Unified PITs via the Jacobian

Manindra Agrawal

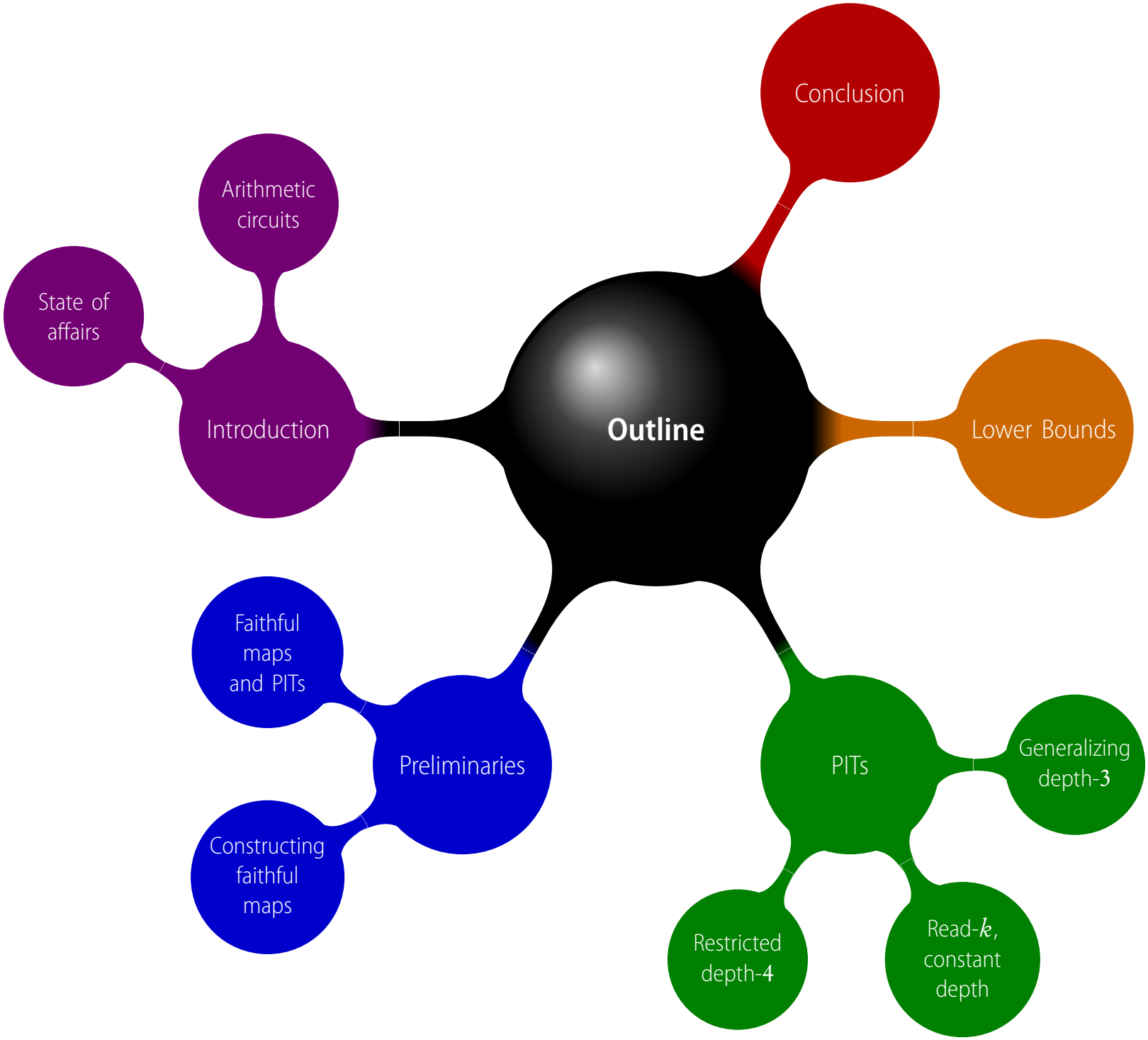
Chandan Saha

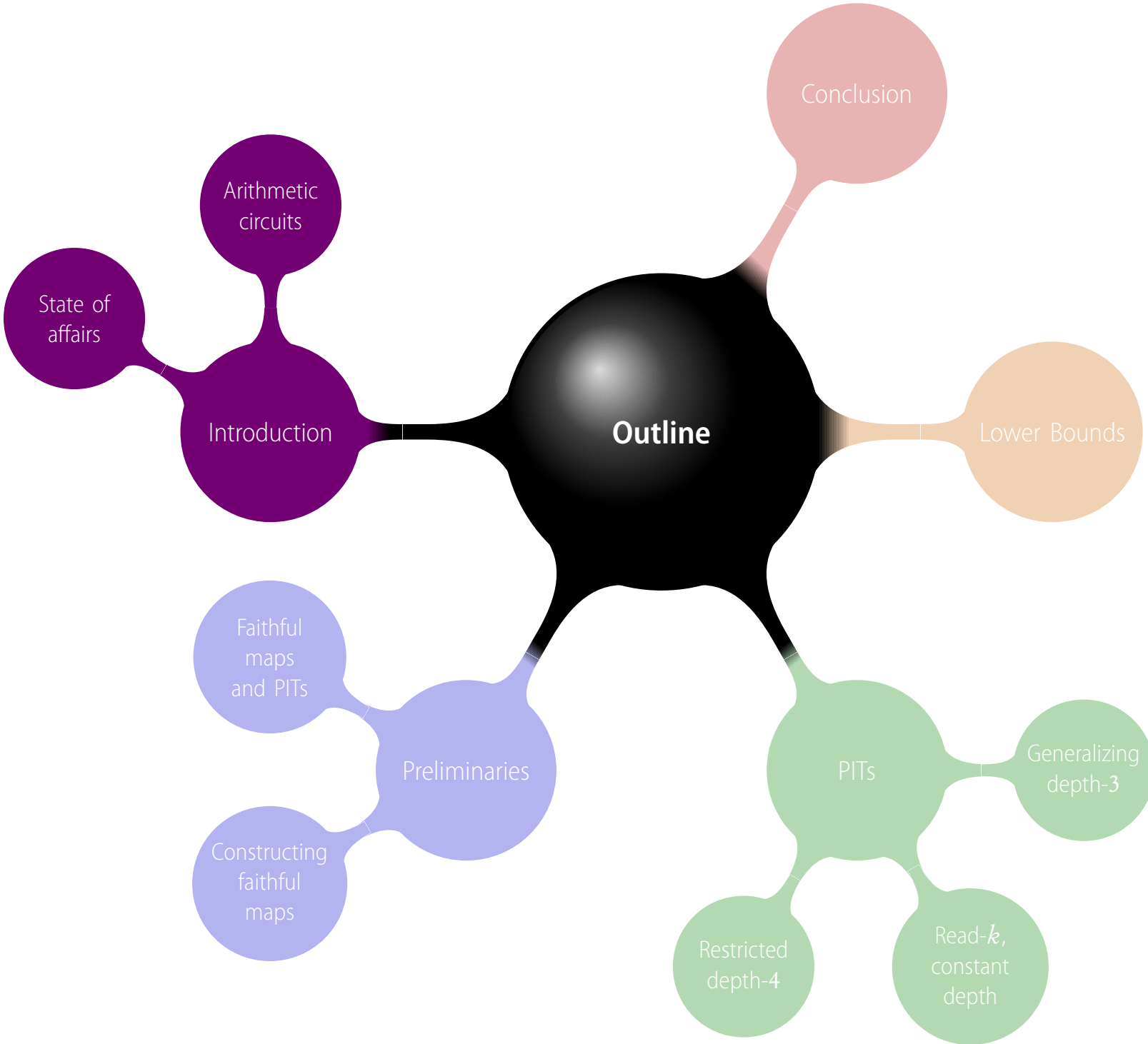
Ramprasad Saptharishi

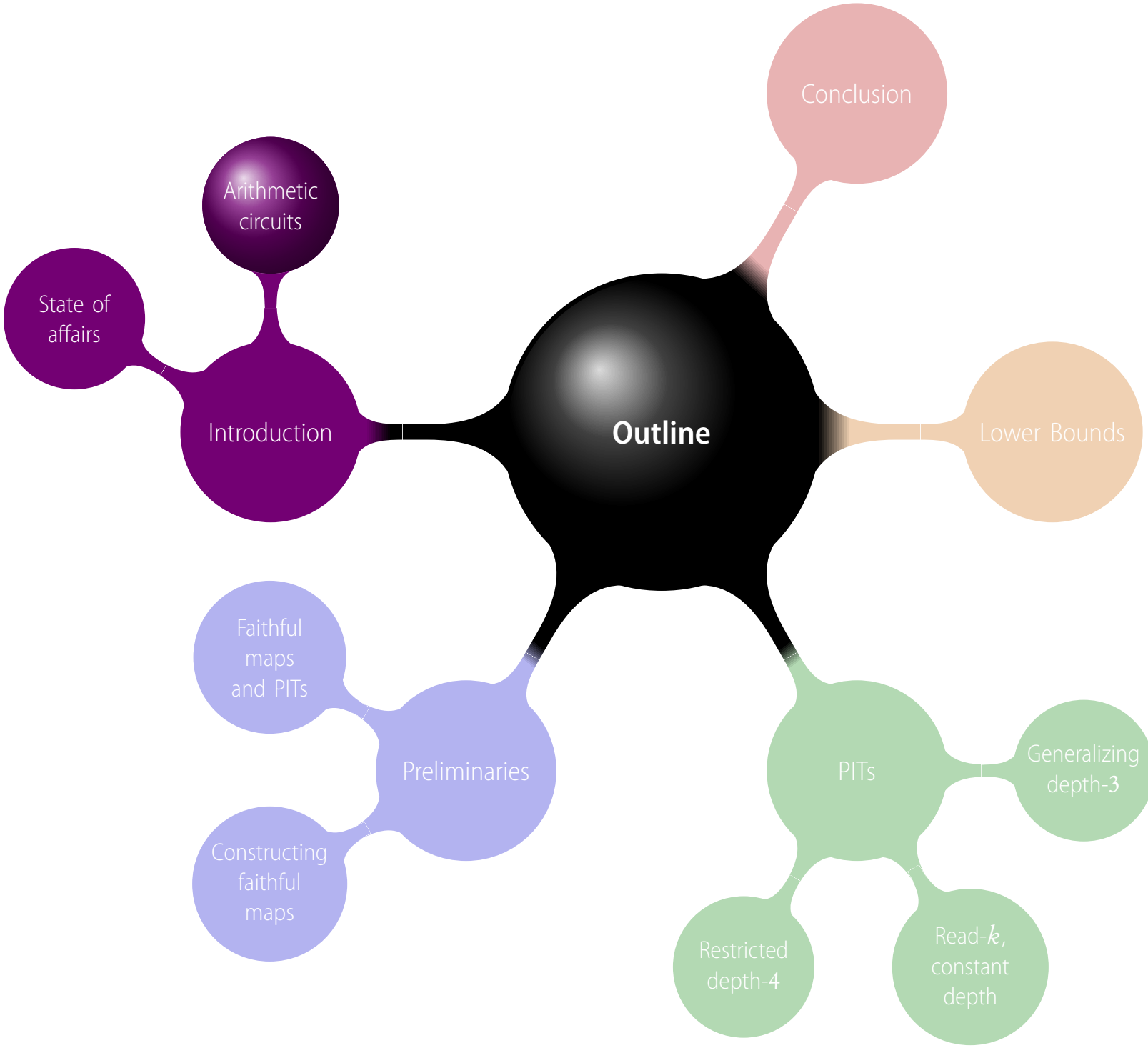
Nitin Saxena

Microsoft Research India

January, 2012


Outline
Introduction
Arithmetic circuits
State of affairs
Preliminaries
Faithful maps and PITs
Constructing faithful maps
Lower Bounds
PITs
Generalizing depth-3
Read-k, constant depth
Restricted depth-4
Conclusion


Outline
Introduction
Arithmetic circuits
State of affairs
Preliminaries
Faithful maps and PITs
Constructing faithful maps
Lower Bounds
PITs
Generalizing depth-3
Read-k, constant depth
Restricted depth-4
Conclusion

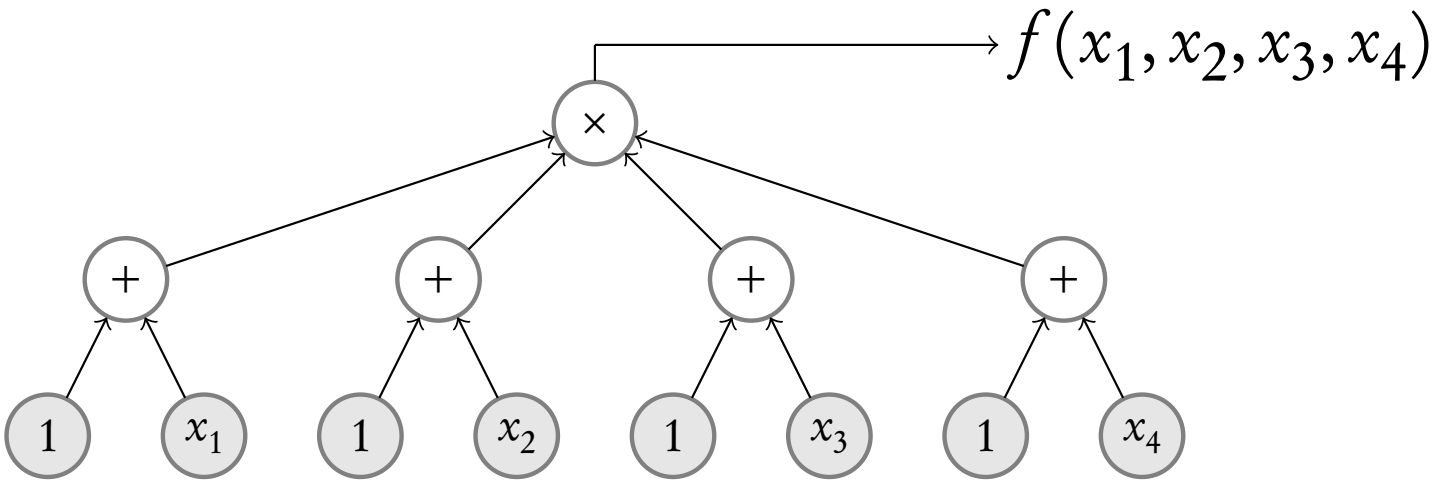

Outline
Introduction
Arithmetic circuits
State of affairs
Preliminaries
Faithful maps and PITs
Constructing faithful maps
Lower Bounds
PITs
Generalizing depth-3
Restricted depth-4
Read-k, constant depth
Conclusion

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

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

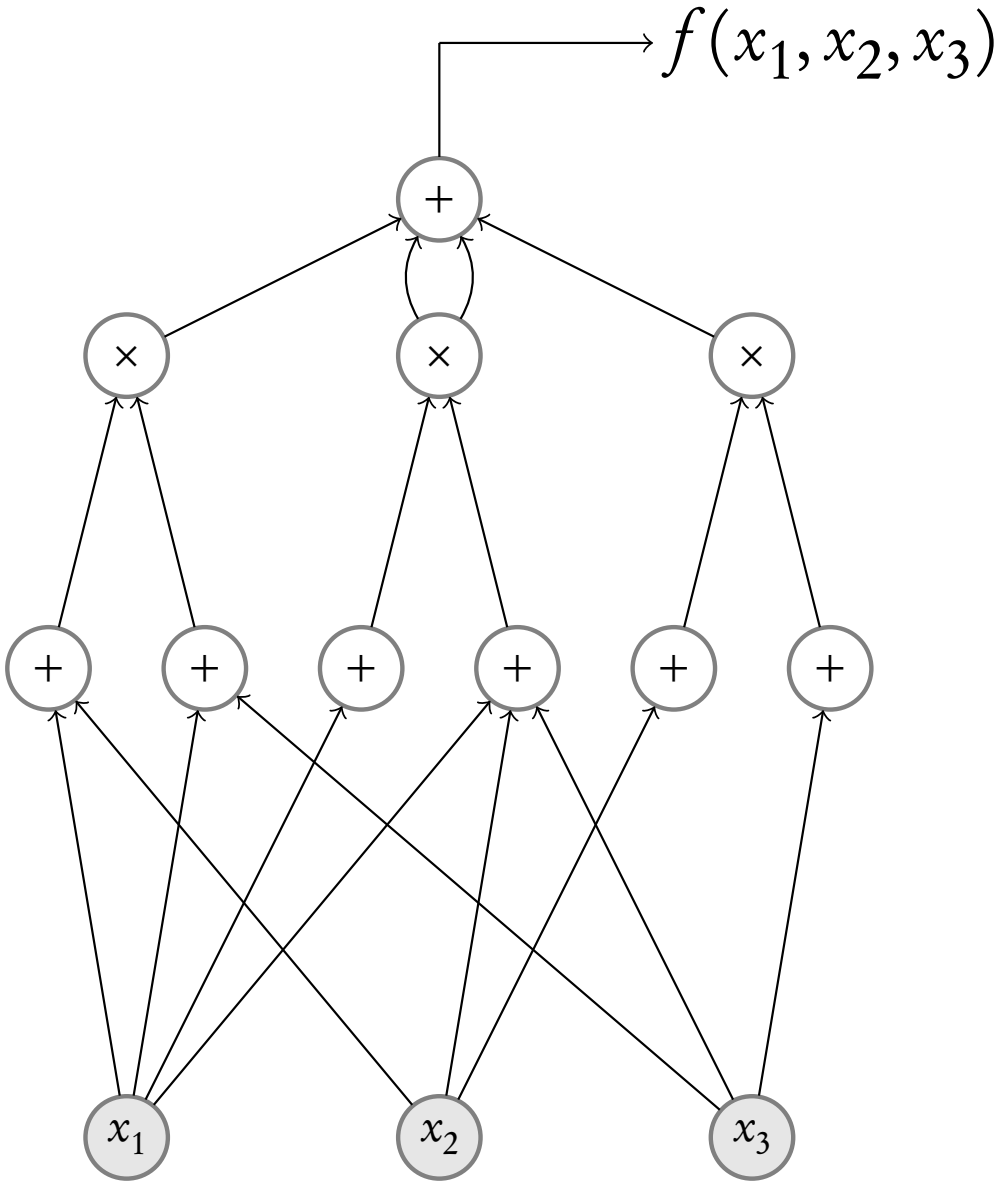
... certainly a more compact representation.

