Polynomial Identity Testing for Read-4 Arithmetic Formulas

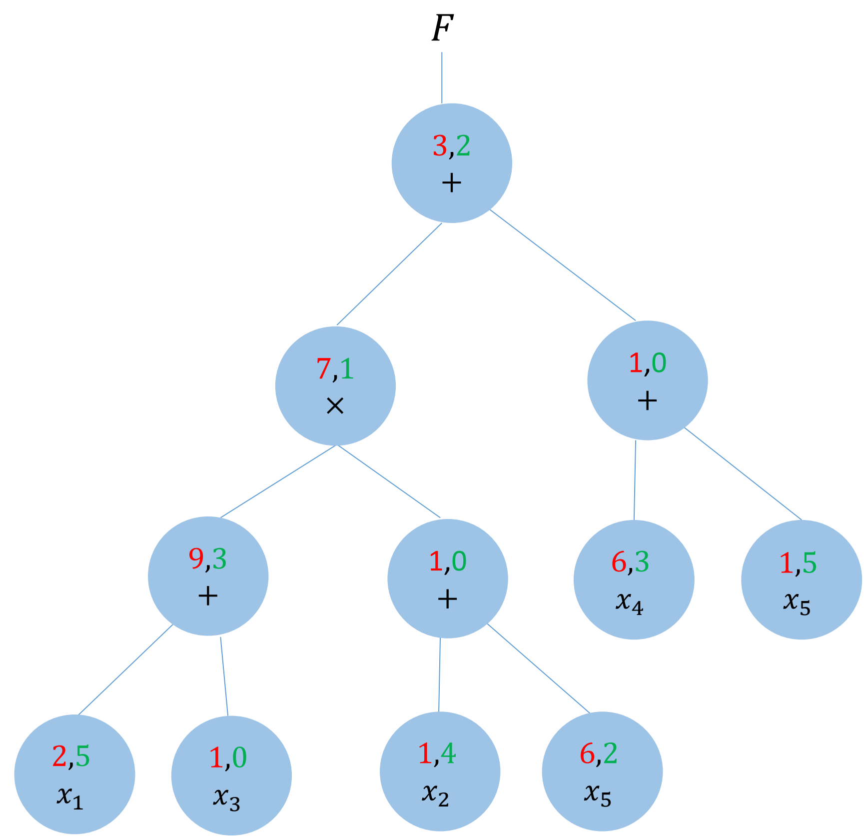
Nimrod Kaplan Amir Shpilka

Overview

- Background
- Our results
- Why is Read-4 interesting?
- Some new techniques
- Future directions – higher reads

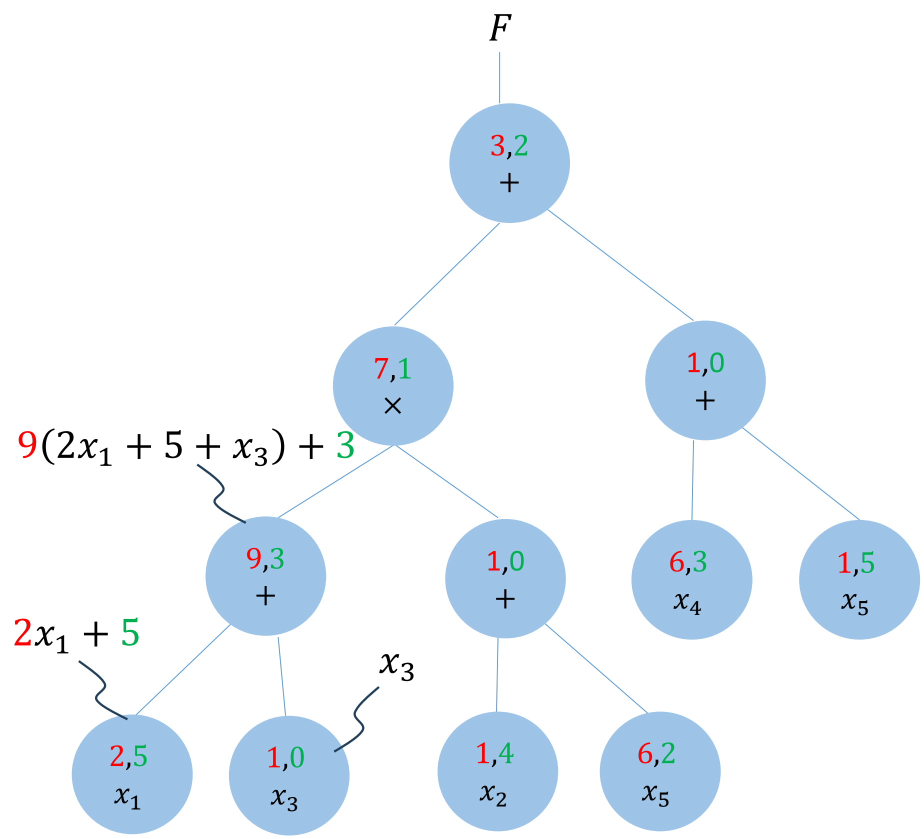
Bounded Read Arithmetic Formulas

- Binary tree
- Inputs are variables $x_1, x_2, \ldots x_n = \vec{x}$
- Gates labeld with $+$ and $\times$
- Multiplicative and additive constants
- Computation in the "natural" way
- Read-k formula: Every variable $x_i \in \boldsymbol{x}$ appears at most k times (RkF, ROF)



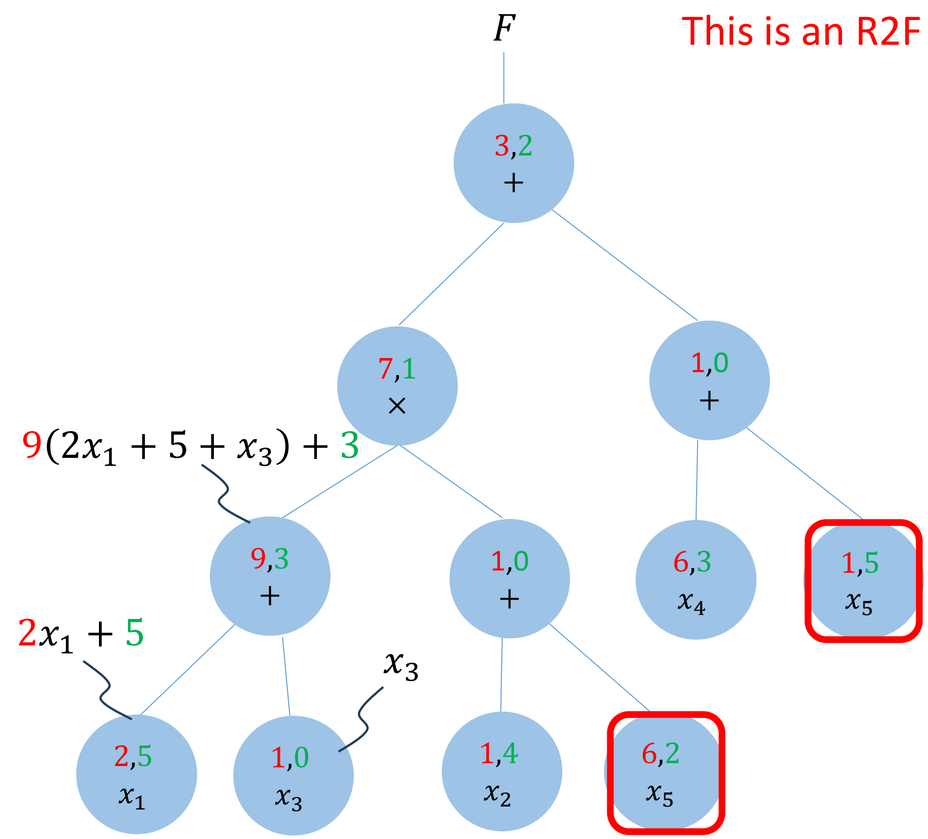
Bounded Read Arithmetic Formulas

- Binary tree
- Inputs are variables $x_1, x_2, \ldots x_n = \vec{x}$
- Gates labeld with $+$ and $\times$
- Multiplicative and additive constants
- Computation in the "natural" way
- Read-k formula: Every variable $x_i \in \boldsymbol{x}$ appears at most k times (RkF, ROF)



Bounded Read Arithmetic Formulas

- Binary tree
- Inputs are variables $x_1, x_2, \ldots x_n = \vec{x}$
- Gates labeld with $+$ and $\times$
- Multiplicative and additive constants
- Computation in the "natural" way
- Read-k formula: Every variable $x_i \in \boldsymbol{x}$ appears at most k times (RkF, ROF)



Polynomial Identity Testing (PIT)

Problem. Given $f \in \mathbb{F}[\boldsymbol{x}]$, check if $f \equiv 0$.

Polynomial Identity Testing (PIT)

Problem. Given $f \in \mathbb{F}[\boldsymbol{x}]$, check if $f \equiv 0$.

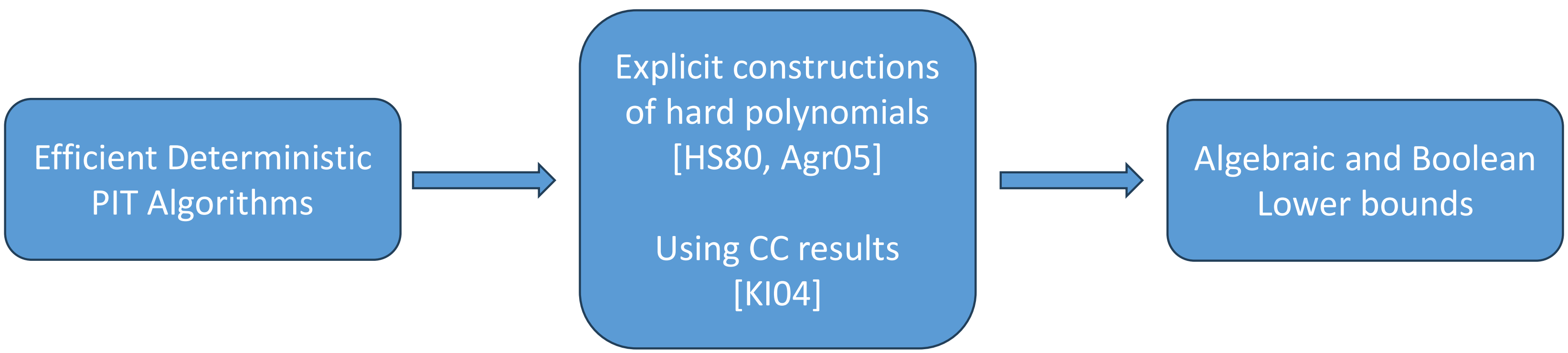
Randomized algorithm: Schwartz-Zippel Lemma. Choose a random point from a “big enough” set.

