Depth reduction in arithmetic circuits

Ramprasad Saptharishi
Tel Aviv University

WACT
March 2015, Saarbrücken

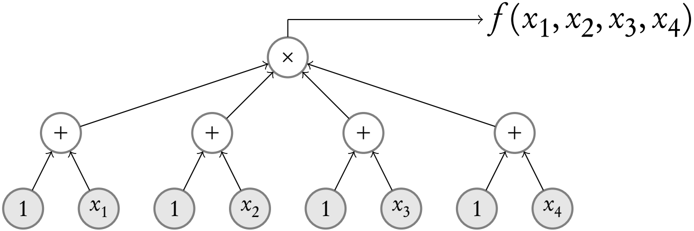
$$\begin{aligned} f(x_1,x_2,x_3,x_4) \quad = \quad & 1+x_1+x_2+x_3+x_4 \\ & +x_1x_2+x_1x_3+x_1x_4+x_2x_3+x_2x_4+x_3x_4 \\ & +x_2x_3x_4+x_1x_3x_4+x_1x_2x_4+x_1x_2x_3 \\ & +x_1x_2x_3x_4 \end{aligned}$$

Polynomials

$$
\begin{aligned}
f(x_1, x_2, x_3, x_4) \quad &= \quad 1 + x_1 + x_2 + x_3 + x_4 \\
&\quad + x_1x_2 + x_1x_3 + x_1x_4 + x_2x_3 + x_2x_4 + x_3x_4 \\
&\quad + x_2x_3x_4 + x_1x_3x_4 + x_1x_2x_4 + x_1x_2x_3 \\
&\quad + x_1x_2x_3x_4 \\
\\
&= \quad (1 + x_1)(1 + x_2)(1 + x_3)(1 + x_4)
\end{aligned}
$$

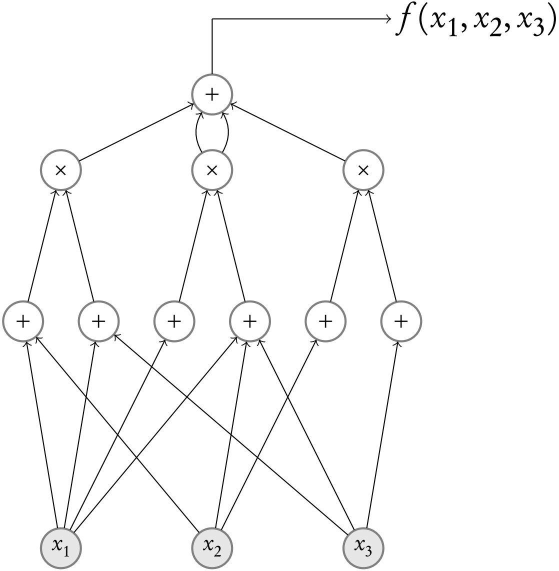
"How hard is it to compute a given n-variate degree d polynomial?"

Arithmetic Formulae

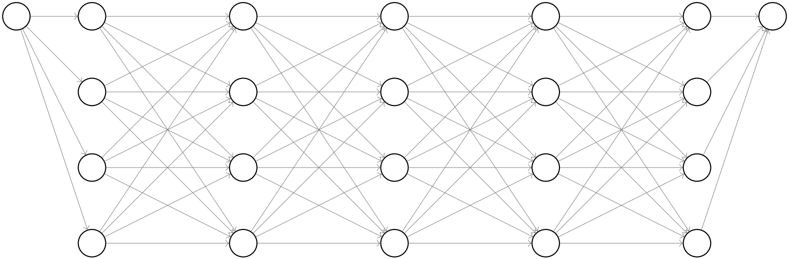


- Tree
- Leaves containing variables or constants

Arithmetic Circuits

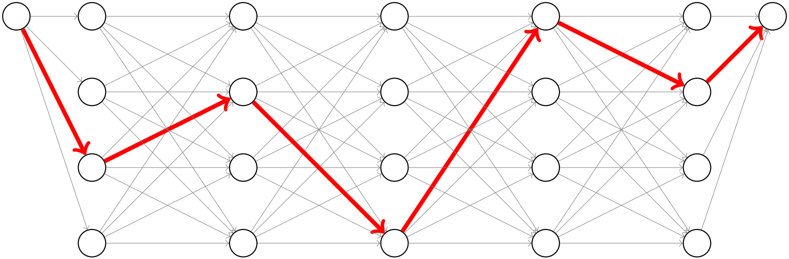


Arithmetic Branching Programs



A directed layered graph with a unique source node s and sink node t.
Each edge e holds a linear polynomial ℓ_e.

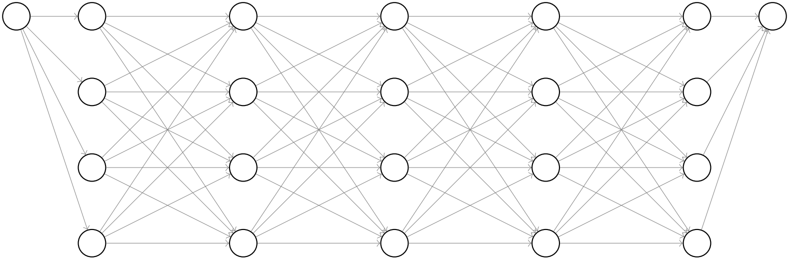
Arithmetic Branching Programs



A directed layered graph with a unique source node s and sink node t.
Each edge e holds a linear polynomial ℓ_e.

$$\mathrm{wt}(P) \quad = \quad \prod_{e \in P} \ell_e$$

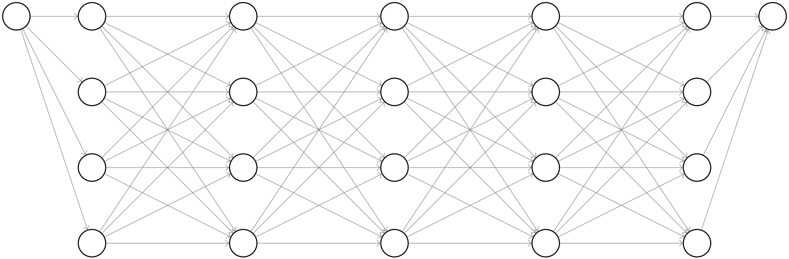
Arithmetic Branching Programs



A directed layered graph with a unique source node s and sink node t.
Each edge e holds a linear polynomial ℓ_e.

$$\text{Computes} \quad f \quad = \quad \sum_{P: s \rightsquigarrow t} \mathrm{wt}(P)$$

Arithmetic Branching Programs



A directed layered graph with a unique source node s and sink node t.
Each edge e holds a linear polynomial ℓ_e.

$$\text{Computes} \quad f \quad = \quad \sum_{P : s \rightsquigarrow t} \mathrm{wt}(P)$$

Equivalent to iterated matrix product

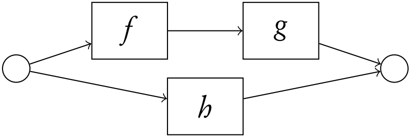
Relationship between these classes

Formulas $\subseteq$ ABP

Relationship between these classes

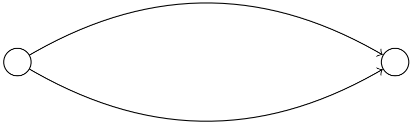
Formulas $\subseteq$ ABP

$f\,g + h$:



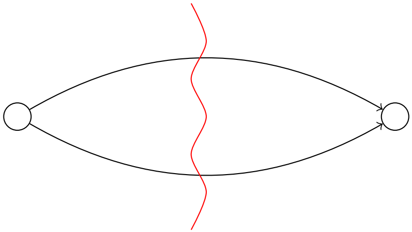
Relationship between these classes

Formulas $\subseteq$ ABP $\subseteq$ Circuits



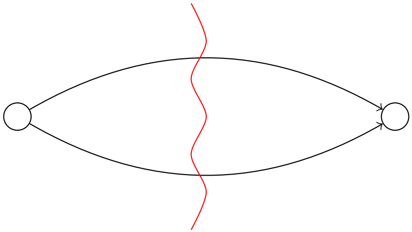
Relationship between these classes

Formulas $\subseteq$ ABP $\subseteq$ Circuits



Relationship between these classes

Formulas $\subseteq$ ABP $\subseteq$ Circuits



Savitch

Homogenization

"Thou shalt not compute polynomials of larger degree than thou needst"

Homogenization

"Thou shalt not compute polynomials of larger degree than thou needst"

- All gates compute *homogeneous polynomials*.

Homogenization

"Thou shalt not compute polynomials of larger degree than thou needst"

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

Homogenization

"Thou shalt not compute polynomials of larger degree than thou needst"

- 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

"Thou shalt not compute polynomials of larger degree than thou needst"

- 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

"Thou shalt not compute polynomials of larger degree than thou needst"

- 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.

Homogenization

"Thou shalt not compute polynomials of larger degree than thou needst"

- 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.

$$g = g_1 + g_2 \quad \longrightarrow \quad g^{(i)} = g_1^{(i)} + g_2^{(i)}$$

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

The illustrious siblings

$$\mathrm{Det}_n(x_{11},\ldots,x_{nn}) = \sum_{\sigma\in S_n} \mathrm{sign}(\sigma)\cdot x_{1\sigma(1)}\ldots x_{n\sigma(n)}$$

$$\mathrm{Perm}_n(x_{11},\ldots,x_{nn}) = \sum_{\sigma\in S_n} x_{1\sigma(1)}\ldots x_{n\sigma(n)}$$

The illustrious siblings

$$\mathrm{Det}_n(x_{11},\ldots,x_{nn}) = \sum_{\sigma\in S_n} \mathrm{sign}(\sigma)\cdot x_{1\sigma(1)}\cdots x_{n\sigma(n)}$$

$$\mathrm{Perm}_n(x_{11},\ldots,x_{nn}) = \sum_{\sigma\in S_n} x_{1\sigma(1)}\cdots x_{n\sigma(n)}$$

Det_n		VP
vs	[Valiant-79] ⟷	vs
Perm_n		VNP

Why?

Err…

Why?

Err...

- We would obtain new algorithms for polynomial identity testing.
- Identity testing has relevence to primality, results such as IP = PSPACE, bipartite matching etc. :-|

Why?

Err...