Arithmetic Formulae

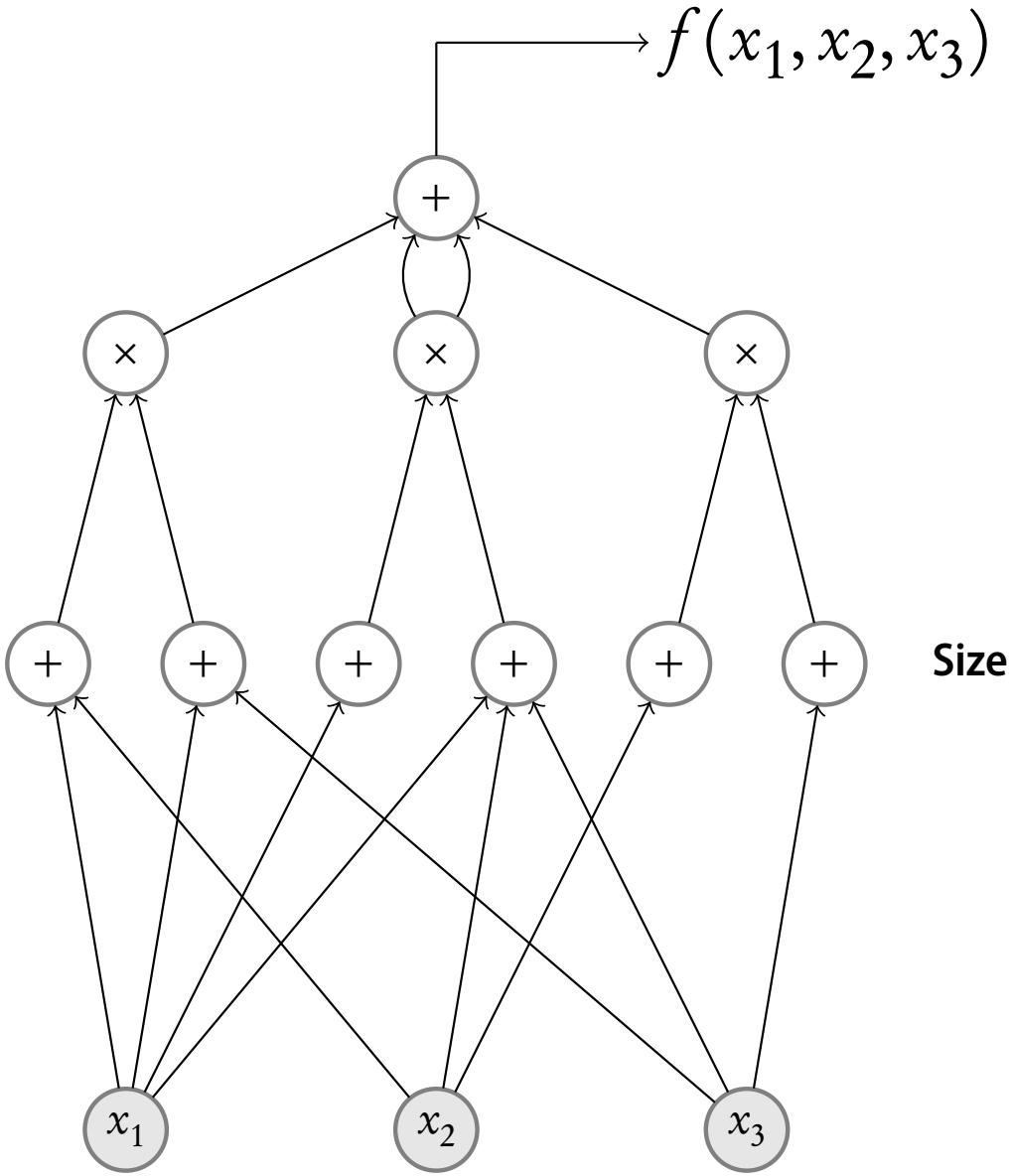


- Tree
- Leaves containing variables or constants

Arithmetic Circuits

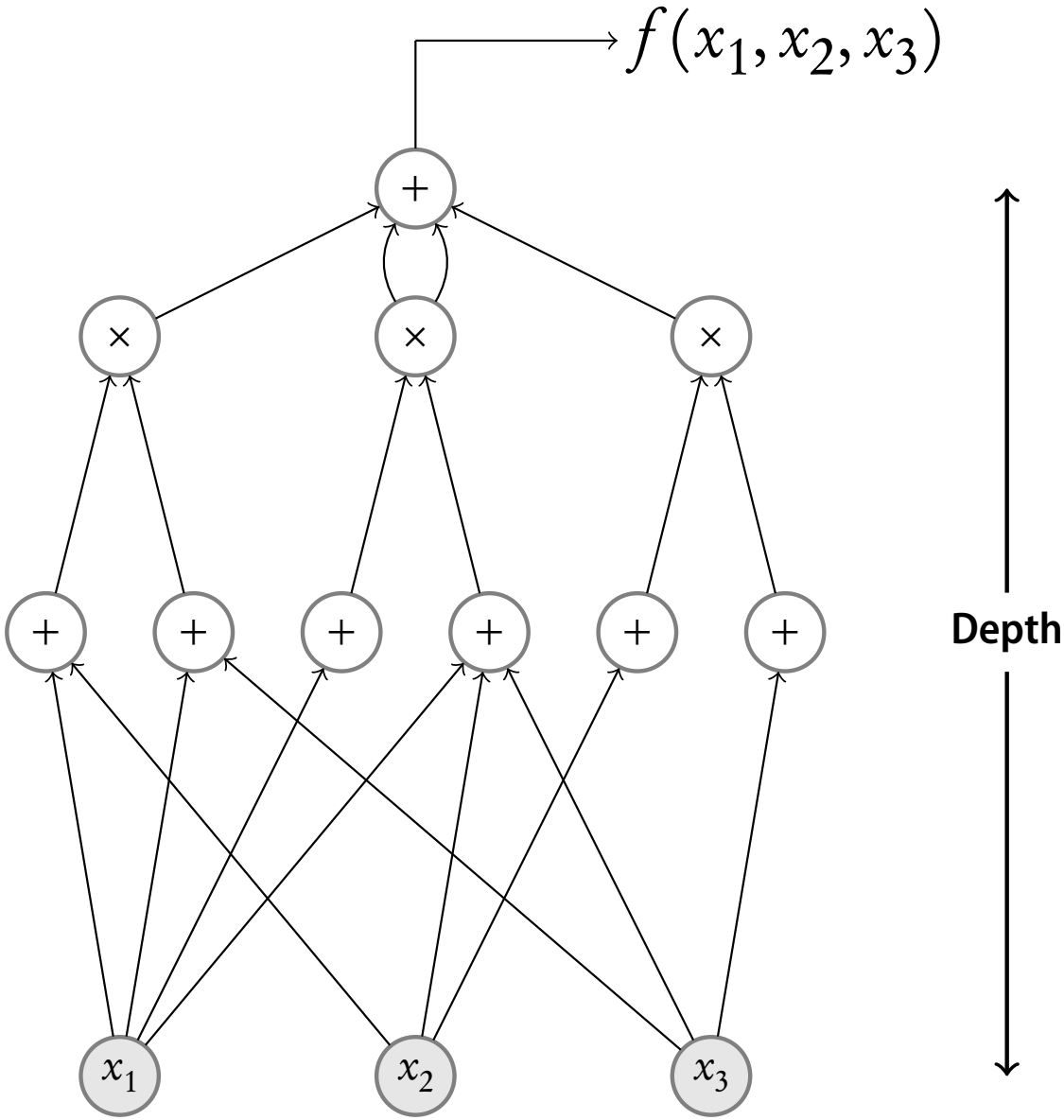


Arithmetic Circuits

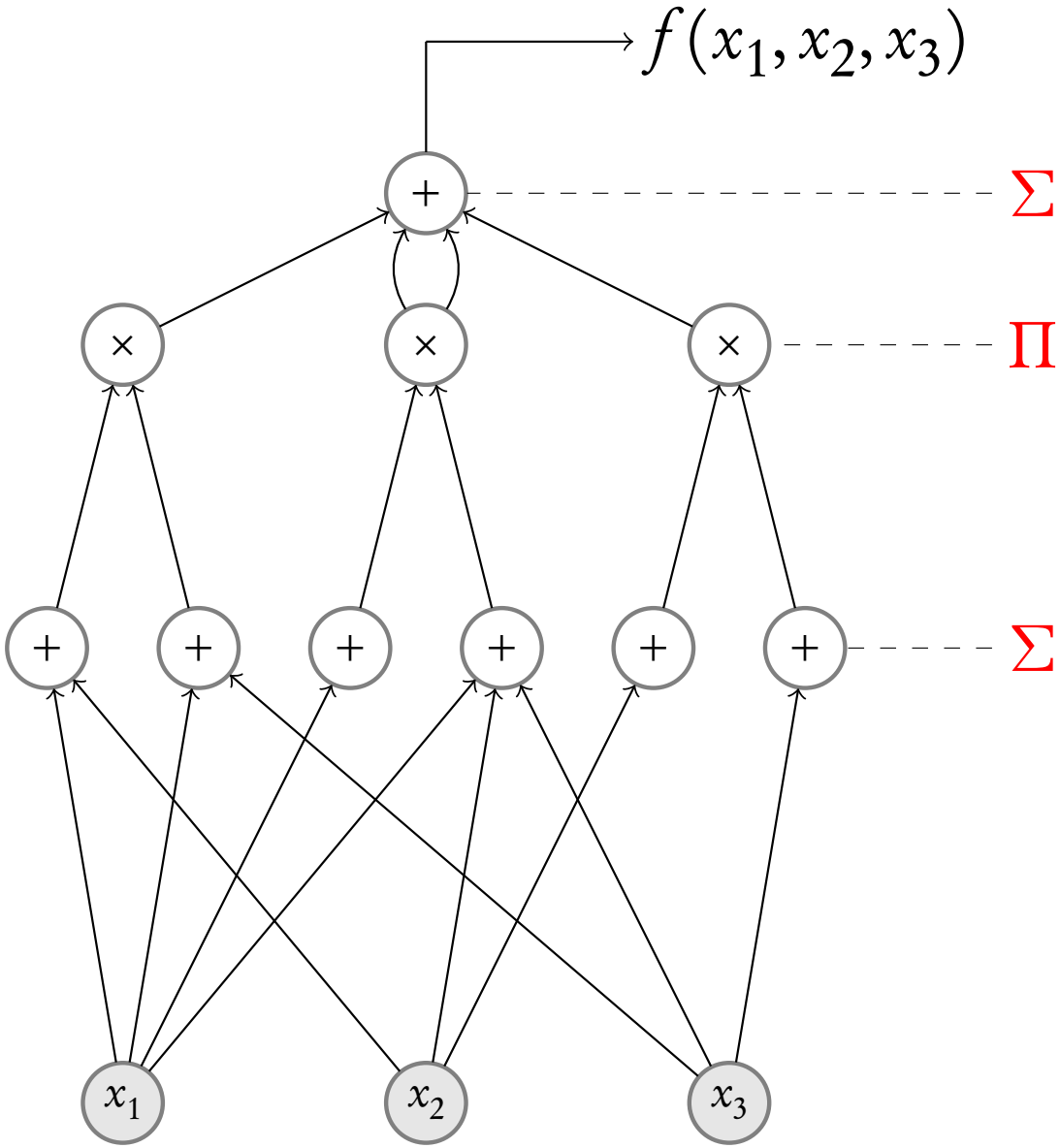


Size = number of gates

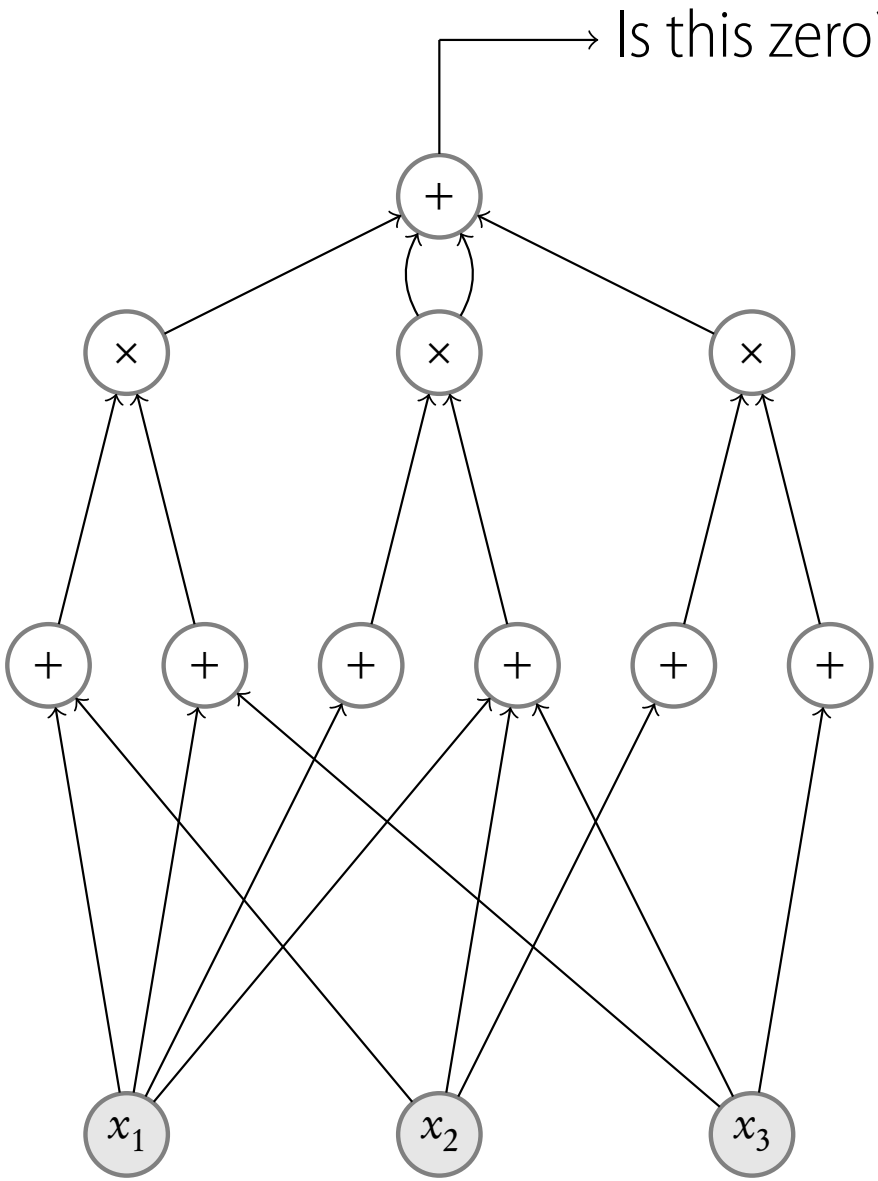
Arithmetic Circuits



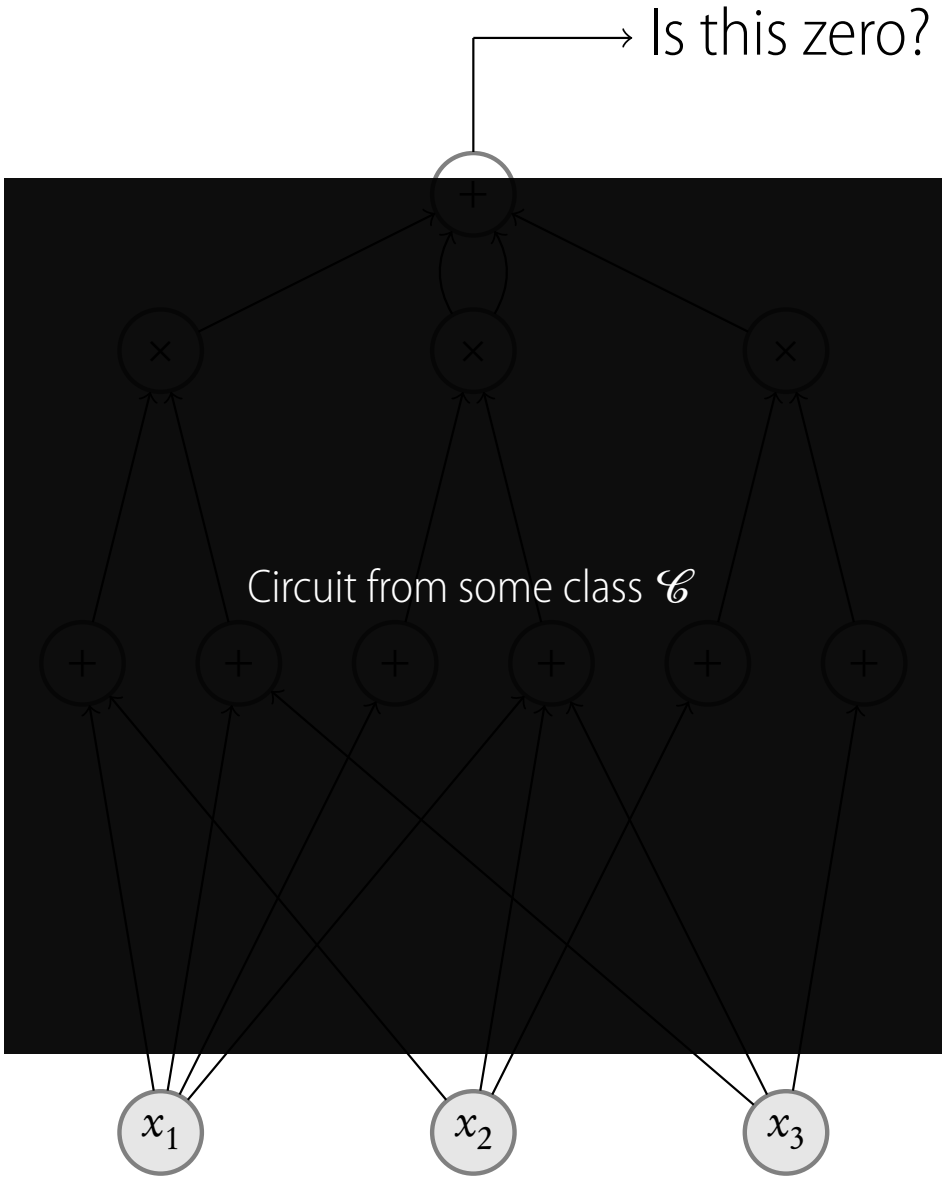
Arithmetic Circuits



Identity Testing of Arithmetic Circuits



Black-box Identity Testing of Arithmetic Circuits



The [Schwartz-Zippel-DeMillo-Lipton] Lemma

Lemma

Let f be a non-zero polynomial of degree d, and let $S \subseteq \mathbb{F}$. Then,

$$\Pr_{a_i \in S}[f(a_1, \cdots, a_n) = 0] \quad \leq \quad \frac{d}{|S|}$$

The [Schwartz-Zippel-DeMillo-Lipton] Lemma

Lemma

Let f be a non-zero polynomial of degree d, and let $S \subseteq \mathbb{F}$. Then,

$$\Pr_{a_i \in S}[f(a_1, \cdots, a_n) = 0] \quad \leq \quad \frac{d}{|S|}$$

Thus, if $|S| \geq d + 1$, then S^n contains a witness.

The [Schwartz-Zippel-DeMillo-Lipton] Lemma

Lemma

Let f be a non-zero polynomial of degree d, and let $S \subseteq \mathbb{F}$. Then,

$$\Pr_{a_i \in S}[f(a_1, \cdots, a_n) = 0] \quad \leq \quad \frac{d}{|S|}$$

Thus, if $|S| \geq d + 1$, then S^n contains a witness.

Big Question: If f is computable by a small circuit, do we have polynomial sized hitting set?

Why do we care?

Part of many important results like $\mathbf{IP} = \mathbf{PSPACE}$, the $\mathbf{PCP}$ theorem, AKS primality test, etc.

Connections with lower bounds. [Kabanets-Impagliazzo03], [Agrawal05]: "Efficient PIT algorithms imply lower bounds"

Why do we care?

Part of many important results like $\mathbf{IP} = \mathbf{PSPACE}$, the $\mathbf{PCP}$ theorem, AKS primality test, etc.

Connections with lower bounds. [Kabanets-Impagliazzo03], [Agrawal05]: "Efficient PIT algorithms imply lower bounds"

> *"For the pessimist, this indicates that derandomizing identity testing is a hopeless problem. For the optimist, this means on the contrary that to obtain an arithmetic circuit lower bound, we 'simply' have to prove a good upper bound on identity testing."*
>
> *- [Kayal-Saraf09]*

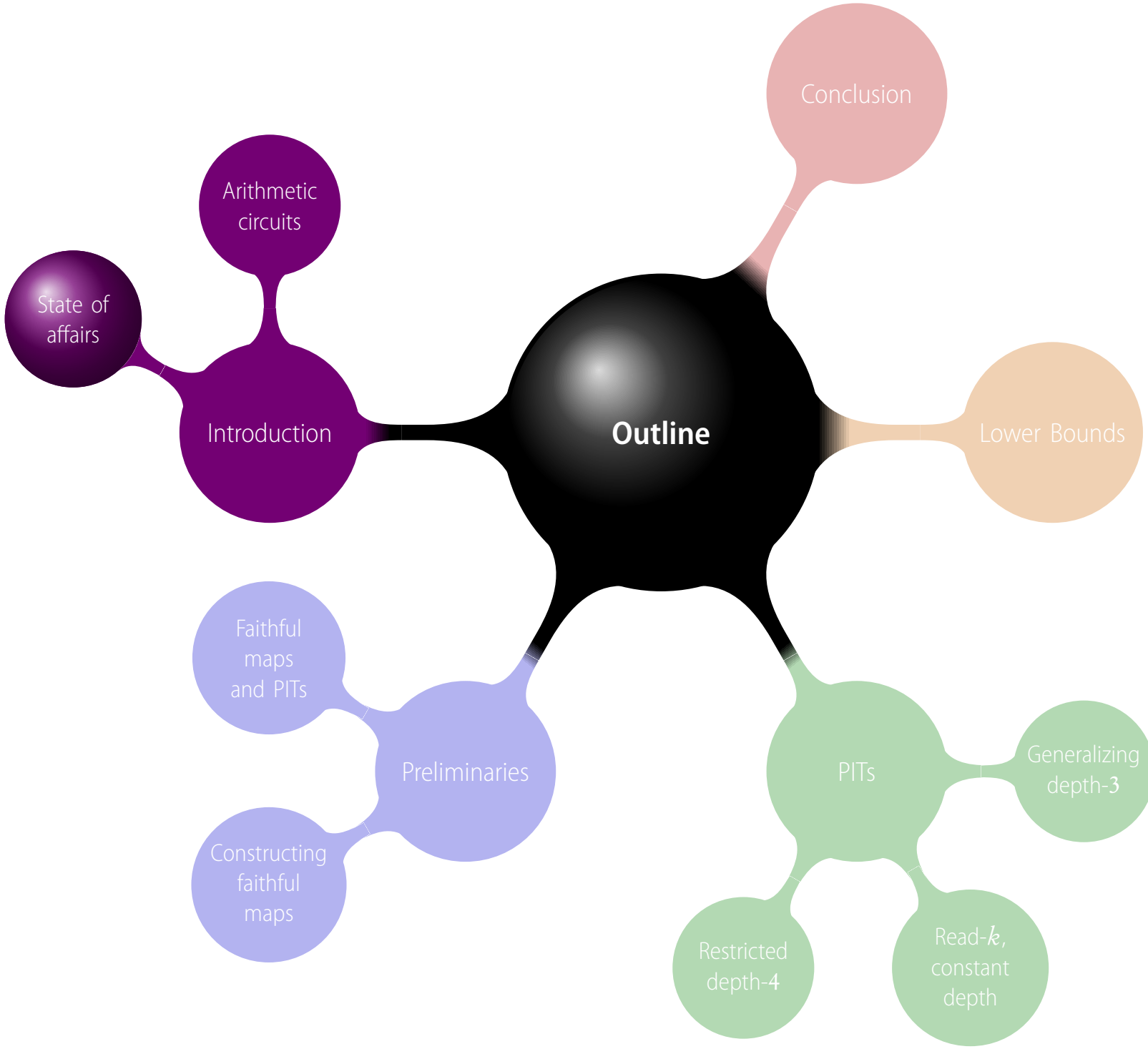
Why do we care?

Part of many important results like $\mathbf{IP} = \mathbf{PSPACE}$, the $\mathbf{PCP}$ theorem, AKS primality test, etc.

Connections with lower bounds. [Kabanets-Impagliazzo03], [Agrawal05]: "Efficient PIT algorithms imply lower bounds"

> *"For the pessimist, this indicates that derandomizing identity testing is a hopeless problem. For the optimist, this means on the contrary that to obtain an arithmetic circuit lower bound, we 'simply' have to prove a good upper bound on identity testing."*
>
> - *[Kayal-Saraf09]*

Of course, it is a natural problem!


Outline
Introduction
State of affairs
Arithmetic circuits
Preliminaries
Faithful maps and PITs
Constructing faithful maps
Lower Bounds
PITs
Generalizing depth-3
Restricted depth-4
Read-k, constant depth
Conclusion

State of affairs

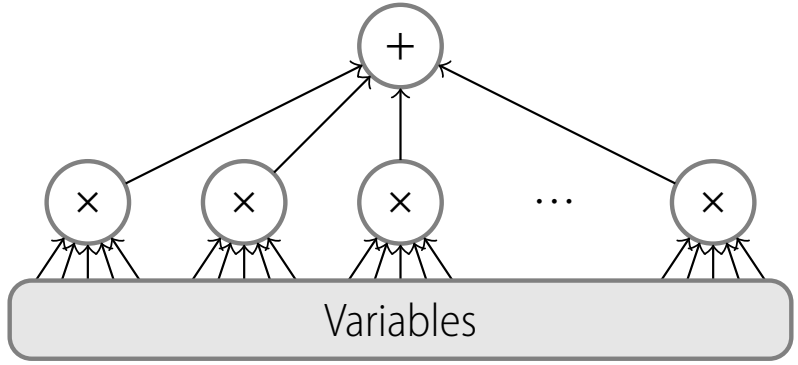
"If you can't solve a problem, then there is an easier problem you can solve: find it."

- George Pólya

Identity tests of restricted types of circuits:

- Formulae:
  1. Bounded depth formulae?
  2. Bounded read formulae?

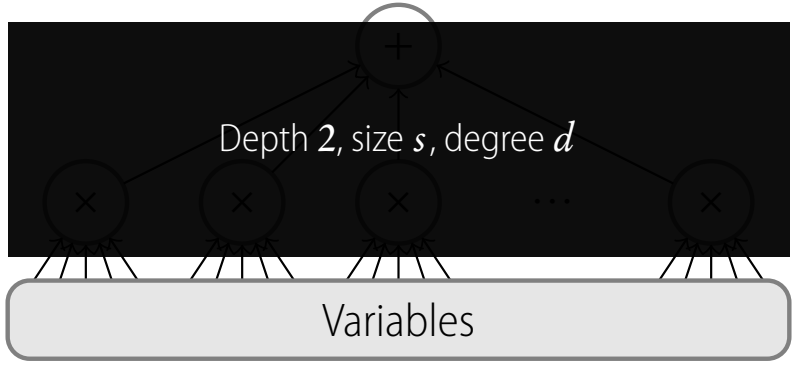
State of affairs for $\Sigma\Pi$ *Circuits*



$$f \quad = \quad \sum_{i=1}^{\text{poly}} \text{monomial}_i$$

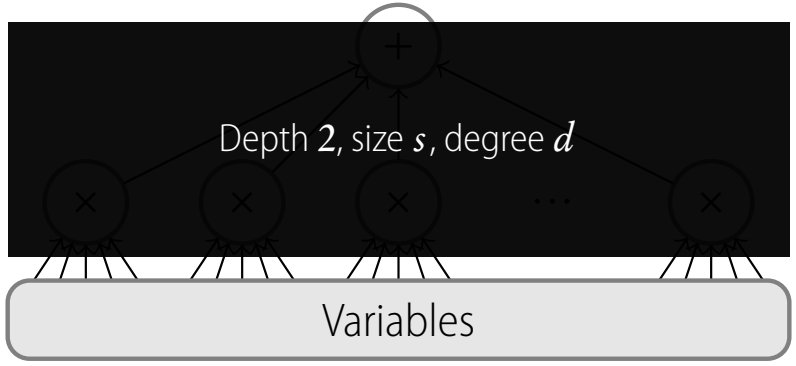
Depth **2** is easy (sparse polynomials)

State of affairs for $\Sigma\Pi$ *Circuits*



Black-box not-too-hard as well.