Polynomial Identity Testing (PIT)

Problem. Given $f \in \mathbb{F}[\boldsymbol{x}]$, check if $f \equiv 0$.

Randomized algorithm: Schwartz-Zippel Lemma. Choose a random point from a "big enough" set.

We want to find a **deterministic** algorithm for PIT.



Polynomial Identity Testing (PIT)

Whitebox

$f \in \mathbb{F}[\boldsymbol{x}]$ is given to us as a circuit\formula.

Polynomial Identity Testing (PIT)

Blackbox

$f \in \mathbb{F}[\boldsymbol{x}]$ is given to us as an oracle.

Whitebox

$f \in \mathbb{F}[\boldsymbol{x}]$ is given to us as a circuit\formula.

Polynomial Identity Testing (PIT)

Blackbox

$f \in \mathbb{F}[\boldsymbol{x}]$ is given to us as an oracle.

We are only allowed to query the polynomial on chosen points.

These sets of points are called *Hitting Sets*

Whitebox

$f \in \mathbb{F}[\boldsymbol{x}]$ is given to us as a circuit\formula.

Polynomial Identity Testing (PIT)

Blackbox

$f \in \mathbb{F}[\boldsymbol{x}]$ is given to us as an oracle.

We are only allowed to query the polynomial on chosen points.

These sets of points are called *Hitting Sets*

We **hit** $\mathcal{C}\backslash f$ – found a HS for $\mathcal{C}\backslash f$.

Whitebox

$f \in \mathbb{F}[\boldsymbol{x}]$ is given to us as a circuit\formula.

Previous Work

Model	Whitebox	Blackbox	BB Orbits
ROF	Poly	Poly [SV15,MV17]	Quasi-poly [MS22]
R2F	Poly [MRS14]	Quasi-Poly [Pra19, Sha23]	?
R3F	Poly [MRS14]	Quasi-Poly [Pra19]	?
R4F	?	?	?
∑ROF	Poly	Poly [SV15,MV17]	?
Multilinear-RkF	Poly [AvMV15]	Quasi-Poly [AvMV15]	?

[SV15] – Amir Shpilka and Ilya Volkovich (STOC)
[MV17] – Daniel Minahan and Ilya Volkovich (CCC)
[AvMV15] - Anderson et al. (CCC)
[Pra19] - Gautam Prakriya (Doctoral Thesis)
[MS22] – Dori Medini and Amir Shpilka (CCC)
[Sha23] – Nadav Shamir (Master Thesis)
[MRS14] – Mahajan et al. (TCS)

Previous Work

$$Orbit(\mathcal{C}) = \left\{ f(\ell_1, \dots, \ell_n) \middle| \begin{array}{c} f \in \mathcal{C}, \\ \ell_i\text{'s linearly independent} \end{array} \right\}$$

Model	Whitebox	Blackbox	BB Orbits
ROF	Poly	Poly [SV15,MV17]	Quasi-poly [MS22]
R2F	Poly [MRS14]	Quasi-Poly [Pra19, Sha23]	?
R3F	Poly [MRS14]	Quasi-Poly [Pra19]	?
R4F	?	?	?
$\sum$ROF	Poly	Poly [SV15,MV17]	?
Multilinear-RkF	Poly [AvMV15]	Quasi-Poly [AvMV15]	?

[SV15] – Amir Shpilka and Ilya Volkovich (STOC)
[MV17] – Daniel Minahan and Ilya Volkovich (CCC)
[AvMV15] - Anderson et al. (CCC)
[Pra19] - Gautam Prakriya (Doctoral Thesis)
[MS22] – Dori Medini and Amir Shpilka (CCC)
[Sha23] – Nadav Shamir (Master Thesis)
[MRS14] – Mahajan et al. (TCS)

Results

$$Orbit(\mathcal{C}) = \left\{ f(\ell_1, \dots, \ell_n) \middle| \begin{array}{c} f \in \mathcal{C}, \\ \ell_i\text{'s linearly independent} \end{array} \right\}$$

Model	Whitebox	Blackbox	BB Orbits
ROF	Poly	Poly [SV15,MV17]	Quasi-poly [MS22]
R2F	Poly [MRS14]	Quasi-poly *	Quasi-poly
R3F	Poly [MRS14]	Quasi-poly *	Quasi-poly
R4F	Poly	Quasi-poly	Quasi-poly
$\sum$ROF	Poly	Poly [SV15,MV17]	Quasi-Poly
Multilinear-RkF	Poly [AvMV15]	Quasi-Poly [AvMV15]	Quasi-poly

* Improved algorithm.

Main Result – PIT for R4Fs

Theorem.

There is a deterministic blackbox PIT algorithm for R4Fs in time:

$$O(2n^{c+18\log n})$$

For constant c.

Main Result – PIT for R4Fs

Theorem.

There is a deterministic blackbox PIT algorithm for R4Fs in time:

$$O(2n^{c+18\log n})$$

For constant c.

What makes read-4 so special?

Known Techniques – Partial Derivatives

Previous results:

PIT for New Model $\xrightarrow[\text{via partial derivatives}]{\text{reduce}}$ Known PIT

Known Techniques – Partial Derivatives

<u>Previous results:</u>

PIT for New Model $\xrightarrow[\text{via partial derivatives}]{\text{reduce}}$ Known PIT

$f \in \mathcal{C}$

Known Techniques – Partial Derivatives

<u>Previous results:</u>

PIT for New Model $\xrightarrow[\text{via partial derivatives}]{\text{reduce}}$ Known PIT

$f \in \mathcal{C} \xrightarrow{\text{reduce}}$ $S = \{\, g_i \mid g_i \text{ a PD of } f \,\}$

g_i's nonzero & simple

Known Techniques – Partial Derivatives

<u>Previous results:</u>

PIT for New Model $\xrightarrow[\text{via partial derivatives}]{\text{reduce}}$ Known PIT

$f \in \mathcal{C} \xrightarrow{\text{reduce}}$ $S = \{\, g_i \mid g_i \text{ a PD of } f \,\}$ $\xrightarrow{\text{hit}}$ Find HS that hits S

${g_i}'$s nonzero & simple

Known Techniques – Partial Derivatives

<u>Previous results:</u>

PIT for New Model $\xrightarrow[\text{via partial derivatives}]{\text{reduce}}$ Known PIT

$f \in \mathcal{C}$ $\xrightarrow{\text{reduce}}$ $S = \{\, g_i \mid g_i \text{ a PD of } f \,\}$

${g_i}'$s nonzero & simple $\xrightarrow{\text{hit}}$ Find HS that hits S

HS' for $\mathcal{C}$ $\xleftarrow{\text{lifting using known results}}$

Known Techniques – Partial Derivatives

Previous results:

$$\text{PIT for New Model} \xrightarrow[\text{via partial derivatives}]{\text{reduce}} \text{Known PIT}$$

- ROF: Homogenization
- $\sum$ROF: PDs + Hardness of Representation → ROF PIT
- R2F: Fragmenting (repeated PDs) → $\sum^2 \text{ROF}$ PIT
- R3F: Fragmenting (repeated PDs) → R2F PIT
- ml-RkF: Fragmenting/Shattering (repeated PDs) → ml-R$(k-1)$F
- Orbits for ROFs: Directional derivatives

Known Techniques – Partial Derivatives

Previous results:

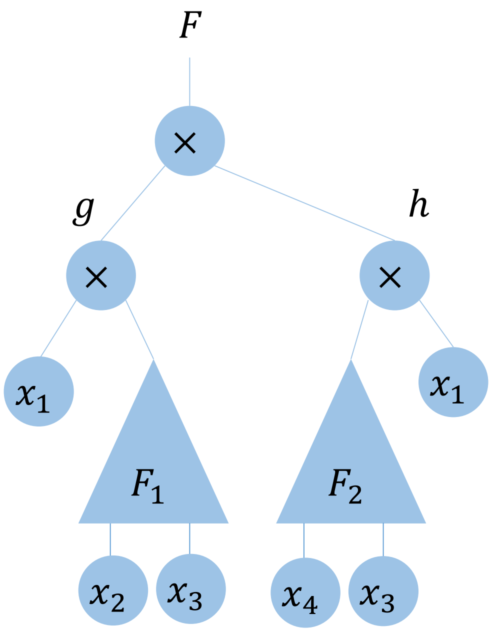
$$\text{PIT for New Model} \xrightarrow[\text{via partial derivatives}]{\text{reduce}} \text{Known PIT}$$

- ROF: Homogenization
- $\sum$ROF: PDs + Hardne
- R2F: Fragmenting (re
- R3F: Fragmenting (re
- ml-RkF: Fragmentin
- Orbits for ROFs: Direc