- We would obtain new algorithms for polynomial identity testing.
- Identity testing has relevence to primality, results such as IP = PSPACE, bipartite matching etc. :-|

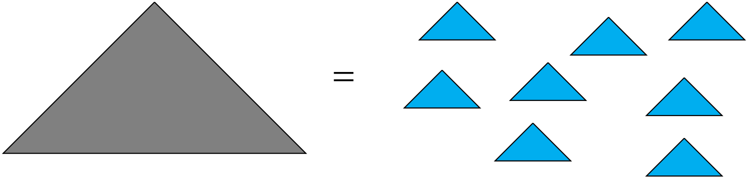
General roadmap for lower bounds

Four steps in most lower bound proofs

Step 1: Finding the right building blocks

Four steps in most lower bound proofs

Step 1: Finding the right building blocks



Meta Theorem 1

Every small circuit can be equivalently computed as a *sum* of few s

Four steps in most lower bound proofs

Step 1: Finding the right building blocks

Meta Theorem 1

Every small circuit can be equivalently computed as a *sum* of few s

Four steps in most lower bound proofs

Step 1: Finding the right building blocks

Meta Theorem 1

Every small circuit can be equivalently computed as a *sum* of few s

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 the right building blocks

Meta Theorem 1

Every small circuit can be equivalently computed as a *sum* of few s

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 the right building blocks

Meta Theorem 1

Every small circuit can be equivalently computed as a *sum* of few s

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

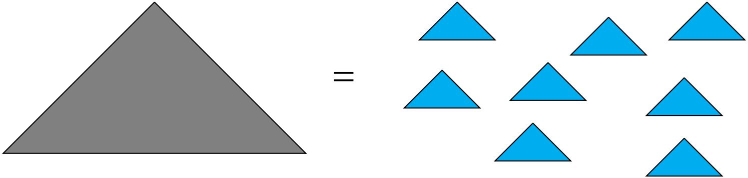
Meta Theorem 2

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

Step 4: Find a hay in the haystack

Four steps in most lower bound proofs

Step 1: Finding the right building blocks

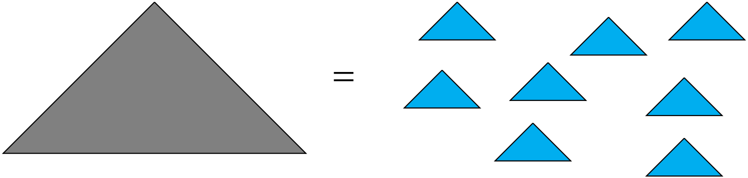


Meta Theorem 1

Every small circuit can be equivalently computed as a *sum* of few s

Four steps in most lower bound proofs

Step 1: Finding the right building blocks



Meta Theorem 1

Every small circuit can be equivalently computed as a *sum* of few s

Depth reduction

A short history of depth reduction

Circuit Class	Depth	Size	
Formulas	$O(\log s)$	poly(s)	[Brent]

A short history of depth reduction

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

A short history of depth reduction

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

A short history of depth reduction

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

A short history of depth reduction

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

A short history of depth reduction

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

A short history of depth reduction

Circuit 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]
		~~$s^{O(\sqrt{d}\log d)}$~~	[Koiran]
		$s^{O(\sqrt{d})}$	[Tavenas]
Circuits	3^*	$s^{O(\sqrt{d})}$	[Gupta-Kamath-Kayal-S]

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]

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

Other depth reductions in lower bounds

Multilinear formulas [Raz, Raz-Yehudayoff]

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

Homogeneous formulas [Hrubes-Yehudayoff]

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

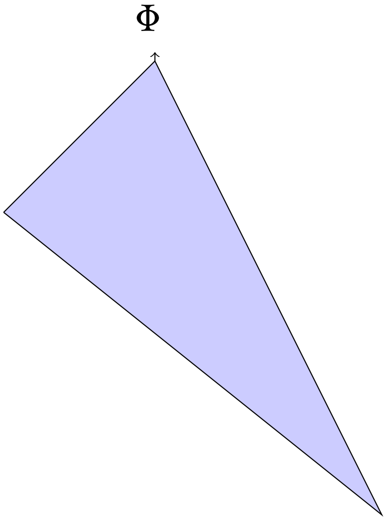
Homogeneous $\Sigma\Pi\Sigma\Pi$ [Kayal-Limaye-Saha-Srinivasan]

$$f = \Sigma\Pi\Sigma\Pi^{[\sqrt{d}]} + \sum_{i=1}^{s} m_i Q_i \quad , \quad \deg(m_i) \geq \sqrt{d}$$

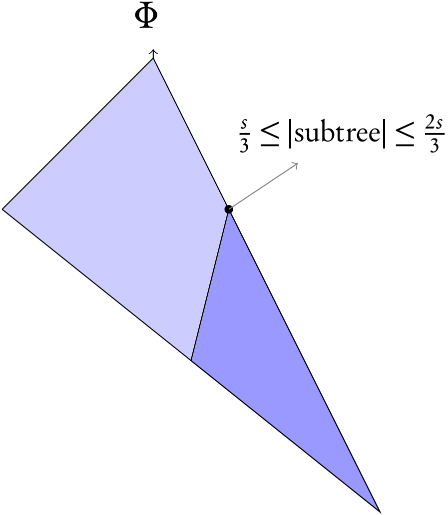
Plan

- Classical depth reductions of [Brent] and [VSBR].
- A *slightly* different proof of [Tavenas]
- Better building blocks for homogeneous formulas
- (depending on time) Reduction to depth three [GKKS2]

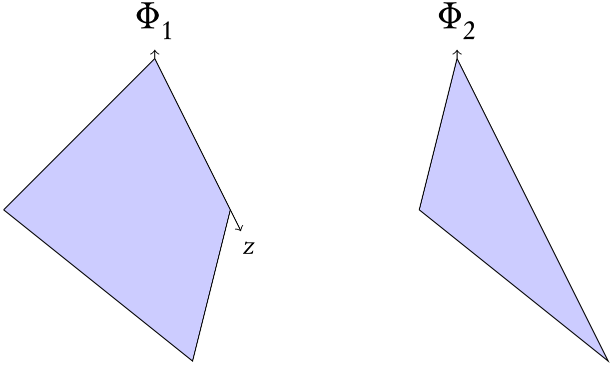
Depth reducing formulas



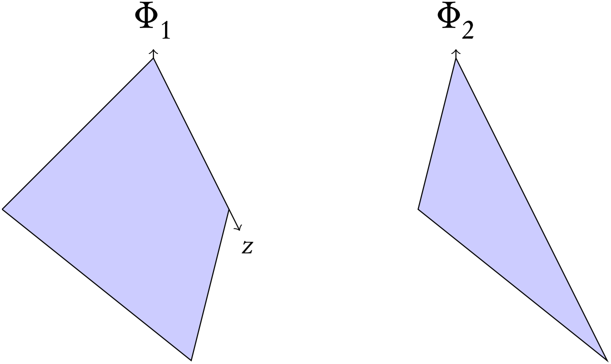
Depth reducing formulas



Depth reducing formulas

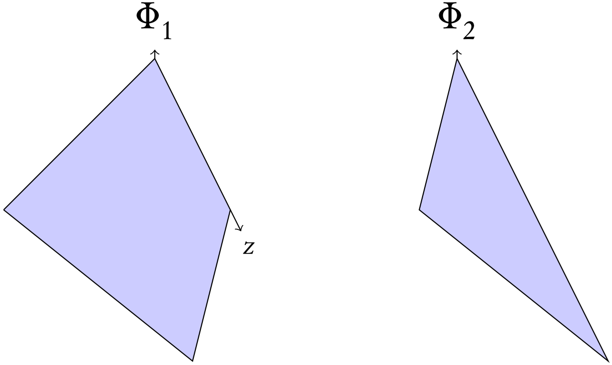


Depth reducing formulas



$$\Phi_1(z) \quad = \quad A \cdot z \quad + \quad B$$
$$\Phi \quad = \quad A \cdot \Phi_2 \quad + \quad B$$

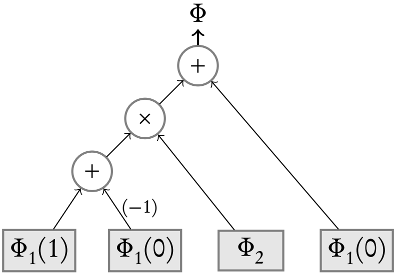
Depth reducing formulas



$$\Phi_1(z) = A \cdot z + B$$

$$\Phi = A \cdot \Phi_2 + B = (\Phi_1(1) - \Phi_1(0)) \cdot \Phi_2 + \Phi_1(0)$$

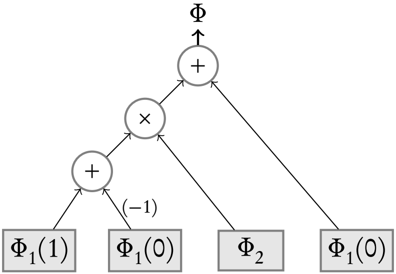
Depth reducing formulas



$$\Phi_1(z) = A \cdot z + B$$

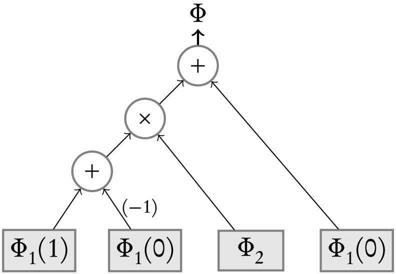
$$\Phi = A \cdot \Phi_2 + B = (\Phi_1(1) - \Phi_1(0)) \cdot \Phi_2 + \Phi_1(0)$$

Depth reducing formulas



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

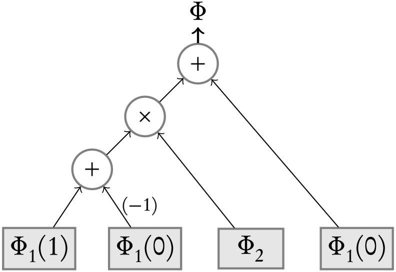
Depth reducing formulas



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

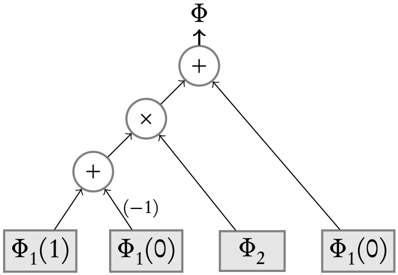
$$\mathrm{Depth}(s) \leq \mathrm{Depth}(2s/3) + O(1)$$