State of affairs for $\Sigma\Pi$ *Circuits*

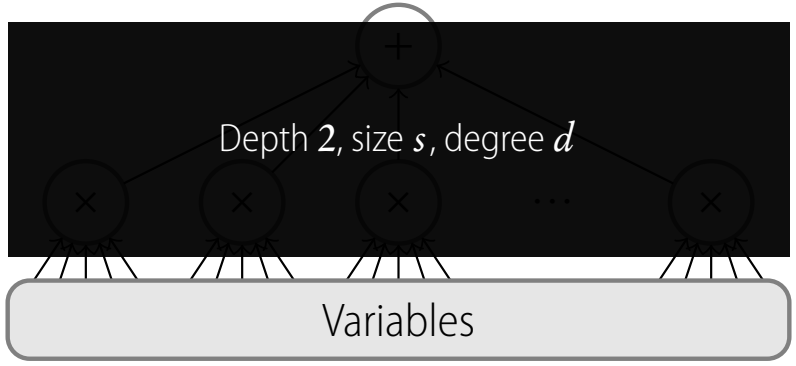


Black-box not-too-hard as well.

$$\Phi: \quad x_i \mapsto u^{(d+1)^i}$$

Works, but exponential degree

State of affairs for $\Sigma\Pi$ Circuits



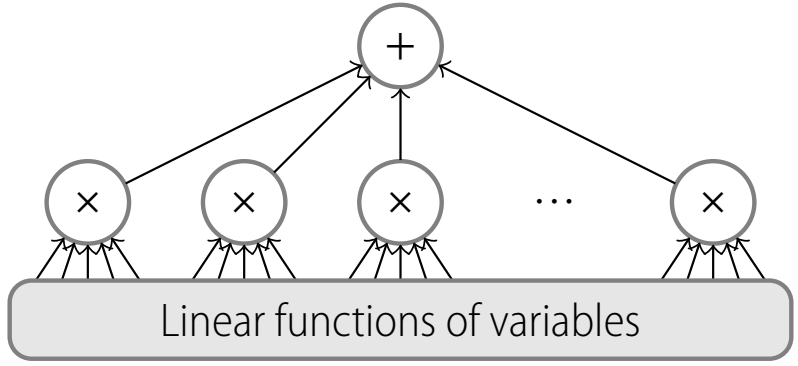
Black-box not-too-hard as well.

$$\Phi_r : \quad x_i \mapsto u^{(d+1)^i \bmod r}$$

Not too many bad r's

Hint: u^a and u^b collide if and only if $r \mid (a-b)$

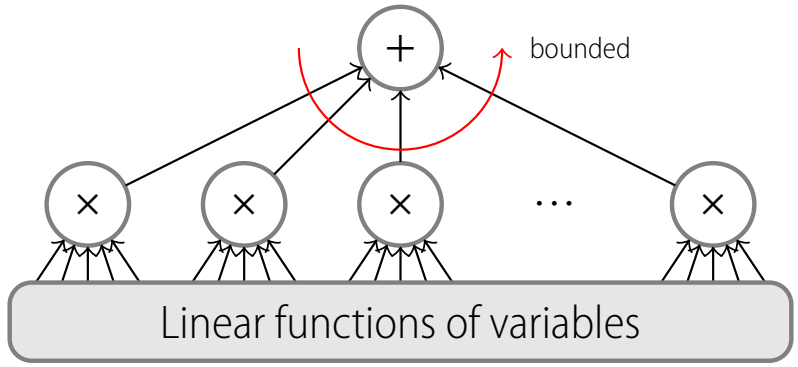
State of affairs for $\Sigma\Pi\Sigma$ *Circuits*



$$f \quad = \quad \sum_{i=1}^{k} \ell_{i1} \cdots \ell_{id}$$

PIT for even depth **3** circuits is open.

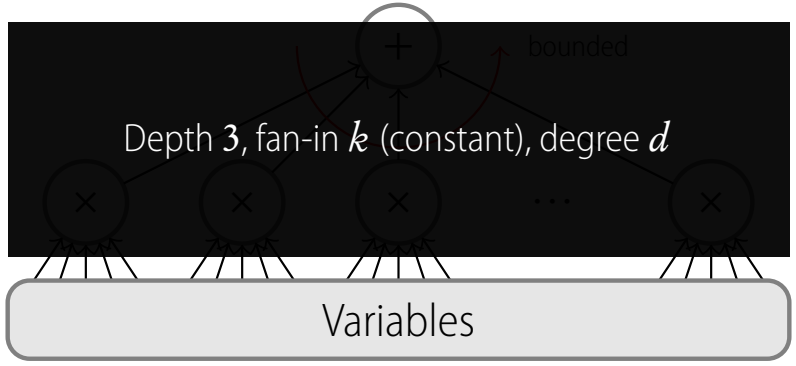
State of affairs for $\Sigma\Pi\Sigma(k)$ *Circuits*



$$f \quad = \quad \sum_{i=1}^{k} \ell_{i1} \cdots \ell_{id}$$

[KayalSaxena07] : PIT in time $\mathbf{poly}(s^k)$

State of affairs for $\Sigma\Pi\Sigma(k)$ *Circuits*

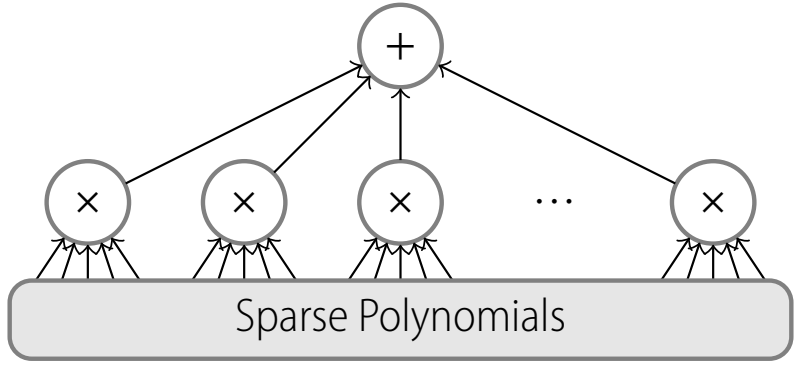


$$f \quad = \quad \sum_{i=1}^{k} \ell_{i1} \cdots \ell_{id}$$

[KayalSaxena07] : PIT in time $\mathbf{poly}(s^k)$

[SaxenaSeshadri11]: Black-box PIT in time $\mathbf{poly}(s^k)$

State of affairs for $\Sigma\Pi\Sigma\Pi$ *Circuits*

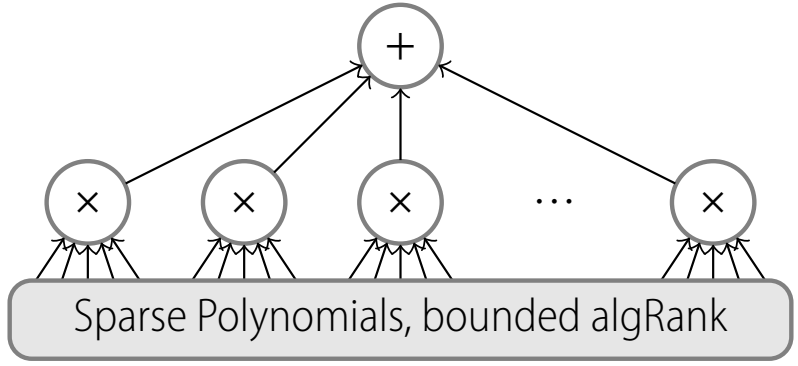


$$f \quad = \quad \sum_{i=1}^{\text{poly}} g_{i1} \cdots g_{id}$$

[AgrawalVinay08] : Black-box PIT for depth **4** implies $n^{O(\log n)}$ black-box PIT for any depth!

Depth **4** is (almost) as hard as the general case.

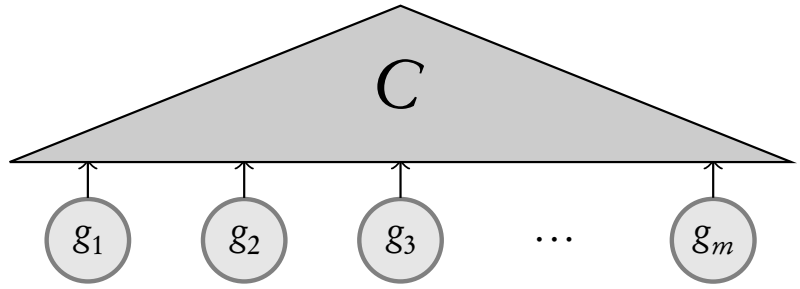
State of affairs for $\Sigma\Pi\Sigma\Pi$ *Circuits*



$$f \quad = \quad \sum_{i=1}^{\text{poly}} g_{i1} \cdots g_{id} \quad \text{with } \mathbf{algRank}\left\{ g_{ij} \right\} \leq k$$

[BeeckenMittmannSaxena11]: Polynomial time black-box PIT

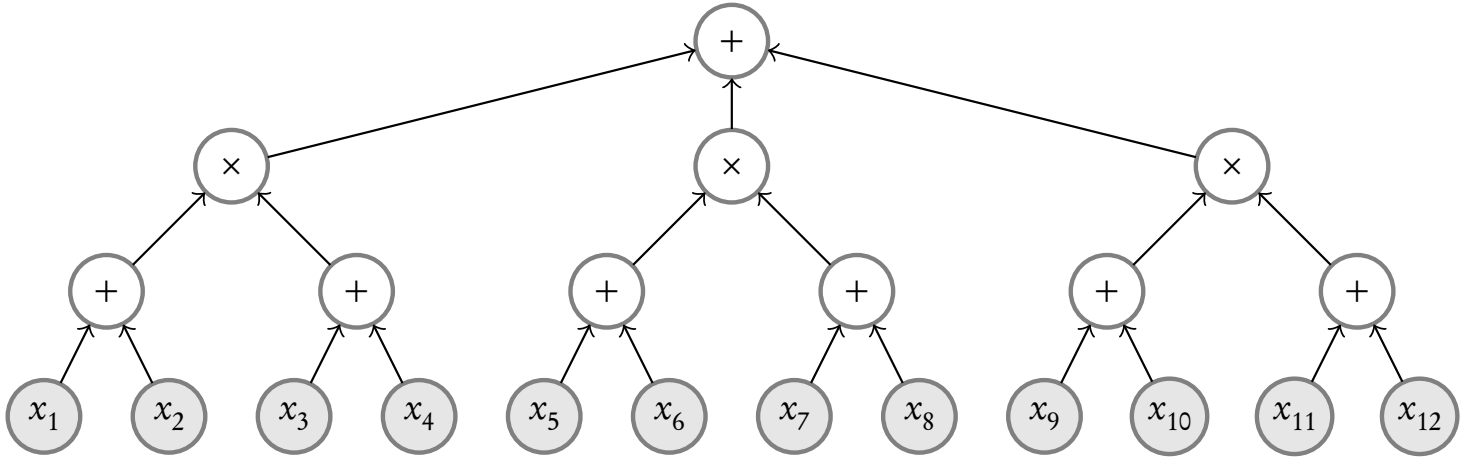
State of affairs for $\Sigma\Pi\Sigma\Pi$ *Circuits*



$$f \quad = \quad C(g_1, \cdots, g_m) \quad \text{with } \mathbf{algRank}\{g_i\} \leq k$$

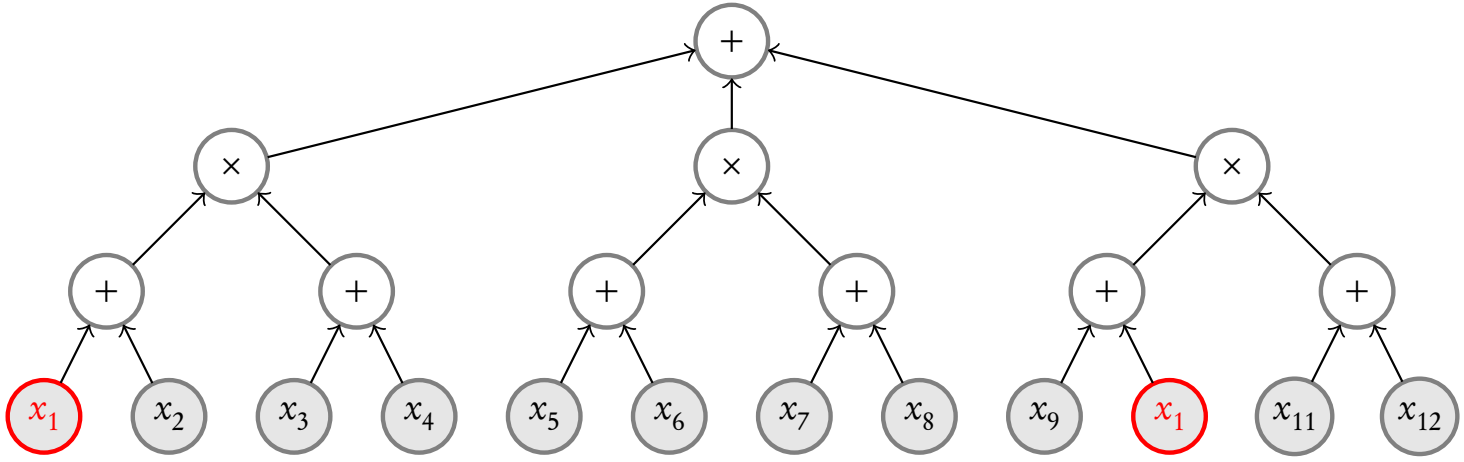
[BeeckenMittmannSaxena11]: Polynomial time black-box PIT

State of affairs for bounded read formulae



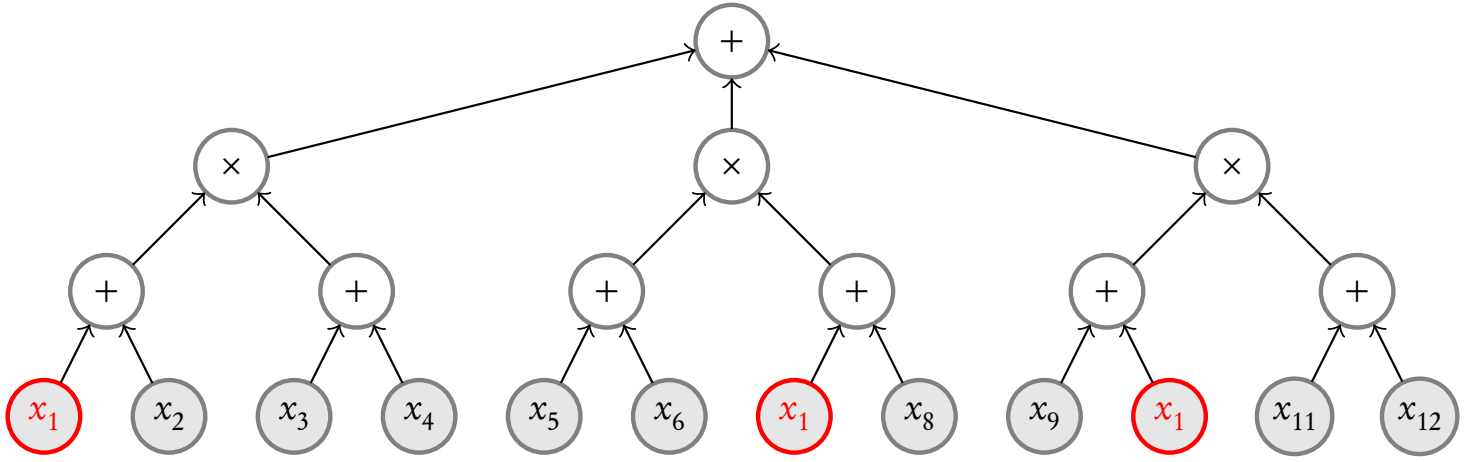
Read-**1** formula

State of affairs for bounded read formulae



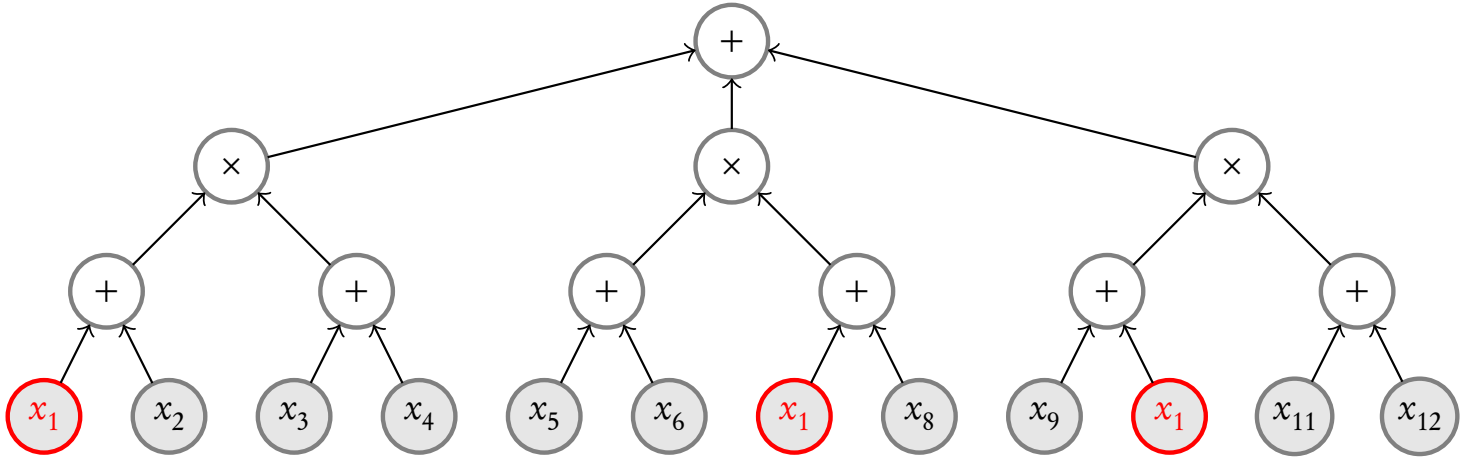
Read-**2** formula

State of affairs for bounded read formulae



Read-**3** formula

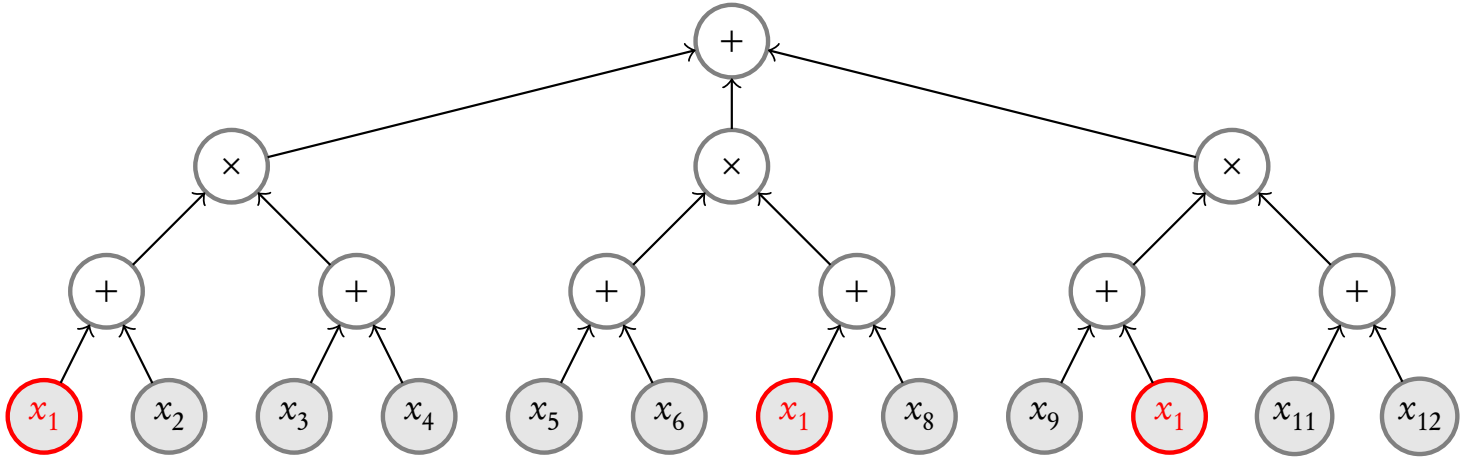
State of affairs for bounded read formulae



Read-k formula

Status of PIT:

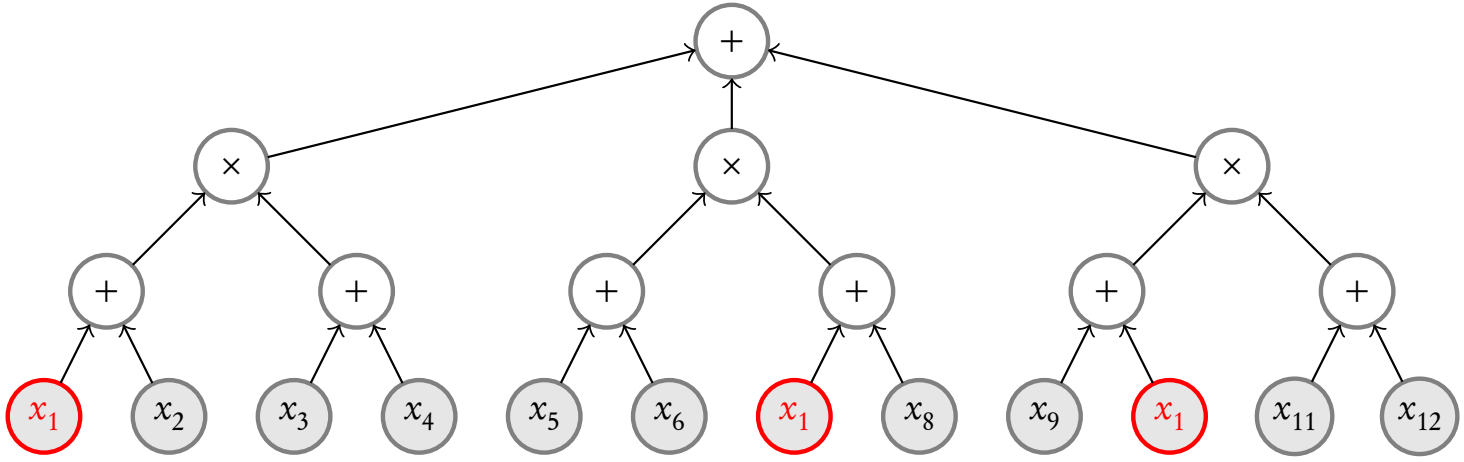
State of affairs for bounded read formulae



Read-k formula

Status of PIT: Open!

State of affairs for bounded read formulae



Read-k multilinear formula

Status of PIT:

- [SarafVolkovich11]: Polytime black-box PIT for multilinear $\Sigma\Pi\Sigma\Pi(k)$.
- [Anderson-vanMelkebeek-Volkovich11]: Polytime black-box PIT for constant depth, multilinear, read-k formulae.
  Quasi-poly black-box PIT for arbitrary depth, multilinear read-k formulae, and polynomial time non-blackbox PIT.