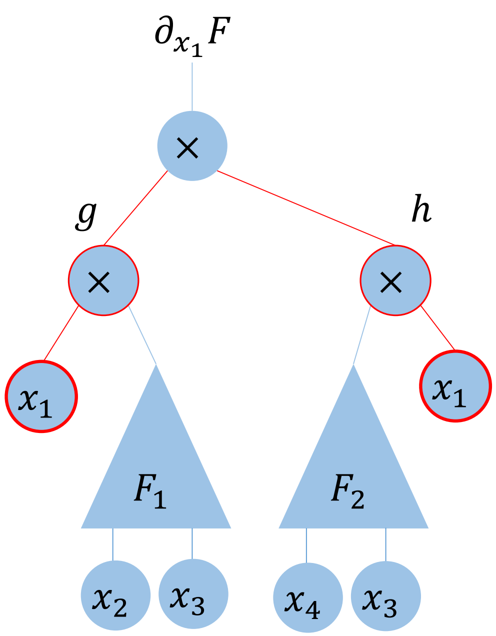
PDs **simplify** the problem

This means, these models are **closed** under PDs

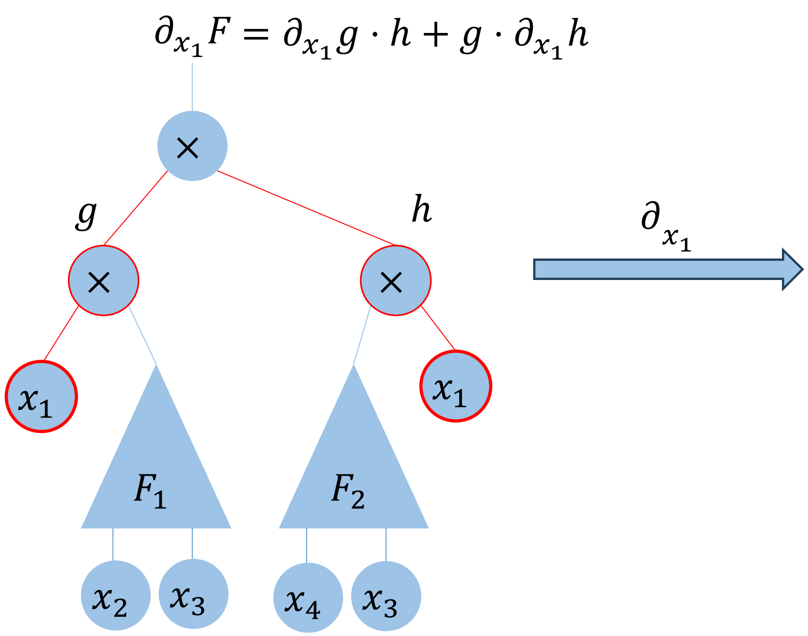
Differentiation Problems



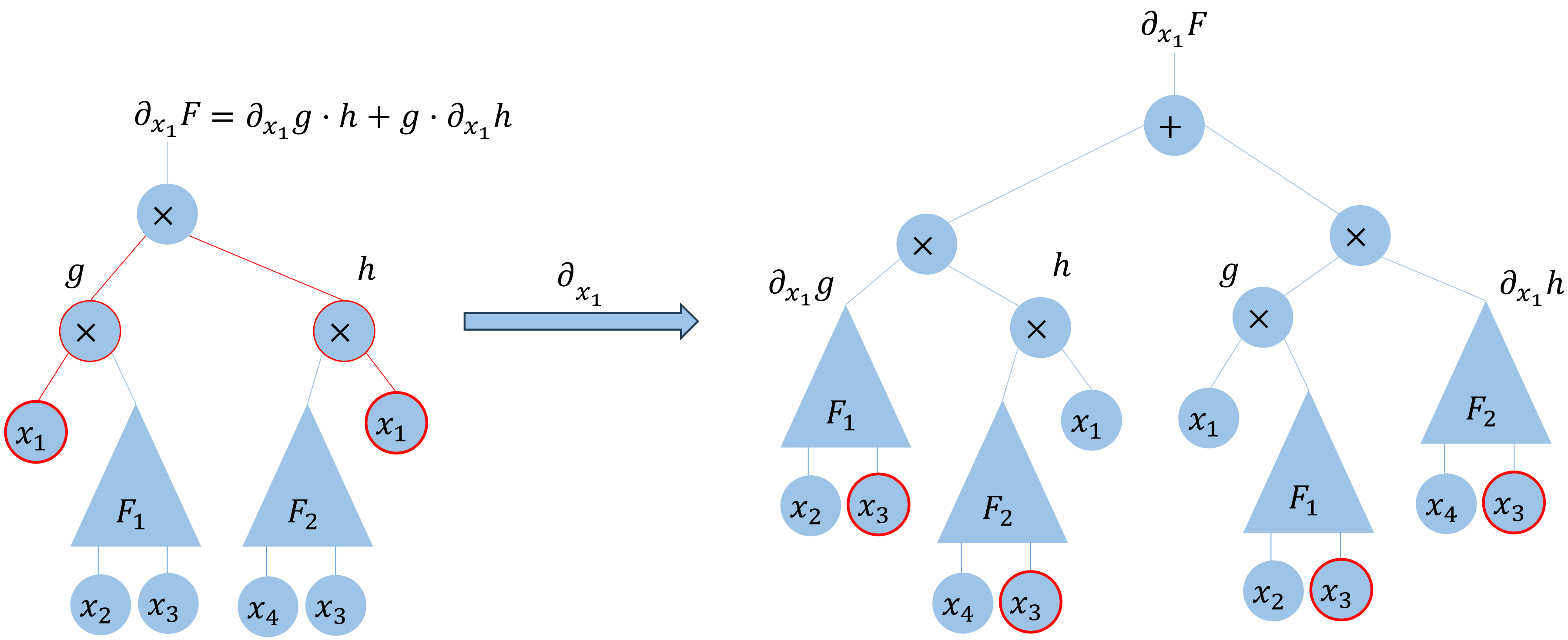
Differentiation Problems



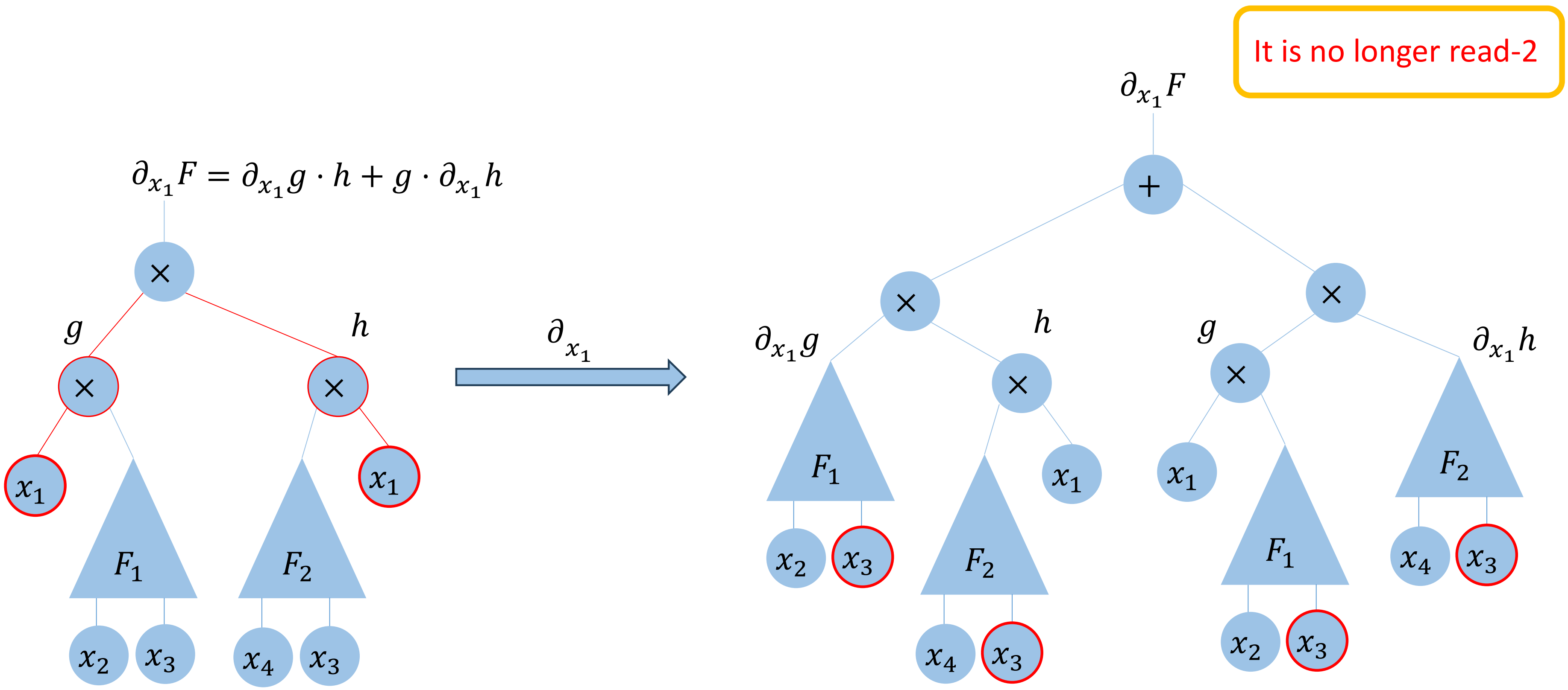
Differentiation Problems



Differentiation Problems