Depth reducing formulas



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

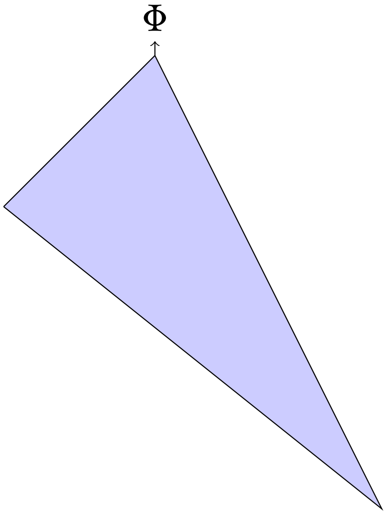
Depth reducing formulas



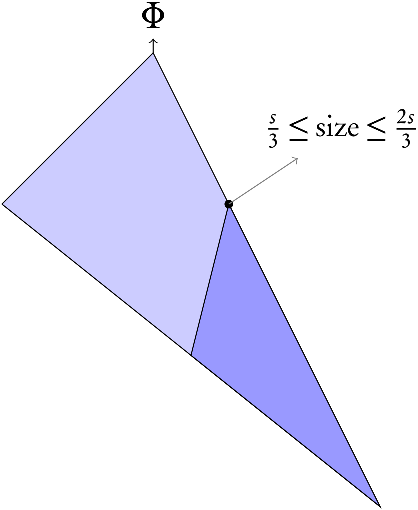
$$\begin{array}{rcccccccc} \mathrm{Size}(s) & \leq & 4 \cdot \mathrm{Size}(2s/3) & + & O(1) & \Longrightarrow & \mathrm{poly}(s) \\ \mathrm{Depth}(s) & \leq & \mathrm{Depth}(2s/3) & + & O(1) & \Longrightarrow & O(\log s) \end{array}$$

□

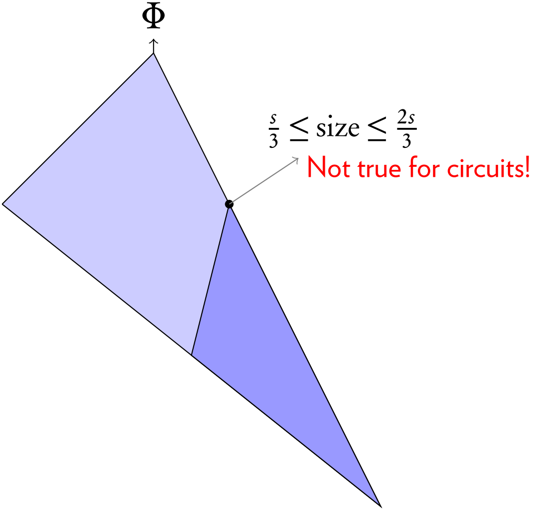
Adapting to circuits



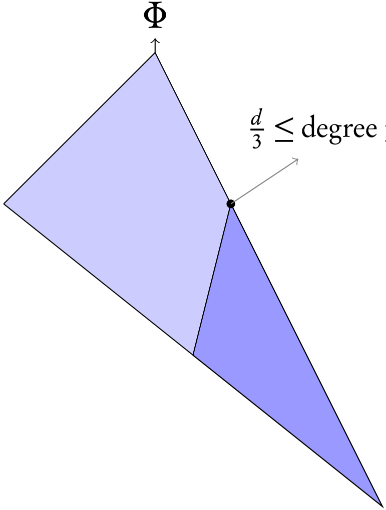
Adapting to circuits



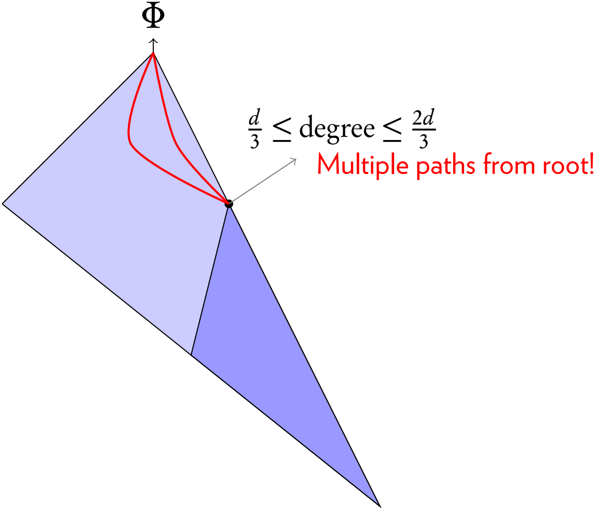
Adapting to circuits



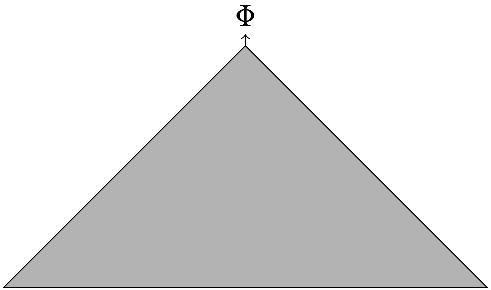
Adapting to circuits



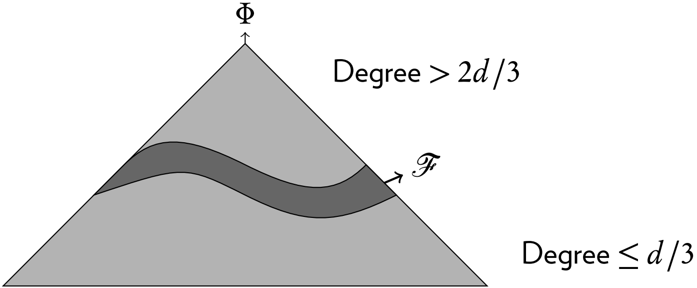
Adapting to circuits



Adapting to circuits: Attempt 1

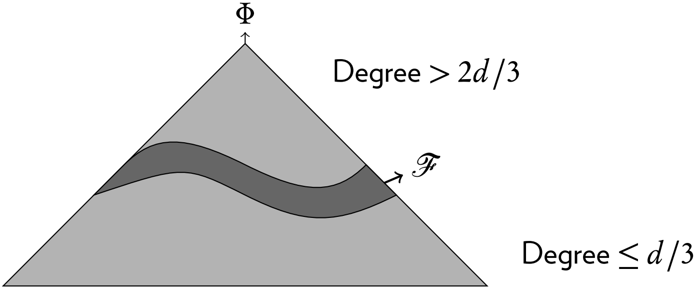


Adapting to circuits: Attempt 1



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

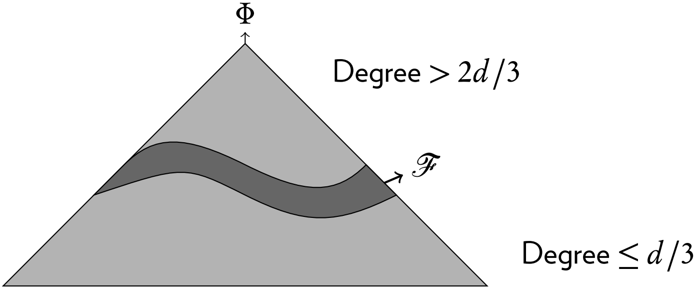
Adapting to circuits: Attempt 1



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

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

Adapting to circuits: Attempt 1

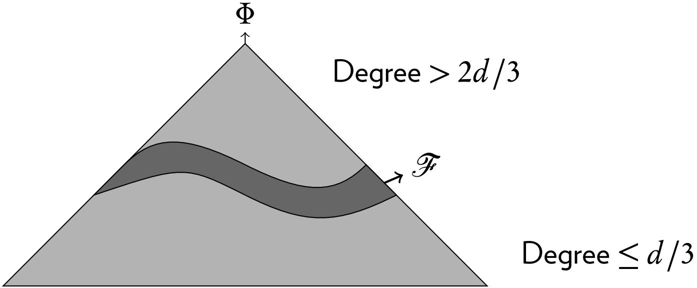


$$\mathscr{F} \quad = \quad \left\{ v \in \Phi \; : \; \frac{d}{3} < \deg(v) \leq \frac{2d}{3} \right\}$$

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

each have degree at most $2d/3$

Adapting to circuits: Attempt 1



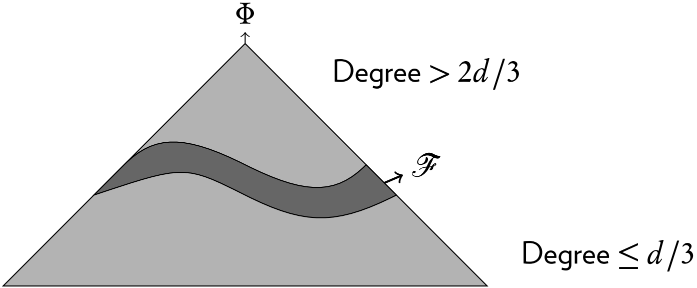
$$\mathscr{F} = \left\{ v \in \Phi : \frac{d}{3} < \deg(v) \leq \frac{2d}{3} \right\}$$

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

each have degree at most $2d/3$

Interpolate!

