Deterministic Polynomial Identity Tests for Multilinear Bounded-Read Formulae

Matthew Anderson
UW - Madison

Dieter van Melkebeek
UW - Madison

Ilya Volkovich
Technion

January 27^{th}, 2012

Arithmetic Formula Identity Testing

Problem (AFIT)

Arithmetic Formula Identity Testing

Problem (AFIT)

Input: $F \in \mathbb{F}[x_1, ..., x_n]$

Arithmetic Formula Identity Testing

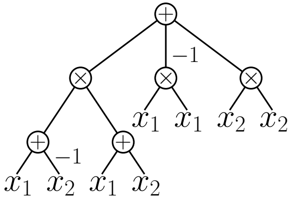
Problem (AFIT)

Input: $F \in \mathbb{F}[x_1, ..., x_n]$, *given as an arithmetic formula.*

Arithmetic Formula Identity Testing

Problem (AFIT)

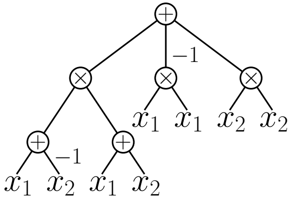
Input: $F \in \mathbb{F}[x_1, ..., x_n]$, *given as an arithmetic formula.*



Arithmetic Formula Identity Testing

Problem (AFIT)

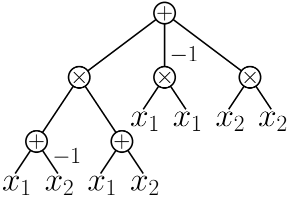
Input: $F \in \mathbb{F}[x_1, ..., x_n]$, *given as an arithmetic formula.*
Question: *Is* $F \equiv 0$*?*



Arithmetic Formula Identity Testing

Problem (AFIT)

Input: $F \in \mathbb{F}[x_1, ..., x_n]$, *given as an arithmetic formula.*
Question: *Is* $F \equiv 0$*?*

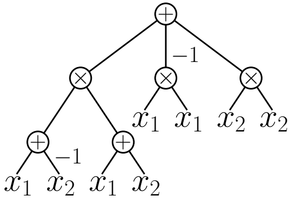


$$(x_1 - x_2)(x_1 + x_2) - x_1^2 + x_2^2 \equiv 0$$

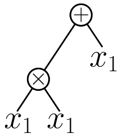
Arithmetic Formula Identity Testing

Problem (AFIT)

Input: $F \in \mathbb{F}[x_1, ..., x_n]$, *given as an arithmetic formula.*
Question: Is $F \equiv 0$*?*



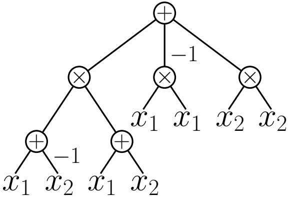
$$(x_1 - x_2)(x_1 + x_2) - x_1^2 + x_2^2 \equiv 0$$



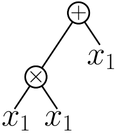
Arithmetic Formula Identity Testing

Problem (AFIT)

Input: $F \in \mathbb{F}[x_1, ..., x_n]$*, given as an arithmetic formula.*
Question: Is $F \equiv 0$*?*



$$(x_1 - x_2)(x_1 + x_2) - x_1^2 + x_2^2 \equiv 0$$



$$x_1^2 + x_1 \not\equiv 0$$

An Algorithm for AFIT

Randomized algorithm:

Pick $a_i \in_U S \subseteq \mathbb{F}$ uniformly, ACCEPT iff $P(a_1, \ldots, a_n) = 0$.

An Algorithm for AFIT

Randomized algorithm:

Pick $a_i \in_U S \subseteq \mathbb{F}$ uniformly, ACCEPT iff $P(a_1, \ldots, a_n) = 0$.

