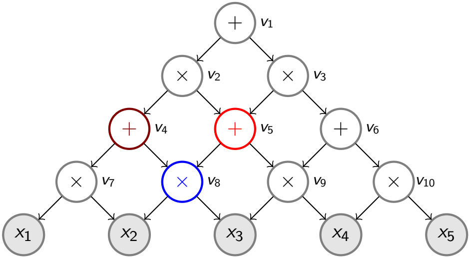
$$[x_3]\cdot\underbrace{[x_4 : v_8]}_{=0} + [x_4]\underbrace{[x_5 : v_8]}_{=0}$$

An example



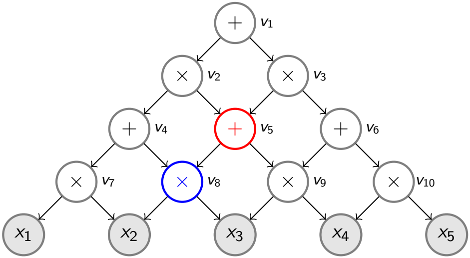
$$[v_1 : v_8] \quad = \quad [v_2 : v_8] \quad + \quad \cancel{[v_3 : v_8]}$$

An example



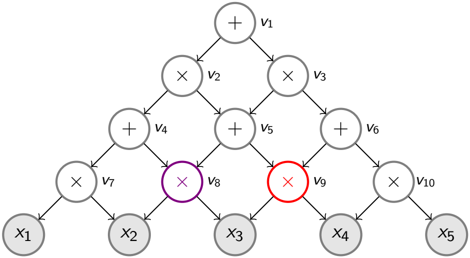
$$\begin{aligned}[v_1 : v_8] &= [v_2 : v_8] \quad + \quad \cancel{[v_3 : v_8]} \\ &= [v_4] \cdot [v_5 : v_8]\end{aligned}$$

An example



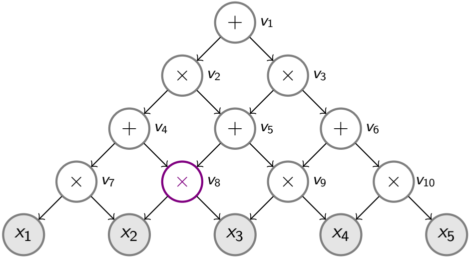
$$
\begin{aligned}
[v_1 : v_8] &= [v_2 : v_8] \quad + \quad \cancel{[v_3 : v_8]} \\
&= [v_4] \cdot [v_5 : v_8] \\
&= (x_1 x_2 + x_2 x_3) \cdot [v_5 : v_8]
\end{aligned}
$$

An example



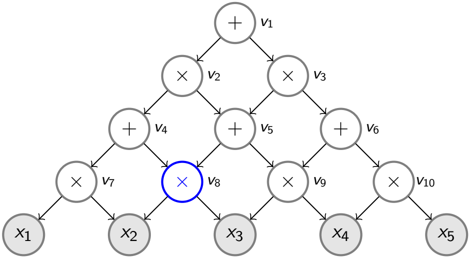
$$
\begin{aligned}
[v_1 : v_8] &= [v_2 : v_8] \quad + \quad \cancel{[v_3 : v_8]} \\
&= [v_4] \cdot [v_5 : v_8] \\
&= (x_1x_2 + x_2x_3) \cdot [v_5 : v_8] \\
&= (x_1x_2 + x_2x_3) \cdot ([v_8 : v_8] + [v_9 : v_8])
\end{aligned}
$$

An example



$$\begin{aligned}
[v_1 : v_8] &= [v_2 : v_8] \quad + \quad \cancel{[v_3 : v_8]} \\
&= [v_4] \cdot [v_5 : v_8] \\
&= (x_1x_2 + x_2x_3) \cdot [v_5 : v_8] \\
&= (x_1x_2 + x_2x_3) \cdot ([v_8 : v_8] + \cancel{[v_9 : v_8]})
\end{aligned}$$

An example



$$
\begin{aligned}
[v_1 : v_8] &= [v_2 : v_8] \quad + \quad \cancel{[v_3 : v_8]} \\
&= [v_4] \cdot [v_5 : v_8] \\
&= (x_1x_2 + x_2x_3) \cdot [v_5 : v_8] \\
&= (x_1x_2 + x_2x_3) \cdot ([v_8 : v_8] + \cancel{[v_9 : v_8]}) \\
&= (x_1x_2 + x_2x_3)
\end{aligned}
$$

[VSBR] continued ...

We want a set of nodes $\mathcal{F}$ such that

$$[u] \quad = \quad \sum_{v \in \mathcal{F}} [u : v] \cdot [v]$$

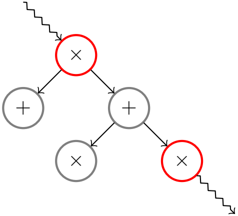
What are candidates for $\mathcal{F}$?

[VSBR] continued ...

We want a set of nodes $\mathcal{F}$ such that

$$[u] \quad = \quad \sum_{v \in \mathcal{F}} [u : v] \cdot [v]$$

What are candidates for $\mathcal{F}$?

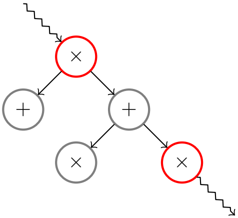


[VSBR] continued ...

We want a set of nodes $\mathcal{F}$ such that

$$[u] \quad = \quad \sum_{v \in \mathcal{F}} [u : v] \cdot [v]$$

What are candidates for $\mathcal{F}$? Every "right-path" must pass through exactly one $v \in \mathcal{F}$



[VSBR] continued ...

We want a set of nodes $\mathcal{F}$ such that

$$[u] \quad = \quad \sum_{v \in \mathcal{F}} [u : v] \cdot [v]$$

What are candidates for $\mathcal{F}$? Every “right-path” must pass through exactly one $v \in \mathcal{F}$

$$\mathcal{F}_a \quad = \quad \{v \mid \deg(v) \geq a \ , \ \deg(v_L), \deg(v_R) < a\}$$

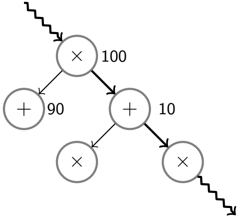
[VSBR] continued ...

We want a set of nodes $\mathcal{F}$ such that

$$[u] \quad = \quad \sum_{v \in \mathcal{F}} [u : v] \cdot [v]$$

What are candidates for $\mathcal{F}$? Every "right-path" must pass through exactly one $v \in \mathcal{F}$

$$\mathcal{F}_a \quad = \quad \{v \mid \deg(v) \geq a \ , \ \deg(v_L), \deg(v_R) < a\}$$



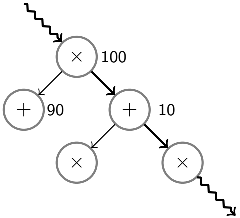
[VSBR] continued ...

We want a set of nodes $\mathcal{F}$ such that

$$[u] \quad = \quad \sum_{v \in \mathcal{F}} [u : v] \cdot [v]$$

What are candidates for $\mathcal{F}$? Every "right-path" must pass through exactly one $v \in \mathcal{F}$

$$\mathcal{F}_a \quad = \quad \{v \mid \deg(v) \geq a \ , \ \deg(v_L), \deg(v_R) < a\}$$



Make the circuit *right heavy*.

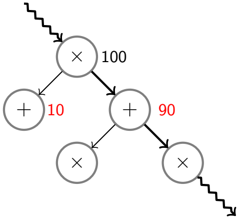
[VSBR] continued ...

We want a set of nodes $\mathcal{F}$ such that

$$[u] \quad = \quad \sum_{v \in \mathcal{F}} [u : v] \cdot [v]$$

What are candidates for $\mathcal{F}$? Every "right-path" must pass through exactly one $v \in \mathcal{F}$

$$\mathcal{F}_a \quad = \quad \{v \mid \deg(v) \geq a \ , \ \deg(v_L), \deg(v_R) < a\}$$



Make the circuit *right heavy*.

[VSBR] continued ...