Adapting to circuits: Attempt 1

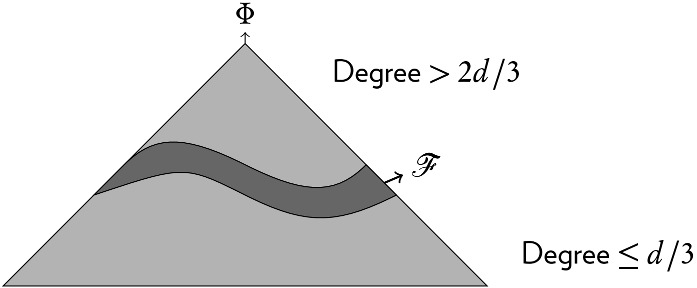


$$\mathscr{F} = \left\{ v \in \Phi : \frac{d}{3} < \deg(v) \leq \frac{2d}{3} \right\}$$

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

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

Adapting to circuits: Attempt 1

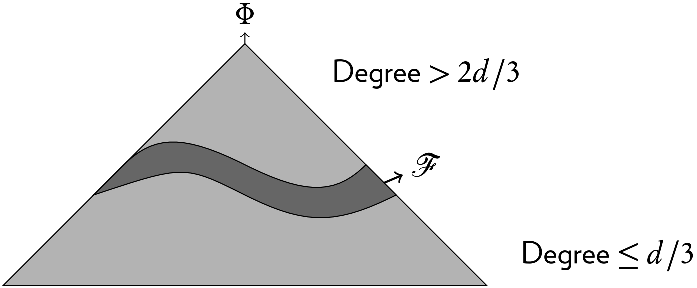


$$\mathscr{F} = \left\{ v \in \Phi : \frac{d}{3} < \deg(v) \leq \frac{2d}{3} \right\}$$

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

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

Adapting to circuits: Attempt 1



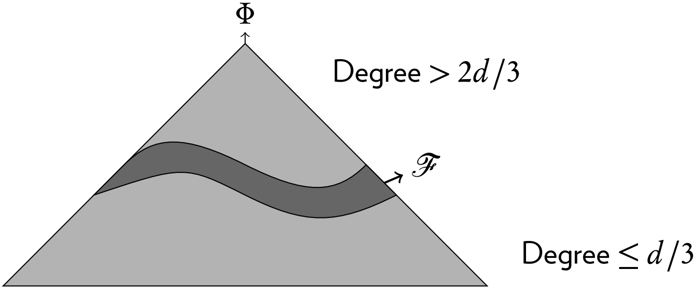
$$\mathscr{F} = \left\{ v \in \Phi : \frac{d}{3} < \deg(v) \leq \frac{2d}{3} \right\}$$

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

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

$$\text{Size}(s, d) = ?$$

Adapting to circuits: Attempt 1



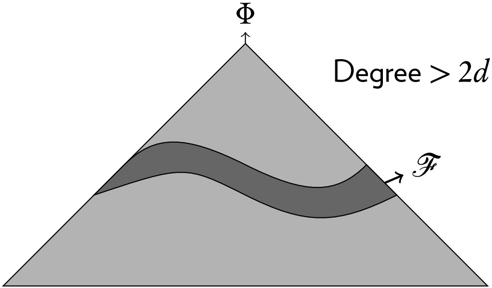
$$\mathscr{F} = \left\{ v \in \Phi : \frac{d}{3} < \deg(v) \leq \frac{2d}{3} \right\}$$

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

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

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

Adapting to circuits: [Hyafil]



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

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

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

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

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 a linear 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 a linear 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 a linear 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 a linear 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"

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

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 a linear 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"

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

Another possibility: Partial Derivatives.

Adapting to circuits: Attempt 2

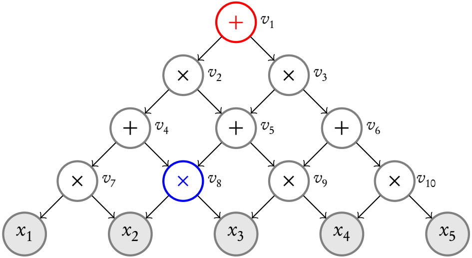
- Want an analogue of $\Phi = A \cdot \Phi_v + B$.
- Problem is that there are multiple paths to v.
  Φ isn't really a linear function in Φ_v.

[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"

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

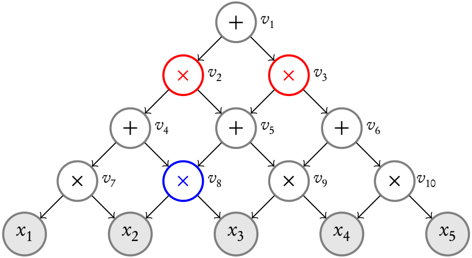
Another possibility: Partial Derivatives.
Works, but one needs to be a little careful with multiple paths. See [Shpilka-Yehudayoff]

An example



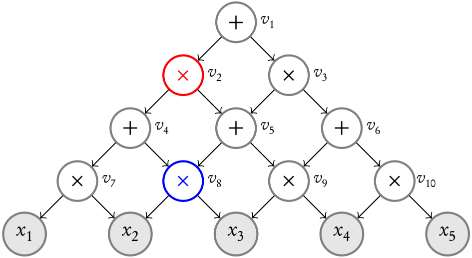
$$[v_1 : v_8] \quad =$$

An example



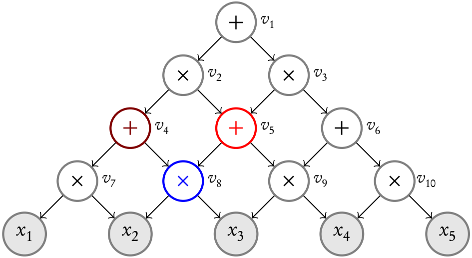
$$[v_1 : v_8] = [v_2 : v_8] + [v_3 : v_8]$$

An example



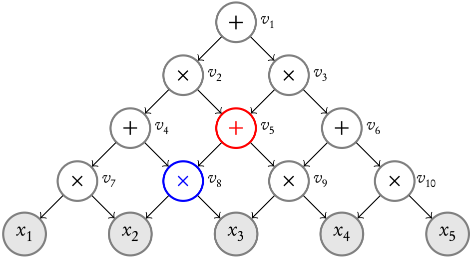
$$[v_1 : v_8] \quad = \quad [v_2 : v_8] \quad + \quad \cancel{[v_3 : v_8]}$$

An example



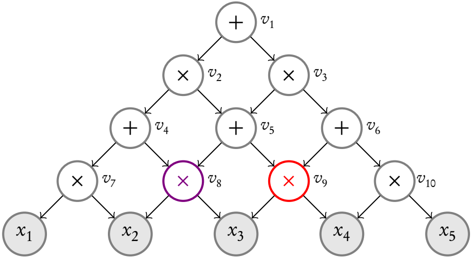
$$
\begin{aligned}
[v_1 : v_8] &= [v_2 : v_8] + \cancel{[v_3 : v_8]} \\
&= [v_4] \cdot [v_5 : v_8]
\end{aligned}
$$

An example



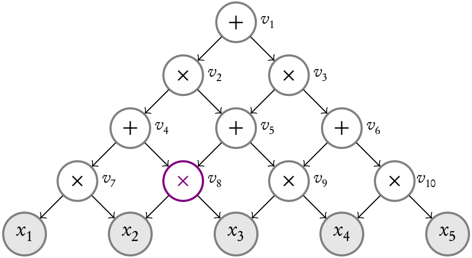
$$
\begin{aligned}
[v_1 : v_8] &= [v_2 : v_8] \quad + \quad \cancel{[v_3 : v_8]} \\
&= [v_4] \cdot [v_5 : v_8] \\
&= (x_1 x_2 + x_2 x_3) \cdot [v_5 : v_8]
\end{aligned}
$$

An example



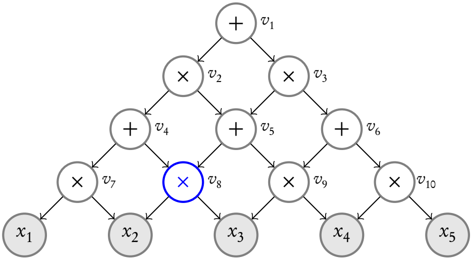
$$\begin{aligned}
[v_1 : v_8] &= [v_2 : v_8] \quad + \quad \cancel{[v_3 : v_8]} \\
&= [v_4] \cdot [v_5 : v_8] \\
&= (x_1 x_2 + x_2 x_3) \cdot [v_5 : v_8] \\
&= (x_1 x_2 + x_2 x_3) \cdot ([v_8 : v_8] + [v_9 : v_8])
\end{aligned}$$

An example



$$
\begin{aligned}
[v_1 : v_8] &= [v_2 : v_8] \quad + \quad \cancel{[v_3 : v_8]} \\
&= [v_4] \cdot [v_5 : v_8] \\
&= (x_1 x_2 + x_2 x_3) \cdot [v_5 : v_8] \\
&= (x_1 x_2 + x_2 x_3) \cdot ([v_8 : v_8] + \cancel{[v_9 : v_8]})
\end{aligned}
$$

An example



$$
\begin{aligned}
[v_1 : v_8] &= [v_2 : v_8] \quad + \quad \cancel{[v_3 : v_8]} \\
&= [v_4] \cdot [v_5 : v_8] \\
&= (x_1 x_2 + x_2 x_3) \cdot [v_5 : v_8] \\
&= (x_1 x_2 + x_2 x_3) \cdot ([v_8 : v_8] + \cancel{[v_9 : v_8]}) \\
&= (x_1 x_2 + x_2 x_3)
\end{aligned}
$$

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

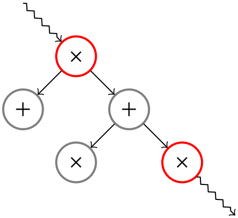
$$[u] \quad = \quad \sum_{v \in \mathscr{F}} [u : v] \cdot [v]$$

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

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

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

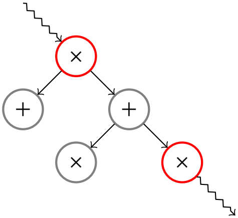
What are candidates for $\mathscr{F}$?



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

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

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



[VSBR] *continued ...*

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

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

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

$$\mathscr{F}_a \quad = \quad \{v \;:\; \deg(v) \geq a \;,\; \deg(v_L), \deg(v_R) < a\}$$

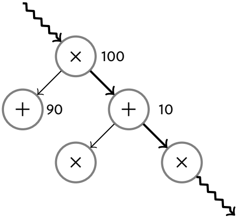
[VSBR] *continued ...*

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

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

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

$$\mathscr{F}_a \quad = \quad \{v \; : \; \deg(v) \geq a \; , \; \deg(v_L), \deg(v_R) < a\}$$

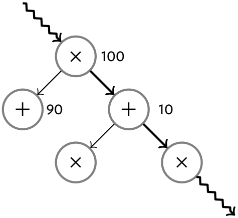


We want a set of nodes $\mathscr{F}$ such that

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

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

$$\mathscr{F}_a \quad = \quad \{v \; : \; \deg(v) \geq a \; , \; \deg(v_L), \deg(v_R) < a\}$$



Make the circuit *right heavy*.