Summary of results

Model	**Best known PIT**	**Idea**
$\Sigma\Pi\Sigma(k)$	s^k black-box	CRT over local rings
bounded $\mathbf{algRank}$ $\Sigma\Pi\Sigma\Pi$	Polytime black-box	Jacobian
$\Sigma\Pi\Sigma\Pi(k)$ multilinear	s^{k^3} black-box	sparsity bounds
multilinear read-k	Quasi-poly black-box	shattering, fragmentation under partial derivatives

Summary of results

Model	**Best known PIT**	**Idea**
$T_1 + \cdots + T_k \stackrel{?}{=} 0$	s^k black-box	CRT over local rings
bounded $\mathbf{algRank}$ $\Sigma\Pi\Sigma\Pi$	Polytime black-box	Jacobian
$\Sigma\Pi\Sigma\Pi(k)$ multilinear	s^{k^3} black-box	sparsity bounds
multilinear read-k	Quasi-poly black-box	shattering, fragmentation under partial derivatives

Summary of our results

Model*	**Best known PIT**	**Idea**
$C(T_1, \cdots, T_m) \stackrel{?}{=} 0$ $\text{algRank}\{T_1, \cdots, T_m\} \leq k$	s^k black-box	CRT over local rings
bounded algRank $\Sigma\Pi\Sigma\Pi$	Polytime black-box	Jacobian
$\Sigma\Pi\Sigma\Pi(k)$ multilinear	s^{k^3} black-box	sparsity bounds
multilinear read-k	Quasi-poly black-box	shattering, fragmentation under partial derivatives

Summary of our results

Model*	**Best known PIT**	**Idea**
$C(T_1, \cdots, T_m) \stackrel{?}{=} 0$ $\text{algRank}\{T_1, \cdots, T_m\} \leq k$	s^k black-box	CRT over local rings
bounded algRank $\Sigma\Pi\Sigma\Pi$	Polytime black-box	Jacobian
$\Sigma\Pi\Sigma\Pi$~~(k)~~ ~~multilinear~~ read-k	s^{k^2} black-box	sparsity bounds
multilinear read-k	Quasi-poly black-box	shattering, fragmentation under partial derivatives

Summary of our results

Model*	**Best known PIT**	**Idea**
$C(T_1, \cdots, T_m) \stackrel{?}{=} 0$ $\text{algRank}\{T_1, \cdots, T_m\} \leq k$	s^k black-box	CRT over local rings
bounded algRank $\Sigma\Pi\Sigma\Pi$	Polytime black-box	Jacobian
$\Sigma\Pi\Sigma\Pi$~~(k)~~ ~~multilinear~~ read-k	s^{k^2} black-box	sparsity bounds
~~multilinear~~ read-k constant depth	Polytime black-box	shattering, fragmentation under partial derivatives

Summary of our results

Model*	**Best known PIT**	**Idea**
$C(T_1, \cdots, T_m) \stackrel{?}{=} 0$ $\text{algRank}\{T_1, \cdots, T_m\} \leq k$	s^k black-box	Jacobian
bounded algRank $\Sigma\Pi\Sigma\Pi$	Polytime black-box	Jacobian
$\Sigma\Pi\Sigma\Pi(k)$ ~~multilinear~~ read-k	s^{k^2} black-box	Jacobian
~~multilinear~~ read-k constant depth	Polytime black-box	Jacobian

Summary of our results

Model*	**Best known PIT**	**Idea**
$C(T_1, \cdots, T_m) \stackrel{?}{=} 0$ $\text{algRank}\{T_1, \cdots, T_m\} \leq k$	s^k black-box	Jacobian
bounded algRank $\Sigma\Pi\Sigma\Pi$	Polytime black-box	Jacobian
$\Sigma\Pi\Sigma\Pi(k)$ ~~multilinear~~ read-k	s^{k^2} black-box	Jacobian
~~multilinear~~ read-k constant depth	Polytime black-box	Jacobian

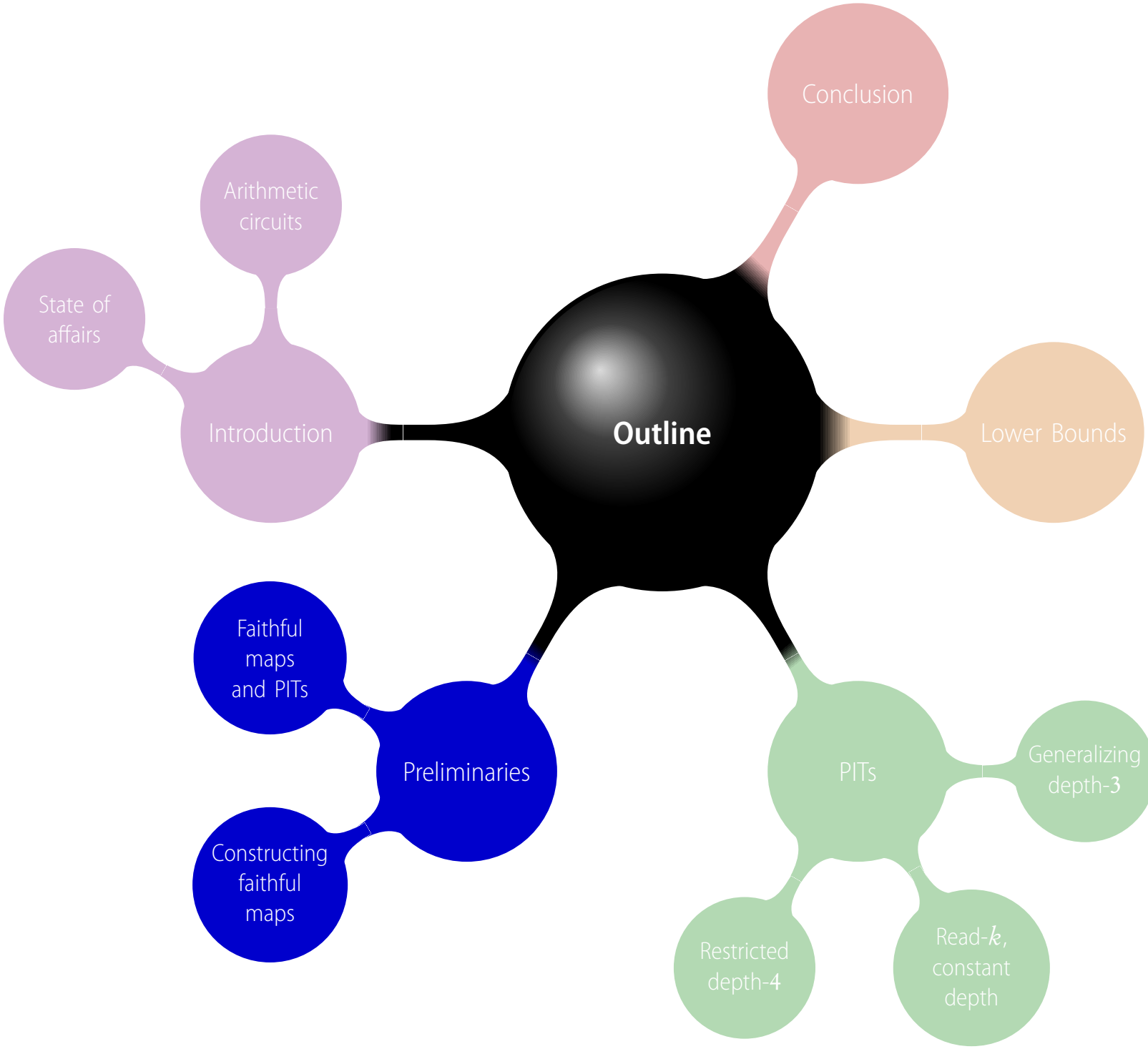
... and some lower bounds

Summary of our results

Model*	**Best known PIT**	**Idea**
$C(T_1, \cdots, T_m) \stackrel{?}{=} 0$ $\text{algRank}\{T_1, \cdots, T_m\} \leq k$	s^k black-box	Jacobian
bounded algRank $\Sigma\Pi\Sigma\Pi$	Polytime black-box	Jacobian
$\Sigma\Pi\Sigma\Pi(k)$ ~~multilinear~~ read-k	s^{k^2} black-box	Jacobian
~~multilinear~~ read-k constant depth	Polytime black-box	Jacobian

... and some lower bounds

*: $\text{char}(\mathbb{F}) = 0$ or large


Outline
Introduction
Arithmetic circuits
State of affairs
Preliminaries
Faithful maps and PITs
Constructing faithful maps
Lower Bounds
PITs
Generalizing depth-3
Restricted depth-4
Read-k, constant depth
Conclusion

Rank of a $\Sigma\Pi\Sigma$ circuit

$$C = \sum_i \prod_j \ell_{ij}$$

Rank of a $\Sigma\Pi\Sigma$ circuit

$$C = \sum_i \prod_j \ell_{ij}$$

$$\text{rank}(C) \stackrel{\text{def}}{=} \dim\left\{\ell_{ij}\right\}$$

the maximum number of linearly independent ℓ_{ij}'s

Rank of a $\Sigma\Pi\Sigma$ circuit

$$C = \sum_i \prod_j \ell_{ij}$$

$$\text{rank}(C) \stackrel{\text{def}}{=} \dim\left\{\ell_{ij}\right\}$$

the maximum number of linearly independent ℓ_{ij}'s

If $\text{rank}(C)$ is "small" (say less than k):

1. Construct a linear transformation $\Psi : \mathbb{F}[x_{[n]}] \mapsto \mathbb{F}[y_{[k]}]$ such that $\dim\left\{\ell_{ij}\right\} = \dim\left\{\Psi(\ell_{ij})\right\}$.
2. Show that this preserves non-zeroness of C.
3. Use [DLSZ] on $\Psi(C)$ to get a hitting set of size $(d+1)^k$.

Rank of a ΣΠΣΠ circuit

$$C = \sum_i \prod_j f_{ij}$$

$$\text{rank}(C) \stackrel{\text{def}}{=} \dim\left\{\ell_{ij}\right\}$$

the maximum number of linearly independent ℓ_{ij}'s

If $\text{rank}(C)$ is "small" (say less than k):

1. Construct a linear transformation $\Psi : \mathbb{F}[x_{[n]}] \mapsto \mathbb{F}[y_{[k]}]$ such that $\dim\left\{\ell_{ij}\right\} = \dim\left\{\Psi(\ell_{ij})\right\}$.
2. Show that this preserves non-zeroness of C.
3. Use [DLSZ] on $\Psi(C)$ to get a hitting set of size $(d+1)^k$.

Rank of a ΣΠΣΠ circuit

$$C = \sum_i \prod_j f_{ij}$$

$\text{rank}(C) \stackrel{\text{def}}{=} \text{algRank}\left\{f_{ij}\right\}$

the maximum number of algebraically independent f_{ij}'s

If $\text{rank}(C)$ is "small" (say less than k):

1. Construct a linear transformation $\Psi : \mathbb{F}[x_{[n]}] \mapsto \mathbb{F}[y_{[k]}]$ such that $\dim\left\{\ell_{ij}\right\} = \dim\left\{\Psi(\ell_{ij})\right\}$.
2. Show that this preserves non-zeroness of C.
3. Use [DLSZ] on $\Psi(C)$ to get a hitting set of size $(d+1)^k$.

Rank of a $\Sigma\Pi\Sigma\Pi$ *circuit*

$$C = \sum_i \prod_j f_{ij}$$

$$\text{rank}(C) \stackrel{\text{def}}{=} \text{algRank}\left\{f_{ij}\right\}$$

the maximum number of algebraically independent f_{ij}'s

If $\text{rank}(C)$ is "small" (say less than k):

1. Construct a homomorphism $\Psi : \mathbb{F}[x_{[n]}] \mapsto \mathbb{F}[y_{[k]}]$ such that $\text{algRank}\left\{f_{ij}\right\} = \text{algRank}\left\{\Psi(f_{ij})\right\}$.
2. Show that this preserves non-zeroness of C.
3. Use [DLSZ] on $\Psi(C)$ to get a hitting set of size $(\text{poly}(d)+1)^k$.

Definition

$\{f_1, \cdots, f_m\}$ are algebraically independent if there is no non-trivial polynomial relation between them. That is,

$$H(f_1, \cdots, f_m) = 0 \quad \Longleftrightarrow \quad H = 0$$

Formal definitions

Definition

$\{f_1, \cdots, f_m\}$ are algebraically independent if there is no non-trivial polynomial relation between them. That is,

$$H(f_1, \cdots, f_m) = 0 \quad \Longleftrightarrow \quad H = 0$$

Definition

The algebraic rank (**algRank**) of $\{f_1, \cdots, f_m\}$ is the size of the largest algebraically independent subset.

Formal definitions

Definition

$\{f_1, \cdots, f_m\}$ are algebraically independent if there is no non-trivial polynomial relation between them. That is,

$$H(f_1, \cdots, f_m) = 0 \quad \Longleftrightarrow \quad H = 0$$

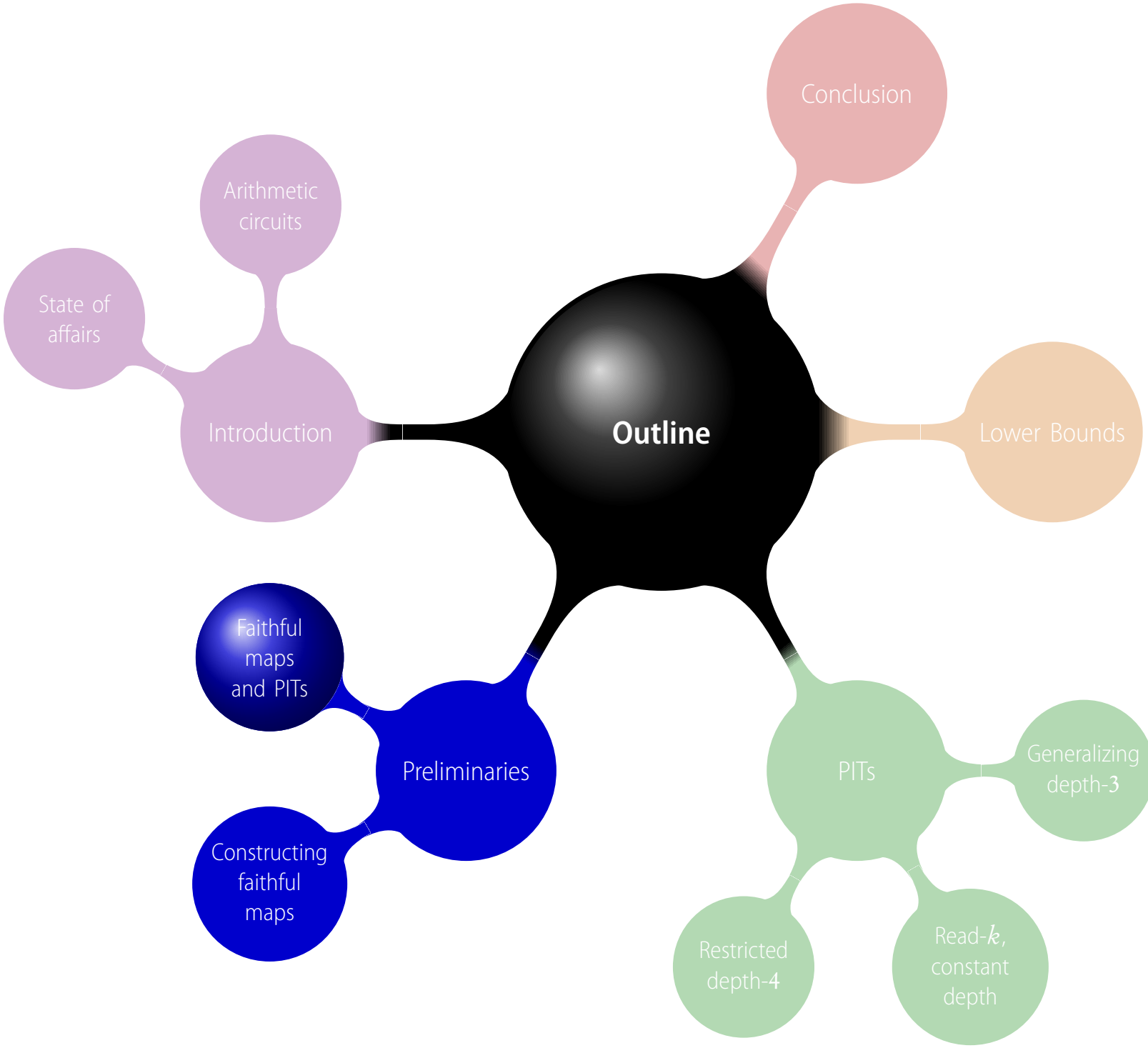
Definition

The algebraic rank (**algRank**) of $\{f_1, \cdots, f_m\}$ is the size of the largest algebraically independent subset.

Definition

A map $\Psi : \mathbb{F}[x_1, \cdots, x_n] \to \mathbb{F}[y_1, \cdots, y_k]$ is faithful for $\{f_1, \cdots, f_m\}$ if

$$\text{algRank}\{f_1, \cdots, f_m\} \quad = \quad \text{algRank}\{\Psi(f_1), \cdots, \Psi(f_m)\}$$


Outline
Introduction
Arithmetic circuits
State of affairs
Preliminaries
Faithful maps and PITs
Constructing faithful maps
Lower Bounds
PITs
Generalizing depth-3
Restricted depth-4
Read-k, constant depth
Conclusion

Faithful maps preserve non-zeroness

Theorem (Beecken-Mittmann-Saxena)

If $\Psi : \mathbb{F}[x_1, \cdots, x_n] \to \mathbb{F}[y_1, \cdots, y_k]$ *is faithful for* $\{f_1, \cdots, f_m\}$*, then for any* C

$$C(f_1, \cdots, f_m) \neq 0 \quad \textit{if and only if} \quad \Psi(C(f_1, \cdots, f_m)) \neq 0$$

Proof.

□

Faithful maps preserve non-zeroness

Theorem (Beecken-Mittmann-Saxena)

If $\Psi : \mathbb{F}[x_1, \cdots, x_n] \to \mathbb{F}[y_1, \cdots, y_k]$ *is faithful for* $\{f_1, \cdots, f_m\}$*, then for any* C

$$C(f_1, \cdots, f_m) \neq 0 \quad \textit{if and only if} \quad \Psi(C(f_1, \cdots, f_m)) \neq 0$$

Proof.

Say $\{f_1, \cdots, f_r\}$ is a maximal algebraically independent set that is preserved by Ψ.

Faithful maps preserve non-zeroness

Theorem (Beecken-Mittmann-Saxena)

If $\Psi : \mathbb{F}[x_1, \cdots, x_n] \to \mathbb{F}[y_1, \cdots, y_k]$ *is faithful for* $\{f_1, \cdots, f_m\}$*, then for any* C

$$C(f_1, \cdots, f_m) \neq 0 \quad \textit{if and only if} \quad \Psi(C(f_1, \cdots, f_m)) \neq 0$$

Proof.

Say $\{f_1, \cdots, f_r\}$ is a maximal algebraically independent set that is preserved by Ψ. Then, $\mathbb{F}(f_1, \cdots, f_m) = \mathbb{F}(f_1, \cdots, f_r)[f_{r+1}, \cdots, f_m]$.

□

Faithful maps preserve non-zeroness

Theorem (Beecken-Mittmann-Saxena)

If $\Psi : \mathbb{F}[x_1, \cdots, x_n] \to \mathbb{F}[y_1, \cdots, y_k]$ *is faithful for* $\{f_1, \cdots, f_m\}$*, then for any* C

$$C(f_1, \cdots, f_m) \neq 0 \quad \textit{if and only if} \quad \Psi(C(f_1, \cdots, f_m)) \neq 0$$

Proof.

Say $\{f_1, \cdots, f_r\}$ is a maximal algebraically independent set that is preserved by Ψ. Then, $\mathbb{F}(f_1, \cdots, f_m) = \mathbb{F}(f_1, \cdots, f_r)[f_{r+1}, \cdots, f_m]$.

$$C(f_1, \cdots, f_m) \cdot Q(f_1, \cdots, f_m) \quad = \quad 1$$

□

Faithful maps preserve non-zeroness

Theorem (Beecken-Mittmann-Saxena)

If $\Psi : \mathbb{F}[x_1, \cdots, x_n] \to \mathbb{F}[y_1, \cdots, y_k]$ *is faithful for* $\{f_1, \cdots, f_m\}$*, then for any* C

$$C(f_1, \cdots, f_m) \neq 0 \quad \textit{if and only if} \quad \Psi(C(f_1, \cdots, f_m)) \neq 0$$

Proof.

Say $\{f_1, \cdots, f_r\}$ is a maximal algebraically independent set that is preserved by Ψ. Then, $\mathbb{F}(f_1, \cdots, f_m) = \mathbb{F}(f_1, \cdots, f_r)[f_{r+1}, \cdots, f_m]$.

$$\begin{aligned} C(f_1, \cdots, f_m) \cdot Q(f_1, \cdots, f_m) &= 1 \\ \implies C(f_1, \cdots f_m) \cdot \tilde{Q}(f_1, \cdots, f_m) &= R(f_1, \cdots, f_r) \end{aligned}$$

□

Faithful maps preserve non-zeroness

Theorem (Beecken-Mittmann-Saxena)

If $\Psi : \mathbb{F}[x_1, \cdots, x_n] \to \mathbb{F}[y_1, \cdots, y_k]$ *is faithful for* $\{f_1, \cdots, f_m\}$*, then for any* C

$$C(f_1, \cdots, f_m) \neq 0 \quad \textit{if and only if} \quad \Psi(C(f_1, \cdots, f_m)) \neq 0$$

Proof.

Say $\{f_1, \cdots, f_r\}$ is a maximal algebraically independent set that is preserved by Ψ. Then, $\mathbb{F}(f_1, \cdots, f_m) = \mathbb{F}(f_1, \cdots, f_r)[f_{r+1}, \cdots, f_m]$.

$$\begin{aligned} C(f_1, \cdots, f_m) \cdot Q(f_1, \cdots, f_m) &= 1 \\ \implies C(f_1, \cdots f_m) \cdot \tilde{Q}(f_1, \cdots, f_m) &= R(f_1, \cdots, f_r) \end{aligned}$$

$$\mathbb{F}[f_1, \cdots, f_m] \qquad \mathbb{F}[f_1, \cdots, f_r]$$

□

Faithful maps preserve non-zeroness

Theorem (Beecken-Mittmann-Saxena)

If $\Psi : \mathbb{F}[x_1, \cdots, x_n] \to \mathbb{F}[y_1, \cdots, y_k]$ *is faithful for* $\{f_1, \cdots, f_m\}$*, then for any* C

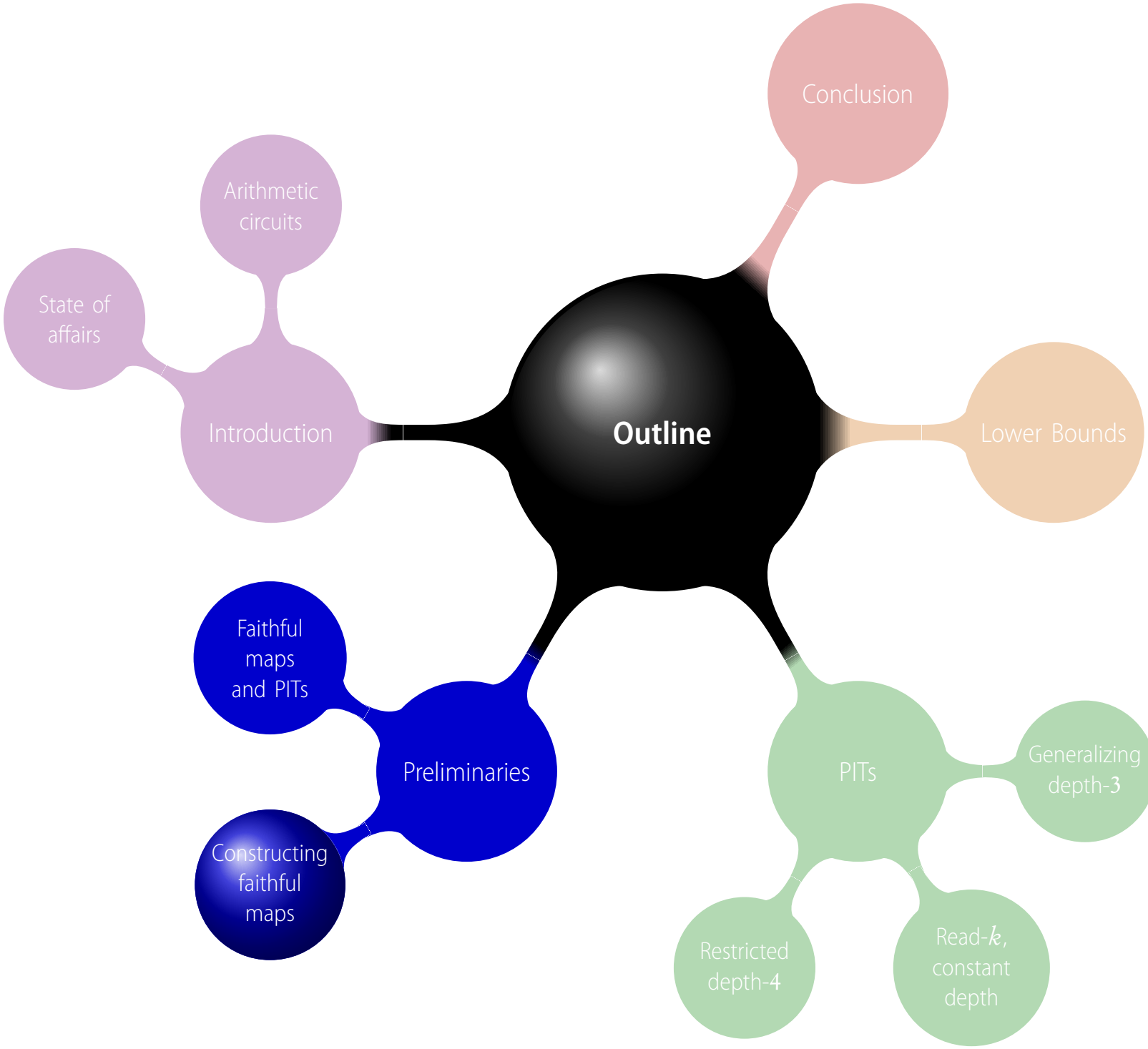
$$C(f_1, \cdots, f_m) \neq 0 \quad \textit{if and only if} \quad \Psi(C(f_1, \cdots, f_m)) \neq 0$$

Proof.