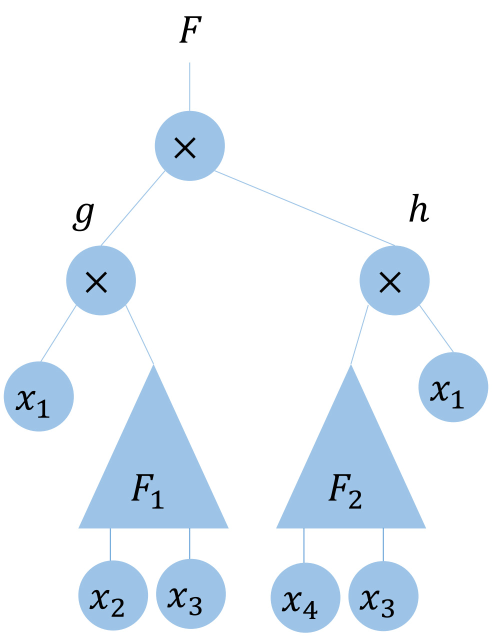


Differentiation Problems



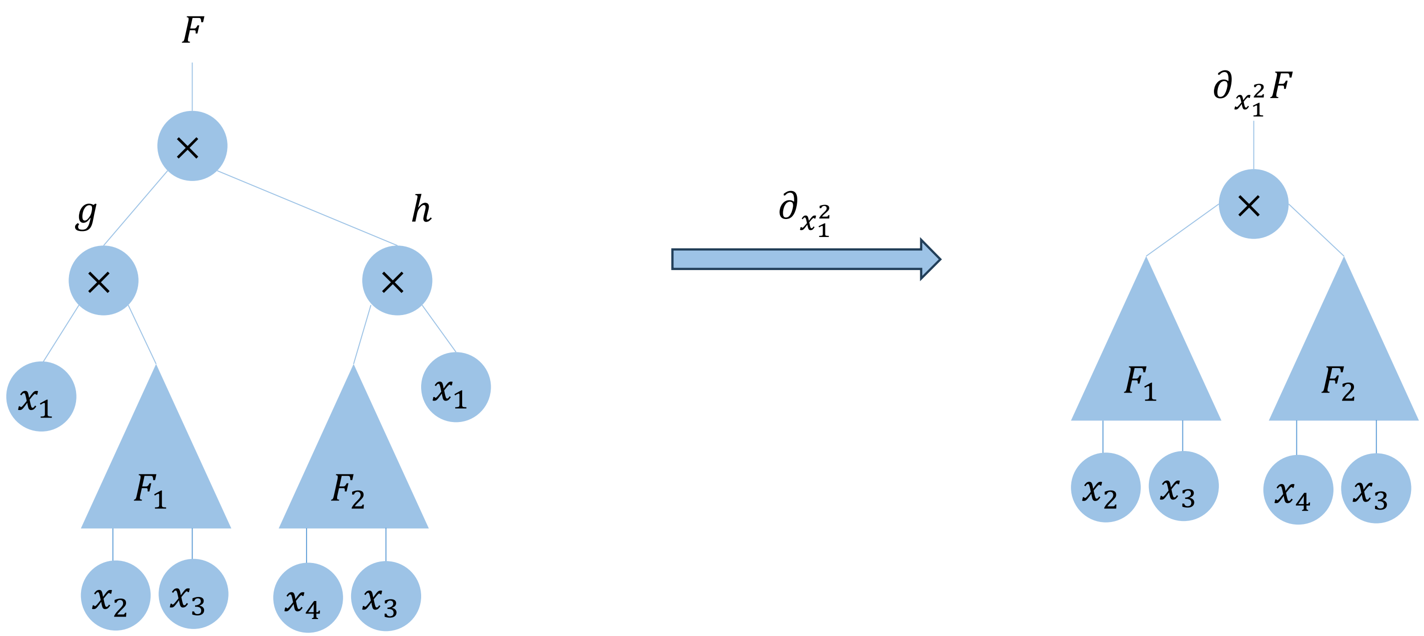
Differentiation Problems

Solution (for R2Fs): Differentiate twice.

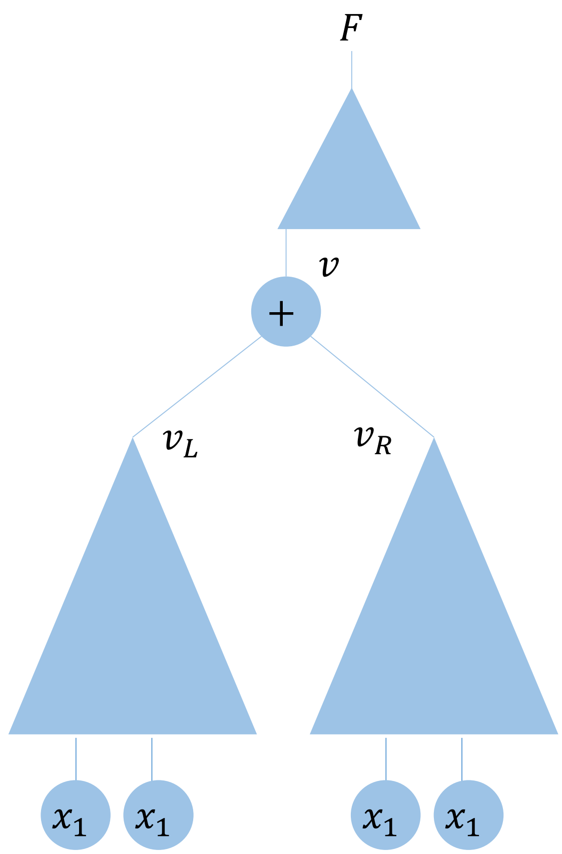


Differentiation Problems

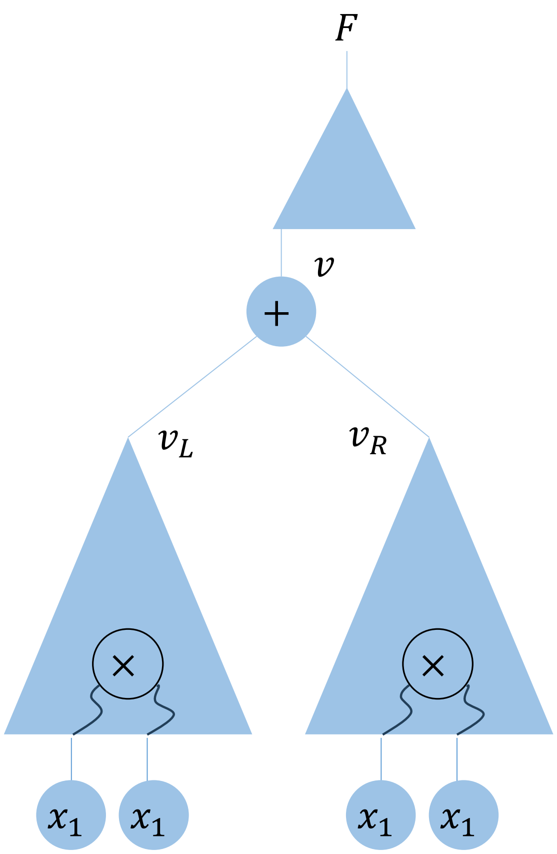
Solution (for R2Fs): Differentiate twice.