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

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

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

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

Lemma

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

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

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

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

$$\mathscr{F}_a \quad = \quad \{v \;:\; \deg(v) \geq a \;,\; \deg(v_L), \deg(v_R) < a\}$$

Lemma

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

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

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

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

[VSBR] *continued ...*

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

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

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

$$[u] \quad = \quad \sum_{v \in \mathscr{F}_a} [u : v] \cdot [v_L] \cdot [v_R] \qquad a = \deg(u)/2$$

[VSBR] *continued ...*

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

$$[u] = \sum_{v \in \mathscr{F}_a} [u : v] \cdot [v_L] \cdot [v_R] \qquad a = \deg(u)/2$$

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

$$[u] = \sum_{v \in \mathscr{F}_a} [u : v] \cdot [v_L] \cdot [v_R] \qquad a = \deg(u)/2$$

$$\mathscr{F}_a \quad = \quad \{v \;:\; \deg(v) \geq a \;,\; \deg(v_L), \deg(v_R) < a\}$$

$$[u] \quad = \quad \sum_{v \in \mathscr{F}_a} [u : v] \cdot [v_L] \cdot [v_R] \qquad a = \deg(u)/2$$

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

$$[u] = \sum_{v \in \mathscr{F}_a} [u : v] \cdot [v_L] \cdot [v_R] \qquad a = \deg(u)/2$$

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

[VSBR] *continued ...*

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

$$[u] = \sum_{v \in \mathscr{F}_a} [u : v] \cdot [v_L] \cdot [v_R] \qquad a = \deg(u)/2$$

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

[VSBR] *continued ...*

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

$$[u] = \sum_{v \in \mathscr{F}_a} [u:v] \cdot [v_L] \cdot [v_R] \qquad a = \deg(u)/2$$

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

[VSBR] *continued ...*

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

$$[u] = \sum_{v \in \mathscr{F}_a} [u : v] \cdot [v_L] \cdot [v_R] \qquad a = \deg(u)/2$$

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

[VSBR] *continued ...*

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

$$[u] = \sum_{v \in \mathscr{F}_a} [u : v] \cdot [v_L] \cdot [v_R] \qquad a = \deg(u)/2$$

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

[VSBR] *continued ...*

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

$$[u] = \sum_{v \in \mathscr{F}_a} [u:v] \cdot [v_L] \cdot [v_R] \qquad a = \deg(u)/2$$

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

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

[VSBR] *continued ...*

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

$$[u] = \sum_{v \in \mathscr{F}_a} [u : v] \cdot [v_L] \cdot [v_R] \qquad a = \deg(u)/2$$

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

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

[VSBR] *continued ...*

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

$$[u] = \sum_{v \in \mathscr{F}_a} [u : v] \cdot [v_L] \cdot [v_R] \qquad a = \deg(u)/2$$

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

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

Summarizing

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

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 some $[u:v]$,*
- *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

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

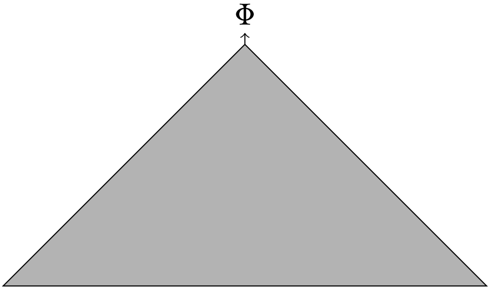
$$[u : w] = \sum_{v \in \mathscr{F}_a} \sum_{q \in \mathscr{F}_a} [u : v] \cdot [v : q] \cdot [q_L] \cdot [q_R] \cdot [v_R : w]$$

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

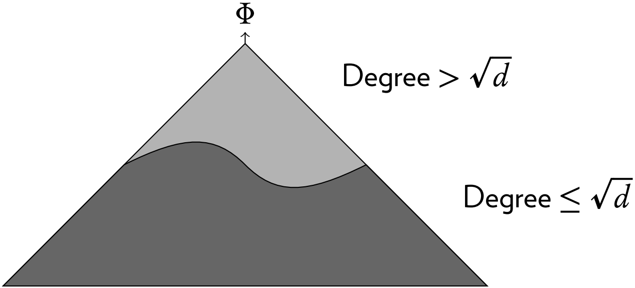
If Φ is a size s circuit computing an n-variate degree d polynomial f, then there is a circuit Φ' computing f with the following properties.

- *Every gate of Φ' computes some $[u : v]$,*
- *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 $O(\log d)$.*

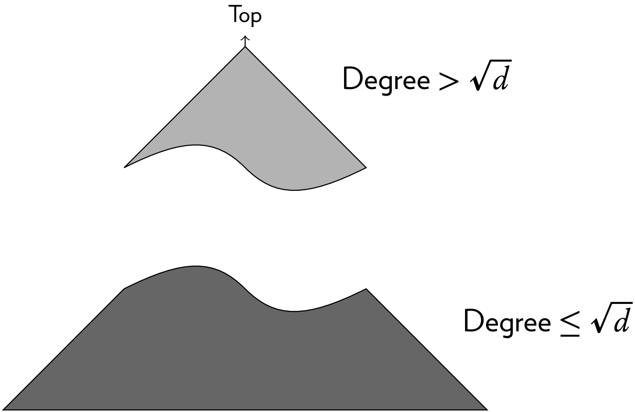
Reducing to depth four



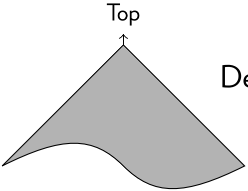
Reducing to depth four



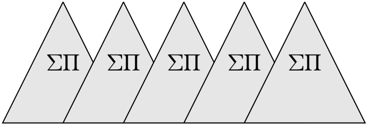
Reducing to depth four



Reducing to depth four



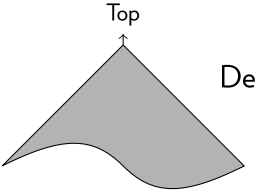
Degree $> \sqrt{d}$



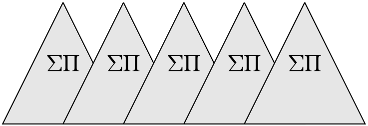
Degree $\leq \sqrt{d}$

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

Reducing to depth four



Degree $> \sqrt{d}$



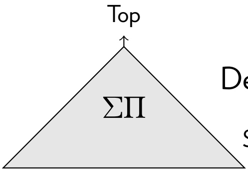
Degree $\leq \sqrt{d}$

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

Lemma ([Tavenas13])

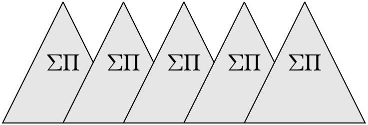
$$\deg(\mathrm{Top}(z_1, \ldots, z_s)) \quad \leq \quad 15\sqrt{d}$$

Reducing to depth four



Degree $> \sqrt{d}$

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



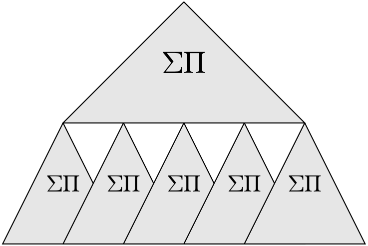
Degree $\leq \sqrt{d}$

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

Lemma ([Tavenas13])

$$\deg(\mathrm{Top}(z_1, \ldots, z_s)) \quad \leq \quad 15\sqrt{d}$$

Reducing to depth four

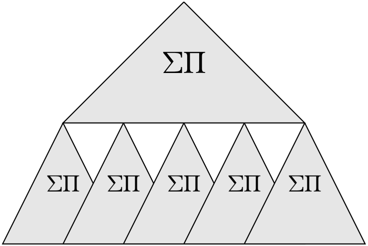


Theorem

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



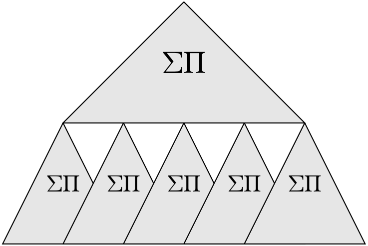
Theorem

Equivalent depth-4 circuit of size

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

□

Reducing to depth four



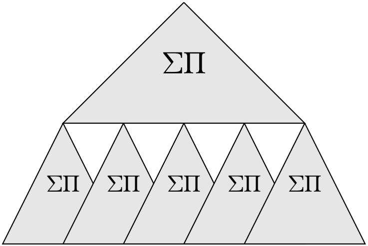
Theorem

Equivalent 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



Theorem

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

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

□

A different perspective

Let's start with [VSBR]

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

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.

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.
Each f_{ij} is also some $[u : v]$. Keep expanding terms of degree more than t.

A different perspective

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. Each f_{ij} is also some $[u:v]$. Keep expanding terms of degree more than t.

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.

Let's start with [VSBR]

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

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.

A different perspective

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.

A different perspective

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.

A different perspective

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

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

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

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

Observation

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

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

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

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 $t/8$?

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

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

Observation

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

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

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

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 $t/8$? At most $8d/t$.

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

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

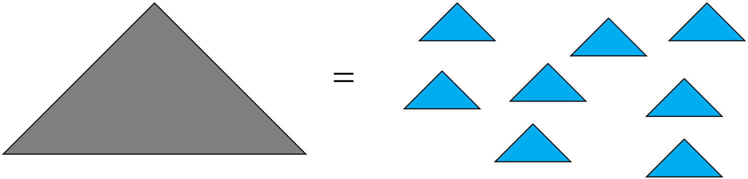
In each summand, at least two terms have degree at least $t/8$.