Say $\{f_1, \cdots, f_r\}$ is a maximal algebraically independent set that is preserved by Ψ. Then, $\mathbb{F}(f_1, \cdots, f_m) = \mathbb{F}(f_1, \cdots, f_r)[f_{r+1}, \cdots, f_m]$.

$$\begin{aligned} C(f_1, \cdots, f_m) \cdot Q(f_1, \cdots, f_m) &= 1 \\ \implies C(f_1, \cdots f_m) \cdot \tilde{Q}(f_1, \cdots, f_m) &= R(f_1, \cdots, f_r) \\ \Psi(C(f_1, \cdots, f_m)) \cdot \Psi(\tilde{Q}(f_1, \cdots, f_m)) &= R(\Psi(f_1), \cdots, \Psi(f_r)) \neq 0 \end{aligned}$$

□


Outline
Introduction
Arithmetic circuits
State of affairs
Preliminaries
Faithful maps and PITs
Constructing faithful maps
Lower Bounds
PITs
Generalizing depth-3
Restricted depth-4
Read-k, constant depth
Conclusion

Constructing faithful maps

Constructing faithful maps

Question: Given polynomials $f_1, \cdots, f_m$ explicitly, can we even compute $\mathrm{algRank}\{f_1, \cdots, f_m\}$?

Constructing faithful maps

Question: Given polynomials $f_1, \cdots, f_m$ explicitly, can we even compute $\mathrm{algRank}\{f_1, \cdots, f_m\}$?

Can we try to somehow find the annihilator polynomials?

Constructing faithful maps

Question: Given polynomials $f_1, \cdots, f_m$ explicitly, can we even compute $\mathrm{algRank}\{f_1, \cdots, f_m\}$?

Can we try to somehow find the annihilator polynomials?
[Kayal09]: NP-hard to even decide if it has a constant term or not!

Constructing faithful maps

Question: Given polynomials $f_1, \cdots, f_m$ explicitly, can we even compute $\mathrm{algRank}\{f_1, \cdots, f_m\}$?

Can we try to somehow find the annihilator polynomials?
[Kayal09]: NP-hard to even decide if it has a constant term or not!

Answer: Use the Jacobian!

The Jacobian

$$\mathscr{J}_{x_1,\cdots,x_n}(f_1,\cdots,f_m)=\begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}_{m\times n}$$

The Jacobian

$$\mathscr{J}_{x_1,\cdots,x_n}(f_1,\cdots,f_m) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}_{m \times n}$$

Theorem (Jacobi Criterion)

If $\mathrm{char}(\mathbb{F}) = 0$ *or "large enough",*

$$\mathrm{algRank}\{f_1,\cdots,f_m\} \quad = \quad \mathrm{rank}(\mathscr{J}(f_1,\cdots,f_m))$$

The Jacobian

$$\mathscr{J}_{x_1,\cdots,x_n}(f_1,\cdots,f_m) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}_{m \times n}$$

Theorem (Jacobi Criterion)

If $\mathbf{char}(\mathbb{F}) = 0$ *or "large enough",*

$$\mathbf{algRank}\{f_1,\cdots,f_m\} \quad = \quad \mathbf{rank}(\mathscr{J}(f_1,\cdots,f_m))$$

algRank can be computed in randomized polynomial time. (how?)

The Jacobian

$$\mathscr{J}_{x_1,\cdots,x_n}(f_1,\cdots,f_m) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}_{m \times n}$$

Theorem (Jacobi Criterion)

If $\mathbf{char}(\mathbb{F}) = 0$ *or "large enough",*

$$\mathbf{algRank}\{f_1,\cdots,f_m\} \quad = \quad \mathbf{rank}(\mathscr{J}(f_1,\cdots,f_m))$$

algRank can be computed in randomized polynomial time. (how?)

How do we use this to construct faithful maps?

Revisiting $\Sigma\Pi\Sigma$ circuits:

$$C = \sum\prod \ell_{ij}$$

Say $\dim \left\{\ell_{ij}\right\}_{i,j} = k$.

How do preserve the rank in a blackbox fashion?

Rank preserving maps

Lemma (GabizonRaz05)

Given n, k, there is a set of $O(nk^2)$ of linear transformations $\{\Psi_t\} : \mathbb{F}^n \to \mathbb{F}^k$ such that for any subspace $V \subset \mathbb{F}^n$ of dimension k, there is at least one Ψ_t that is an isomorphism between V and $\mathbb{F}^k$.

Rank preserving maps

Lemma (GabizonRaz05)

Given n, k, there is a set of $O(nk^2)$ of linear transformations $\{\Psi_t\} : \mathbb{F}^n \to \mathbb{F}^k$ such that for any subspace $V \subset \mathbb{F}^n$ of dimension k, there is at least one Ψ_t that is an isomorphism between V and $\mathbb{F}^k$.

$$\Psi_t = \begin{bmatrix} t & t^2 & \cdots & t^n \\ t^2 & t^4 & \cdots & t^{2n} \\ \vdots & \vdots & \ddots & \vdots \\ t^k & t^{2k} & \cdots & t^{nk} \end{bmatrix}_{n \times k}$$

Rank preserving maps

Lemma (GabizonRaz05)

Given n, k, there is a set of $O(nk^2)$ of linear transformations $\{\Psi_t\} : \mathbb{F}^n \to \mathbb{F}^k$ such that for any subspace $V \subset \mathbb{F}^n$ of dimension k, there is at least one Ψ_t that is an isomorphism between V and $\mathbb{F}^k$.

$$\begin{bmatrix} t & t^2 & \cdots & t^n \\ t^2 & t^4 & \cdots & t^{2n} \\ \vdots & \vdots & \ddots & \vdots \\ t^k & t^{2k} & \cdots & t^{nk} \end{bmatrix} \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ f_1 & f_2 & \cdots & f_k \\ \downarrow & \downarrow & & \downarrow \end{bmatrix} = \begin{bmatrix} f_1(t) & \cdots & f_k(t) \\ f_1(t^2) & \cdots & f_k(t^2) \\ \cdots & \ddots & \vdots \\ f_1(t^k) & \cdots & f_k(t^k) \end{bmatrix}$$

Rank preserving maps

Lemma (GabizonRaz05)

Given n, k, there is a set of $O(nk^2)$ of linear transformations $\{\Psi_t\} : \mathbb{F}^n \to \mathbb{F}^k$ such that for any subspace $V \subset \mathbb{F}^n$ of dimension k, there is at least one Ψ_t that is an isomorphism between V and $\mathbb{F}^k$.

$$\begin{bmatrix} t & t^2 & \cdots & t^n \\ t^2 & t^4 & \cdots & t^{2n} \\ \vdots & \vdots & \ddots & \vdots \\ t^k & t^{2k} & \cdots & t^{nk} \end{bmatrix} \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ f_1 & f_2 & \cdots & f_k \\ \downarrow & & & \\ & \ddots & & \\ & & & \downarrow \end{bmatrix} = \begin{bmatrix} f_1(t) & \cdots & f_k(t) \\ f_1(t^2) & \cdots & f_k(t^2) \\ \cdots & \ddots & \vdots \\ f_1(t^k) & \cdots & f_k(t^k) \end{bmatrix}$$

Rank preserving maps

Lemma (GabizonRaz05)

Given n, k, there is a set of $O(nk^2)$ of linear transformations $\{\Psi_t\} : \mathbb{F}^n \to \mathbb{F}^k$ such that for any subspace $V \subset \mathbb{F}^n$ of dimension k, there is at least one Ψ_t that is an isomorphism between V and $\mathbb{F}^k$.

$$\begin{bmatrix} t & t^2 & \cdots & t^n \\ t^2 & t^4 & \cdots & t^{2n} \\ \vdots & \vdots & \ddots & \vdots \\ t^k & t^{2k} & \cdots & t^{nk} \end{bmatrix} \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ f_1 & f_2 & \cdots & f_k \\ \downarrow & & & \\ & \ddots & & \\ & & & \downarrow \end{bmatrix} = \begin{bmatrix} f_1(t) & \cdots & f_k(t) \\ f_1(t^2) & \cdots & f_k(t^2) \\ \cdots & \ddots & \vdots \\ f_1(t^k) & \cdots & f_k(t^k) \end{bmatrix}$$

□

What do we need?

$$\begin{bmatrix} \partial_{x_1} f_1 & \cdots & \partial_{x_n} f_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} f_m & \cdots & \partial_{x_n} f_m \end{bmatrix}_{m \times n}$$

What do we need?

$$\begin{bmatrix} \partial_{x_1} f_1 & \cdots & \partial_{x_n} f_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} f_m & \cdots & \partial_{x_n} f_m \end{bmatrix}_{m \times n}$$

$$\Psi \quad : \quad \mathbb{F}[x_1, \cdots, x_n] \quad \longrightarrow \quad \mathbb{F}[y_1, \cdots, y_n]$$

What do we need?

$$\begin{bmatrix} \partial_{x_1} f_1 & \cdots & \partial_{x_n} f_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} f_m & \cdots & \partial_{x_n} f_m \end{bmatrix}_{m \times n}$$

$$\Psi \quad : \quad \mathbb{F}[x_1, \cdots, x_n] \quad \longrightarrow \quad \mathbb{F}[y_1, \cdots, y_n]$$

$$\begin{bmatrix} \partial_{y_1} \Psi(f_1) & \cdots & \partial_{y_k} \Psi(f_1) \\ \vdots & \ddots & \vdots \\ \partial_{y_1} \Psi(f_m) & \cdots & \partial_{y_k} \Psi(f_m) \end{bmatrix}_{m \times k}$$

What do we need?

$$\begin{bmatrix} \partial_{x_1} f_1 & \cdots & \partial_{x_n} f_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} f_m & \cdots & \partial_{x_n} f_m \end{bmatrix}_{m \times n}$$

$$\Psi \quad : \quad \mathbb{F}[x_1, \cdots, x_n] \quad \longrightarrow \quad \mathbb{F}[y_1, \cdots, y_n]$$

$$\begin{bmatrix} \partial_{y_1} \Psi(f_1) & \cdots & \partial_{y_k} \Psi(f_1) \\ \vdots & \ddots & \vdots \\ \partial_{y_1} \Psi(f_m) & \cdots & \partial_{y_k} \Psi(f_m) \end{bmatrix}_{m \times k}$$

How does the Jacobian evolve?

Evolution of the Jacobian under homomorphism

$$\frac{\partial \Psi(f)}{\partial y} = \frac{\partial}{\partial y} f(\overline{\Psi(x)})$$

Evolution of the Jacobian under homomorphism

$$\begin{aligned}
\frac{\partial \Psi(f)}{\partial y} &= \frac{\partial}{\partial y} f(\overline{\Psi(x)}) \\
&= \sum_{i=1}^{n} \frac{\partial f}{\partial x_i} \left[\overline{\Psi(x)}\right] \quad \cdot \quad \frac{\partial \Psi(x_i)}{\partial y}
\end{aligned}$$

Evolution of the Jacobian under homomorphism

$$\begin{aligned}\frac{\partial \Psi(f)}{\partial y} &= \frac{\partial}{\partial y} f(\overline{\Psi(x)}) \\ &= \sum_{i=1}^{n} \Psi\left(\frac{\partial f}{\partial x_i}\right) \cdot \frac{\partial \Psi(x_i)}{\partial y}\end{aligned}$$

Evolution of the Jacobian under homomorphism

$$\begin{bmatrix} \partial_{y_1}\Psi(f_1) & \cdots & \partial_{y_k}\Psi(f_1) \\ \vdots & \ddots & \vdots \\ \partial_{y_1}\Psi(f_m) & \cdots & \partial_{y_k}\Psi(f_m) \end{bmatrix} =$$

$$\Psi \circ \begin{bmatrix} \partial_{x_1} f_1 & \cdots & \partial_{x_n} f_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} f_m & \cdots & \partial_{x_n} f_m \end{bmatrix} \cdot \begin{bmatrix} \partial_{y_1}\Psi(x_1) & \cdots & \partial_{y_k}\Psi(x_1) \\ \vdots & \ddots & \vdots \\ \partial_{y_1}\Psi(x_n) & \cdots & \partial_{y_k}\Psi(x_n) \end{bmatrix}$$

Evolution of the Jacobian under homomorphism

$$\begin{bmatrix} \partial_{y_1}\Psi(f_1) & \cdots & \partial_{y_k}\Psi(f_1) \\ \vdots & \ddots & \vdots \\ \partial_{y_1}\Psi(f_m) & \cdots & \partial_{y_k}\Psi(f_m) \end{bmatrix} =$$

$$\Psi \circ \begin{bmatrix} \partial_{x_1} f_1 & \cdots & \partial_{x_n} f_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} f_m & \cdots & \partial_{x_n} f_m \end{bmatrix} \cdot \begin{bmatrix} \partial_{y_1}\Psi(x_1) & \cdots & \partial_{y_k}\Psi(x_1) \\ \vdots & \ddots & \vdots \\ \partial_{y_1}\Psi(x_n) & \cdots & \partial_{y_k}\Psi(x_n) \end{bmatrix}$$

$$\Psi: \quad x_i \quad \mapsto \quad \sum_{j=1}^{k} y_j t^{ij} \quad + \Phi(x_i)$$

Evolution of the Jacobian under homomorphism

$$\begin{bmatrix} \partial_{y_1}\Psi(f_1) & \cdots & \partial_{y_k}\Psi(f_1) \\ \vdots & \ddots & \vdots \\ \partial_{y_1}\Psi(f_m) & \cdots & \partial_{y_k}\Psi(f_m) \end{bmatrix} =$$

$$\Psi \circ \begin{bmatrix} \partial_{x_1} f_1 & \cdots & \partial_{x_n} f_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} f_m & \cdots & \partial_{x_n} f_m \end{bmatrix} \cdot \begin{bmatrix} t & t^2 & \cdots & t^k \\ \vdots & \vdots & \ddots & \vdots \\ t^n & t^{2n} & \cdots & t^{nk} \end{bmatrix}$$

$$\Psi: \quad x_i \quad \mapsto \quad \sum_{j=1}^{k} y_j t^{ij} \quad + \Phi(x_i)$$

Evolution of the Jacobian under homomorphism

$$\begin{bmatrix} \partial_{y_1}\Psi(f_1) & \cdots & \partial_{y_k}\Psi(f_1) \\ \vdots & \ddots & \vdots \\ \partial_{y_1}\Psi(f_m) & \cdots & \partial_{y_k}\Psi(f_m) \end{bmatrix} =$$

$$\Psi \circ \begin{bmatrix} \partial_{x_1} f_1 & \cdots & \partial_{x_n} f_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} f_m & \cdots & \partial_{x_n} f_m \end{bmatrix} \cdot \begin{bmatrix} t & t^2 & \cdots & t^k \\ \vdots & \vdots & \ddots & \vdots \\ t^n & t^{2n} & \cdots & t^{nk} \end{bmatrix}$$

$$\Psi: \quad x_i \quad \mapsto \quad \sum_{j=1}^{k} y_j t^{ij} \quad + \Phi(x_i)$$

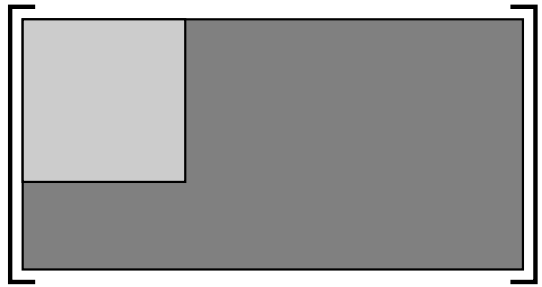
If $\operatorname{rank}\left(\mathscr{J}(f_1,\cdots,f_m)\right) = \operatorname{rank}\left(\Phi \circ \mathscr{J}(f_1,\cdots,f_m)\right)$, we are done.

Recipe for constructing faithful maps

$$\mathscr{J}(f_1,\cdots,f_m)=$$

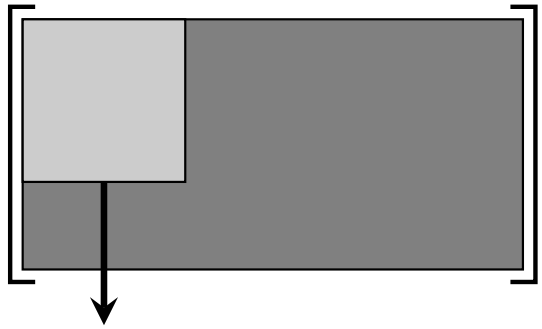
Recipe for constructing faithful maps

$$\mathscr{J}(f_1,\cdots,f_m)=$$



Recipe for constructing faithful maps

$$\mathcal{J}(f_1,\cdots,f_m)=$$



$$J=\begin{vmatrix} \partial_{x_1} f_1 & \cdots & \partial_{x_k} f_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} f_k & \cdots & \partial_{x_k} f_k \end{vmatrix}$$

Recipe for constructing faithful maps

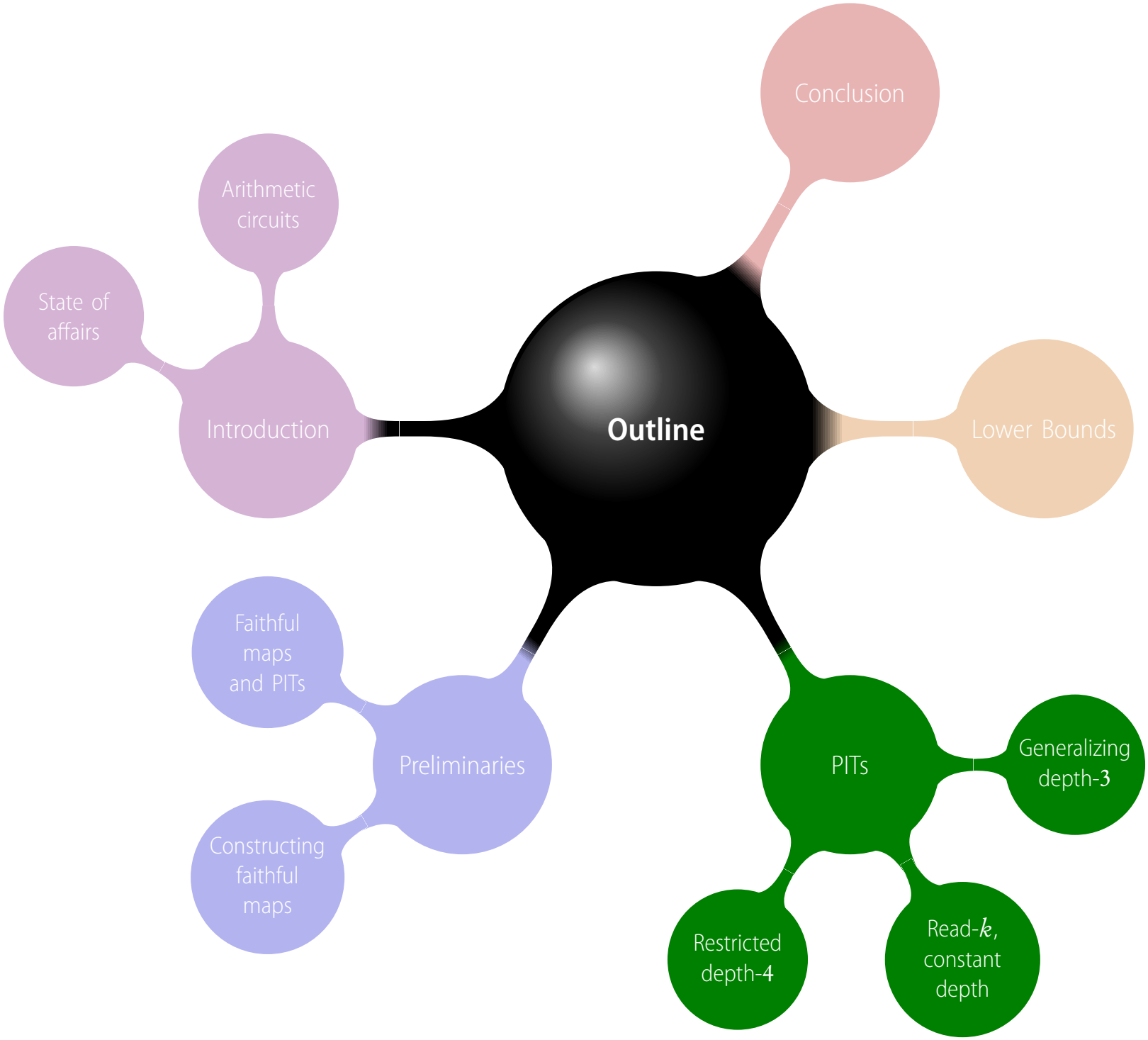
$$\mathcal{J}(f_1,\cdots,f_m) = \left[\begin{array}{c} \\ \\ \\ \end{array}\right]$$

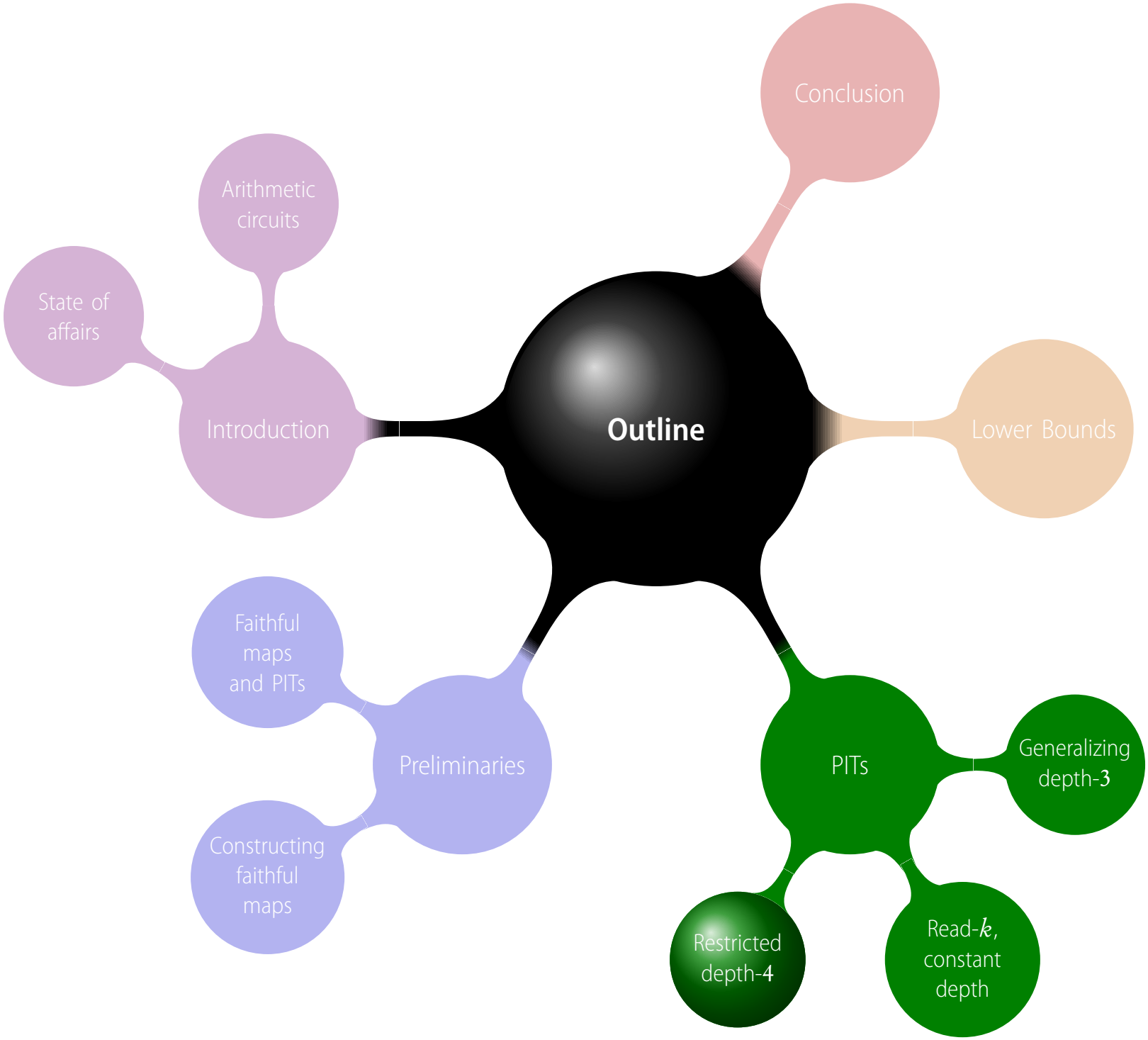
$$J = \left|\begin{array}{ccc} \partial_{x_1} f_1 & \cdots & \partial_{x_k} f_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} f_k & \cdots & \partial_{x_k} f_k \end{array}\right| \longleftarrow \text{Preserve this determinant}$$

Lemma (Composition Lemma)

Let Φ *be a map such that* $\Phi(J) \neq 0$. *Then the map* Ψ *is faithful to* $\{f_1,\cdots,f_m\}$:

$$\Psi : x_i \quad \mapsto \quad \sum_{j=1}^{k} y_j t^{ij} \quad + \quad \Phi(x_i)$$


Outline
Introduction
Arithmetic circuits
State of affairs
Preliminaries
Faithful maps and PITs
Constructing faithful maps
Lower Bounds
PITs
Generalizing depth-3
Restricted depth-4
Read-k, constant depth
Conclusion


Outline
Introduction
Arithmetic circuits
State of affairs
Preliminaries
Faithful maps and PITs
Constructing faithful maps
Lower Bounds
PITs
Generalizing depth-3
Read-k, constant depth
Restricted depth-4
Conclusion

[BeeckenMittmannSaxena]'s PIT

Theorem

There is a black-box PIT for circuits of the form $C(f_1, \cdots, f_m)$ *where each* f_i *is "sparse" and* $\mathrm{algRank}\{f_1, \cdots, f_m\} \leq k$.