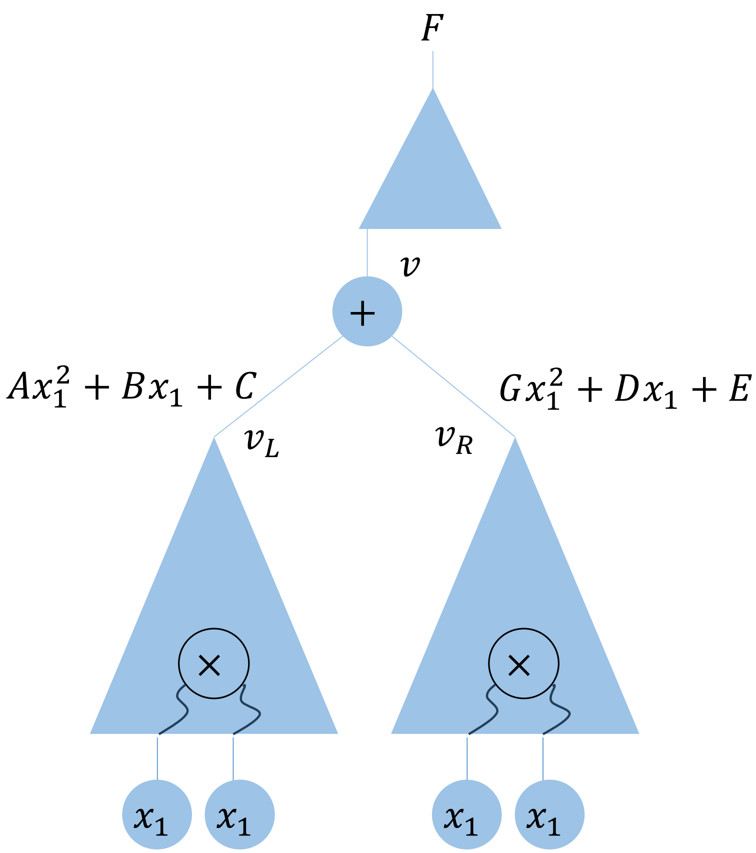
Differentiation Problems



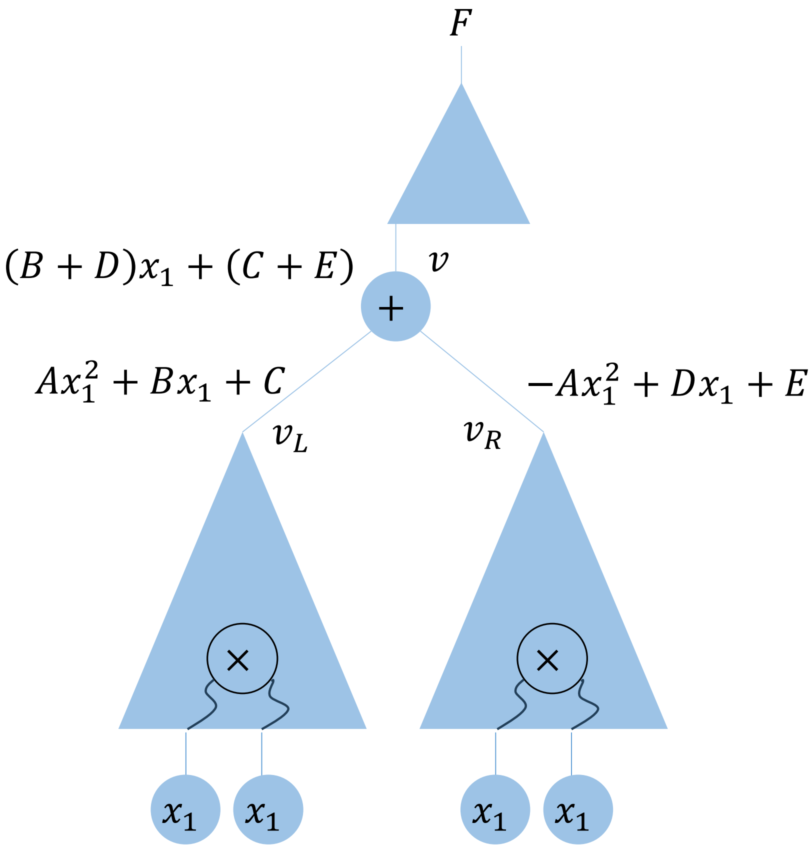
Differentiation Problems



Differentiation Problems

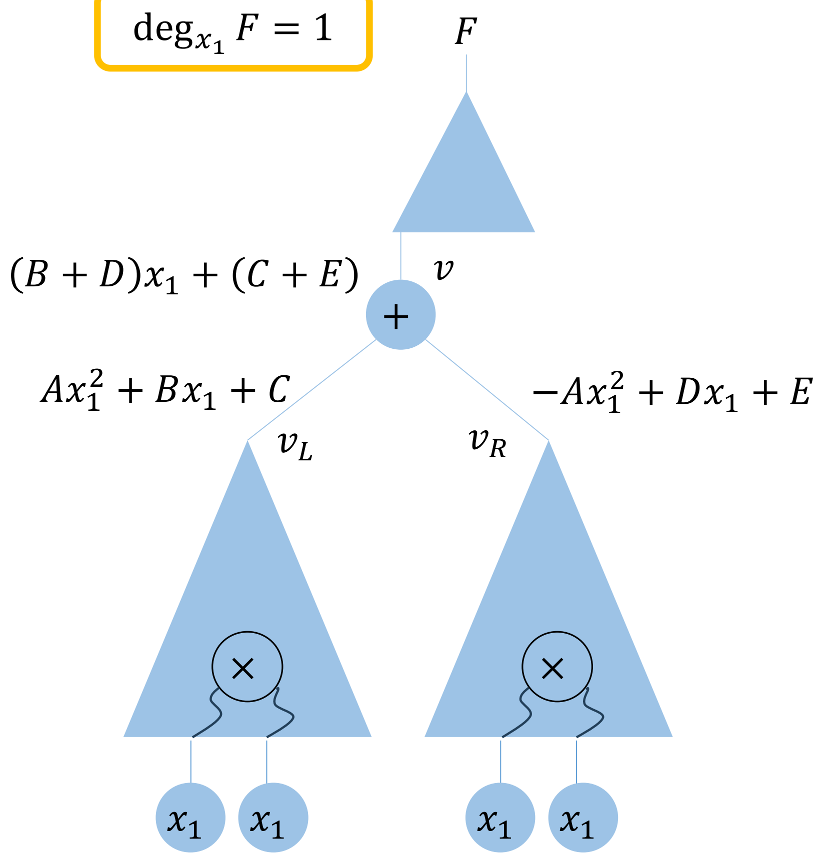


Differentiation Problems

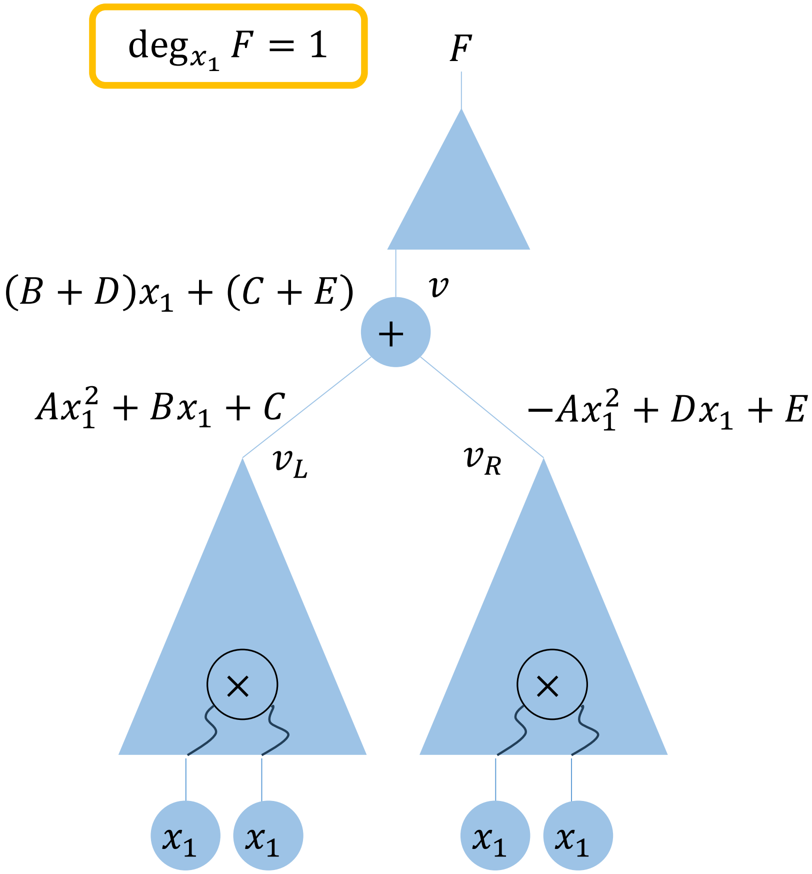


Differentiation Problems

$\boxed{\deg_{x_1} F = 1}$

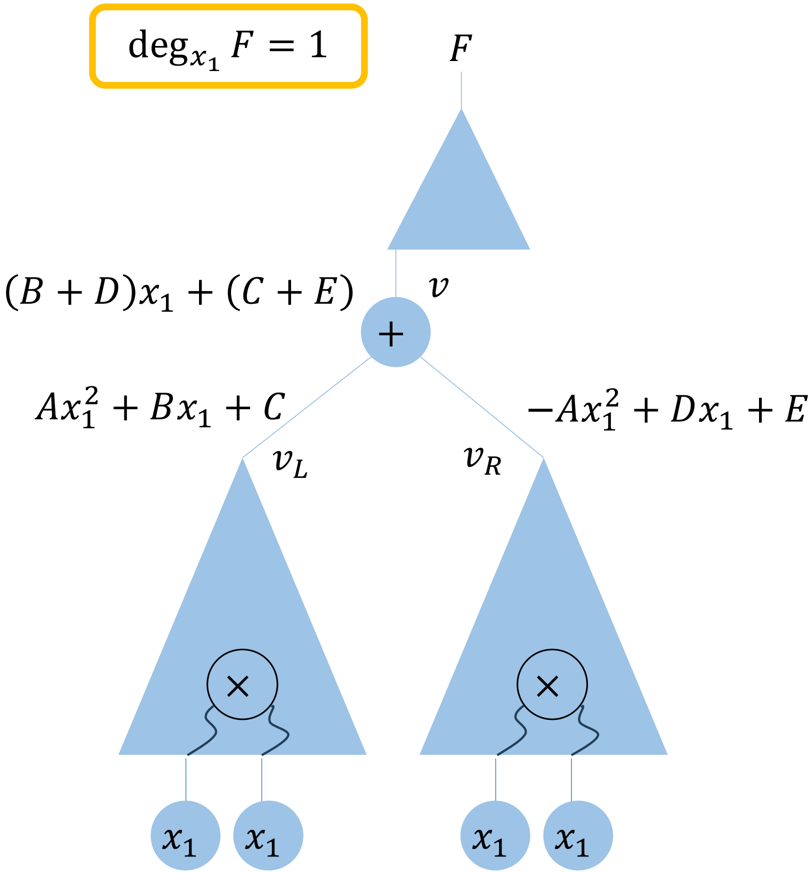


Differentiation Problems



Can we use ∂_{x_1}?

Differentiation Problems

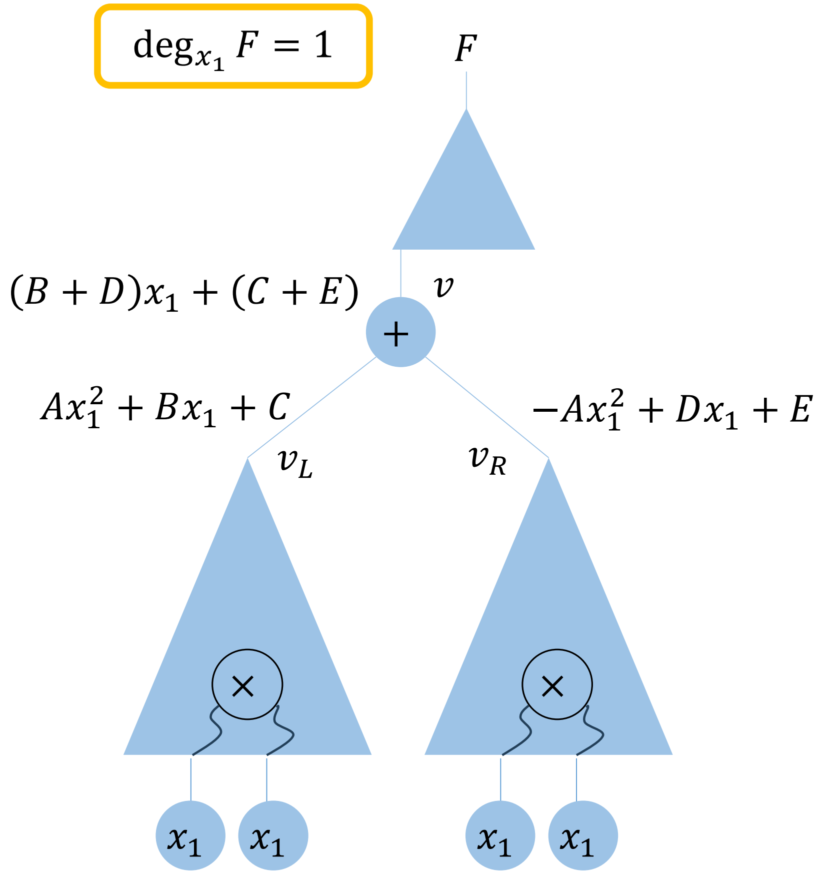


No! we increase reads

Can we use ∂_{x_1}?

Differentiation Problems

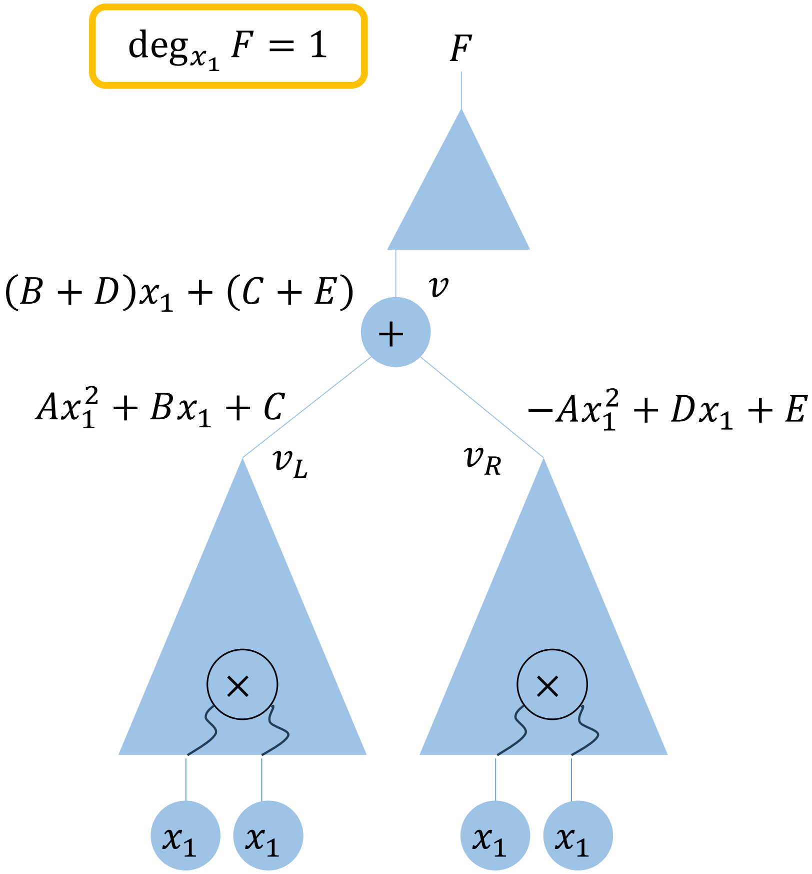
$\boxed{\deg_{x_1} F = 1}$



No! we increase reads

Can we use ∂_{x_1}? $\partial_{x_1^2}$?