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

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

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

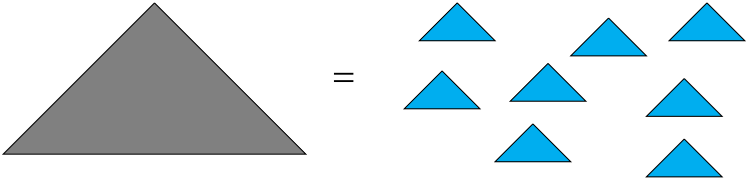
Recap



Theorem

Every small circuit can be equivalently computed as a *sum* of few s

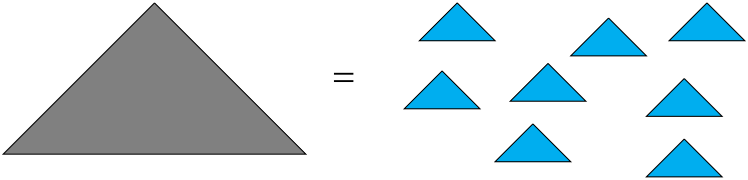
Recap



Theorem

Every circuit of size s can be equivalently computed as a *sum* of few s

Recap

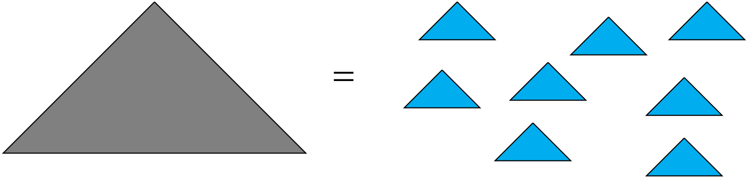


Theorem

Every circuit of size s can be equivalently computed as a *sum* of $s^{O(d/t)}$

s

Recap

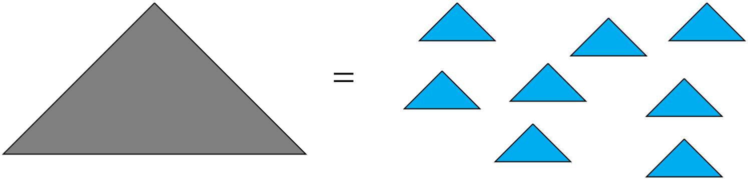


Theorem

Every circuit of size s can be equivalently computed as a *sum* of $s^{O(d/t)}$ s , where

 $$= \prod_{i=1}^{O(d/t)} Q_i \qquad \deg(Q_i) \leq t$$

Recap



Theorem

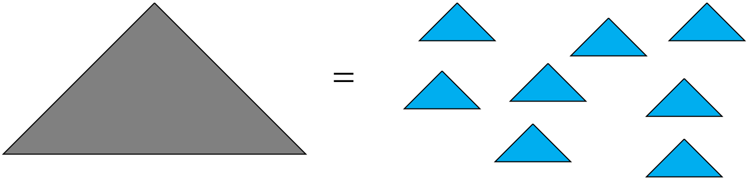
Every circuit of size s can be equivalently computed as a *sum* of $s^{O(d/t)}$ s , where

$$= \prod_{i=1}^{O(d/t)} Q_i \qquad \deg(Q_i) \leq t$$

Question

What are the s if we start with a homogeneous formula of size s?

Recap



Theorem

Every circuit of size s can be equivalently computed as a *sum* of $s^{O(d/t)}$ s , where

 $$= \prod_{i=1}^{O(d/t)} Q_i \qquad \deg(Q_i) \le t$$

Question

What are the s if we start with a depth 100 formula of size s?

A better starting point?

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

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

A better starting point?

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

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

Lemma ([Hrubes-Yehudayoff])

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

A better starting point?

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

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

Lemma ([Hrubes-Yehudayoff])

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

Proof

$$f = A \cdot \Phi_v + \Phi_{v=0}$$

A better starting point?

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

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

Lemma ([Hrubes-Yehudayoff])

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

Proof

$$\begin{aligned} f \quad &= \quad A \cdot \Phi_v + \Phi_{v=0} \\ &= \quad A \cdot \left(\sum_{i=1}^{s_1} g_{i1} \cdots g_{i\ell} \right) + \left(\sum_{j=1}^{s_2} h_{j1} \cdots h_{j\ell} \right) \end{aligned}$$

A better starting point?

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

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

Lemma ([Hrubes-Yehudayoff])

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

Proof

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

Reduction to depth four, again

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

Reduction to depth four, again

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

Fact: Each f_{ij} is also computable by size s homogeneous formulas

Reduction to depth four, again

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

Fact: Each f_{ij} is also computable by size s homogeneous formulas

- The above expression is a $\Sigma\Pi\Sigma\Pi^{[2d/3]}$ circuit. We want a $\Sigma\Pi\Sigma\Pi^{[t]}$ circuit.

Reduction to depth four, again

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

Fact: Each f_{ij} is also computable by size s homogeneous formulas

- The above expression is a $\Sigma\Pi\Sigma\Pi^{[2d/3]}$ circuit. We want a $\Sigma\Pi\Sigma\Pi^{[t]}$ circuit.
- In each iteration, if the highest degree factor has degree more than t, expand again.

Reduction to depth four, again

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

Fact: Each f_{ij} is also computable by size s homogeneous formulas

- The above expression is a $\Sigma\Pi\Sigma\Pi^{[2d/3]}$ circuit. We want a $\Sigma\Pi\Sigma\Pi^{[t]}$ circuit.
- In each iteration, if the highest degree factor has degree more than t, expand again.

Question: How many iterations?

Reduction to depth four, again

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

Fact: Each f_{ij} is also computable by size s homogeneous formulas

- The above expression is a $\Sigma\Pi\Sigma\Pi^{[2d/3]}$ circuit. We want a $\Sigma\Pi\Sigma\Pi^{[t]}$ circuit.
- In each iteration, if the highest degree factor has degree more than t, expand again.

Question: How many iterations? $O(d/t)$ again.

Reduction to depth four, again

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

Fact: Each f_{ij} is also computable by size s homogeneous formulas

- The above expression is a $\Sigma\Pi\Sigma\Pi^{[2d/3]}$ circuit. We want a $\Sigma\Pi\Sigma\Pi^{[t]}$ circuit.
- In each iteration, if the highest degree factor has degree more than t, expand again.

Question: How many iterations? $O(d/t)$ again.

Proof: There are at least two terms of degree $t/9$. Yada Yada Yada □

Reduction to depth four, again

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

Fact: Each f_{ij} is also computable by size s homogeneous formulas

- The above expression is a $\Sigma\Pi\Sigma\Pi^{[2d/3]}$ circuit. We want a $\Sigma\Pi\Sigma\Pi^{[t]}$ circuit.
- In each iteration, if the highest degree factor has degree more than t, expand again.

Question: How many iterations? $O(d/t)$ again.

Proof: There are at least two terms of degree $t/9$. Yada Yada Yada □

Yields a $\Sigma\Pi\Sigma\Pi^{[t]}$ circuit of top fan-in $s^{O(d/t)}$.

Wait... what's different?

For circuits:

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

For homogeneous formulas:

$$f = \sum_{i=1}^{s} f_{i1} \cdot f_{i2} \cdots f_{i\ell}$$

Wait... what's different?

For circuits:

$$f = \sum_{i=1}^{s} \left(\sum_{j=1}^{s} g_{i1} \cdots g_{i5} \right) \cdot f_{i2} \cdots f_{i5}$$

For homogeneous formulas:

$$f = \sum_{i=1}^{s} \left(\sum_{j=1}^{s} g_{i1} \cdots g_{i\ell} \right) \cdot f_{i2} \cdots f_{i\ell}$$

Wait... what's different?

For circuits:

$$f = \sum_{i=1}^{s^2} f_{i1} \cdot f_{i2} \cdots f_{i9}$$

For homogeneous formulas:

$$f = \sum_{i=1}^{s^2} f_{i1} \cdot f_{i2} \cdots f_{i(2\ell)}$$

Wait... what's different?

For circuits:

$$f \quad = \quad \sum_{i=1}^{s^4} f_{i1} \cdot f_{i2} \cdots f_{i13}$$

For homogeneous formulas:

$$f \quad = \quad \sum_{i=1}^{s^4} f_{i1} \cdot f_{i2} \cdots f_{i(3\ell)}$$

Wait... what's different?

For circuits:

$$f \quad = \quad \sum_{i=1}^{s^r} f_{i1} \cdot f_{i2} \cdots f_{i(4r+1)}$$

For homogeneous formulas:

$$f \quad = \quad \sum_{i=1}^{s^r} f_{i1} \cdot f_{i2} \cdots f_{i(r\ell)}$$

Wait... what's different?

For circuits:

$$f = \sum_{i=1}^{s^r} f_{i1} \cdot f_{i2} \cdots f_{i(4r+1)}$$

a $\Sigma\Pi^{[O(d/t)]}\Sigma\Pi^{[t]}$ circuit

For homogeneous formulas:

$$f = \sum_{i=1}^{s^r} f_{i1} \cdot f_{i2} \cdots f_{i(r\ell)}$$

a $\Sigma\Pi^{[O(d/t)\cdot \log d]}\Sigma\Pi^{[t]}$ circuit

Wait... what's different?

For circuits:

$$f \quad = \quad \sum_{i=1}^{s^r} f_{i1} \cdot f_{i2} \cdots f_{i(4r+1)}$$

a $\Sigma\Pi^{[O(d/t)]}\Sigma\Pi^{[t]}$ circuit

For homogeneous formulas:

$$f \quad = \quad \sum_{i=1}^{s^r} f_{i1} \cdot f_{i2} \cdots f_{i(r\ell)}$$

a $\Sigma\Pi^{[O(d/t)\cdot \log d]}\Sigma\Pi^{[t]}$ circuit

Here, $\triangle$s factorize more

Is that a big deal?

Circuit class	No. factors of 	Lower bound

Is that a big deal?

Circuit class	No. factors of 	Lower bound
Hom. $\Sigma\Pi\Sigma$	$O(n)$	$n^{\Omega(n)}$

Is that a big deal?

Circuit class	**No. factors of** 	**Lower bound**
Hom. $\Sigma\Pi\Sigma$	$O(n)$	$n^{\Omega(n)}$
Multilinear formulas	$O(\log n)$	$n^{\Omega(\log n)}$

Is that a big deal?

Circuit class	No. factors of 	Lower bound
Hom. $\Sigma\Pi\Sigma$	$O(n)$	$n^{\Omega(n)}$
Multilinear formulas	$O(\log n)$	$n^{\Omega(\log n)}$
Multilinear, depth-Δ	$n^{O(1/\Delta)}$	$n^{n^{\Omega(1/\Delta)}}$

Is that a big deal?

Circuit class	**No. factors of** 	**Lower bound**
Hom. $\Sigma\Pi\Sigma$	$O(n)$	$n^{\Omega(n)}$
Multilinear formulas	$O(\log n)$	$n^{\Omega(\log n)}$
Multilinear, depth-Δ	$n^{O(1/\Delta)}$	$n^{n^{\Omega(1/\Delta)}}$
$\Sigma\Pi\Sigma\Pi^{[t]}$	$O(d/t)$	$n^{\Omega(d/t)}$

Is that a big deal?