Proof.

[BeeckenMittmannSaxena]'s PIT

Theorem

There is a black-box PIT for circuits of the form $C(f_1, \cdots, f_m)$ *where each* f_i *is "sparse" and* $\mathrm{algRank}\{f_1, \cdots, f_m\} \leq k$.

Proof.

$$\mathcal{J}(f_1, \cdots, f_m) = \left[\begin{array}{c} \\ \\ \\ \end{array}\right]$$

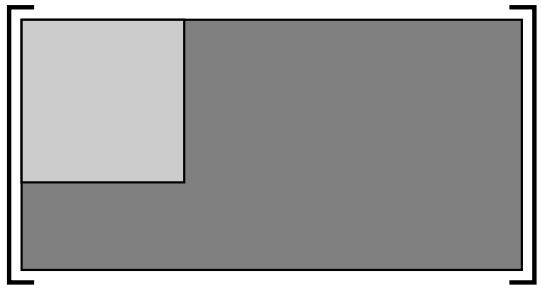
□

[BeeckenMittmannSaxena]'s PIT

Theorem

There is a black-box PIT for circuits of the form $C(f_1, \cdots, f_m)$ *where each* f_i *is "sparse" and* $\mathrm{algRank}\{f_1, \cdots, f_m\} \leq k$.

Proof.

$\mathcal{J}(f_1, \cdots, f_m) =$ 

□

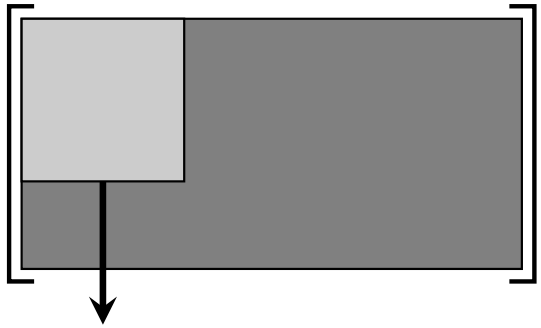
[BeeckenMittmannSaxena]'s PIT

Theorem

There is a black-box PIT for circuits of the form $C(f_1, \cdots, f_m)$ *where each* f_i *is "sparse" and* $\mathrm{algRank}\{f_1, \cdots, f_m\} \leq k$.

Proof.

$$\mathcal{J}(f_1, \cdots, f_m) =$$



$$J = \begin{vmatrix} \partial_{x_1} f_1 & \cdots & \partial_{x_k} f_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} f_k & \cdots & \partial_{x_k} f_k \end{vmatrix}$$

□

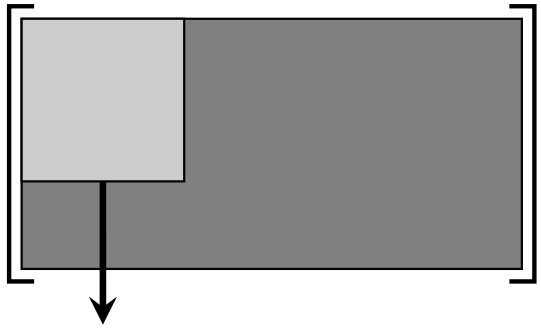
[BeeckenMittmannSaxena]'s PIT

Theorem

There is a black-box PIT for circuits of the form $C(f_1, \cdots, f_m)$ *where each* f_i *is "sparse" and* $\mathrm{algRank}\{f_1, \cdots, f_m\} \leq k$.

Proof.

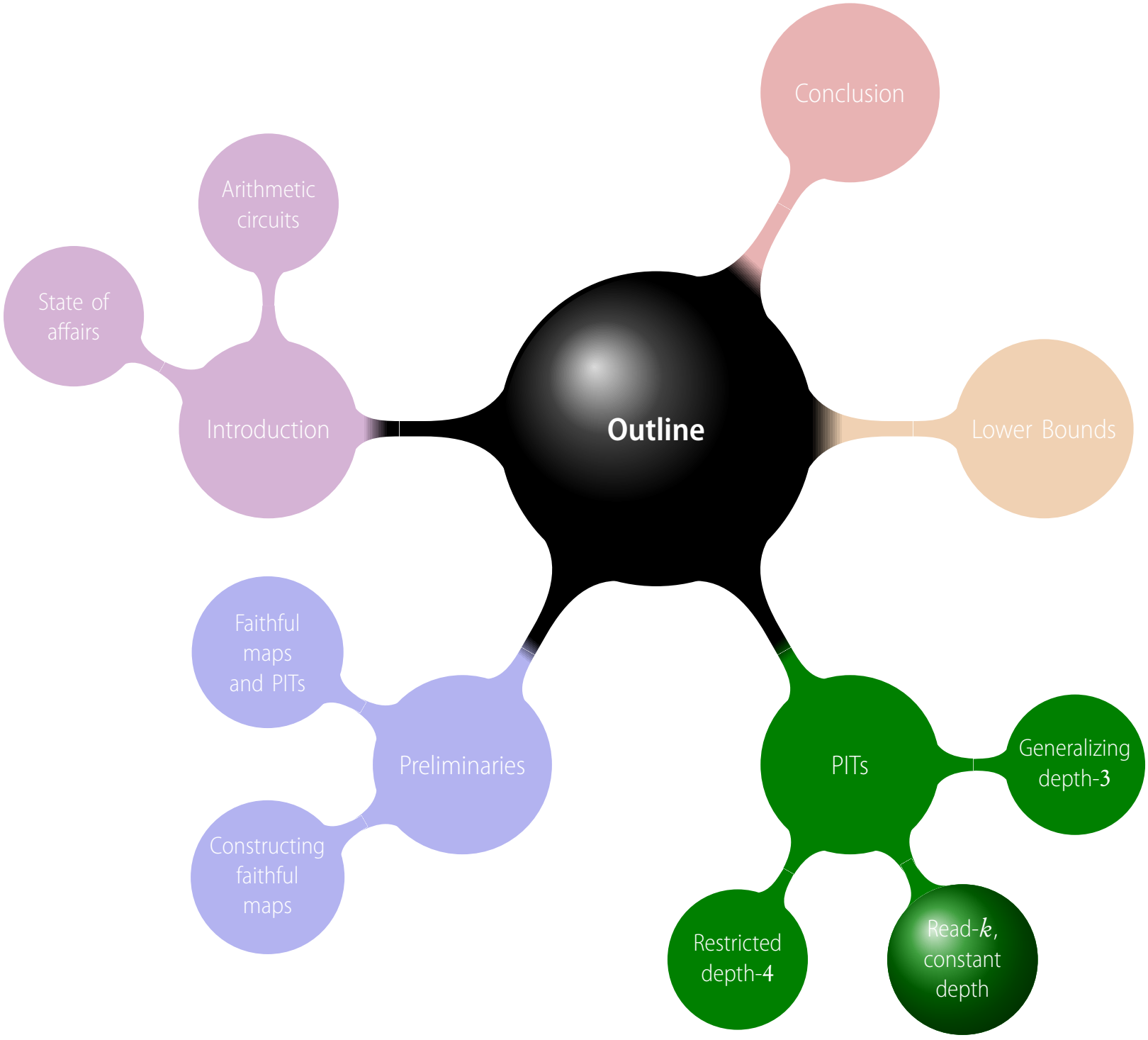
$$\mathcal{J}(f_1, \cdots, f_m) =$$



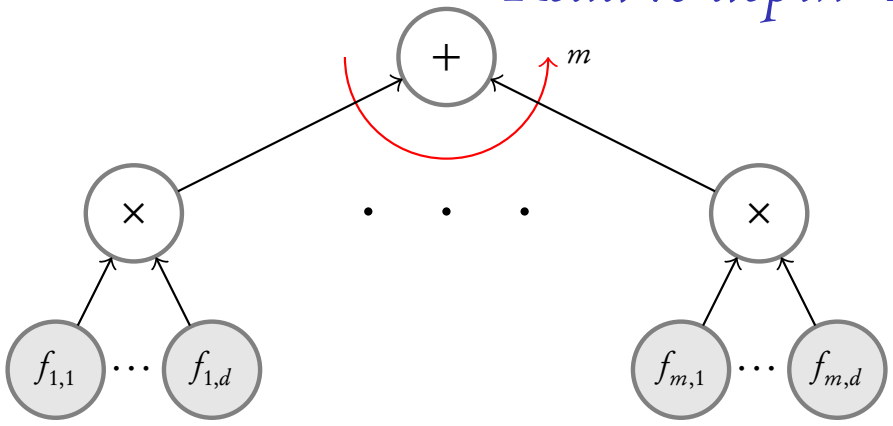
$$J = \begin{vmatrix} \partial_{x_1} f_1 & \cdots & \partial_{x_k} f_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} f_k & \cdots & \partial_{x_k} f_k \end{vmatrix}$$

which is a sparse poly!

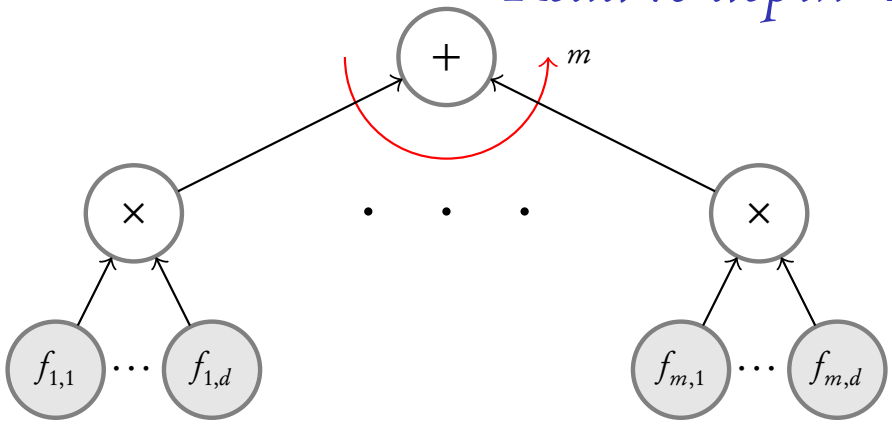
□


Outline
Introduction
Arithmetic circuits
State of affairs
Preliminaries
Faithful maps and PITs
Constructing faithful maps
Lower Bounds
PITs
Generalizing depth-3
Restricted depth-4
Read-k, constant depth
Conclusion

Read-k depth-4 formulae



Read-k depth-4 formulae



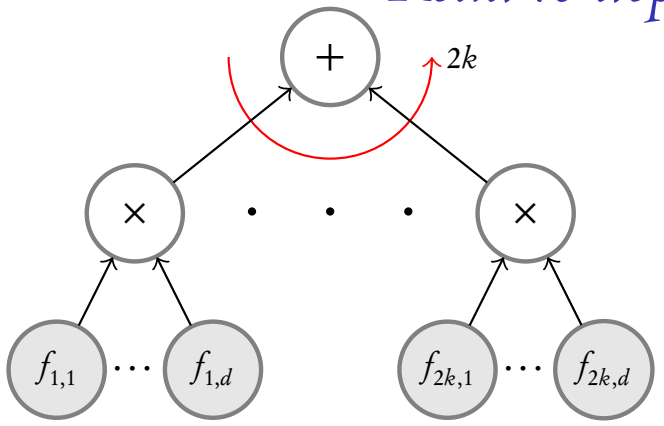
Observation

If $C(x_1, \cdots, x_n) \neq 0$, *then there exists an* i *such that*

$$C(x_1, \cdots, x_i + 1, \cdots, x_n) - C(x_1, \cdots, x_i, \cdots, x_n) \neq 0$$

In fact, any x_i that C non-trivially depends on.

Read-k depth-4 formulae



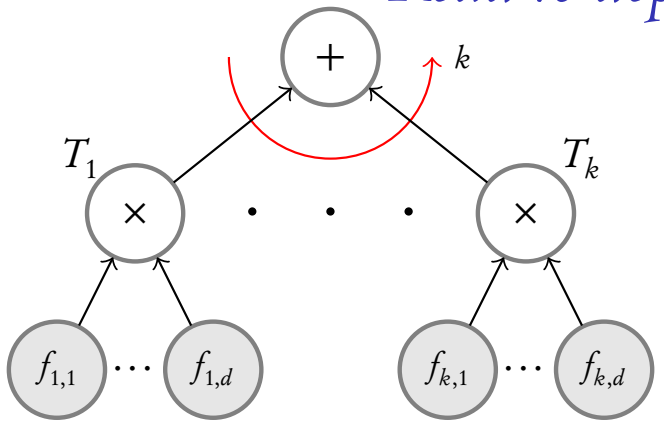
Observation

If $C(x_1, \cdots, x_n) \neq 0$, *then there exists an* i *such that*

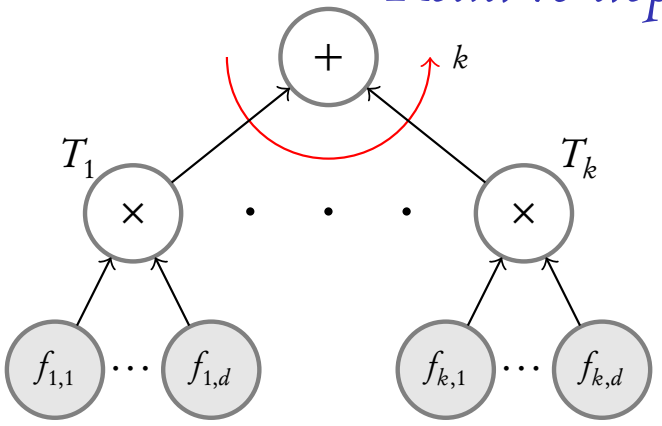
$$C(x_1, \cdots, x_i + 1, \cdots, x_n) - C(x_1, \cdots, x_i, \cdots, x_n) \neq 0$$

In fact, any x_i that C non-trivially depends on.

Read-k depth-4 formulae



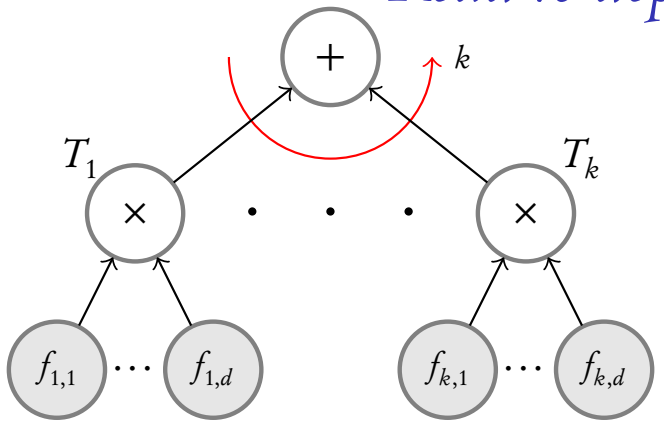
Read-k depth-4 formulae



$$\mathcal{J}(T_1,\cdots,T_k) = \left[\begin{array}{c} \\ \\ \\ \\ \\ \end{array}\right]$$

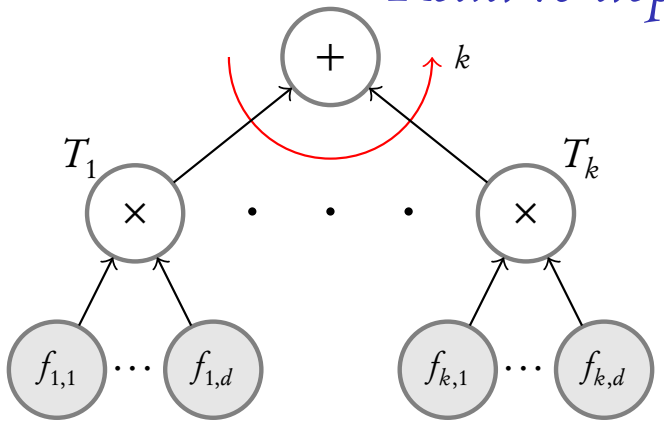
$$J = \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix}$$

Read-k depth-4 formulae



$$J = \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix}$$

Read-k depth-4 formulae

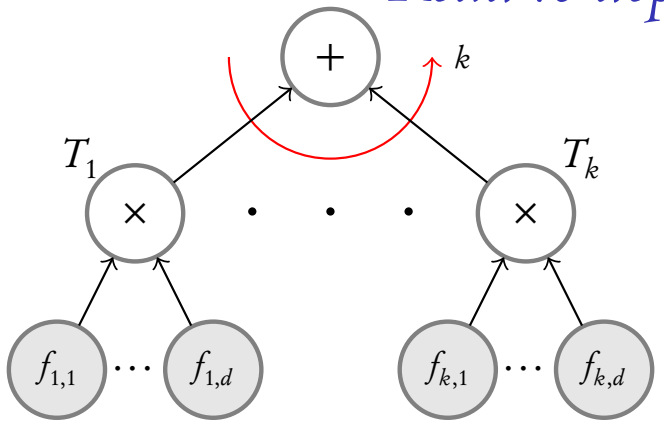


$$J = \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix}$$

Observation

At most rk *of the* f_{ij}*'s depend on* $x_1, \cdots, x_r$.

Read-k depth-4 formulae

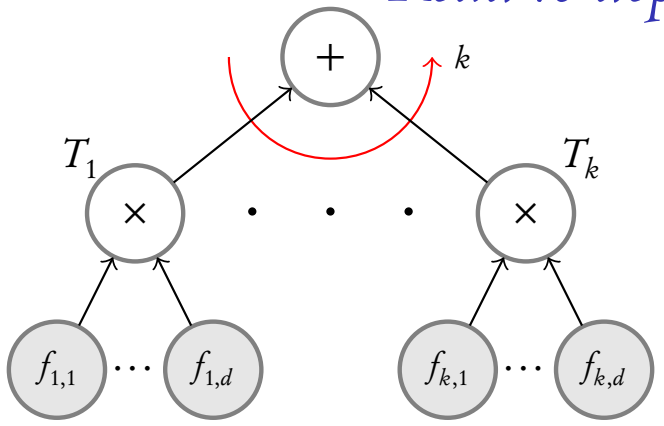


$$J = \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix} = \left(\prod f_{ij}\right) \cdot \begin{vmatrix} \partial_{x_1} T'_1 & \cdots & \partial_{x_r} T'_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T'_r & \cdots & \partial_{x_r} T'_r \end{vmatrix}$$

... a product of sparse polys!

□

Read-k depth-4 formulae

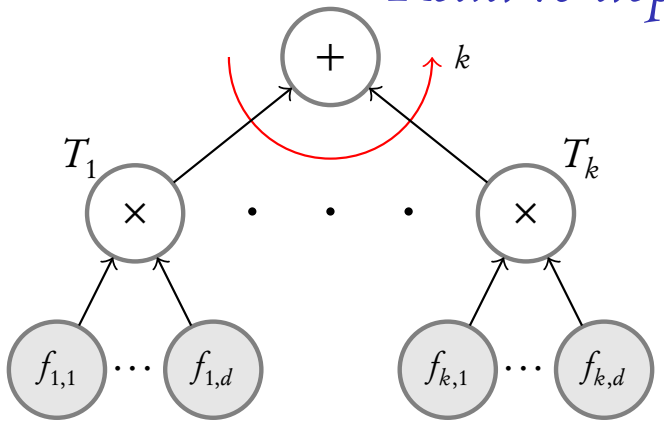


$$J = \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix} = \left(\prod f_{ij}\right) \cdot \begin{vmatrix} \partial_{x_1} T'_1 & \cdots & \partial_{x_r} T'_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T'_r & \cdots & \partial_{x_r} T'_r \end{vmatrix}$$

... a product of sparse polys!

$\Psi_r : x_i \mapsto u^{d^i \bmod r}$ preserves J

Read-k depth-4 formulae



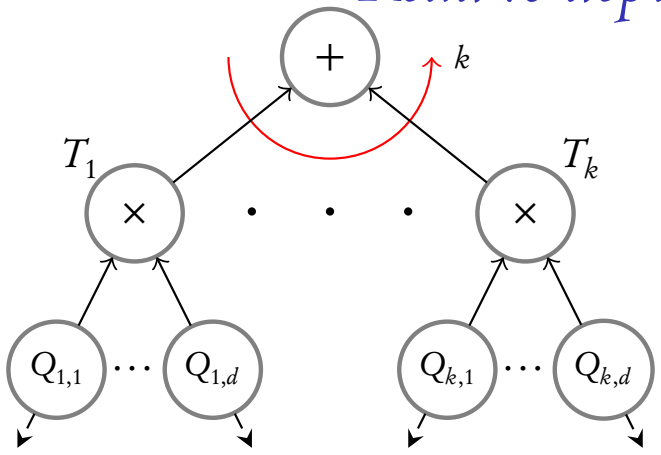
$$J = \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix} = \left(\prod f_{ij}\right) \cdot \begin{vmatrix} \partial_{x_1} T'_1 & \cdots & \partial_{x_r} T'_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T'_r & \cdots & \partial_{x_r} T'_r \end{vmatrix}$$

... a product of sparse polys!