We want a set of nodes $\mathcal{F}$ such that

$$[u] \quad = \quad \sum_{v \in \mathcal{F}} [u : v] \cdot [v]$$

What are candidates for $\mathcal{F}$? Every "right-path" must pass through exactly one $v \in \mathcal{F}$

$$\mathcal{F}_a \quad = \quad \{v \mid \deg(v) \geq a\ ,\ \deg(v_L), \deg(v_R) < a\}$$

Lemma

$$[u] \quad = \quad \sum_{v \in \mathcal{F}_a} [u : v] \cdot [v]$$

$$[u : w] \quad = \quad \sum_{v \in \mathcal{F}_a} [u : v] \cdot [v : w]$$

[VSBR] continued ...

$$\mathcal{F}_a = \{v \mid \deg(v) \geq a\ ,\ \deg(v_L), \deg(v_R) < a\}$$

$$[u] = \sum_{v \in \mathcal{F}_{a_{[u]}}} [u : v] \cdot [v]$$

[VSBR] continued ...

$$\mathcal{F}_a = \{v \mid \deg(v) \geq a\ ,\ \deg(v_L), \deg(v_R) < a\}$$

$$[u] = \sum_{v \in \mathcal{F}_{a_{[u]}}} [u : v] \cdot [v_L] \cdot [v_R]$$

[VSBR] continued ...

$$\mathcal{F}_a = \{v \mid \deg(v) \geq a\ ,\ \deg(v_L), \deg(v_R) < a\}$$

$$[u] = \sum_{v \in \mathcal{F}_{a_{[u]}}} [u : v] \cdot [v_L] \cdot [v_R] \qquad a_{[u]} = \deg(u)/2$$

[VSBR] continued ...

$$\mathcal{F}_a = \{v \mid \deg(v) \geq a\ ,\ \deg(v_L), \deg(v_R) < a\}$$

$$[u] = \sum_{v \in \mathcal{F}_{a_{[u]}}} [u : v] \cdot [v_L] \cdot [v_R] \qquad a_{[u]} = \deg(u)/2$$

[VSBR] continued ...

$$\mathcal{F}_a = \{v \mid \deg(v) \geq a\ ,\ \deg(v_L), \deg(v_R) < a\}$$

$$[u] = \sum_{v \in \mathcal{F}_{a_{[u]}}} [u : v] \cdot [v_L] \cdot [v_R] \qquad a_{[u]} = \deg(u)/2$$

[VSBR] continued ...

$$\mathcal{F}_a = \{v \mid \deg(v) \geq a\ ,\ \deg(v_L), \deg(v_R) < a\}$$

$$[u] = \sum_{v \in \mathcal{F}_{a_{[u]}}} [u : v] \cdot [v_L] \cdot [v_R] \qquad a_{[u]} = \deg(u)/2$$

[VSBR] continued ...

$$\mathcal{F}_a = \{v \mid \deg(v) \geq a\ ,\ \deg(v_L), \deg(v_R) < a\}$$

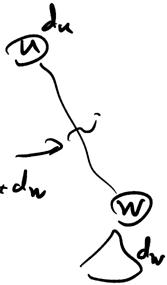
$$[u] = \sum_{v \in \mathcal{F}_{a_{[u]}}} [u:v] \cdot [v_L] \cdot [v_R] \qquad a_{[u]} = \deg(u)/2$$

$$[u:w] = \sum_{v \in \mathcal{F}_{a_{[u:w]}}} [u:v] \cdot [v:w]$$

[VSBR] continued ...

$$\mathcal{F}_a = \{ v \mid \deg(v) \geq a \, , \ \deg(v_L), \deg(v_R) < a \}$$

$$[u] = \sum_{v \in \mathcal{F}_{a_{[u]}}} [u : v] \cdot [v_L] \cdot [v_R] \qquad a_{[u]} = \deg(u)/2 \quad \frac{d_u - d_w}{2} + d_w$$



$$[u : w] = \sum_{v \in \mathcal{F}_{a_{[u:w]}}} [u : v] \cdot [v_L] \cdot [v_R : w]$$

[VSBR] continued ...

$$\mathcal{F}_a = \{v \mid \deg(v) \geq a\ ,\ \deg(v_L), \deg(v_R) < a\}$$

$$[u] = \sum_{v \in \mathcal{F}_{a_{[u]}}} [u : v] \cdot [v_L] \cdot [v_R] \qquad a_{[u]} = \deg(u)/2$$

$$[u : w] = \sum_{v \in \mathcal{F}_{a_{[u:w]}}} [u : v] \cdot [v_L] \cdot [v_R : w] \qquad a_{[u:w]} = \frac{\deg(u) + \deg(w)}{2}$$

[VSBR] continued ...

$$\mathcal{F}_a = \{v \mid \deg(v) \geq a\ ,\ \deg(v_L), \deg(v_R) < a\}$$

$$[u] = \sum_{v \in \mathcal{F}_{a_{[u]}}} [u : v] \cdot [v_L] \cdot [v_R] \qquad a_{[u]} = \deg(u)/2$$

$$[u : w] = \sum_{v \in \mathcal{F}_{a_{[u:w]}}} \underbrace{[u : v]} \cdot [v_L] \cdot [v_R : w] \qquad a_{[u:w]} = \frac{\deg(u) + \deg(w)}{2}$$

$$d_u - d_v \leq d_u - \frac{d_u + d_w}{2} = \frac{d_u}{2} - \frac{d_w}{2}$$

[VSBR] continued ...

$$\mathcal{F}_a = \{v \mid \deg(v) \geq a\ ,\ \deg(v_L), \deg(v_R) < a\}$$

$$[u] = \sum_{v \in \mathcal{F}_{a_{[u]}}} [u : v] \cdot [v_L] \cdot [v_R] \qquad a_{[u]} = \deg(u)/2$$

$$[u : w] = \sum_{v \in \mathcal{F}_{a_{[u:w]}}} [u : v] \cdot \overbrace{[v_L]}^{\frac{d_u + d_w}{2}} \cdot \underbrace{[v_R : w]}_{\le \frac{d_u + d_w}{2} - d_w = \frac{d_u - d_w}{2}} \qquad a_{[u:w]} = \frac{\deg(u) + \deg(w)}{2}$$

[VSBR] continued ...

$$\mathcal{F}_a = \{v \mid \deg(v) \geq a\ ,\ \deg(v_L), \deg(v_R) < a\}$$

$$[u] = \sum_{v \in \mathcal{F}_{a_{[u]}}} [u : v] \cdot [v_L] \cdot [v_R] \qquad a_{[u]} = \deg(u)/2$$

$$[u : w] = \sum_{v \in \mathcal{F}_{a_{[u:w]}}} [u : v] \cdot [v_L] \cdot [v_R : w] \qquad a_{[u:w]} = \frac{\deg(u) + \deg(w)}{2}$$

$$\leq \frac{d_u - d_w}{2}$$

$$= \sum_{v \in \mathcal{F}_{a_{[u:w]}}} [u : v] \cdot \left(\sum_{q \in \mathcal{F}_{a_{[v_L]}}} [v_L : q] \cdot [q_L] \cdot [q_R] \right) \cdot [v_R : w]$$

[VSBR] continued ...

$$\mathcal{F}_a = \{v \mid \deg(v) \geq a\ ,\ \deg(v_L), \deg(v_R) < a\}$$

$$[u] = \sum_{v \in \mathcal{F}_{a_{[u]}}} [u : v] \cdot [v_L] \cdot [v_R] \qquad a_{[u]} = \deg(u)/2$$

$$[u : w] = \sum_{v \in \mathcal{F}_{a_{[u:w]}}} [u : v] \cdot [v_L] \cdot [v_R : w] \qquad a_{[u:w]} = \frac{\deg(u) + \deg(w)}{2}$$

$$= \sum_{v \in \mathcal{F}_{a_{[u:w]}}} [u : v] \cdot \left(\sum_{q \in \mathcal{F}_{a_{[v]}}} [v_L : q] \cdot [q_L] \cdot [q_R] \right) \cdot [v_R : w]$$

[VSBR] continued ...

$$\mathcal{F}_a = \{v \mid \deg(v) \geq a\ ,\ \deg(v_L), \deg(v_R) < a\}$$

$$[u] = \sum_{v \in \mathcal{F}_{a_{[u]}}} [u : v] \cdot [v_L] \cdot [v_R] \qquad a_{[u]} = \deg(u)/2$$

$$[u : w] = \sum_{v \in \mathcal{F}_{a_{[u:w]}}} [u : v] \cdot [v_L] \cdot [v_R : w] \qquad a_{[u:w]} = \frac{\deg(u) + \deg(w)}{2}$$

$$= \sum_{v \in \mathcal{F}_{a_{[u:w]}}} [u : v] \cdot \left(\sum_{q \in \mathcal{F}_{a_{[v]}}} [v_L : q] \cdot [q_L] \cdot [q_R] \right) \cdot [v_R : w]$$

Summarizing

$$[u] = \sum_{v \in \mathcal{F}_a} [u : v] \cdot [v_L] \cdot [v_R]$$

$$[u : w] = \sum_{v \in \mathcal{F}_a} \sum_{q \in \mathcal{F}_a} [u : v] \cdot [v : q] \cdot [q_L] \cdot [q_R] \cdot [v_R : w]$$

Theorem ([Valiant-Skyum-Berkowitz-Rackoff])

If Φ is a size s circuit computing an n-variate degree d polynomial f, then there is a circuit Φ' computing f with the following properties.