Circuit class	No. factors of	Lower bound
Hom. $\Sigma\Pi\Sigma$	$O(n)$	$n^{\Omega(n)}$
Multilinear formulas	$O(\log n)$	$n^{\Omega(\log n)}$
Multilinear, depth-Δ	$n^{O(1/\Delta)}$	$n^{n^{\Omega(1/\Delta)}}$
$\Sigma\Pi\Sigma\Pi^{[t]}$	$O(d/t)$	$n^{\Omega(d/t)}$
Hom. $\Sigma\Pi\Sigma\Pi$	$O(\sqrt{d})$	$n^{O(\sqrt{d})}$

Is that a big deal?

Circuit class	No. factors of △	Lower bound
Hom. $\Sigma\Pi\Sigma$	$O(n)$	$n^{\Omega(n)}$
Multilinear formulas	$O(\log n)$	$n^{\Omega(\log n)}$
Multilinear, depth-Δ	$n^{O(1/\Delta)}$	$n^{n^{\Omega(1/\Delta)}}$
$\Sigma\Pi\Sigma\Pi^{[t]}$	$O(d/t)$	$n^{\Omega(d/t)}$
Hom. $\Sigma\Pi\Sigma\Pi$	$O(\sqrt{d})$	$n^{O(\sqrt{d})}$

Wishful Question: Can we get an $n^{\Omega(\log n)}$ lower bound for homogeneous formulas, using current techniques (with slight modifications)?

Reduction to Depth-3 Circuits

No time!

Road map

$$\Sigma\Pi^{\sqrt{d}}\Sigma\Pi^{\sqrt{d}}$$

circuits

$$\downarrow$$

$$\Sigma\wedge^{\sqrt{d}}\Sigma\wedge^{\sqrt{d}}\Sigma$$

circuits

$$\downarrow$$

$$\Sigma\Pi\Sigma$$

circuits

Road map

$\Sigma\Pi^{\sqrt{d}}\Sigma\Pi^{\sqrt{d}}$ circuits

↓ App. of Ryser's formula

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

↓

$\Sigma\Pi\Sigma$ circuits

Road map

$$\Sigma \overset{\sqrt{d}}{\Pi} \Sigma \overset{\sqrt{d}}{\Pi}$$

circuits

↓ App. of Ryser's formula

$$\Sigma \overset{\sqrt{d}}{\bigwedge} \Sigma \overset{\sqrt{d}}{\bigwedge} \Sigma$$

circuits

↓ [Saxena]'s duality trick

$$\Sigma \Pi \Sigma$$

circuits

Road map

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

circuits

↓

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

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

circuits

↓

Heavily non-homogeneous — [Saxena]'s duality trick

$$\Sigma\Pi\Sigma$$

circuits

Step 1: $\Sigma\Pi\Sigma\Pi$ *to* $\Sigma\wedge\Sigma\wedge\Sigma$

Step 1*:* $\Sigma\Pi\Sigma\Pi$ *to* $\Sigma\wedge\Sigma\wedge\Sigma$

Recall Ryser's formula:

$$\mathrm{Perm}_n \begin{bmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nn} \end{bmatrix} = \sum_{S \subseteq [n]} (-1)^{n-|S|} \prod_{i=1}^{n} \sum_{j \in S} x_{ij}$$

Step 1*:* $\Sigma\Pi\Sigma\Pi$ *to* $\Sigma\wedge\Sigma\wedge\Sigma$

Recall Ryser's formula:

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

Step 1*:* $\Sigma\Pi\Sigma\Pi$ *to* $\Sigma\wedge\Sigma\wedge\Sigma$

Recall Ryser's formula:

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

Step 1*:* $\Sigma\Pi\Sigma\Pi$ *to* $\Sigma\wedge\Sigma\wedge\Sigma$

Recall Ryser's formula:

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

Step 1*:* $\Sigma\Pi\Sigma\Pi$ *to* $\Sigma\wedge\Sigma\wedge\Sigma$

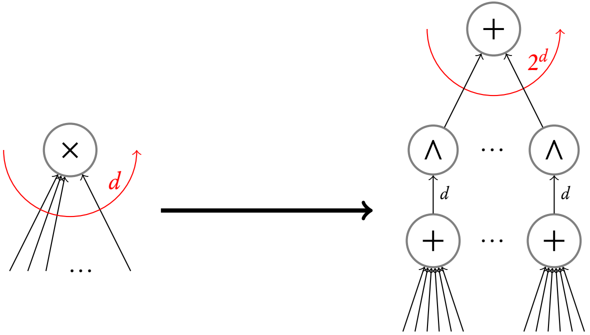
[Fischer]:

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

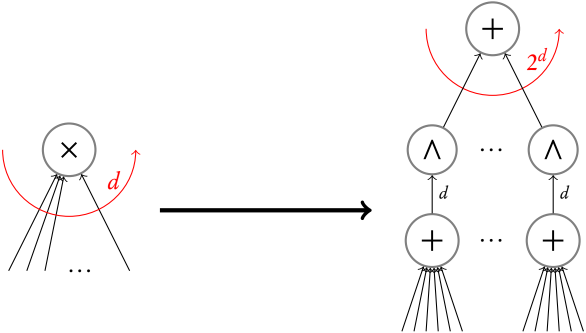
Step 1: $\Sigma\Pi\Sigma\Pi$ *to* $\Sigma\wedge\Sigma\wedge\Sigma$

[Fischer]:

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

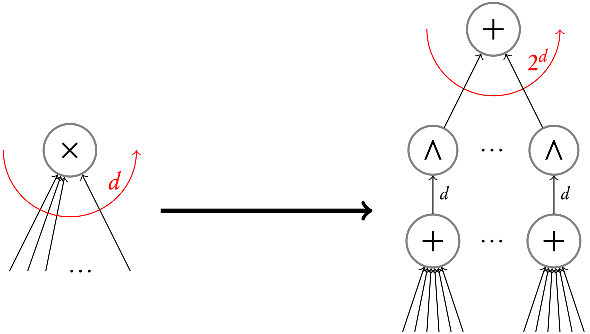


Step 1: $\Sigma\Pi\Sigma\Pi$ *to* $\Sigma\wedge\Sigma\wedge\Sigma$



$$\overset{d}{\prod} \longrightarrow \overset{2^d}{\Sigma}\overset{d}{\wedge}\overset{d}{\Sigma}$$

Step 1: $\Sigma\Pi\Sigma\Pi$ *to* $\Sigma\Lambda\Sigma\Lambda\Sigma$



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

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

Road map

$$\Sigma\Pi^{\sqrt{d}}\Sigma\Pi^{\sqrt{d}}$$

circuits

↓

$$\Sigma\wedge^{\sqrt{d}}\Sigma\wedge^{\sqrt{d}}\Sigma$$

circuits

↓

$$\Sigma\Pi\Sigma$$

circuits

Road map

$$\Sigma\Pi^{\sqrt{d}}\Sigma\Pi^{\sqrt{d}}$$

circuits

✓

$$\Sigma\wedge^{\sqrt{d}}\Sigma\wedge^{\sqrt{d}}\Sigma$$

circuits

$$\Sigma\Pi\Sigma$$

circuits

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ *to* $\Sigma\Pi\Sigma$

$$\Sigma \overset{\sqrt{d}}{\wedge} \Sigma \overset{\sqrt{d}}{\wedge} \Sigma$$

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ *to* $\Sigma\Pi\Sigma$

$$\Sigma \overset{\sqrt{d}}{\wedge} \Sigma \overset{\sqrt{d}}{\wedge} \Sigma$$

$$\ell$$

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ *to* $\Sigma\Pi\Sigma$

$$\Sigma \overset{\sqrt{d}}{\wedge} \Sigma \overset{\sqrt{d}}{\wedge} \Sigma$$

$$\ell^{\sqrt{d}}$$

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ *to* $\Sigma\Pi\Sigma$

$$\Sigma \overset{\sqrt{d}}{\wedge} \Sigma \overset{\sqrt{d}}{\wedge} \Sigma$$

$$\ell_1^{\sqrt{d}} + \ldots + \ell_s^{\sqrt{d}}$$

Step 2: $\Sigma\Lambda\Sigma\Lambda\Sigma$ *to* $\Sigma\Pi\Sigma$

$$\Sigma \overset{\sqrt{d}}{\Lambda} \Sigma \overset{\sqrt{d}}{\Lambda} \Sigma$$

$$\left(\ell_1^{\sqrt{d}} + \ldots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}}$$

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ *to* $\Sigma\Pi\Sigma$

$$\Sigma \overset{\sqrt{d}}{\wedge} \Sigma \overset{\sqrt{d}}{\wedge} \Sigma$$

$$\sum_i \left(\ell_{i1}^{\sqrt{d}} + \ldots + \ell_{is}^{\sqrt{d}} \right)^{\sqrt{d}}$$

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ *to* $\Sigma\Pi\Sigma$

$$C = \sum_i \left(\ell_{i1}^{\sqrt{d}} + \cdots + \ell_{is}^{\sqrt{d}}\right)^{\sqrt{d}}$$

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ *to* $\Sigma\Pi\Sigma$

$$T = \left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}}$$

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ *to* $\Sigma\Pi\Sigma$

$$T \quad = \quad \left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}}$$

Lemma ([Saxena])

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

$$(x_1 + \cdots + x_s)^d \quad = \quad \sum_{i=1}^{sd+1} \prod_{j=1}^{s} f_{ij}(x_j)$$

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ to $\Sigma\Pi\Sigma$

$$T = \left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}}$$

Lemma ([Saxena])

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

$$(x_1 + \cdots + x_s)^d = \sum_{i=1}^{sd+1} \prod_{j=1}^{s} f_{ij}(x_j)$$

Sketch of a proof by Shpilka

$$P_{\mathbf{x}}(t) = (1 + x_1 t)\ldots(1 + x_s t) = 1 + \ell t + \text{(higher degree terms)}$$

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ to $\Sigma\Pi\Sigma$

$$T \quad = \quad \left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}}$$

Lemma ([Saxena])

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

$$(x_1 + \cdots + x_s)^d \quad = \quad \sum_{i=1}^{sd+1} \prod_{j=1}^{s} f_{ij}(x_j)$$

Sketch of a proof by Shpilka

$$P_{\mathbf{x}}(t) - 1 \quad = \quad \ell t \quad + \quad \text{(higher degree terms)}$$

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ to $\Sigma\Pi\Sigma$

$$T \quad = \quad \left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}}$$

Lemma ([Saxena])

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

$$(x_1 + \cdots + x_s)^d \quad = \quad \sum_{i=1}^{sd+1} \prod_{j=1}^{s} f_{ij}(x_j)$$

Sketch of a proof by Shpilka

$$(P_{\mathbf{x}}(t) - 1)^d \quad = \quad \ell^d t^d \quad + \quad \text{(higher degree terms)}$$

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ to $\Sigma\Pi\Sigma$

$$T \quad = \quad \left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}}$$

Lemma ([Saxena])

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

$$(x_1 + \cdots + x_s)^d \quad = \quad \sum_{i=1}^{sd+1} \prod_{j=1}^{s} f_{ij}(x_j)$$

Sketch of a proof by Shpilka

$$(P_{\mathbf{x}}(t) - 1)^d \quad = \quad \ell^d t^d \quad + \quad \text{(higher degree terms)}$$

Interpolate!