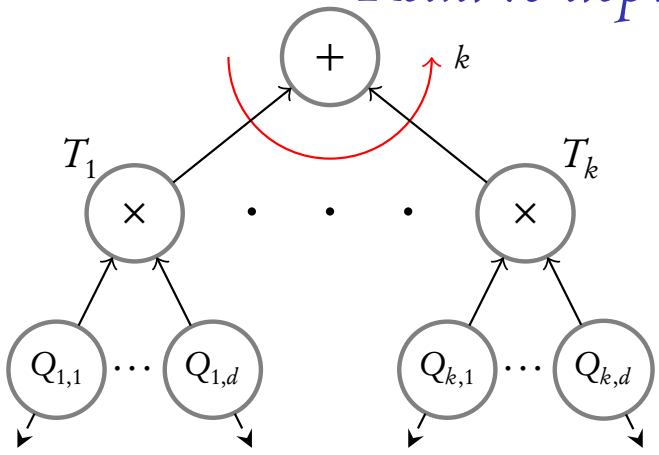
$\Psi_r : x_i \mapsto u^{d^i \bmod r}$ preserves J

$\Phi_r : x_i \mapsto \sum_{j=1}^{k} y_j t^{ij} + u^{d^i \bmod r}$ is a black-box PIT

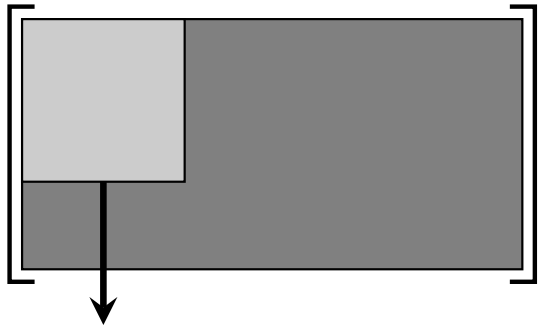
Read-k depth-D formulae



Read-k depth-D formulae

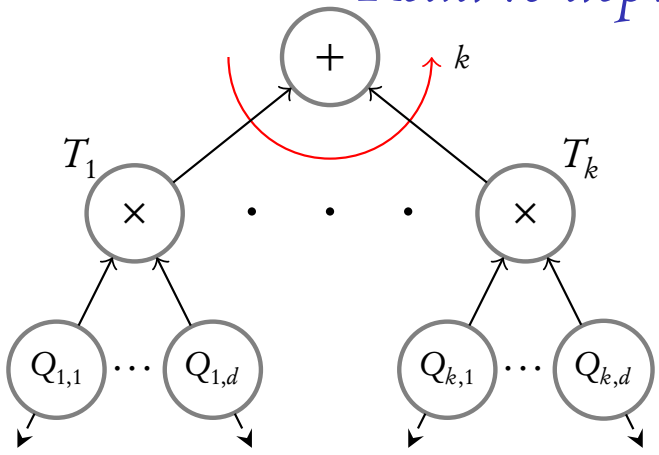


$$\mathscr{J}(T_1,\cdots,T_k)=$$



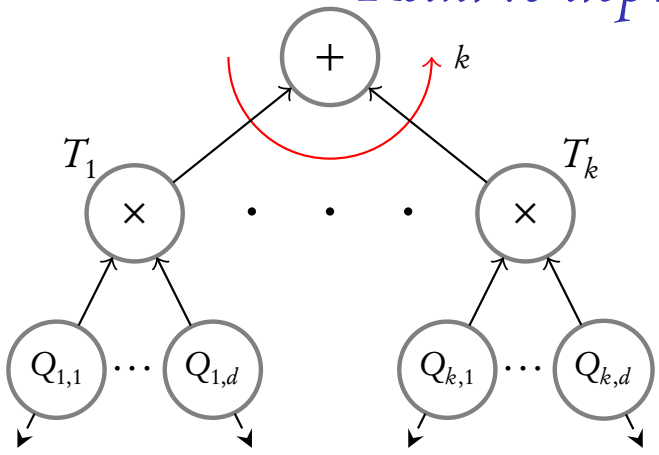
$$J \quad = \quad \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix}$$

Read-k depth-D formulae



$$J = \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix}$$

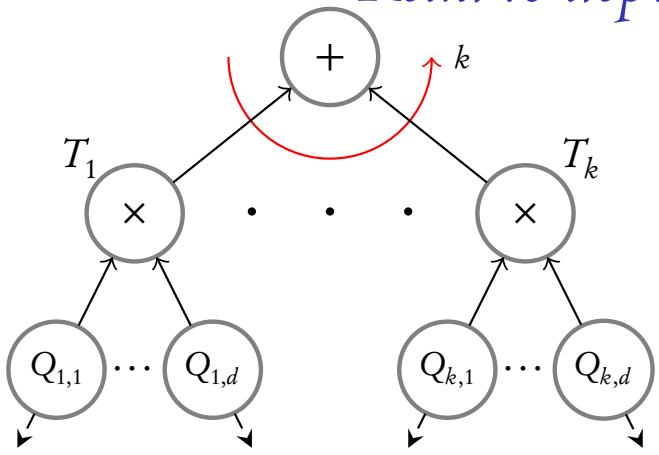
Read-k depth-D formulae



$$J = \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix} = \left(\prod Q_{ij}\right) \cdot \begin{vmatrix} \partial_{x_1} T_1' & \cdots & \partial_{x_r} T_1' \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r' & \cdots & \partial_{x_r} T_r' \end{vmatrix}$$

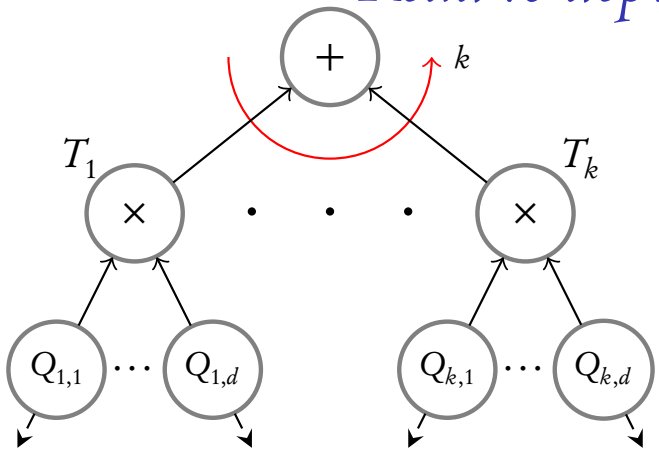
Function of "few" Q_{ij}'s

Read-k depth-D formulae



$$J = \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix} = \underbrace{\left(\prod Q_{ij}\right) \cdot \begin{vmatrix} \partial_{x_1} T'_1 & \cdots & \partial_{x_r} T'_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T'_r & \cdots & \partial_{x_r} T'_r \end{vmatrix}}_{\text{Product of functions of "few" } Q_{ij}\text{'s}}$$

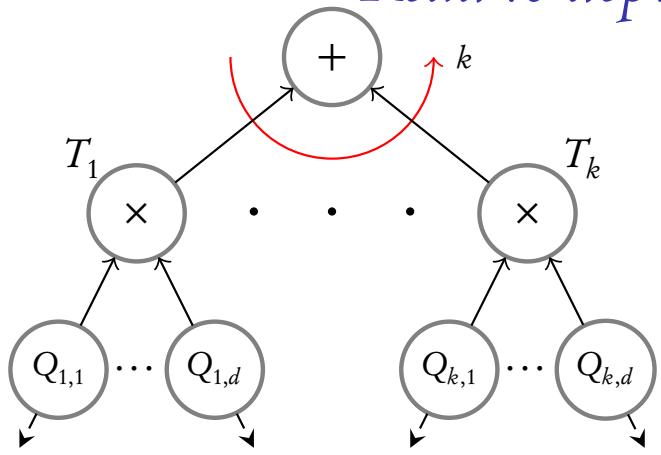
Read-k depth-D formulae



$$J = \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix} = \underbrace{\left(\prod Q_{ij}\right) \cdot \begin{vmatrix} \partial_{x_1} T'_1 & \cdots & \partial_{x_r} T'_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T'_r & \cdots & \partial_{x_r} T'_r \end{vmatrix}}_{\text{Product of functions of "few" } Q_{ij}\text{'s}}$$

To preserve non-zeroness of C it suffices to preserve non-zeroness of J.

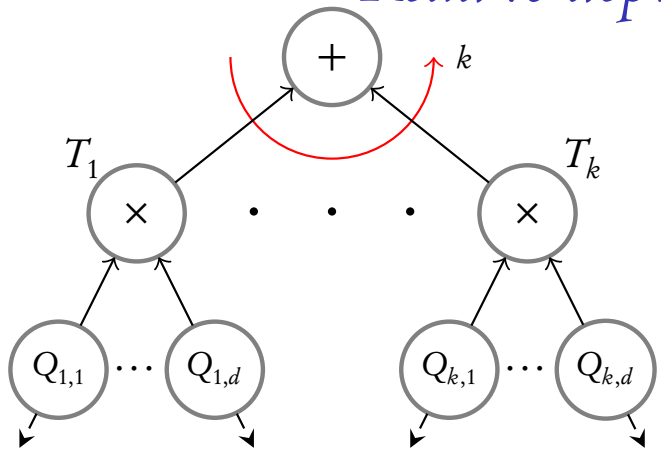
Read-k depth-D formulae



$$J = \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix} = \underbrace{\left(\prod Q_{ij}\right) \cdot \begin{vmatrix} \partial_{x_1} T'_1 & \cdots & \partial_{x_r} T'_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T'_r & \cdots & \partial_{x_r} T'_r \end{vmatrix}}_{\text{Product of functions of "few" } Q_{ij}\text{'s}}$$

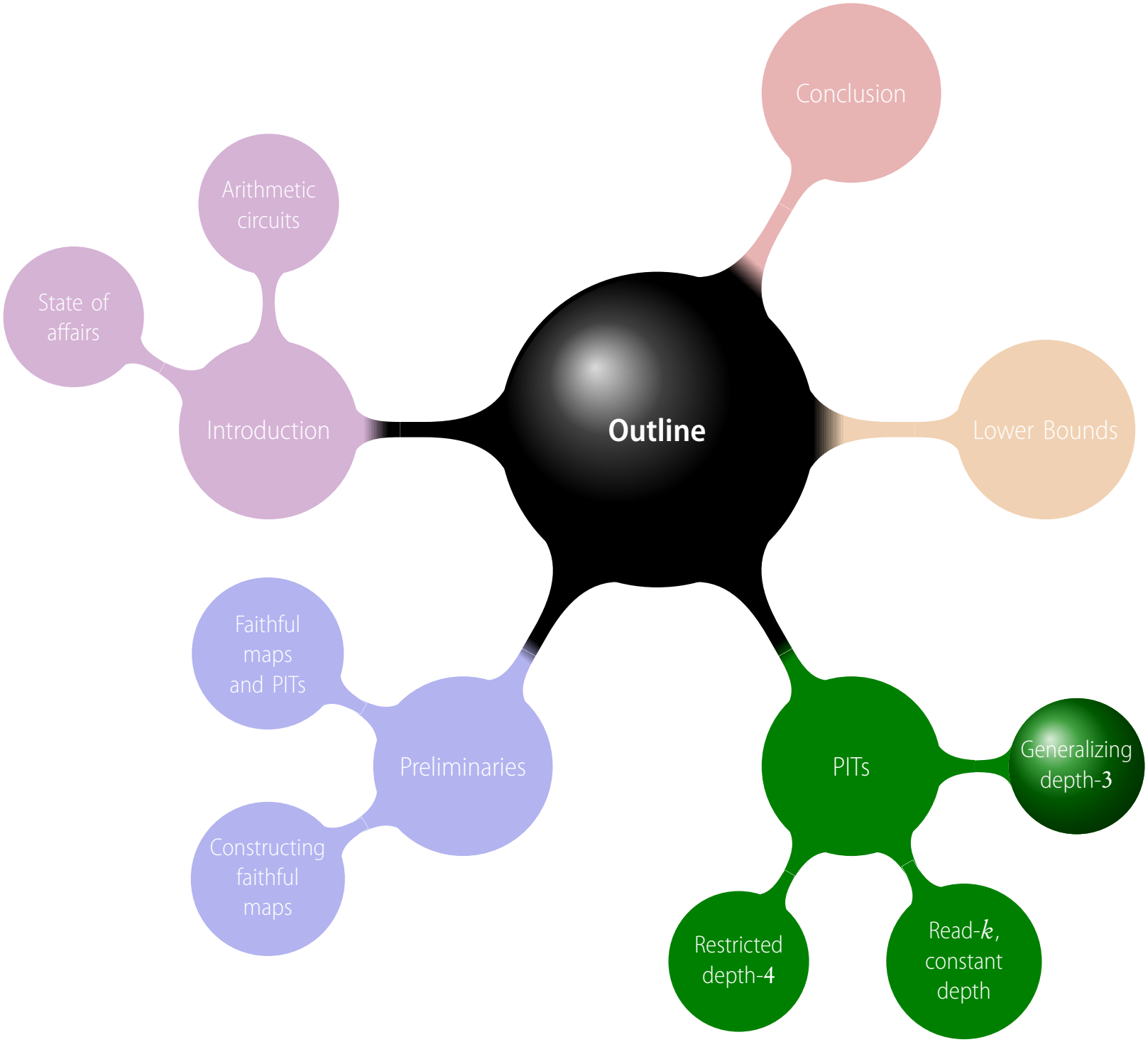
To preserve non-zeroness of C it suffices to preserve non-zeroness of J.
Hence, suffices to preserve the Jacobian of the Q_{ij}'s.

Read-k depth-D formulae



$$J = \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix} = \underbrace{\left(\prod Q_{ij}\right) \cdot \begin{vmatrix} \partial_{x_1} T'_1 & \cdots & \partial_{x_r} T'_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T'_r & \cdots & \partial_{x_r} T'_r \end{vmatrix}}_{\text{Product of functions of "few" } Q_{ij}\text{'s}}$$

To preserve non-zeroness of C it suffices to preserve non-zeroness of J.
Hence, suffices to preserve the Jacobian of the Q_{ij}'s. Recurse! □


Outline
Introduction
Arithmetic circuits
State of affairs
Preliminaries
Faithful maps and PITs
Constructing faithful maps
Lower Bounds
PITs
Generalizing depth-3
Restricted depth-4
Read-k, constant depth
Conclusion

Generalizing $\Sigma\Pi\Sigma(k)$ *PITs*

$$f \quad = \quad C(T_1, \cdots, T_k) \qquad \text{where } T_i = \prod_{j=1}^{d} \ell_{ij}$$

Generalizing $\Sigma\Pi\Sigma(k)$ *PITs*

$$J(T_1, \cdots, T_m) = \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix}$$

Generalizing $\Sigma\Pi\Sigma(k)$ *PITs*

$$J(T_1, \cdots, T_m) = \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix}$$

Lemma

$$\partial_x P \cdot Q = PQ \cdot \left(\frac{\partial_x P}{P} + \frac{\partial_x Q}{Q} \right)$$

Generalizing $\Sigma\Pi\Sigma(k)$ *PITs*

$$J(T_1,\cdots,T_m) = \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix}$$

Lemma

$$\partial_x P \cdot Q = PQ \cdot \left(\frac{\partial_x P}{P} + \frac{\partial_x Q}{Q} \right)$$

Lemma

$$\det \begin{bmatrix} \mathbf{b_1} + \mathbf{b'_1} \\ \mathbf{a_2} \\ \vdots \\ \mathbf{a_n} \end{bmatrix} = \det \begin{bmatrix} \mathbf{b_1} \\ \mathbf{a_2} \\ \vdots \\ \mathbf{a_n} \end{bmatrix} + \det \begin{bmatrix} \mathbf{b'_1} \\ \mathbf{a_2} \\ \vdots \\ \mathbf{a_n} \end{bmatrix}$$

Generalizing $\Sigma\Pi\Sigma(k)$ *PITs*

$$
\begin{aligned}
J(T_1,\cdots,T_m) &= \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix} \\
&= T_1 \cdots T_k \cdot \sum_{\ell_i \in T_i} \frac{J(\ell_1,\cdots,\ell_k)}{\ell_1 \cdots \ell_k}
\end{aligned}
$$

Generalizing $\Sigma\Pi\Sigma(k)$ *PITs*

$$
\begin{aligned}
J(T_1, \cdots, T_m) &= \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix} \\
&= T_1 \cdots T_k \cdot \sum_{\ell_i \in T_i} \frac{\alpha_L}{\ell_1 \cdots \ell_k}
\end{aligned}
$$

Generalizing $\Sigma\Pi\Sigma(k)$ *PITs*

$$
\begin{aligned}
J(T_1, \cdots, T_m) &= \begin{vmatrix} \partial_{x_1} T_1 & \cdots & \partial_{x_r} T_1 \\ \vdots & \ddots & \vdots \\ \partial_{x_1} T_r & \cdots & \partial_{x_r} T_r \end{vmatrix} \\
&= T_1 \cdots T_k \cdot \sum_{\ell_i \in T_i} \frac{\alpha_L}{\ell_1 \cdots \ell_k}
\end{aligned}
$$

A similar analysis (slightly simpler due to lack of multiplicities)

$$C \quad = \quad T \cdot \sum \frac{\alpha_L}{\ell_1 \cdots \ell_k}$$

$$C \quad = \quad T \cdot \sum \frac{\alpha_i}{\ell_i}$$

Generalizing $\Sigma\Pi\Sigma(k)$ *PITs*

$$C \quad = \quad T \cdot \sum \frac{\alpha_i}{\ell_i}$$

- Degree $= |T| - 1$.

Generalizing $\Sigma\Pi\Sigma(k)$ *PITs*

$$C \quad = \quad T \cdot \sum \frac{\alpha_i}{\ell_i}$$

- Degree $= |T| - 1$.
- Hence $C \bmod \ell \neq 0$ for some $\ell \in T$. [CRT]

Generalizing $\Sigma\Pi\Sigma(k)$ *PITs*

$$C \quad = \quad T \cdot \sum \frac{\alpha_i}{\ell_i}$$

- Degree $= |T| - 1$.
- Hence $C \bmod \ell \neq 0$ for some $\ell \in T$. [CRT]

$$C \bmod \ell \quad = \quad \frac{T}{\ell}$$

$$C \quad = \quad T \cdot \sum \frac{\alpha_{ij}}{\ell_i \ell_j}$$

Generalizing $\Sigma\Pi\Sigma(k)$ *PITs*

$$C \quad = \quad T \cdot \sum \frac{\alpha_{ij}}{\ell_i \ell_j}$$

- Degree $= |T| - 2$.

Generalizing $\Sigma\Pi\Sigma(k)$ *PITs*

$$C \quad = \quad T \cdot \sum \frac{\alpha_{ij}}{\ell_i \ell_j}$$

- Degree $= |T| - 2$.
- Hence $C \bmod \ell \neq 0$ for some $\ell \in T$.[CRT]

Generalizing $\Sigma\Pi\Sigma(k)$ *PITs*

$$C \quad = \quad T \cdot \sum \frac{\alpha_{ij}}{\ell_i \ell_j}$$

- Degree $= |T| - 2$.
- Hence $C \bmod \ell \neq 0$ for some $\ell \in T$.[CRT]

$$C \bmod \ell \quad = \quad \frac{T}{\ell} \cdot \sum \frac{\alpha_i}{\ell_i}$$

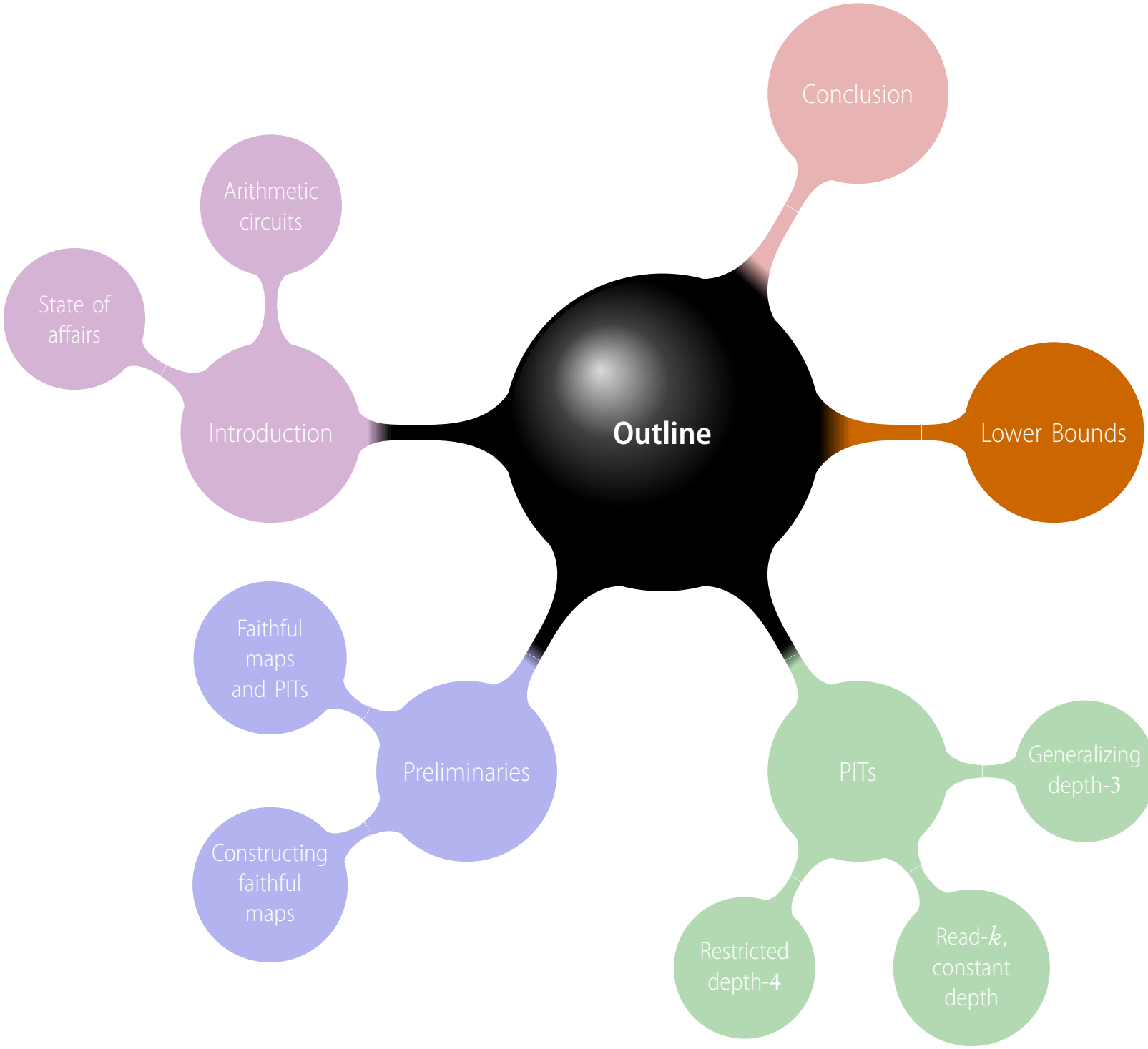
Generalizing $\Sigma\Pi\Sigma(k)$ PITs

$$C \quad = \quad T \cdot \sum \frac{\alpha_{ij}}{\ell_i \ell_j}$$

- Degree $= |T| - 2$.
- Hence $C \bmod \ell \neq 0$ for some $\ell \in T$.[CRT]

$$C \bmod \ell \quad = \quad \frac{T}{\ell} \cdot \sum \frac{\alpha_i}{\ell_i}$$

...recurse


Outline
Introduction
Arithmetic circuits
State of affairs
Preliminaries
Faithful maps and PITs
Constructing faithful maps
Lower Bounds
PITs
Generalizing depth-3
Restricted depth-4
Read-k, constant depth
Conclusion

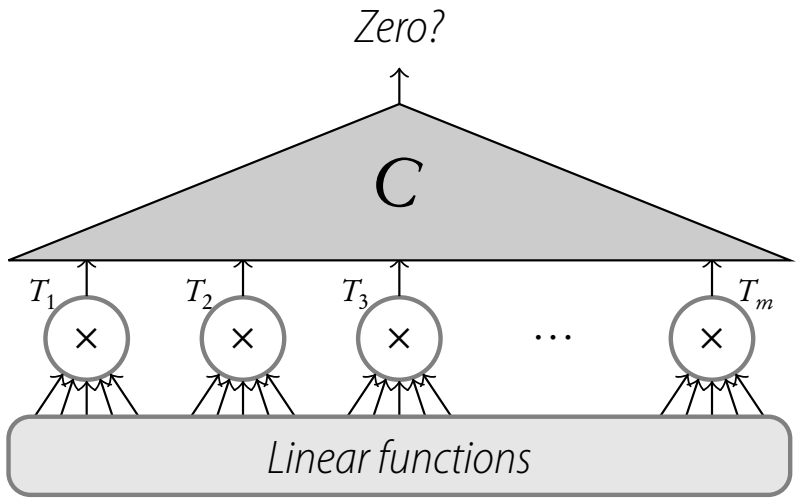
General philosophy

If you have black-box PITs for a class $\mathscr{C}$, then you have determinant/permanent lower bounds for (almost) $\mathscr{C}'$.

[KabanetsImpagliazzo03], [Agrawal05], [DvirShpilkaYehudayoff08] etc...