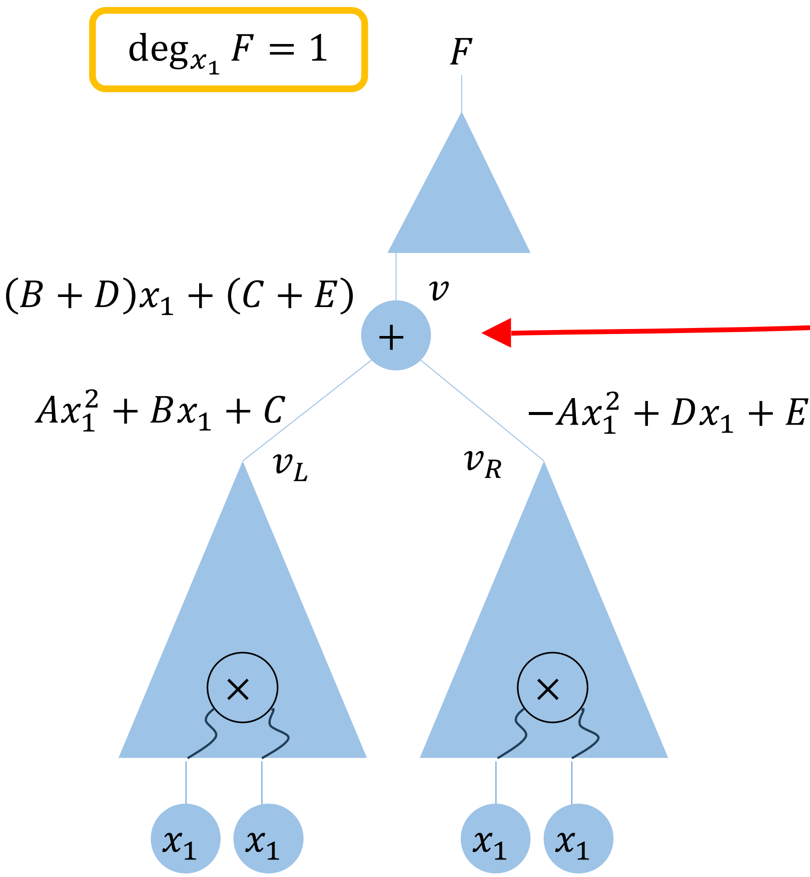
Differentiation Problems



No! we increase reads

Can we use ∂_{x_1}? $\partial_{x_1^2}$? No! we get zero

Differentiation Problems



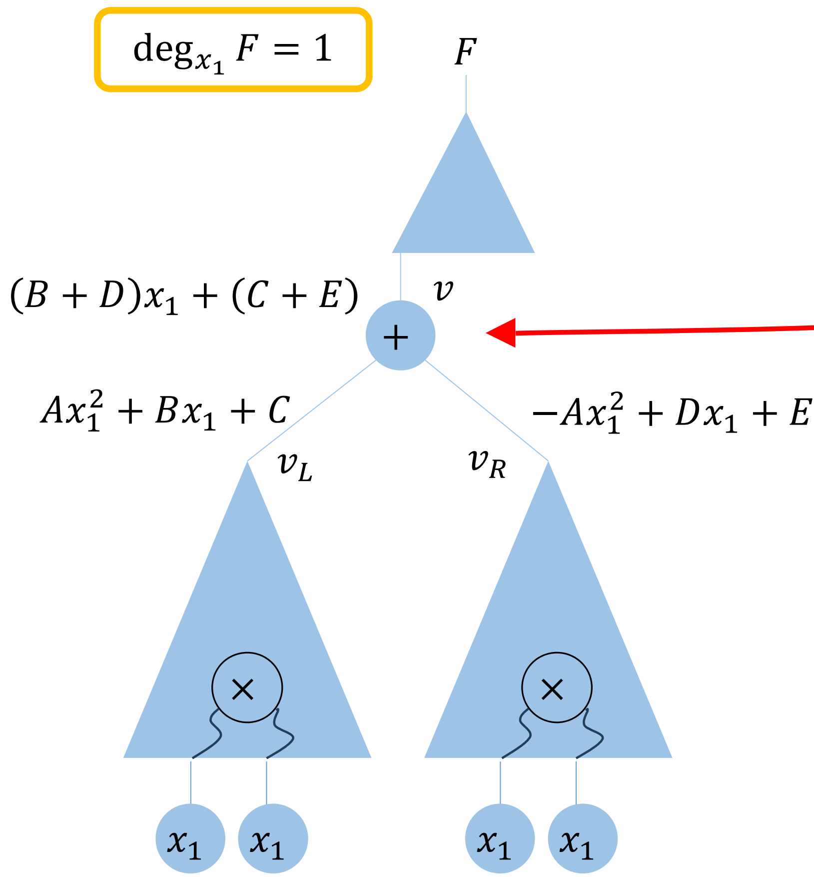
No! we increase reads

Can we use ∂_{x_1}? $\partial_{x_1^2}$? No! we get zero

The problem is here

$$0 < \deg_{x_1}(v) < \deg_{x_1}(v_L), \deg_{x_1}(v_R)$$

Differentiation Problems



No! we increase reads

Can we use ∂_{x_1}? $\partial_{x_1^2}$? No! we get zero

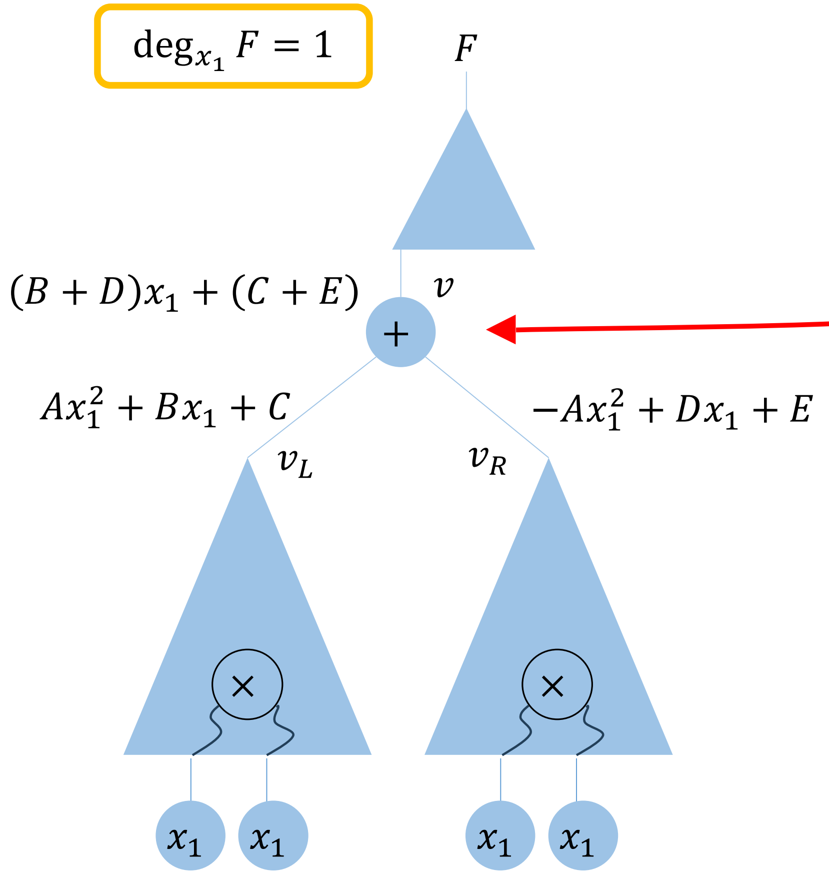
The problem is here

$$0 < \deg_{x_1}(v) < \deg_{x_1}(v_L), \deg_{x_1}(v_R)$$

Requires read ≥ 4!

Differentiation Problems

$\boxed{\deg_{x_1} F = 1}$



No! we increase reads

Can we use ∂_{x_1}? $\partial_{x_1^2}$? No! we get zero

The problem is here

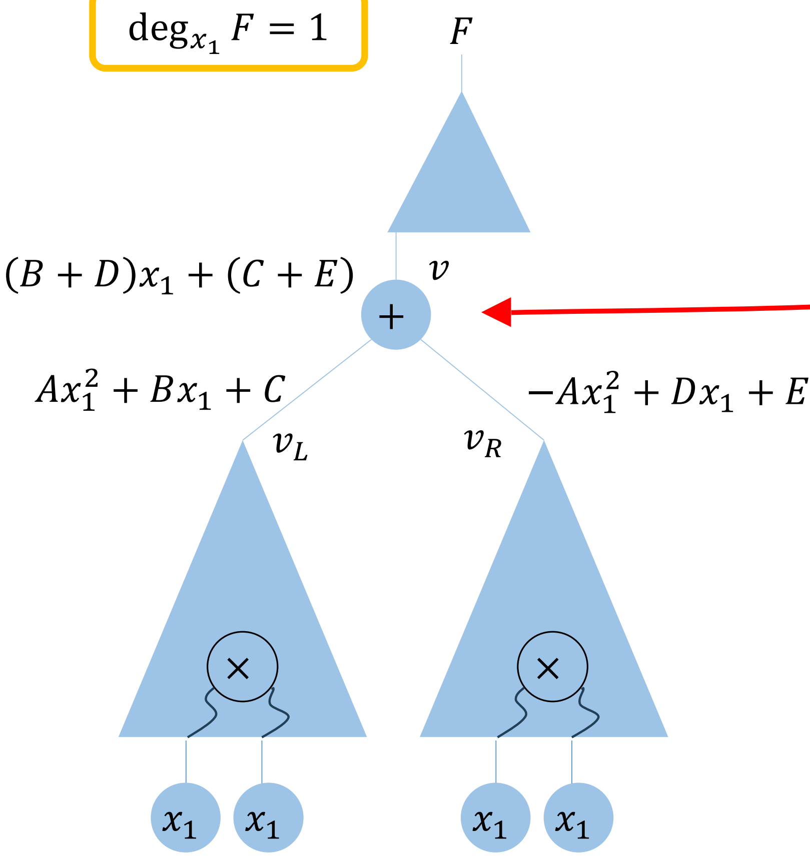
$$0 < \deg_{x_1}(v) < \deg_{x_1}(v_L), \deg_{x_1}(v_R)$$

Requires read ≥ 4!