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

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ *to* $\Sigma\Pi\Sigma$

$$T = \left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}}$$

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

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ *to* $\Sigma\Pi\Sigma$

$$T \quad = \quad \left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}}$$

$$\left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}} \quad = \quad \sum_{i}^{\mathrm{poly}(s,d)} \prod_{j=1}^{s} f_{ij}\left(\ell_j^{\sqrt{d}}\right)$$

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ *to* $\Sigma\Pi\Sigma$

$$T \quad = \quad \left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}}$$

$$\left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}} \quad = \quad \sum_{i}^{\mathrm{poly}(s,d)} \prod_{j=1}^{s} f_{ij}\left(\ell_j^{\sqrt{d}}\right)$$

$$= \quad \sum_{i}^{\mathrm{poly}(s,d)} \prod_{j=1}^{s} \tilde{f}_{ij}(\ell_j)$$

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

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ *to* $\Sigma\Pi\Sigma$

$$T \quad = \quad \left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}}$$

$$\left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}} \quad = \quad \sum_{i}^{\text{poly}(s,d)} \prod_{j=1}^{s} f_{ij}\left(\ell_j^{\sqrt{d}}\right)$$

$$= \quad \sum_{i}^{\text{poly}(s,d)} \prod_{j=1}^{s} \tilde{f}_{ij}(\ell_j)$$

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

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ *to* $\Sigma\Pi\Sigma$

$$T = \left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}}$$

$$\left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}} = \sum_{i}^{\mathrm{poly}(s,d)} \prod_{j=1}^{s} f_{ij}\left(\ell_j^{\sqrt{d}}\right)$$

$$= \sum_{i}^{\mathrm{poly}(s,d)} \prod_{j=1}^{s} \tilde{f}_{ij}(\ell_j)$$

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

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

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ *to* $\Sigma\Pi\Sigma$

$$T \quad = \quad \left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}}$$

$$\left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}} \quad = \quad \sum_i^{\mathrm{poly}(s,d)} \prod_{j=1}^{s} f_{ij}\left(\ell_j^{\sqrt{d}}\right)$$

$$= \quad \sum_i^{\mathrm{poly}(s,d)} \prod_{j=1}^{s} \tilde{f}_{ij}(\ell_j)$$

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

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

Step 2: $\Sigma\wedge\Sigma\wedge\Sigma$ *to* $\Sigma\Pi\Sigma$

$$T = \left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}}$$

$$\begin{aligned}\left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}} &= \sum_{i}^{\mathrm{poly}(s,d)} \prod_{j=1}^{s} f_{ij}\left(\ell_j^{\sqrt{d}}\right) \\ &= \sum_{i}^{\mathrm{poly}(s,d)} \prod_{j=1}^{s} \tilde{f}_{ij}(\ell_j) \\ &= \sum_{i}^{\mathrm{poly}(s,d)} \prod_{j=1}^{s} \prod_{k=1}^{d} \left(\ell_j - \zeta_{ijk}\right)\end{aligned}$$

Step 2*:* $\Sigma\wedge\Sigma\wedge\Sigma$ *to* $\Sigma\Pi\Sigma$

$$T = \left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}}$$

$$\begin{aligned}
\left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}} &= \sum_i^{\text{poly}(s,d)} \prod_{j=1}^{s} f_{ij}\left(\ell_j^{\sqrt{d}}\right) \\
&= \sum_i^{\text{poly}(s,d)} \prod_{j=1}^{s} \tilde{f}_{ij}(\ell_j) \\
&= \sum_i^{\text{poly}(s,d)} \prod_{j=1}^{s} \prod_{k=1}^{d} \left(\ell_j - \zeta_{ijk}\right)
\end{aligned}$$

... a $\Sigma\Pi\Sigma$ circuit of poly(s,d) size.

Step 2*:* $\Sigma\wedge\Sigma\wedge\Sigma$ *to* $\Sigma\Pi\Sigma$

$$T \quad = \quad \left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}}$$

$$\left(\ell_1^{\sqrt{d}} + \cdots + \ell_s^{\sqrt{d}}\right)^{\sqrt{d}} \quad = \quad \sum_{i}^{\mathrm{poly}(s,d)} \prod_{j=1}^{s} f_{ij}\left(\ell_j^{\sqrt{d}}\right)$$

$$= \quad \sum_{i}^{\mathrm{poly}(s,d)} \prod_{j=1}^{s} \tilde{f}_{ij}(\ell_j)$$

$$= \quad \sum_{i}^{\mathrm{poly}(s,d)} \prod_{j=1}^{s} \prod_{k=1}^{d} \left(\ell_j - \zeta_{ijk}\right)$$

... a $\Sigma\Pi\Sigma$ circuit of $\mathrm{poly}(s,d)$ size and degree sd.

Putting it together

general circuit
of size s

Putting it together

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

Putting it together

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

$\downarrow$

$\Sigma\bigwedge^{\sqrt{d}}\Sigma\bigwedge^{\sqrt{d}}\Sigma$ circuits of size $s^{O(\sqrt{d})} \cdot 2^{O(\sqrt{d})}$

Putting it together

general circuit of size s $\longrightarrow$ $\Sigma\overset{\sqrt{d}}{\Pi}\Sigma\overset{\sqrt{d}}{\Pi}$ circuit of size $s^{O(\sqrt{d})}$

$\downarrow$

$\Sigma\overset{\sqrt{d}}{\bigwedge}\Sigma\overset{\sqrt{d}}{\bigwedge}\Sigma$ circuits of size $s^{O(\sqrt{d})}$

Putting it together

general circuit of size s

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

$\downarrow$ $\Sigma \bigwedge^{\sqrt{d}} \Sigma \bigwedge^{\sqrt{d}} \Sigma$ circuits of size $s^{O(\sqrt{d})}$

$\longleftarrow$ $\Sigma\Pi\Sigma$ circuits of size $s^{O(\sqrt{d})}$

Putting it together

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

$\downarrow$

$\Sigma\Pi\Sigma$ circuits of size $s^{O(\sqrt{d})}$ $\longleftarrow$ $\Sigma\bigwedge^{\sqrt{d}}\Sigma\bigwedge^{\sqrt{d}}\Sigma$ circuits of size $s^{O(\sqrt{d})}$

Question: Where should one try to prove lower bounds?

Putting it together

general hom. circuit of size s

$\longrightarrow$

$\Sigma\prod^{\sqrt{d}}\Sigma\prod^{\sqrt{d}}$ circuit of size $s^{O(\sqrt{d})}$

$\downarrow$

$\Sigma\bigwedge^{\sqrt{d}}\Sigma\bigwedge^{\sqrt{d}}\Sigma$ circuits of size $s^{O(\sqrt{d})}$

$\longrightarrow$

$\Sigma\Pi\Sigma$ circuits of size $s^{O(\sqrt{d})}$

Question: Where should one try to prove lower bounds?

Putting it together

general hom. circuit of size s $\longrightarrow$ $\Sigma\Pi^{\sqrt{d}}\Sigma\Pi^{\sqrt{d}}$ hom. circuit of size $s^{O(\sqrt{d})}$

$\downarrow$

$\Sigma\bigwedge^{\sqrt{d}}\Sigma\bigwedge^{\sqrt{d}}\Sigma$ circuits of size $s^{O(\sqrt{d})}$ $\longrightarrow$ $\Sigma\Pi\Sigma$ circuits of size $s^{O(\sqrt{d})}$

Question: Where should one try to prove lower bounds?

Putting it together

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

$\downarrow$

$\Sigma\bigwedge^{\sqrt{d}}\Sigma\bigwedge^{\sqrt{d}}\Sigma$ hom. circuits of size $s^{O(\sqrt{d})}$ $\longrightarrow$ $\Sigma\prod\Sigma$ circuits of size $s^{O(\sqrt{d})}$

Question: Where should one try to prove lower bounds?

Putting it together

general hom. circuit of size s $\longrightarrow$ $\Sigma\Pi^{\sqrt{d}}\Sigma\Pi^{\sqrt{d}}$ hom. circuit of size $s^{O(\sqrt{d})}$

$\downarrow$

$\Sigma\bigwedge^{\sqrt{d}}\Sigma\bigwedge^{\sqrt{d}}\Sigma$ hom. circuits of size $s^{O(\sqrt{d})}$ $\longrightarrow$ $\Sigma\Pi\Sigma$ **non-hom.** circuits of size $s^{O(\sqrt{d})}$

Question: Where should one try to prove lower bounds?

Putting it together

general hom. circuit of size s $\longrightarrow$ $\Sigma\Pi^{\sqrt{d}}\Sigma\Pi^{\sqrt{d}}$ hom. circuit of size $s^{O(\sqrt{d})}$

$\downarrow$

$\Sigma\bigwedge^{\sqrt{d}}\Sigma\bigwedge^{\sqrt{d}}\Sigma$ hom. circuits of size $s^{O(\sqrt{d})}$ $\longrightarrow$ $\Sigma\Pi\Sigma$ **non-hom.** circuits of size $s^{O(\sqrt{d})}$

Question: Where should one try to prove lower bounds?

Summary

- Depth reduction can manifest in many forms. Finding the right building block is sometimes crucial.
- A *slightly* different proof of [Tavenas] yields a possible useful building block for homogeneous formulas with more factors.
- *Maybe* we can get $n^{\Omega(\log n)}$ lower bounds via modified shifted-partials.
- Can we say something similar about $\Sigma\Pi\Sigma\Pi^{[t]}$ circuits obtained from ABPs?

Call for contributors

A git survey on arithmetic circuit lower bounds:

`https://github.com/dasarpmar/lowerbounds-survey/`

Dankeschön