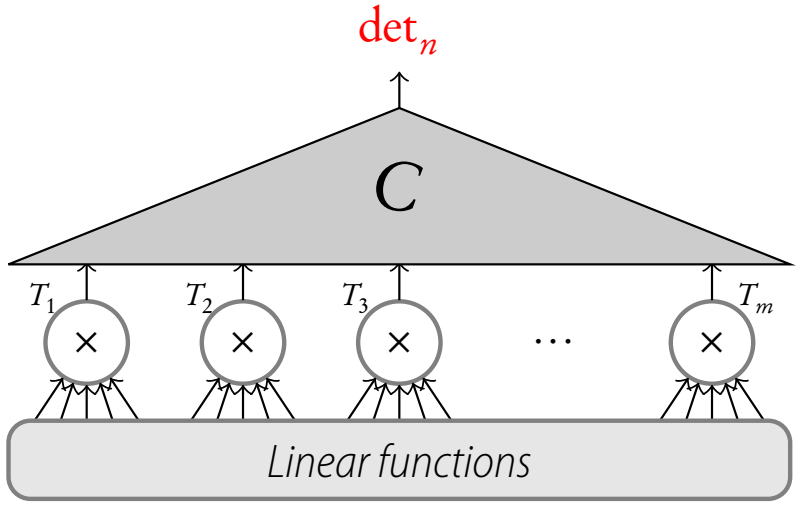
PITs & Lower bounds

Theorem



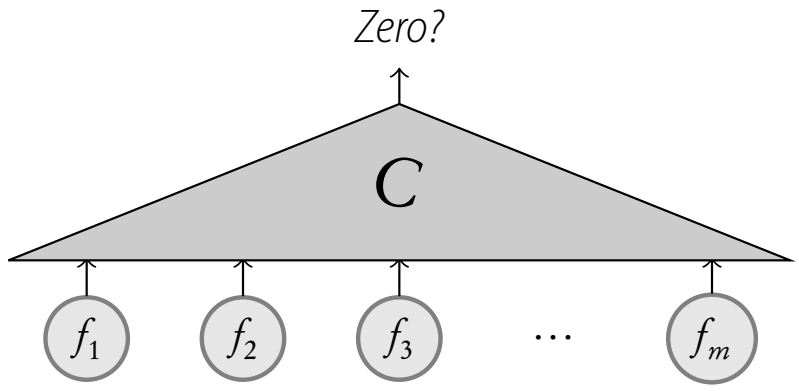
Black-box PIT if $\mathbf{algRank}\{T_1, \cdots, T_m\} = O(1)$.

Theorem



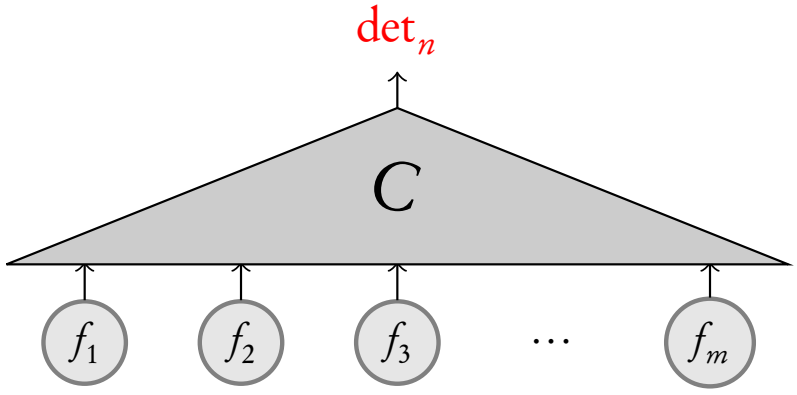
Then, $\text{algRank}\{T_1, \cdots, T_m\} = \Omega(n)$.

Theorem



Black-box PIT if $\mathbf{algRank}\{f_1, \cdots, f_m\} = O(1)$.

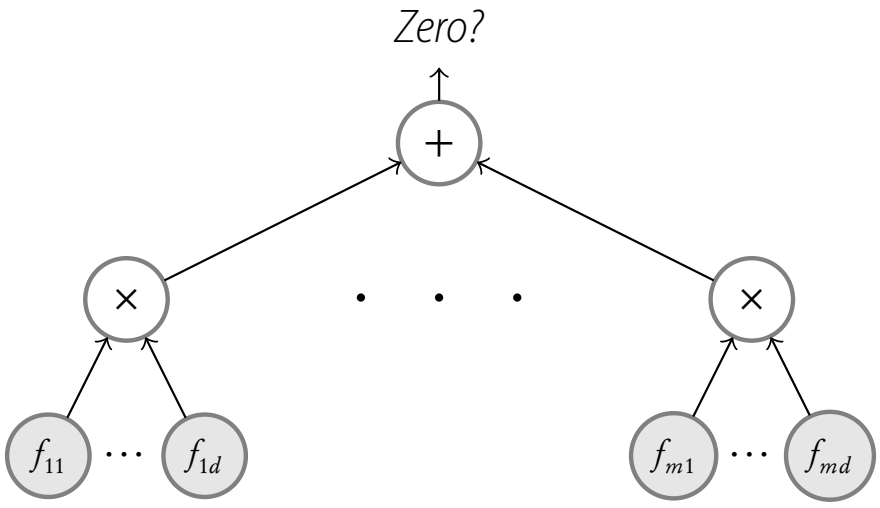
Theorem



If $\text{algRank}\{f_1, \cdots, f_m\} = k$, *then* $\text{size}(f_i) \geq 2^{n/k^2}$

PITs & Lower bounds

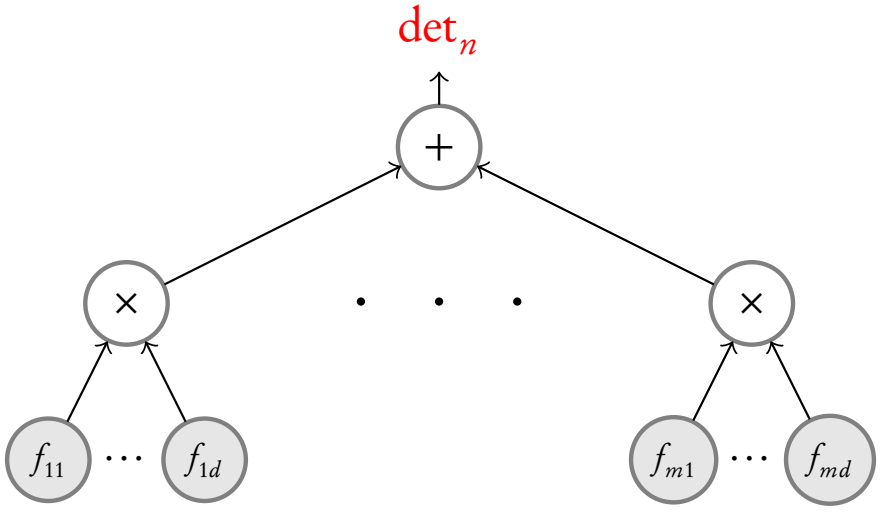
Theorem



Black-box PIT if at most $O(1)$ *of the* f_i*'s depend on any* x_j.

PITs & Lower bounds

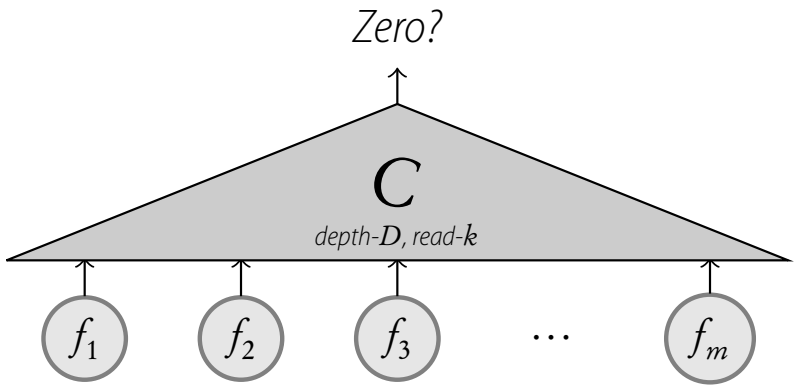
Theorem



If at most k *of the* f_i*'s depend on any* x_j*, then* $\text{size}(f_i) \geq 2^{n/k^3}$

PITs & Lower bounds

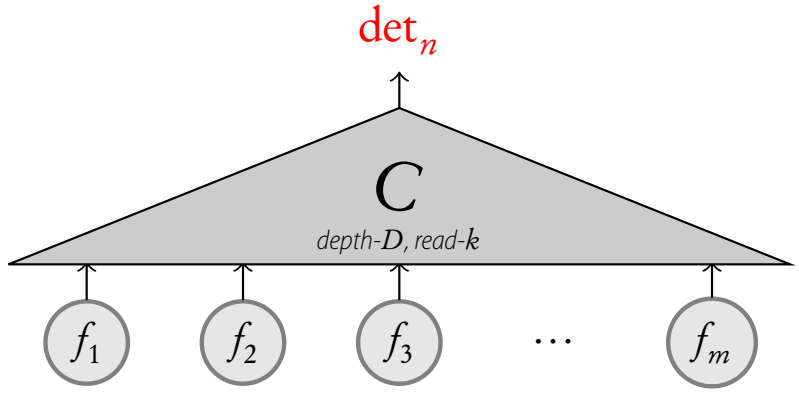
Theorem



Black-box PIT if at most $O(1)$ *of the* f_i*'s depend on any* x_j.

PITs & Lower bounds

Theorem



If at most $O(1)$ *of the* f_i*'s depend on any* x_j*, then* $\text{size}(f_i) \geq 2^{\Omega(n)}$*, assuming a conjecture about determinants is true.*

An illustrative example

Theorem

If $\det_n = C(f_1, \cdots, f_k)$, *then one of the* f_i*'s has* $2^{n/k^2}$ *monomials .*

Proof

$$\mathscr{J}(\det_n, f_1, \cdots, f_k) \quad =$$

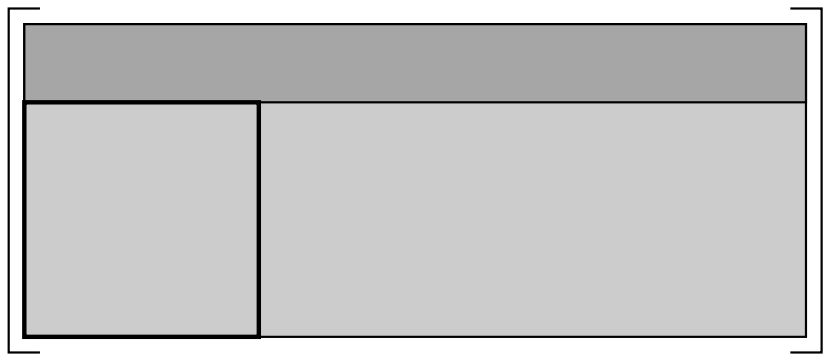
An illustrative example

Theorem

If $\det_n = C(f_1, \cdots, f_k)$, *then one of the* f_i*'s has* $2^{n/k^2}$ *monomials .*

Proof

$$\mathcal{J}(\det_n, f_1, \cdots, f_k) \quad =$$



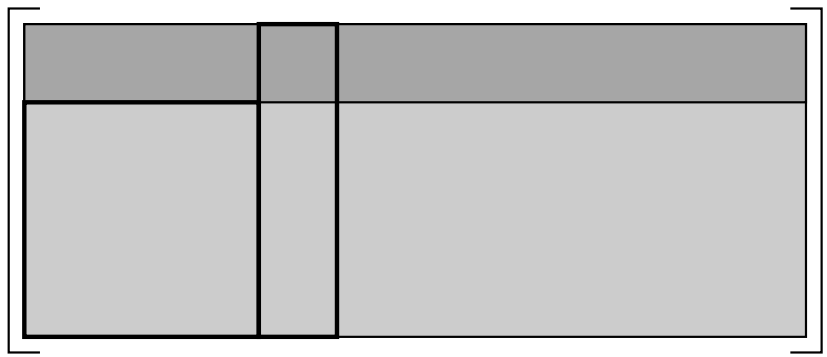
An illustrative example

Theorem

If $\det_n = C(f_1, \cdots, f_k)$, *then one of the* f_i*'s has* $2^{n/k^2}$ *monomials .*

Proof

$$\mathcal{J}(\det_n, f_1, \cdots, f_k) \quad =$$



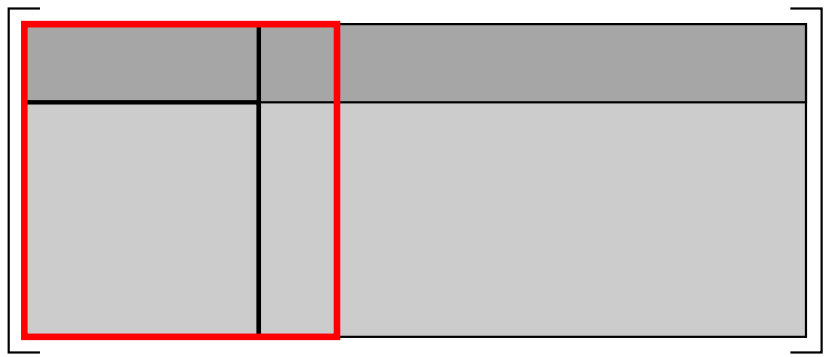
An illustrative example

Theorem

If $\det_n = C(f_1, \cdots, f_k)$, *then one of the* f_i*'s has* $2^{n/k^2}$ *monomials .*

Proof

$$\mathcal{J}(\det_n, f_1, \cdots, f_k) \quad =$$



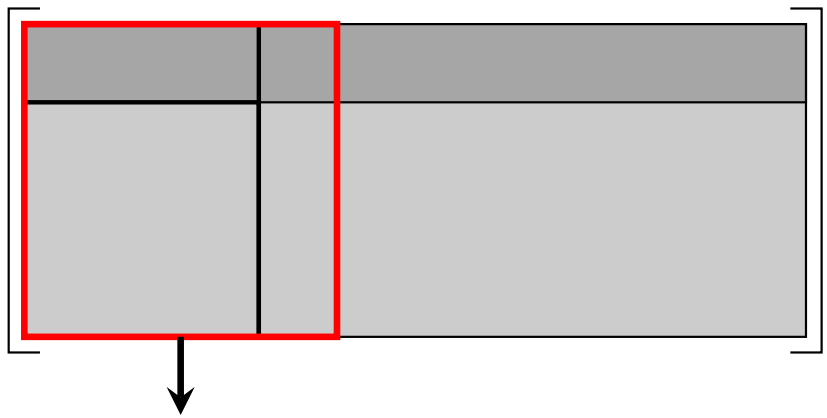
An illustrative example

Theorem

If $\det_n = C(f_1, \cdots, f_k)$, *then one of the* f_i*'s has* $2^{n/k^2}$ *monomials .*

Proof

$$\mathcal{J}(\det_n, f_1, \cdots, f_k) =$$



$$\sum_{i=1}^{k} M_i \cdot g_i = 0$$

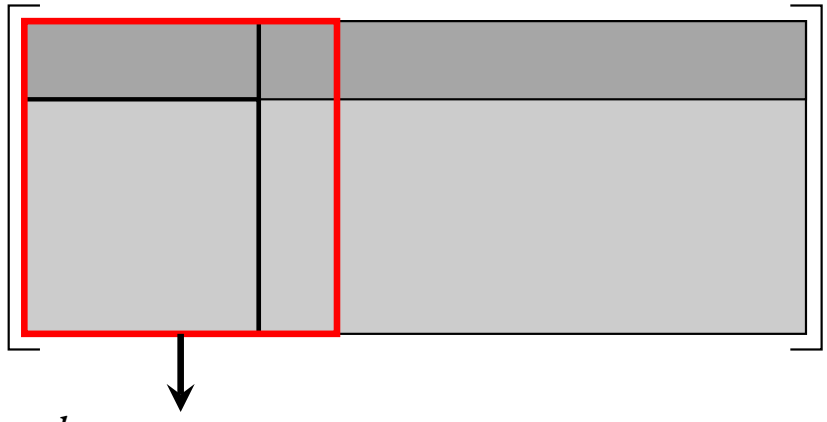
An illustrative example

Theorem

If $\det_n = C(f_1, \cdots, f_k)$, *then one of the* f_i*'s has* $2^{n/k^2}$ *monomials .*

Proof

$$\mathcal{J}(\det_n, f_1, \cdots, f_k) \quad =$$



$$\sum_{i=1}^{k} M_i \cdot g_i \quad = \quad 0$$

Is this possible?!

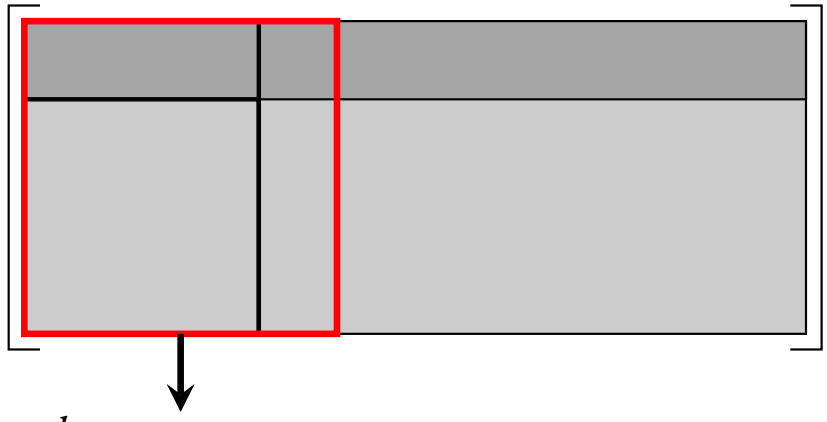
An illustrative example

Theorem

If $\det_n = C(f_1, \cdots, f_k)$, *then one of the* f_i*'s has* $2^{n/k^2}$ *monomials .*

Proof

$$\mathcal{J}(\det_n, f_1, \cdots, f_k) \quad =$$



$$\sum_{i=1}^{k} M_i \cdot g_i \quad = \quad 0$$

Is this possible?!

Not unless $\mathbf{size}(g_i) > 2^{n/k}$

Hanc marginis exiguitas non caperet.

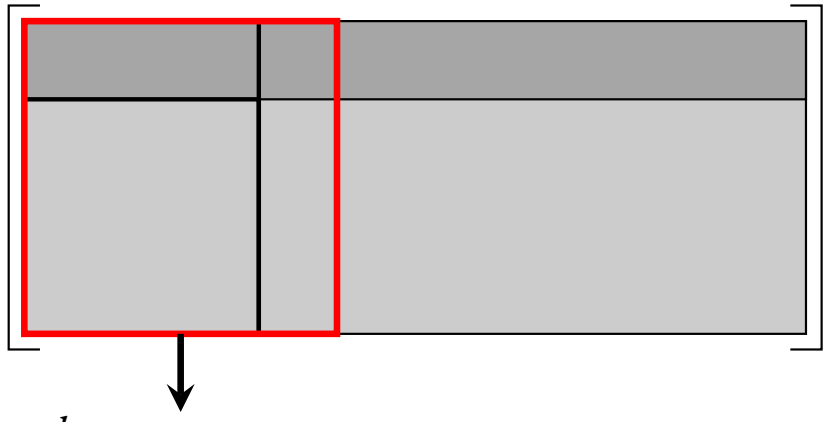
An illustrative example

Theorem

If $\det_n = C(f_1, \cdots, f_k)$, *then one of the* f_i*'s has* $2^{n/k^2}$ *monomials .*

Proof

$$\mathcal{J}(\det_n, f_1, \cdots, f_k) \quad =$$



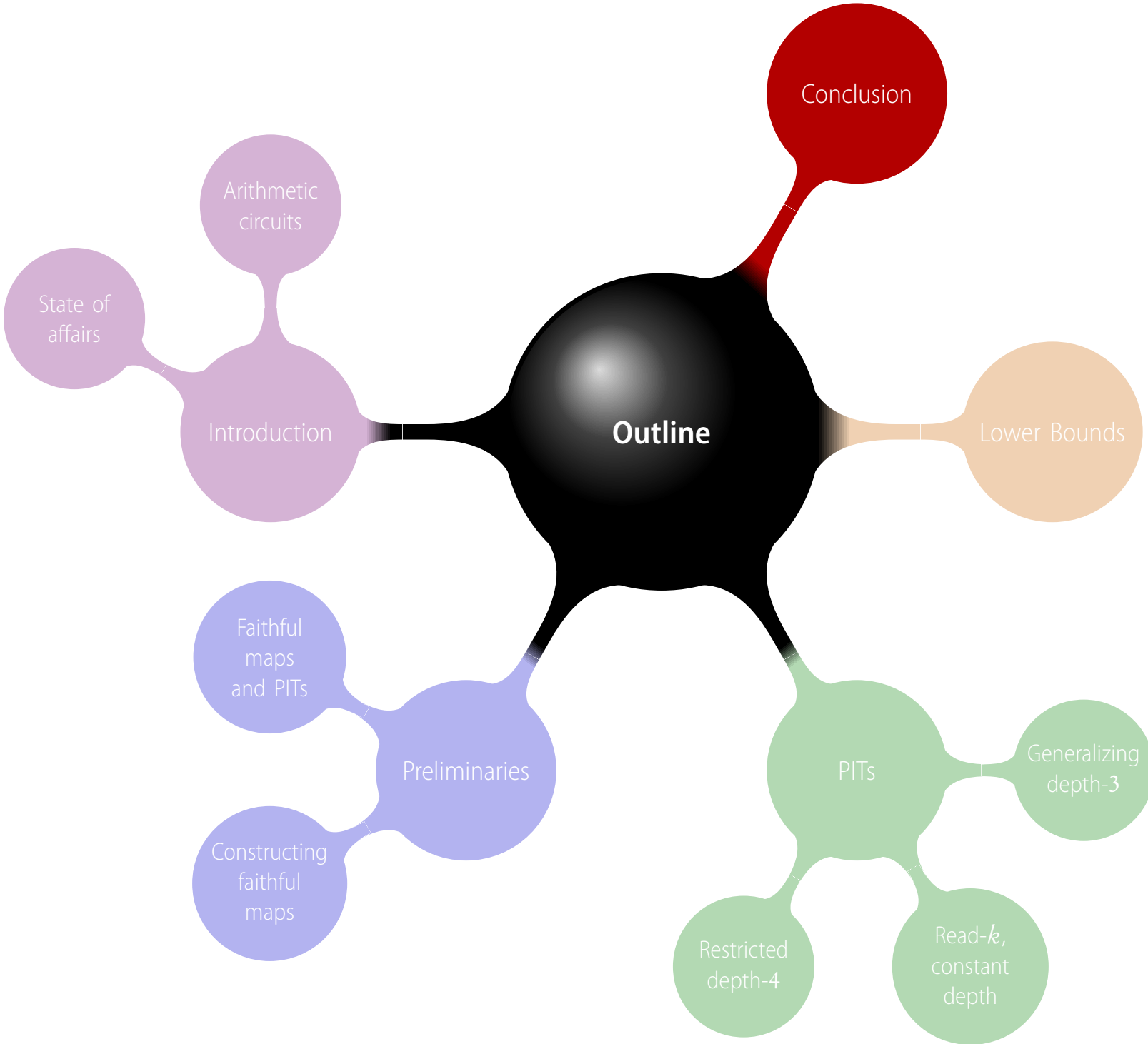
$$\sum_{i=1}^{k} M_i \cdot g_i \quad = \quad 0$$

Is this possible?!

Not unless $\mathbf{size}(g_i) > 2^{n/k}$

Hanc marginis exiguitas non caperet.

□


Outline
Introduction
Arithmetic circuits
State of affairs
Preliminaries
Faithful maps and PITs
Constructing faithful maps
Lower Bounds
PITs
Generalizing depth-3
Restricted depth-4
Read-k, constant depth
Conclusion

Concluding Remarks

- Generalizes all known polynomial time black-box PITs for sub-classes of constant depth formulae.
- Unified approach.
- Simpler proofs.

Open Problems

Models where PIT is known but not hit by the Jacobian (yet!)

- Arbitrary depth, read-k formulae

  However, Jacobian gives a quasipoly blackbox test for arbitrary depth read-1 formulae

- Diagonal circuits: $\ell_1^d + \cdots + \ell_m^d \stackrel{?}{=} 0$

  Polynomial time non-blackbox known [Kayal09,Saxena08]

Others problems

- PIT for bounded fan-in depth-4 circuits? With constant degree sparse polyomials?
- PITs for polynomials with low dimension partial-derivative space?
- Conjecture on independence of minors
- Fields of small characteristic

Open Problems

Models where PIT is known but not hit by the Jacobian (yet!)

- Arbitrary depth, read-k formulae

  However, Jacobian gives a quasipoly blackbox test for arbitrary depth read-1 formulae

- Diagonal circuits: $\ell_1^d + \cdots + \ell_m^d \stackrel{?}{=} 0$

  Polynomial time non-blackbox known [Kayal09,Saxena08]

Thanks!

Others problems

- PIT for bounded fan-in depth-4 circuits? With constant degree sparse polyomials?
- PITs for polynomials with low dimension partial-derivative space?
- Conjecture on independence of minors
- Fields of small characteristic