- *Every gate of Φ' computes either $[u]$, $[u : v]$, or on of the above products, (so size $O(s^4)$)*
- *All addition gates have fan-in at most s^2,*
- *All multiplication gates have fan-in at most 5, and*
- *If v_1 is a child of a $\times$-gate v in Φ', then $\deg(v_1) \leq \deg(v)/2$.*

Summarizing

$$[u] = \sum_{v \in \mathcal{F}_a} [u : v] \cdot [v_L] \cdot [v_R]$$

$$[u : w] = \sum_{v \in \mathcal{F}_a} \sum_{q \in \mathcal{F}_a} [u : v] \cdot [v : q] \cdot [q_L] \cdot [q_R] \cdot [v_R : w]$$

Theorem ([Valiant-Skyum-Berkowitz-Rackoff])

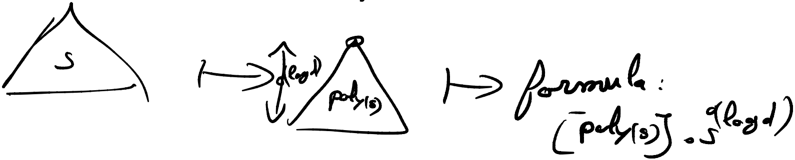
If Φ is a size s circuit computing an n-variate degree d polynomial f, then there is a circuit Φ' computing f with the following properties.

- *Every gate of Φ' computes either $[u]$, $[u : v]$, or on of the above products, (so size $O(s^4)$)*
- *All addition gates have fan-in at most s^2,*
- *All multiplication gates have fan-in at most 5, and*
- *If v_1 is a child of a $\times$-gate v in Φ', then $\deg(v_1) \leq \deg(v)/2$. Hence, the depth of Φ' is $O(\log d)$.*

First consequences of [VSBR]

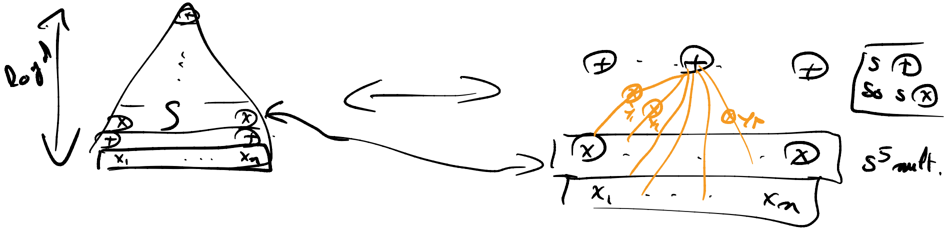
First consequences of [VSBR]

- A sized-s circuit can be simulated by a formula of size $s^{O(\log d)}$.



First consequences of [VSBR]

- A sized-s circuit can be simulated by a formula of size $s^{O(\log d)}$.
- Easy way to construct universal circuits.



Reducing to depth four

Can we reduce the depth further?

Reducing to depth four

Can we reduce the depth further?

Theorem (Koiran)

If f is computed by a circuit of size s,
then it is computed by a $\Sigma\Pi\Sigma\Pi$ of size $s^{O(\sqrt{d}\log d)}$.

Reducing to depth four

Can we reduce the depth further?

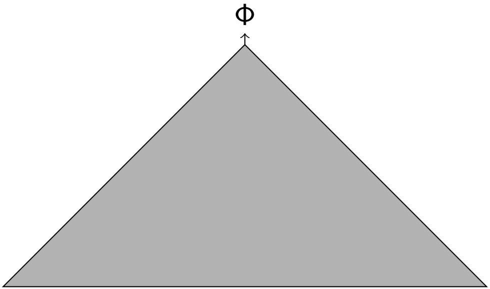
Theorem (Koiran)

If f is computed by a circuit of size s,
then it is computed by a $\Sigma\Pi\Sigma\Pi$ of size $s^{O(\sqrt{d}\log d)}$.

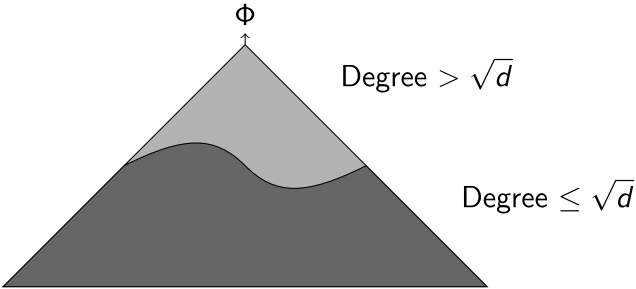
Lemma

If f is computed by an ABP of size s,
then it is computed by a $\Sigma\Pi\Sigma\Pi$ of size $s^{O(\sqrt{d})}$.

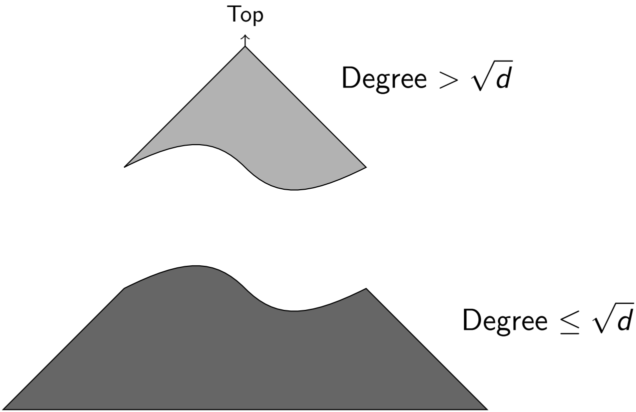
Reducing to depth four : starting from circuits



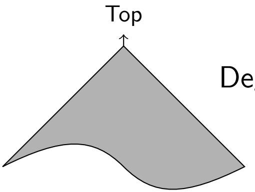
Reducing to depth four : starting from circuits



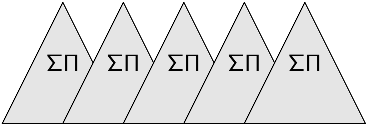
Reducing to depth four : starting from circuits



Reducing to depth four : starting from circuits



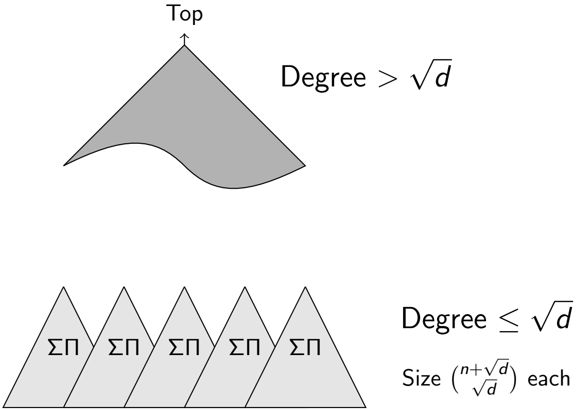
Degree $> \sqrt{d}$



Degree $\leq \sqrt{d}$

Size $\binom{n+\sqrt{d}}{\sqrt{d}}$ each

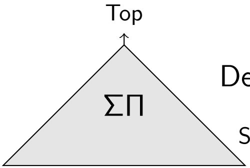
Reducing to depth four : starting from circuits



Lemma ([T.])

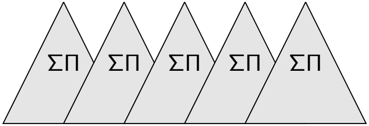
If the circuit has [VSBR] properties,
then $\deg(\mathrm{Top}(z_1, \ldots, z_s)) \leq 15\sqrt{d}$

Reducing to depth four : starting from circuits



Degree $> \sqrt{d}$

Size $\binom{s+15\sqrt{d}}{15\sqrt{d}}$



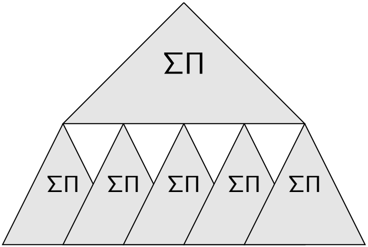
Degree $\leq \sqrt{d}$

Size $\binom{n+\sqrt{d}}{\sqrt{d}}$ each

Lemma ([T.])

If the circuit has [VSBR] properties,
then $\deg(\mathrm{Top}(z_1, \ldots, z_s)) \leq 15\sqrt{d}$

Reducing to depth four : starting from circuits

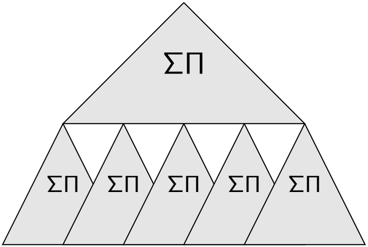


Theorem

Equivalent depth-4 circuit of size

$$s\binom{n+\sqrt{d}}{n} \quad + \quad \binom{s+15\sqrt{d}}{s} \quad = \quad s^{O(\sqrt{d})}$$

Reducing to depth four : starting from circuits



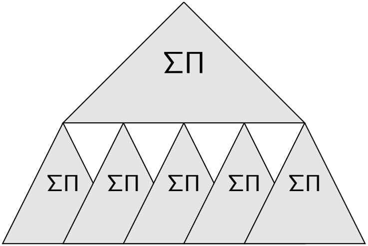
Theorem

Equivalent depth-4 circuit of size

$$s\binom{n+\sqrt{d}}{n} \quad + \quad \binom{s+15\sqrt{d}}{s} \quad = \quad s^{O(\sqrt{d})}$$