This is the barrier that breaks the previous methods

Differentiation Problems

$\boxed{\deg_{x_1} F = 1}$



No! we increase reads

Can we use ∂_{x_1}? $\partial_{x_1^2}$? No! we get zero

The problem is here

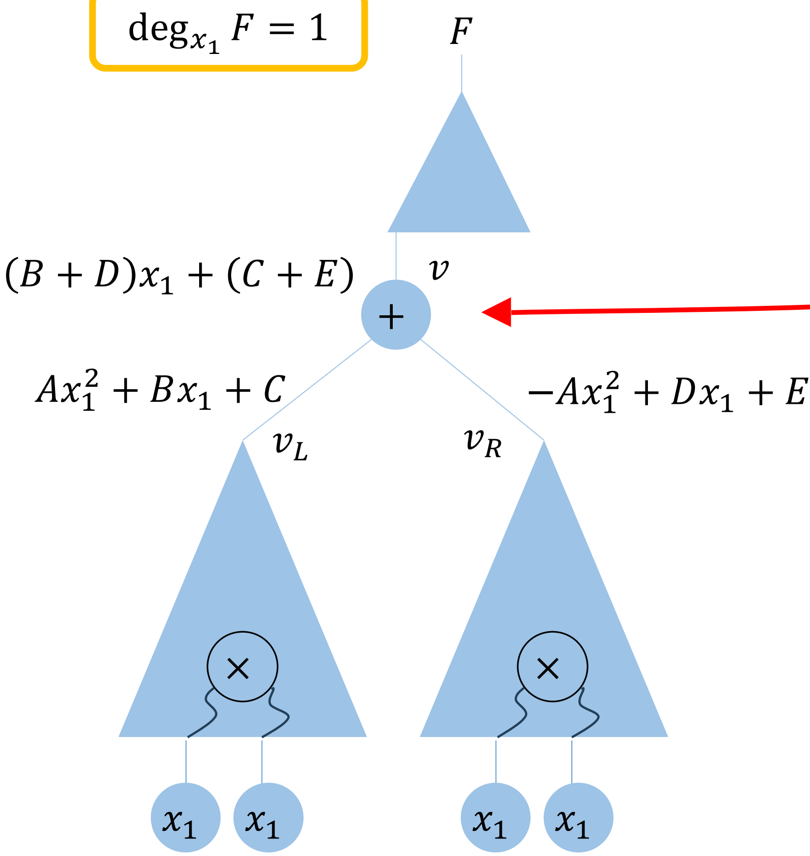
$$0 < \deg_{x_1}(v) < \deg_{x_1}(v_L), \deg_{x_1}(v_R)$$

Requires read ≥ 4!

Cancellations are exactly what makes PIT hard in general

Differentiation Problems

$\boxed{\deg_{x_1} F = 1}$



No! we increase reads

Can we use ∂_{x_1}? $\partial_{x_1^2}$? No! we get zero

The problem is here

$$0 < \deg_{x_1}(v) < \deg_{x_1}(v_L), \deg_{x_1}(v_R)$$

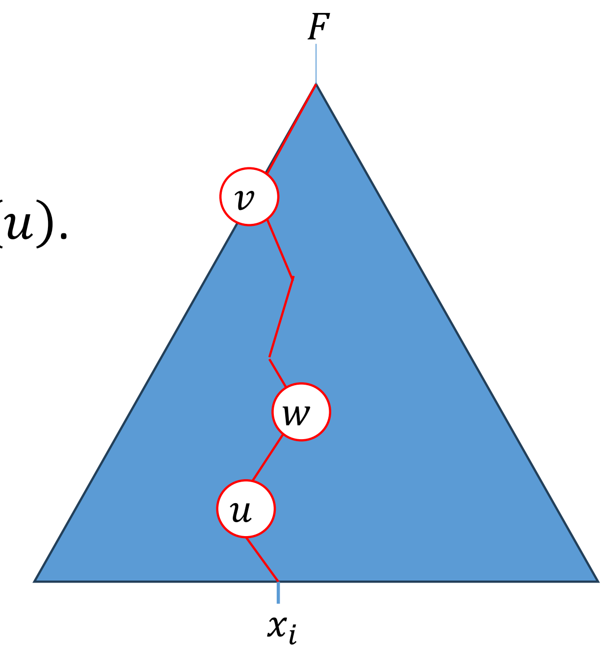
Requires read ≥ 4!

Can also happen in $\sum^2 \mathrm{R2F}$

The Structural Property

x_i is ***structural*** in F if:

$\forall$ gates v, u: v is an **ancestor** of u $\Rightarrow \deg_{x_i}(v) \geq \deg_{x_i}(u)$.



$$\deg_{x_i} u \leq \deg_{x_i} w \leq \deg_{x_i} v$$

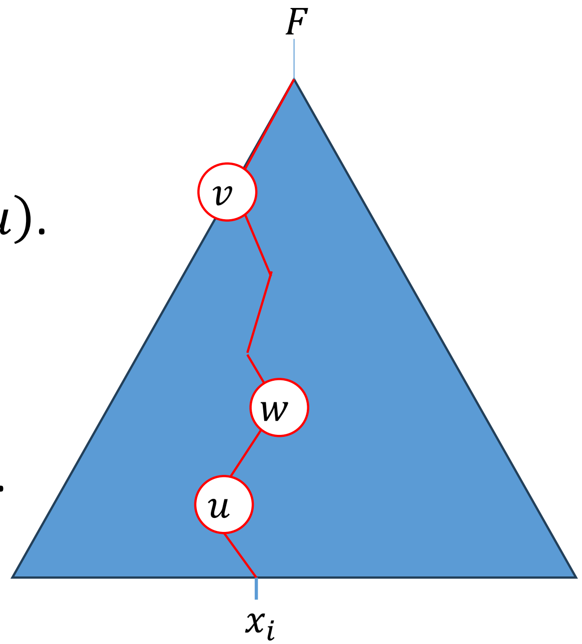
The Structural Property

x_i is ***structural*** in F if:

$\forall$ gates v, u: v is an **ancestor** of u $\Rightarrow \deg_{x_i}(v) \geq \deg_{x_i}(u)$.

F is structural if it is structural $\forall x_i$.

F is **totally** non-structural (TNS) if it is non-structural $\forall x_i$.



$\deg_{x_i} u \leq \deg_{x_i} w \leq \deg_{x_i} v$

The Structural Property

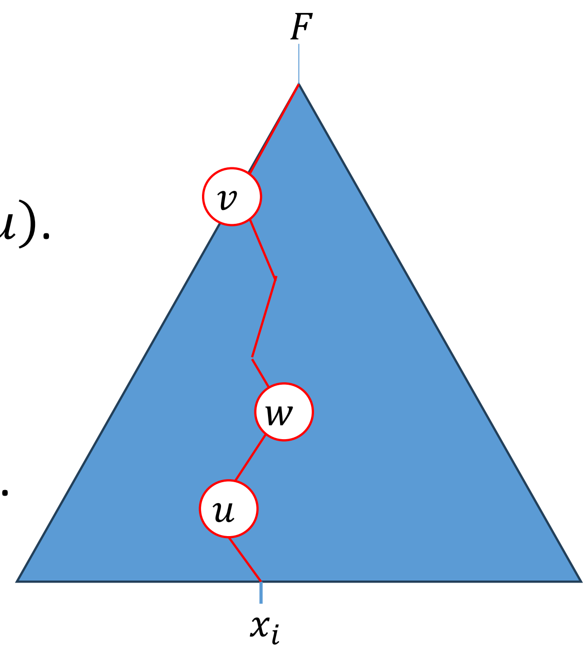
x_i is ***structural*** in F if:

$\forall$ gates v, u: v is an **ancestor** of u $\Rightarrow \deg_{x_i}(v) \geq \deg_{x_i}(u)$.

F is structural if it is structural $\forall x_i$.

F is **totally** non-structural (TNS) if it is non-structural $\forall x_i$.

F is **structural** $\Rightarrow$ previous methods **work** (mostly)



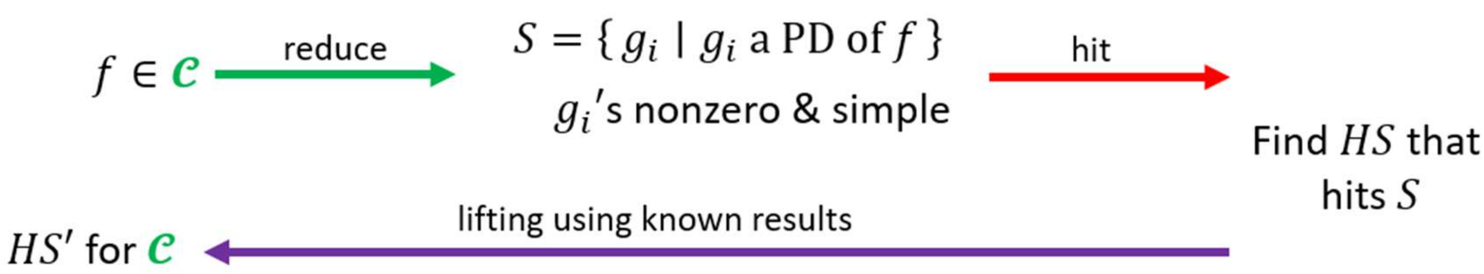
$\deg_{x_i} u \leq \deg_{x_i} w \leq \deg_{x_i} v$

New Techniques – Two Examples

1. Restrictions + Hardness of Representation.
   Divide and conquer

New Techniques – Two Examples

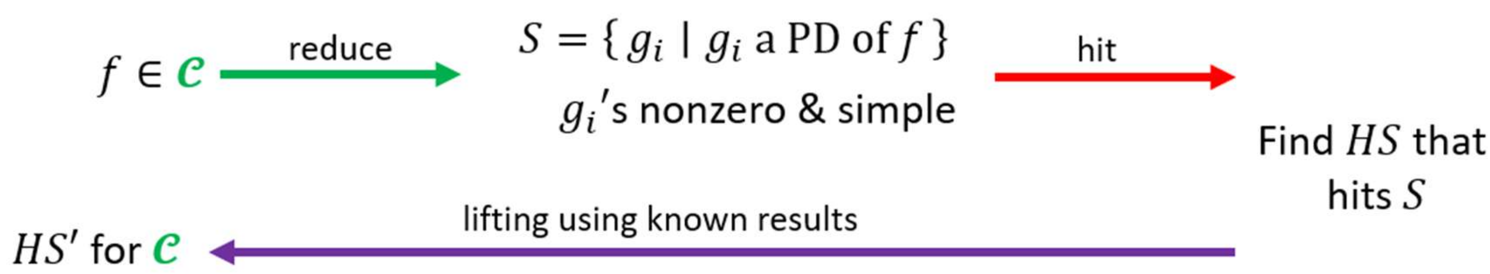
1. Restrictions + Hardness of Representation.
   Divide and conquer, replaces old framework.