$\square$

Reducing to depth four : starting from circuits



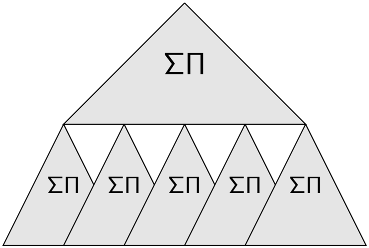
Theorem

Equivalent homogeneous depth-4 circuit with bottom fan-in at most $\sqrt{d}$ of size

$$s\binom{n+\sqrt{d}}{n} \quad + \quad \binom{s+15\sqrt{d}}{s} \quad = \quad s^{O(\sqrt{d})}$$

□

Reducing to depth four : starting from circuits



Theorem

Equivalent homogeneous $\Sigma\Pi\Sigma\Pi^{[\sqrt{d}]}$ *circuit of size*

$$s\binom{n+\sqrt{d}}{n} + \binom{s+15\sqrt{d}}{s} = s^{O(\sqrt{d})}$$

□

[SV]'s proof

Let's start with [VSBR]

$$f = \sum_{i=1}^{s} f_{i1} \cdot f_{i2} \cdot f_{i3} \cdot f_{i4} \cdot f_{i5}$$

[SV]'s proof

Let's start with [VSBR]

$$f = \sum_{i=1}^{s} f_{i1} \cdot f_{i2} \cdot f_{i3} \cdot f_{i4} \cdot f_{i5}$$

This is a $\Sigma\Pi\Sigma\Pi^{[d/2]}$ circuit. We want to obtain a $\Sigma\Pi\Sigma\Pi^{[t]}$ circuit.

[SV]'s proof

Let's start with [VSBR]

$$f \quad = \quad \sum_{i=1}^{s} f_{i1} \cdot f_{i2} \cdot f_{i3} \cdot f_{i4} \cdot f_{i5}$$

This is a $\Sigma\Pi\Sigma\Pi^{[d/2]}$ circuit. We want to obtain a $\Sigma\Pi\Sigma\Pi^{[t]}$ circuit.
Each f_{ij} is also some $[u : v]$. Keep expanding terms of degree more than t.

[SV]'s proof

Let's start with [VSBR]

$$f \quad = \quad \sum_{i=1}^{s} f_{i1} \cdot f_{i2} \cdot f_{i3} \cdot f_{i4} \cdot f_{i5}$$

This is a $\Sigma\Pi\Sigma\Pi^{[d/2]}$ circuit. We want to obtain a $\Sigma\Pi\Sigma\Pi^{[t]}$ circuit.
Each f_{ij} is also some $[u : v]$. Keep expanding terms of degree more than t.

[SV]'s proof

Let's start with [VSBR]

$$f = \sum_{i=1}^{s} \left(\sum_{j=1}^{s} g_{j1} \cdots g_{j5} \right) \cdot f_{i2} \cdot f_{i3} \cdot f_{i4} \cdot f_{i5}$$

This is a $\Sigma\Pi\Sigma\Pi^{[d/2]}$ circuit. We want to obtain a $\Sigma\Pi\Sigma\Pi^{[t]}$ circuit. Each f_{ij} is also some $[u : v]$. Keep expanding terms of degree more than t.

[SV]'s proof

Let's start with [VSBR]

$$f \quad = \quad \sum_{i=1}^{s^2} f_{i1} \cdots f_{i9}$$

This is a $\Sigma\Pi\Sigma\Pi^{[d/2]}$ circuit. We want to obtain a $\Sigma\Pi\Sigma\Pi^{[t]}$ circuit. Each f_{ij} is also some $[u : v]$. Keep expanding terms of degree more than t.

[SV]'s proof

Let's start with [VSBR]

$$f = \sum_{i=1}^{s^3} f_{i1} \cdots f_{i13}$$

This is a $\Sigma\Pi\Sigma\Pi^{[d/2]}$ circuit. We want to obtain a $\Sigma\Pi\Sigma\Pi^{[t]}$ circuit. Each f_{ij} is also some $[u:v]$. Keep expanding terms of degree more than t.

[SV]'s proof

Let's start with [VSBR]

$$f = \sum_{i=1}^{s^4} f_{i1} \cdots f_{i17}$$

This is a $\Sigma\Pi\Sigma\Pi^{[d/2]}$ circuit. We want to obtain a $\Sigma\Pi\Sigma\Pi^{[t]}$ circuit.
Each f_{ij} is also some $[u : v]$. Keep expanding terms of degree more than t.

[SV]'s proof

Let's start with [VSBR]

$$f \quad = \quad \sum_{i=1}^{s^4} f_{i1} \cdots f_{i17}$$

This is a $\Sigma\Pi\Sigma\Pi^{[d/2]}$ circuit. We want to obtain a $\Sigma\Pi\Sigma\Pi^{[t]}$ circuit.
Each f_{ij} is also some $[u : v]$. Keep expanding terms of degree more than t.

How many iterations until all degrees are at most t?

Number of iterations

$$g = \sum_{j=1}^{s} g_{j1} \cdot g_{j2} \cdot g_{j3} \cdot g_{j4} \cdot g_{j5}$$

Number of iterations

$$g = \sum_{j=1}^{s} g_{j1} \cdot g_{j2} \cdot g_{j3} \cdot g_{j4} \cdot g_{j5}$$

Observation

In each summand, at least two terms have degree at least $t/8$.

Number of iterations

$$g \quad = \quad \sum_{j=1}^{s} \underbrace{g_{j1}}_{\geq t/5} \cdot g_{j2} \cdot g_{j3} \cdot g_{j4} \cdot g_{j5}$$

Observation

In each summand, at least two terms have degree at least $t/8$.

Number of iterations

$$g = \sum_{j=1}^{s} \underbrace{g_{j1}}_{\geq t/5} \cdot \underbrace{g_{j2}}_{\geq t/8} \cdot g_{j3} \cdot g_{j4} \cdot g_{j5}$$

Observation

In each summand, at least two terms have degree at least $t/8$.

Number of iterations

$$g = \sum_{j=1}^{s} \underbrace{g_{j1}}_{\geq t/5} \cdot \underbrace{g_{j2}}_{\geq t/8} \cdot g_{j3} \cdot g_{j4} \cdot g_{j5}$$

Observation

In each summand, at least two terms have degree at least $t/8$.

How many factors of degree at least $t/8$?

$$f = \sum_{i=1}^{s} f_{i1} \cdot f_{i2} \cdot f_{i3} \cdot f_{i4} \cdot f_{i5}$$

Number of iterations

$$g \quad = \quad \sum_{j=1}^{s} \underbrace{g_{j1}}_{\geq t/5} \cdot \underbrace{g_{j2}}_{\geq t/8} \cdot g_{j3} \cdot g_{j4} \cdot g_{j5}$$

Observation

In each summand, at least two terms have degree at least $t/8$.

How many factors of degree at least $t/8$?

$$f \quad = \quad \sum_{i=1}^{s} f_{i1} \cdot f_{i2} \cdot f_{i3} \cdot f_{i4} \cdot f_{i5}$$

Number of iterations

$$g \quad = \quad \sum_{j=1}^{s} \underbrace{g_{j1}}_{\geq t/5} \cdot \underbrace{g_{j2}}_{\geq t/8} \cdot g_{j3} \cdot g_{j4} \cdot g_{j5}$$

Observation

In each summand, at least two terms have degree at least $t/8$.

How many factors of degree at least $t/8$?

$$f \quad = \quad \sum_{i=1}^{s} \left(\sum_{j=1}^{s} g_{j1} g_{j2} g_{j3} g_{j4} g_{j5} \right) \cdot f_{i2} \cdot f_{i3} \cdot f_{i4} \cdot f_{i5}$$

Number of iterations

$$g \quad = \quad \sum_{j=1}^{s} \underbrace{g_{j1}}_{\geq t/5} \cdot \underbrace{g_{j2}}_{\geq t/8} \cdot g_{j3} \cdot g_{j4} \cdot g_{j5}$$

Observation

In each summand, at least two terms have degree at least $t/8$.

How many factors of degree at least $t/8$?

$$f \quad = \quad \sum_{i=1}^{s} \left(\sum_{j=1}^{s} g_{j1} g_{j2} g_{j3} g_{j4} g_{j5} \right) \cdot f_{i2} \cdot f_{i3} \cdot f_{i4} \cdot f_{i5}$$

Number of iterations

$$g \quad = \quad \sum_{j=1}^{s} \underbrace{g_{j1}}_{\geq t/5} \cdot \underbrace{g_{j2}}_{\geq t/8} \cdot g_{j3} \cdot g_{j4} \cdot g_{j5}$$

Observation

In each summand, at least two terms have degree at least $t/8$.

How many factors of degree at least $t/8$?

$$f \quad = \quad \sum_{i=1}^{s^2} f_{i1} \cdot f_{i12} \cdot f_{i3} \cdot f_{i4} \cdots f_{i9}$$

Number of iterations

$$g \quad = \quad \sum_{j=1}^{s} \underbrace{g_{j1}}_{\geq t/5} \cdot \underbrace{g_{j2}}_{\geq t/8} \cdot g_{j3} \cdot g_{j4} \cdot g_{j5}$$

Observation

In each summand, at least two terms have degree at least $t/8$.

How many factors of degree at least $t/8$? At most $8d/t$.

$$f \quad = \quad \sum_{i=1}^{s^2} f_{i1} \cdot f_{i12} \cdot f_{i3} \cdot f_{i4} \cdots f_{i9}$$

Number of iterations

$$g \quad = \quad \sum_{j=1}^{s} \underbrace{g_{j1}}_{\geq t/5} \cdot \underbrace{g_{j2}}_{\geq t/8} \cdot g_{j3} \cdot g_{j4} \cdot g_{j5}$$

Observation

In each summand, at least two terms have degree at least $t/8$.

How many factors of degree at least $t/8$? At most $8d/t$.

$$f \quad = \quad \sum_{i=1}^{s^2} f_{i1} \cdot f_{i12} \cdot f_{i3} \cdot f_{i4} \cdots f_{i9}$$

$$\sum^{s^{O(\sqrt{d})}} \prod^{O(\sqrt{d})} \sum^{s^{O(\sqrt{d})}} \prod^{\sqrt{d}}$$

Final $\Sigma\Pi\Sigma\Pi^{[t]}$ circuit has top fan-in at most $s^{O(d/t)}$.

A better starting point?

Recall

If f has a sized-s circuit, then it has a $\Sigma\Pi\Sigma\Pi^{[\sqrt{d}]}$ of size $s^{O(\sqrt{d})}$.

$$f \quad = \quad \sum_{i=1}^{s} f_{i1} \cdot f_{i2} \cdot f_{i3} \cdot f_{i4} \cdot f_{i5}$$

If we start with a homogeneous formula, can we do better?

A better starting point?

Recall

If f has a sized-s circuit, then it has a $\Sigma\Pi\Sigma\Pi^{[\sqrt{d}]}$ of size $s^{O(\sqrt{d})}$.

$$f \quad = \quad \sum_{i=1}^{s} f_{i1} \cdot f_{i2} \cdot f_{i3} \cdot f_{i4} \cdot f_{i5}$$

If we start with a homogeneous formula, can we do better?
[Hrubes-Yehudayoff]: Yes!

A better starting point?

Recall

If f has a sized-s circuit, then it has a $\Sigma\Pi\Sigma\Pi^{[\sqrt{d}]}$ of size $s^{O(\sqrt{d})}$.

$$f \quad = \quad \sum_{i=1}^{s} f_{i1} \cdot f_{i2} \cdot f_{i3} \cdot f_{i4} \cdot f_{i5}$$

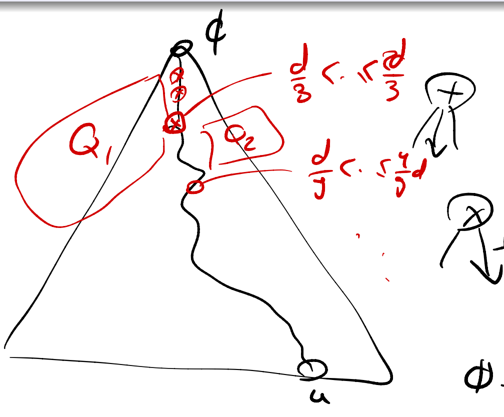
If we start with a homogeneous formula, can we do better?
[Hrubes-Yehudayoff]: Yes!

Lemma ([Hrubes-Yehudayoff])

$$f \quad = \quad \sum_{i=1}^{s} f_{i1} \cdot f_{i2} \cdots f_{i\ell} \qquad \textit{with } \left(\frac{1}{3}\right)^j \cdot d < \deg(f_{ij}) \leq \left(\frac{2}{3}\right)^j \cdot d$$

Lemma ([Hrubes-Yehudayoff])

$$f \quad = \quad \sum_{i=1}^{s} f_{i1} \cdot f_{i2} \cdots f_{i\ell} \qquad \textit{with} \left(\frac{1}{3}\right)^{j} \cdot d < \deg(f_{ij}) \leq \left(\frac{2}{3}\right)^{j} \cdot d$$



larger degree

$$\Phi = \Phi_{u \leftarrow 0} + Q_1 \cdot Q_2 \cdots Q_p \cdot u$$

A better starting point?

Recall

If f has a sized-s circuit, then it has a $\Sigma\Pi\Sigma\Pi^{[\sqrt{d}]}$ of size $s^{O(\sqrt{d})}$.

Theorem (Saptharishi?)

If f has a homogeneous sized-s formula,
then it has a homogeneous $\Sigma\Pi^{[\Omega(d \log t/t)]}\Sigma\Pi^{[\sqrt{t}]}$ t.

A better starting point?

Recall

If f has a sized-s circuit, then it has a $\Sigma\Pi\Sigma\Pi^{[\sqrt{d}]}$ of size $s^{O(\sqrt{d})}$.

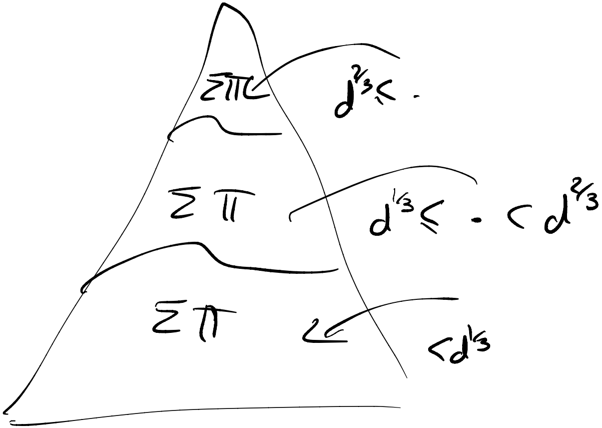
Theorem (Saptharishi?)

If f has a homogeneous sized-s formula, then it has a homogeneous $\Sigma\Pi^{[\Omega(d \log t/t)]}\Sigma\Pi^{[\sqrt{t}]}$.

Theorem (KOS)

If f has a syntactically multilinear sized-s circuit, then it has a $\Sigma\Pi\Sigma\Pi$ of size $2^{O(\sqrt{N \log s})}$.

Generalization to homogeneous depth-2Δ



$\Sigma\Pi\Sigma\Pi\Sigma\Pi$

$s^{O(d^{1/3})}$

Generalization to homogeneous depth-2Δ

Theorem

If f has a sized-s circuit, then it has a depth-2Δ $\underbrace{\Sigma\Pi^{[O(d^{1/\Delta})]}\Sigma\Pi^{[O(d^{1/\Delta})]}\dots\Sigma\Pi^{[O(d^{1/\Delta})]}}_{\left(\Sigma\Pi^{[O(d^{1/\Delta})]}\right)^{\Delta}}$ *of size* $s^{O(\Delta\cdot d^{1/\Delta})}$.

Generalization to homogeneous depth-2Δ

Theorem

If f has a sized-s circuit,
then it has a depth-2Δ $\Sigma\Pi^{[O(d^{1/\Delta})]}\Sigma\Pi^{[O(d^{1/\Delta})]}\dots\Sigma\Pi^{[O(d^{1/\Delta})]}$ of size $s^{O(\Delta\cdot d^{1/\Delta})}$.

Theorem

If f has a sized-$\mathrm{poly}(N)$ syntactically multilinear circuit,
then it has a $(\Sigma\Pi)^{\Delta}$ of size $s^{O\left(\Delta\cdot(n/\log s)^{1/\Delta}\right)}$.

Reduction to Depth-3 Circuits

(or, "can we do better if we allow the final circuit to be highly inhomogeneous?")

Road map [GKKS]

$$\sum \prod^{\sqrt{d}} \sum \prod^{\sqrt{d}}$$
circuits

$$\downarrow$$

$$\sum \bigwedge^{\sqrt{d}} \sum \bigwedge^{\sqrt{d}} \sum$$
circuits

$$\downarrow$$

$$\sum \prod \sum$$
circuits

Road map [GKKS]

$$\sum \prod^{\sqrt{d}} \sum \prod^{\sqrt{d}}$$
circuits

$$\Big\downarrow \text{ App. of Ryser's formula}$$

$$\sum \bigwedge^{\sqrt{d}} \sum \bigwedge^{\sqrt{d}} \sum$$
circuits

$$\Big\downarrow$$

$$\sum \prod \sum$$
circuits

Road map [GKKS]

$$\sum \prod^{\sqrt{d}} \sum \prod^{\sqrt{d}}$$

circuits

$$\Big\downarrow \text{App. of Ryser's formula}$$

$$\sum \bigwedge^{\sqrt{d}} \sum \bigwedge^{\sqrt{d}} \sum$$

circuits

$$\Big\downarrow \text{[Saxena]'s duality trick}$$

$$\sum \prod \sum$$

circuits

Road map [GKKS]

$$\sum \prod^{\sqrt{d}} \sum \prod^{\sqrt{d}} \text{ circuits}$$

Only over $\mathbb{Q}$,$\mathbb{R}$ etc. $\Big\downarrow$ App. of Ryser's formula

$$\sum \bigwedge^{\sqrt{d}} \sum \bigwedge^{\sqrt{d}} \sum \text{ circuits}$$

Heavily non-homogeneous $\Big\downarrow$ [Saxena]'s duality trick

$$\sum \prod \sum \text{ circuits}$$

$$\prod^{d} \Big\downarrow \sum^{2^d} \bigwedge^{d} \sum^{d}$$

$$\bigwedge^{a} \sum^{m} \bigwedge^{b} \Big\downarrow \sum^{\mathrm{poly}(a,m,b)} \prod^{[a\,m\,b]} \sum^{[2]}$$