New Techniques – Two Examples

1. Restrictions + Hardness of Representation.

   Divide and conquer, replaces old framework.



2. Dominating Degree Patterns.

   New closure rules (under taking PDs).

Restrictions and HoR

Idea: Divide and conquer

Restrictions and HoR

Idea: Divide and conquer

$f|_S$: Substituting a generic\random assignment to variables outside of S.
Such an assignment maintains "any property" of f.

Restrictions and HoR

Idea: Divide and conquer

$f|_S$: Substituting a generic\random assignment to variables outside of S.
Such an assignment maintains “any property” of f.

$f \in \mathcal{C}$

Restrictions and HoR

Idea: Divide and conquer

$f|_S$: Substituting a generic\random assignment to variables outside of S.
Such an assignment maintains "any property" of f.

$$f \in \mathcal{C} \xrightarrow{\text{divide}} \begin{array}{c}\text{partition } [n] = S_1 \sqcup \cdots \sqcup S_k \\ S_i\text{'s disjoint and complementing}\end{array}$$

Restrictions and HoR

Idea: Divide and conquer

$f|_S$: Substituting a generic\random assignment to variables outside of S.
Such an assignment maintains “any property” of f.

$f \in \mathcal{C} \xrightarrow{\text{divide}}$ partition $[n] = S_1 \sqcup \cdots \sqcup S_k$ $\xrightarrow{\text{conquer}}$ find efficient PIT for every $f|_{S_i}$

S_i’s disjoint and complementing

Restrictions and HoR

Idea: Divide and conquer

$f|_S$: Substituting a generic\random assignment to variables outside of S.

Such an assignment maintains "any property" of f.

$f \in \mathcal{C}$ $\xrightarrow{\text{divide}}$ partition $[n] = S_1 \sqcup \cdots \sqcup S_k$

S_i's disjoint and complementing $\xrightarrow{\text{conquer}}$

find efficient PIT

for every $f|_{S_i}$

More structure

Restrictions and HoR

Idea: Divide and conquer

$f|_S$: Substituting a generic\random assignment to variables outside of S.

Such an assignment maintains “any property” of f.

$f \in \mathcal{C} \xrightarrow{\text{divide}}$ partition $[n] = S_1 \sqcup \cdots \sqcup S_k$

S_i’s disjoint and complementing $\xrightarrow{\text{conquer}}$

find efficient PIT for every $f|_{S_i}$

More structure

gives efficient PIT for $\mathcal{C}$ $\xleftarrow{\text{using our new lemma}}$ find efficient PIT for every $f|_{S_i}$

Restrictions and HoR – in Our Case

Take $f \in \sum^2 \mathrm{R2F}$.

Restrictions and HoR – in Our Case

Take $f \in \sum^2 \mathrm{R2F}$.

S_1 - set of structural variables, S_2 - set of non-structural variables.

Restrictions and HoR – in Our Case

Take $f \in \sum^2 \mathrm{R2F}$.

S_1 - set of structural variables, S_2 - set of non-structural variables.

1. $f|_{S_1} \in$ "Structural" $\sum^2 \mathrm{R2F}$

2. $f|_{S_2} \in$ "Totally non-structural" $\sum^2 \mathrm{R2F}$

Restrictions and HoR – in Our Case

Take $f \in \sum^2 \mathrm{R2F}$.

S_1 - set of structural variables, S_2 - set of non-structural variables.

1. $f|_{S_1} \in$ "Structural" $\sum^2 \mathrm{R2F}$ - old techniques work (mostly).

2. $f|_{S_2} \in$ "Totally non-structural" $\sum^2 \mathrm{R2F}$

Restrictions and HoR – in Our Case

Take $f \in \sum^2 \mathrm{R2F}$.

S_1 - set of structural variables, S_2 - set of non-structural variables.

1. $f|_{S_1} \in$ "Structural" $\sum^2 \mathrm{R2F}$ - old techniques work (mostly).

2. $f|_{S_2} \in$ "Totally non-structural" $\sum^2 \mathrm{R2F}$ - strong structural constraints.

Dominating Degree Patterns

Gives more closure rules under PDs.

Dominating Degree Patterns

Gives more closure rules under PDs.

Old closure lemmas:

$$f \in \mathrm{R}k\mathrm{F} \quad \Rightarrow \quad \partial_{x^d}(f) \in \mathrm{R}k\mathrm{F}$$

This is local: **one** variable at a time.

Dominating Degree Patterns

Gives more closure rules under PDs.

Old closure lemmas:

$$f \in \mathrm{R}k\mathrm{F} \quad \Rightarrow \quad \partial_{x^d}(f) \in \mathrm{R}k\mathrm{F}$$

This is local: **one** variable at a time.

New idea:

What about global conditions involving **several** variables?

Choose $J \subseteq [n]$ and differentiate by a maximal exponent vector on J.

Dominating Degree Patterns - Example

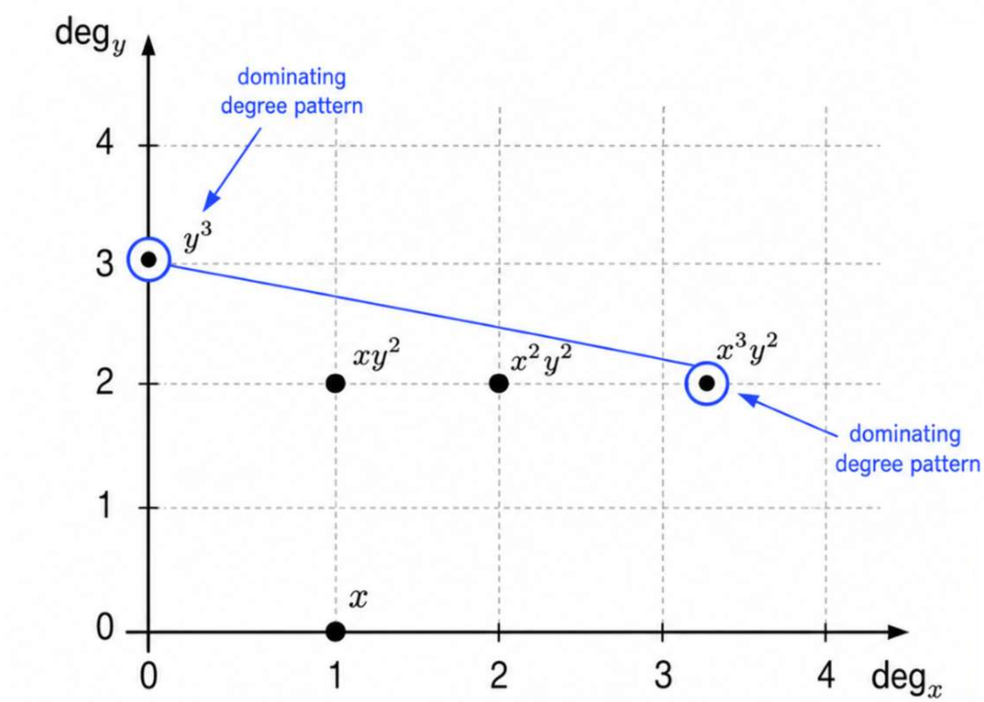
$$f(x, y, z, w) = (5zw^3)xy^2 + (2z^2w)x^3y^2 + (5)x^2y^2 + (zw)x + (7w)y^3$$

Take $J = \{x, y\}$;

Dominating Degree Patterns - Example

$$f(x, y, z, w) = (5zw^3)xy^2 + (2z^2w)x^3y^2 + (5)x^2y^2 + (zw)x + (7w)y^3$$

Take $J = \{x, y\}$;



Dominating Degree Patterns

Informal:

Dominating Degree Patterns

Informal:

Let $f \in \mathrm{R2F}$.

Dominating Degree Patterns

<u>Informal</u>:

Let $f \in \mathrm{R2F}$.

Let $J \subseteq [n]$ of at most 3 variables.

Dominating Degree Patterns

<u>Informal</u>:

Let $f \in \mathrm{R2F}$.

Let $J \subseteq [n]$ of at most 3 variables.

Let $\vec{e} \in \mathbb{N}^J$.

Dominating Degree Patterns

<u>Informal</u>:

Let $f \in \mathrm{R2F}$.

Let $J \subseteq [n]$ of at most 3 variables.

Let $\vec{e} \in \mathbb{N}^J$.

$\vec{e}$ dominating in $f \quad \Rightarrow \quad \partial_{\vec{e}} f \in \mathrm{R2F}$.

Dominating Degree Patterns

Informal:

Let $f \in \mathrm{R2F}$.

Let $J \subseteq [n]$ of at most 3 variables.

Let $\vec{e} \in \mathbb{N}^J$.

Current limitations:

- Variables in J are read-2
- Size of J is at most 3

$$\vec{e} \text{ dominating in } f \quad \Rightarrow \quad \partial_{\vec{e}} f \in \mathrm{R2F}.$$

Open Problems

Open Problems

Can our techniques be used for higher reads? What about R5Fs?

Open Problems

Can our techniques be used for higher reads? What about R5Fs?

More concretely:

1. Can a dominating degree pattern lemma be shown for general RkFs and general set size?

Open Problems

Can our techniques be used for higher reads? What about R5Fs?

More concretely:

1. Can a dominating degree pattern lemma be shown for general RkFs and general set size?
2. Can a better Hardness of Representation result be shown for Resultants of ROFs?

Thanks!