Step 1: $\Pi^{[d]}$ to $\Sigma^{[2^d]}\wedge^{[d]}\Sigma^{[d]}$

Step 1: $\Pi^{[d]}$ to $\Sigma^{[2^d]}\wedge^{[d]}\Sigma^{[d]}$

Recall Ryser's formula:

$$\mathrm{Perm}_d \begin{bmatrix} x_{11} & \cdots & x_{1d} \\ \vdots & \ddots & \vdots \\ x_{d1} & \cdots & x_{dd} \end{bmatrix} = \sum_{S \subseteq [d]} (-1)^{d-|S|} \prod_{i=1}^{d} \sum_{j \in S} x_{ij}$$

Step 1: $\Pi^{[d]}$ to $\Sigma^{[2^d]}\wedge^{[d]}\Sigma^{[d]}$

Recall Ryser's formula:

$$\mathrm{Perm}_d \begin{bmatrix} x_1 & \cdots & x_d \\ \vdots & \ddots & \vdots \\ x_1 & \cdots & x_d \end{bmatrix} = \sum_{S\subseteq[d]} (-1)^{d-|S|} \prod_{i=1}^{d} \sum_{j\in S} x_j$$

Step 1: $\Pi^{[d]}$ to $\Sigma^{[2^d]}\wedge^{[d]}\Sigma^{[d]}$

Recall Ryser's formula:

$$\mathrm{Perm}_d \begin{bmatrix} x_1 & \dots & x_d \\ \vdots & \ddots & \vdots \\ x_1 & \dots & x_d \end{bmatrix} = \sum_{S\subseteq[d]} (-1)^{d-|S|} \left(\sum_{j\in S} x_j \right)^d$$

Step 1: $\Pi^{[d]}$ to $\Sigma^{[2^d]}\wedge^{[d]}\Sigma^{[d]}$

Recall Ryser's formula:

$$d! \cdot x_1 \ldots x_d \quad = \quad \sum_{S \subseteq [d]} (-1)^{d-|S|} \left(\sum_{j \in S} x_j \right)^d$$

Step 1: $\Pi^{[d]}$ to $\Sigma^{[2^d]}\wedge^{[d]}\Sigma^{[d]}$

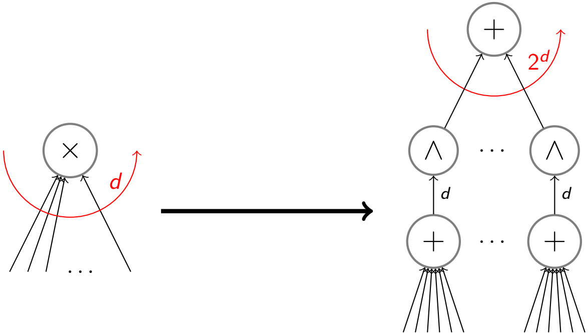
[Fischer]:

$$d! \cdot x_1 \dots x_d \quad = \quad \sum_{S \subseteq [d]} (-1)^{d-|S|} \left(\sum_{j \in S} x_j \right)^d$$

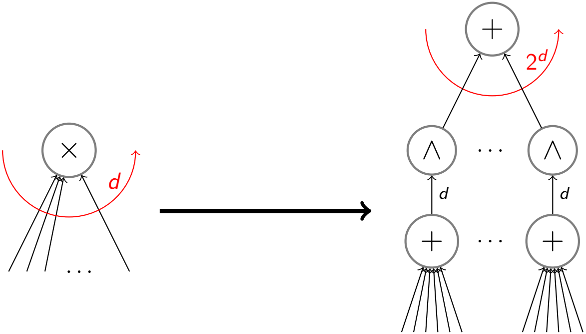
Step 1: $\Pi^{[d]}$ to $\Sigma^{[2^d]}\wedge^{[d]}\Sigma^{[d]}$

[Fischer]:

$$d! \cdot x_1 \ldots x_d \quad = \quad \sum_{S \subseteq [d]} (-1)^{d-|S|} \left(\sum_{j \in S} x_j \right)^d$$

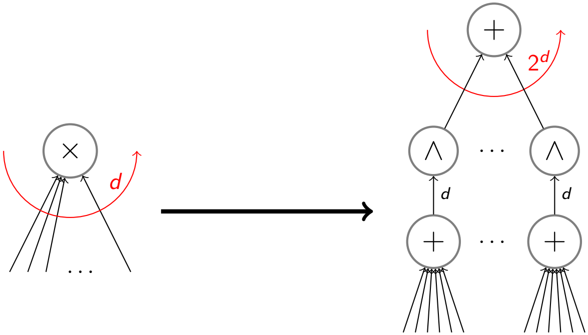


Step 1: $\Pi^{[d]}$ to $\Sigma^{[2^d]}\wedge^{[d]}\Sigma^{[d]}$



$$\prod^{d} \longrightarrow \sum^{2^d}\bigwedge^{d}\sum^{d}$$

Step 1: $\Pi^{[d]}$ to $\Sigma^{[2^d]}\wedge^{[d]}\Sigma^{[d]}$



$$\prod^{d} \longrightarrow \sum^{2^d}\bigwedge^{d}\sum^{d}$$

$$\sum\underbrace{\prod^{\sqrt{d}}}_{\sum^{2^{\sqrt{d}}}\bigwedge^{\sqrt{d}}\sum^{\sqrt{d}}}\sum\underbrace{\prod^{\sqrt{d}}}_{\sum^{2^{\sqrt{d}}}\bigwedge^{\sqrt{d}}\sum^{\sqrt{d}}} \text{ of size } s \longrightarrow \sum\bigwedge^{\sqrt{d}}\sum\bigwedge^{\sqrt{d}}\sum \text{ of size } 2^{O(\sqrt{d})}\cdot s$$

Road map

$$\sum \prod^{\sqrt{d}} \sum \prod^{\sqrt{d}}$$

circuits

$$\downarrow$$

$$\sum \bigwedge^{\sqrt{d}} \sum \bigwedge^{\sqrt{d}} \sum$$

circuits

$$\downarrow$$

$$\sum \prod \sum$$

circuits

Road map

$$\sum \prod^{\sqrt{d}} \sum \prod^{\sqrt{d}}$$

circuits

$$\downarrow$$

$$\sum \bigwedge^{\sqrt{d}} \sum \bigwedge^{\sqrt{d}} \sum$$

circuits

$$\downarrow$$

$$\sum \prod \sum$$

circuits

Step 2: $\wedge^{[a]}\Sigma^{[s]}\wedge^{[b]}$ to $\Sigma^{[\mathrm{poly}(s,a,b)]}\Pi^{[sbd]}\Sigma^{[2]}$

Step 2: $\wedge^{[a]}\sum^{[s]}\wedge^{[b]}$ to $\sum^{[\mathrm{poly}(s,a,b)]}\prod^{[sbd]}\sum^{[2]}$

$$T = \left(x_1^b + \cdots + x_s^b\right)^a$$

Step 2: $\wedge^{[a]}\Sigma^{[s]}\wedge^{[b]}$ to $\Sigma^{[\mathrm{poly}(s,a,b)]}\Pi^{[sbd]}\Sigma^{[2]}$

$$T \quad = \quad \left(x_1^b + \cdots + x_s^b\right)^a$$

Lemma ([Saxena])

There exists univariate polynomials f_{ij}'s of degree at most a such that

$$\ell^a = (y_1 + \cdots + y_s)^a \quad = \quad \sum_{i=1}^{O(sa^2)} \prod_{j=1}^{s} f_{ij}(x_j)$$

Step 2: $\wedge^{[a]}\Sigma^{[s]}\wedge^{[b]}$ to $\Sigma^{[\mathrm{poly}(s,a,b)]}\Pi^{[sbd]}\Sigma^{[2]}$

$$T \quad = \quad \left(x_1^b + \cdots + x_s^b\right)^a$$

Lemma ([Saxena])

There exists univariate polynomials f_{ij}'s of degree at most a such that

$$\ell^a = (y_1 + \cdots + y_s)^a \quad = \quad \sum_{i=1}^{O(sa^2)} \prod_{j=1}^{s} f_{ij}(x_j)$$

Sketch of a proof by Gupta-Forbes-Shpilka

$$P_{\mathbf{y}}(t) \;=\; (1 + y_1 t)\ldots(1 + y_s t) = \;\; 1 + \ell t + (\text{higher degree terms}) \to s$$

Step 2: $\wedge^{[a]}\Sigma^{[s]}\wedge^{[b]}$ to $\Sigma^{[\mathrm{poly}(s,a,b)]}\Pi^{[sbd]}\Sigma^{[2]}$

$$T \quad = \quad \left(x_1^b + \cdots + x_s^b\right)^a$$

Lemma ([Saxena])

There exists univariate polynomials f_{ij}'s of degree at most a such that

$$\ell^a = (y_1 + \cdots + y_s)^a \quad = \quad \sum_{i=1}^{O(sa^2)} \prod_{j=1}^{s} f_{ij}(x_j)$$

Sketch of a proof by Gupta-Forbes-Shpilka

$$P_{\mathbf{y}}(t) - 1 = \qquad \ell t \quad + \quad \text{(higher degree terms)} \to s$$

Step 2: $\wedge^{[a]}\Sigma^{[s]}\wedge^{[b]}$ to $\Sigma^{[\mathrm{poly}(s,a,b)]}\Pi^{[sbd]}\Sigma^{[2]}$

$$T \quad = \quad \left(x_1^b + \cdots + x_s^b\right)^a$$

Lemma ([Saxena])

There exists univariate polynomials f_{ij}'s of degree at most a such that

$$\ell^a = (y_1 + \cdots + y_s)^a \quad = \quad \sum_{i=1}^{O(sa^2)} \prod_{j=1}^{s} f_{ij}(x_j)$$

Sketch of a proof by Gupta-Forbes-Shpilka

$$(P_{\mathbf{y}}(t) - 1)^a = \qquad \ell^a t^a \quad + \quad \text{(higher degree terms)} \to sa$$

Step 2: $\wedge^{[a]}\Sigma^{[s]}\wedge^{[b]}$ to $\Sigma^{[\mathrm{poly}(s,a,b)]}\Pi^{[sbd]}\Sigma^{[2]}$

$$T \quad = \quad \left(x_1^b + \cdots + x_s^b\right)^a$$

Lemma ([Saxena])

There exists univariate polynomials f_{ij}'s of degree at most a such that

$$\ell^a = (y_1 + \cdots + y_s)^a \quad = \quad \sum_{i=1}^{O(sa^2)} \prod_{j=1}^{s} f_{ij}(x_j)$$

Sketch of a proof by Gupta-Forbes-Shpilka

$$(P_{\mathbf{y}}(t) - 1)^a = \qquad \ell^a t^a \quad + \quad \text{(higher degree terms)} \rightarrow sa$$

Interpolate!

$P_y(t)^\ell$ $(0 \le \ell \le a+1)$

$(P_{\mathbf{y}}(t) - 1)^a$ expanded is a sum of $(a+1)$ product of univariates. □

Step 2: $\wedge^{[a]}\Sigma^{[s]}\wedge^{[b]}$ to $\Sigma^{[\mathrm{poly}(s,a,b)]}\Pi^{[sbd]}\Sigma^{[2]}$

$$T = \left(x_1^b + \cdots + x_s^b\right)^a$$

$$(y_1 + \cdots + y_s)^a = \sum_i^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} f_{ij}(y_j)$$

Step 2: $\wedge^{[a]}\Sigma^{[s]}\wedge^{[b]}$ to $\Sigma^{[\mathrm{poly}(s,a,b)]}\Pi^{[sbd]}\Sigma^{[2]}$

$$T = \left(x_1^b + \cdots + x_s^b\right)^a$$

$$\left(x_1^b + \cdots + x_s^b\right)^a = \sum_{i}^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} f_{ij}\left(x_j^b\right)$$

Step 2: $\wedge^{[a]}\Sigma^{[s]}\wedge^{[b]}$ to $\Sigma^{[\mathrm{poly}(s,a,b)]}\Pi^{[sbd]}\Sigma^{[2]}$

$$T \quad = \quad \left(x_1^b + \cdots + x_s^b\right)^a$$

$$\begin{aligned}\left(x_1^b + \cdots + x_s^b\right)^a &= \sum_i^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} f_{ij}\left(x_j^b\right) \\ &= \sum_i^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} \tilde{f}_{ij}(x_j)\end{aligned}$$

where $\tilde{f}_{ij}(t) \; := \; f_{ij}(t^{\sqrt{d}})$

Step 2: $\wedge^{[a]}\Sigma^{[s]}\wedge^{[b]}$ to $\Sigma^{[\mathrm{poly}(s,a,b)]}\Pi^{[sbd]}\Sigma^{[2]}$

$$T \quad = \quad \left(x_1^b + \cdots + x_s^b\right)^a$$

$$\begin{aligned}\left(x_1^b + \cdots + x_s^b\right)^a &= \sum_i^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} f_{ij}\left(x_j^b\right) \\ &= \sum_i^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} \tilde{f}_{ij}(x_j)\end{aligned}$$

Note that $\tilde{f}_{ij}(t)$ is a univariate polynomial

Step 2: $\wedge^{[a]}\Sigma^{[s]}\wedge^{[b]}$ to $\Sigma^{[\mathrm{poly}(s,a,b)]}\Pi^{[sbd]}\Sigma^{[2]}$

$$T = \left(x_1^b + \cdots + x_s^b\right)^a$$

$$\left(x_1^b + \cdots + x_s^b\right)^a = \sum_i^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} f_{ij}\left(x_j^b\right) = \sum_i^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} \tilde{f}_{ij}(x_j)$$

Note that $\tilde{f}_{ij}(t)$ is a univariate polynomial that can be factorized over $\mathbb{C}$:

$$\tilde{f}_{ij}(t) = \prod_{k=1}^{ab}(t - \zeta_{ijk})$$

Step 2: $\wedge^{[a]}\Sigma^{[s]}\wedge^{[b]}$ to $\Sigma^{[\mathrm{poly}(s,a,b)]}\Pi^{[sbd]}\Sigma^{[2]}$

$$T = \left(x_1^b + \cdots + x_s^b\right)^a$$

$$\begin{aligned}\left(x_1^b + \cdots + x_s^b\right)^a &= \sum_i^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} f_{ij}\left(x_j^b\right)\\ &= \sum_i^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} \tilde{f}_{ij}(x_j)\end{aligned}$$

Note that $\tilde{f}_{ij}(t)$ is a univariate polynomial that can be factorized over $\mathbb{C}$:

$$\tilde{f}_{ij}(\ell_j) = \prod_{k=1}^{ab}(\ell_j - \zeta_{ijk})$$

Step 2: $\wedge^{[a]}\Sigma^{[s]}\wedge^{[b]}$ to $\Sigma^{[\mathrm{poly}(s,a,b)]}\Pi^{[sbd]}\Sigma^{[2]}$

$$T \quad = \quad \left(x_1^b + \cdots + x_s^b\right)^a$$

$$\begin{aligned}
\left(x_1^b + \cdots + x_s^b\right)^a &= \sum_i^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} f_{ij}\left(x_j^b\right) \\
&= \sum_i^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} \tilde{f}_{ij}(x_j) \\
&= \sum_i^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} \prod_{k=1}^{ab} (x_j - \zeta_{ijk})
\end{aligned}$$

Step 2: $\wedge^{[a]}\Sigma^{[s]}\wedge^{[b]}$ to $\Sigma^{[\mathrm{poly}(s,a,b)]}\Pi^{[sbd]}\Sigma^{[2]}$

$$T = \left(x_1^b + \cdots + x_s^b\right)^a$$

$$\begin{aligned}
\left(x_1^b + \cdots + x_s^b\right)^a &= \sum_i^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} f_{ij}\left(x_j^b\right) \\
&= \sum_i^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} \tilde{f}_{ij}(x_j) \\
&= \sum_i^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} \prod_{k=1}^{ab} (x_j - \zeta_{ijk})
\end{aligned}$$

... a $\Sigma\Pi\Sigma$ circuit of $\mathrm{poly}(s, a, b)$ size.

Step 2: $\wedge^{[a]}\Sigma^{[s]}\wedge^{[b]}$ to $\Sigma^{[\mathrm{poly}(s,a,b)]}\Pi^{[sbd]}\Sigma^{[2]}$

$$T = \left(x_1^b + \cdots + x_s^b\right)^a$$

$$\begin{aligned}
\left(x_1^b + \cdots + x_s^b\right)^a &= \sum_i^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} f_{ij}\left(x_j^b\right) \\
&= \sum_i^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} \tilde{f}_{ij}(x_j) \\
&= \sum_i^{\mathrm{poly}(s,a)} \prod_{j=1}^{s} \prod_{k=1}^{ab} (x_j - \zeta_{ijk})
\end{aligned}$$

... a $\Sigma\Pi\Sigma$ circuit of poly(s, a, b) size and degree sab.

Putting it together

general circuit
of size s

Putting it together

general circuit of size s $\longrightarrow$ $\sum\prod^{\sqrt{d}}\sum\prod^{\sqrt{d}}$ circuit of size $s^{O(\sqrt{d})}$

Putting it together

$$
\begin{tikzcd}
\begin{array}{c}\text{general circuit}\\ \text{of size } s\end{array} \arrow[r] & \begin{array}{c}\sum\prod^{\sqrt{d}}\sum\prod^{\sqrt{d}} \text{ circuit}\\ \text{of size } s^{O(\sqrt{d})}\end{array} \arrow[d] \\
 & \begin{array}{c}\sum\bigwedge^{\sqrt{d}}\sum\bigwedge^{\sqrt{d}}\sum \text{ circuits}\\ \text{of size } s^{O(\sqrt{d})}\cdot 2^{O(\sqrt{d})}\end{array}
\end{tikzcd}
$$

Putting it together

$$
\begin{tikzcd}
\begin{array}{c}\text{general circuit}\\ \text{of size } s\end{array} \arrow[r] & \begin{array}{c}\sum\prod^{\sqrt{d}}\sum\prod^{\sqrt{d}} \text{ circuit}\\ \text{of size } s^{O(\sqrt{d})}\end{array} \arrow[d] \\
 & \begin{array}{c}\sum\bigwedge^{\sqrt{d}}\sum\bigwedge^{\sqrt{d}}\sum \text{ circuits}\\ \text{of size } s^{O(\sqrt{d})}\end{array}
\end{tikzcd}
$$

Putting it together

$$
\begin{tikzcd}
\begin{array}{c}\text{general circuit} \\ \text{of size } s\end{array} \arrow[r] & \begin{array}{c}\sum\prod^{\sqrt{d}}\sum\prod^{\sqrt{d}} \text{ circuit} \\ \text{of size } s^{O(\sqrt{d})}\end{array} \arrow[d] \\
\begin{array}{c}\sum\prod\sum \text{ circuits} \\ \text{of size } s^{O(\sqrt{d})}\end{array} & \begin{array}{c}\sum\bigwedge^{\sqrt{d}}\sum\bigwedge^{\sqrt{d}}\sum \text{ circuits} \\ \text{of size } s^{O(\sqrt{d})}\end{array} \arrow[l]
\end{tikzcd}
$$

Putting it together

$$\begin{tikzcd}
\begin{array}{c}\text{general circuit}\\ \text{of size } s\end{array} \arrow[r] & \begin{array}{c}\sum\prod^{\sqrt{d}}\sum\prod^{\sqrt{d}} \text{ circuit}\\ \text{of size } s^{O(\sqrt{d})}\end{array} \arrow[d] \\
\begin{array}{c}\sum\prod\sum \text{ circuits}\\ \text{of size } s^{O(\sqrt{d})}\end{array} & \begin{array}{c}\sum\bigwedge^{\sqrt{d}}\sum\bigwedge^{\sqrt{d}}\sum \text{ circuits}\\ \text{of size } s^{O(\sqrt{d})}\end{array} \arrow[l]
\end{tikzcd}$$

Question: Where should one try to prove lower bounds?

Putting it together

$$
\begin{tikzcd}
\text{general hom. circuit of size } s \arrow[r] & \sum \prod^{\sqrt{d}} \sum \prod^{\sqrt{d}} \text{ circuit of size } s^{O(\sqrt{d})} \arrow[d] \\
\sum \prod \sum \text{ circuits of size } s^{O(\sqrt{d})} & \sum \bigwedge^{\sqrt{d}} \sum \bigwedge^{\sqrt{d}} \sum \text{ circuits of size } s^{O(\sqrt{d})} \arrow[l]
\end{tikzcd}
$$

Question: Where should one try to prove lower bounds?

Putting it together

$$\begin{tikzcd} \text{general hom. circuit of size } s \arrow[r] & \sum\prod^{\sqrt{d}}\sum\prod^{\sqrt{d}} \text{ hom. circuit of size } s^{O(\sqrt{d})} \arrow[d] \\ \sum\prod\sum \text{ circuits of size } s^{O(\sqrt{d})} & \sum\bigwedge^{\sqrt{d}}\sum\bigwedge^{\sqrt{d}}\sum \text{ circuits of size } s^{O(\sqrt{d})} \arrow[l] \end{tikzcd}$$

Question: Where should one try to prove lower bounds?

Putting it together

$$
\begin{tikzcd}
\begin{array}{c}\text{general hom. circuit} \\ \text{of size } s\end{array} \arrow[r] & \begin{array}{c}\sum\prod^{\sqrt{d}}\sum\prod^{\sqrt{d}} \text{ hom. circuit} \\ \text{of size } s^{O(\sqrt{d})}\end{array} \arrow[d] \\
\begin{array}{c}\sum\prod\sum \text{ circuits} \\ \text{of size } s^{O(\sqrt{d})}\end{array} & \begin{array}{c}\sum\bigwedge^{\sqrt{d}}\sum\bigwedge^{\sqrt{d}}\sum \text{ hom. circuits} \\ \text{of size } s^{O(\sqrt{d})}\end{array} \arrow[l]
\end{tikzcd}
$$

Question: Where should one try to prove lower bounds?

Putting it together

$$\begin{tikzcd}
\text{general hom. circuit of size } s \arrow[r] & \sum\prod^{\sqrt{d}}\sum\prod^{\sqrt{d}} \text{ hom. circuit of size } s^{O(\sqrt{d})} \arrow[d] \\
\sum\prod\sum \textbf{ non-hom. circuits of size } s^{O(\sqrt{d})} & \sum\bigwedge^{\sqrt{d}}\sum\bigwedge^{\sqrt{d}}\sum \text{ hom. circuits of size } s^{O(\sqrt{d})} \arrow[l]
\end{tikzcd}$$

Question: Where should one try to prove lower bounds?

Other constants for the depth?

Recall

If f has a sized-s circuit,
then it has a depth-2Δ $\left(\Sigma\Pi^{[O(d^{1/\Delta})]}\right)^{\Delta}$ of size $s^{O(\Delta \cdot d^{1/\Delta})}$.

$\Sigma\Pi^{d^{1/3}}\Sigma\Pi^{d^{1/3}}\Sigma\Pi^{d^{1/3}}$ $\quad s^{O(d^{1/3})}$

$\Sigma\wedge^{d^{1/3}}\Sigma \quad \Sigma\wedge^{d^{1/3}}\Sigma$

$\Sigma\Pi\Sigma$

$\Sigma\Pi^{s^{O(d^{1/3})}}\Sigma\Pi^{d^{1/3}} \quad \mapsto s^{O(d^{1/3})}$

Other constants for the depth?

Recall

If f has a sized-s circuit,
then it has a depth-2Δ $\left(\Sigma\Pi^{[O(d^{1/\Delta})]}\right)^{\Delta}$ of size $s^{O(\Delta \cdot d^{1/\Delta})}$.

Theorem

If f has a sized-s circuit,
then it has a depth-p circuit of size $s^{O(p \cdot d^{1/(p-1)})}$.

Other constants for the depth?

Recall

If f has a sized-s circuit,
then it has a depth-2Δ $\left(\Sigma\Pi^{[O(d^{1/\Delta})]}\right)^{\Delta}$ of size $s^{O(\Delta\cdot d^{1/\Delta})}$.

Theorem

If f has a sized-s circuit,
then it has a depth-p circuit of size $s^{O(p\cdot d^{1/(p-1)})}$.

Corollary

- Det_n *has a* $\Sigma\Pi\Sigma\Pi$ *of size* $n^{O(\sqrt[3]{n})}$.
- $\mathrm{IMM}_{n,d}$ *has a* $\Sigma\Pi\Sigma\Pi$ *of size* $n^{O(\sqrt[3]{d})}$.
- *If* Perm_n *needs* $\Sigma\Pi\Sigma\Pi$ *of size* $n^{\omega(\sqrt[3]{n})}$, *then* $VP \neq VNP$.

Back to the homogeneization (case of constant depth)

- All gates compute *homogeneous polynomials*.
- Hence, no gate can compute polynomials of degree larger than output.
- For circuits and ABPs, homogeneity can be assumed without loss of generality.
  For formulas, probably not.
  For constant depth formulas, certainly not.

Back to the homogeneization (case of constant depth)

- All gates compute *homogeneous polynomials*.
- Hence, no gate can compute polynomials of degree larger than output.
- For circuits and ABPs, homogeneity can be assumed without loss of generality.
  For formulas, probably not.
  For constant depth formulas, certainly not.

What happens if we allow some subexponential blow up?

Back to the homogeneization (case of constant depth)

- All gates compute *homogeneous polynomials*.
- Hence, no gate can compute polynomials of degree larger than output.
- For circuits and ABPs, homogeneity can be assumed without loss of generality.
  For formulas, probably not.
  For constant depth formulas, certainly not.

Theorem (Raz)

If f computed by a formula of size s,
then it is computed by a homogeneous one of size $2^{O(d \log \log s)}$.

Back to the homogeneization (case of constant depth)

Theorem (GKKS)

If f computed by a circuit of size s and depth 3, then it is computed by a homogeneous one of size $\mathrm{poly}(s)2^{O(\sqrt{d})}$ and depth 5.

Back to the homogeneization (case of constant depth)

Theorem (GKKS)

If f computed by a circuit of size s and depth 3, then it is computed by a homogeneous one of size $\mathrm{poly}(s)2^{O(\sqrt{d})}$ and depth 5.

Theorem (LST)

If f computed by a circuit of size s and depth Γ, then it is computed by a homogeneous one of size $\mathrm{poly}(s)2^{O(\sqrt{d})}$ and depth 2Γ-1.

Thank you.