Correctness:

Schwartz-Zippel Lemma

For $d := \deg(P)$,

$$\Pr_{a_i \in_U S}[P(a_1, \ldots, a_n) = 0 \mid P \not\equiv 0] \leq \frac{d}{|S|}.$$

An Algorithm for AFIT

Randomized algorithm:

Pick $a_i \in_U S \subseteq \mathbb{F}$ uniformly, ACCEPT iff $P(a_1, \ldots, a_n) = 0$.

Correctness:

Schwartz-Zippel Lemma

For $d := \deg(P)$,

$$\Pr_{a_i \in_U S}[P(a_1, \ldots, a_n) = 0 \mid P \not\equiv 0] \leq \frac{d}{|S|}.$$

Proof.

An Algorithm for AFIT

Randomized algorithm:

Pick $a_i \in_U S \subseteq \mathbb{F}$ uniformly, ACCEPT iff $P(a_1, \ldots, a_n) = 0$.

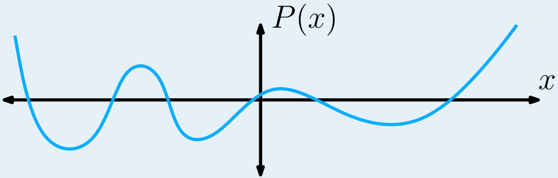
Correctness:

Schwartz-Zippel Lemma

For $d := \deg(P)$,

$$\Pr_{a_i \in_U S}[P(a_1, \ldots, a_n) = 0 \mid P \not\equiv 0] \leq \frac{d}{|S|}.$$

Proof.



∎

Is Randomness Essential?

Open Problem

Is there an efficient **deterministic** identity test?

Is Randomness Essential?

Open Problem

Is there an efficient **deterministic** identity test?

Motivation:

Is Randomness Essential?

Open Problem

Is there an efficient **deterministic** identity test?

Motivation:

- Yes $\Rightarrow$ formula lower bounds.

Is Randomness Essential?

Open Problem

Is there an efficient **deterministic** identity test?

Motivation:

- Yes $\Rightarrow$ formula lower bounds.
- It is a subroutine in other results:
  - primality testing,
  - bipartite perfect matching,
  - PCP theorem,
  - ...

Is Randomness Essential?

Open Problem

Is there an efficient **deterministic** identity test?

Motivation:

- Yes $\Rightarrow$ formula lower bounds.
- It is a subroutine in other results:
  - primality testing,
  - bipartite perfect matching,
  - PCP theorem,
  - ...
- It is a next natural candidate problem to derandomize.

Connections with Lower Bounds

Theorem (Kabanets-Impagliazzo)

If $\mathrm{AFIT} \in \mathrm{NSUBEXP}$, *then either*

Connections with Lower Bounds

Theorem (Kabanets-Impagliazzo)

If $\mathrm{AFIT} \in \mathrm{NSUBEXP}$*, then either*

1. NEXP *does not have poly-size Boolean circuits, or*

Connections with Lower Bounds

Theorem (Kabanets-Impagliazzo)

If $\mathrm{AFIT} \in \mathrm{NSUBEXP}$, *then either*

1. NEXP *does not have poly-size Boolean circuits, or*
2. Perm *does not have poly-size arithmetic formulae.*

Connections with Lower Bounds

Theorem (Kabanets-Impagliazzo)

If $\mathrm{AFIT} \in \mathrm{NSUBEXP}$, *then either*

1. NEXP *does not have poly-size Boolean circuits, or*
2. Perm *does not have poly-size arithmetic formulae.*

Theorem (Agrawal-Vinay)

A polynomial-time identity test for depth-4 formula implies a subexponential-time identity test for arithmetic formula.

Connections with Lower Bounds

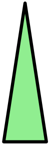
Theorem (Kabanets-Impagliazzo)

If $\mathrm{AFIT} \in \mathrm{NSUBEXP}$, *then either*

1. NEXP *does not have poly-size Boolean circuits, or*
2. Perm *does not have poly-size arithmetic formulae.*

Theorem (Agrawal-Vinay)

A polynomial-time identity test for depth-4 formula implies a subexponential-time identity test for arithmetic formula.



Connections with Lower Bounds

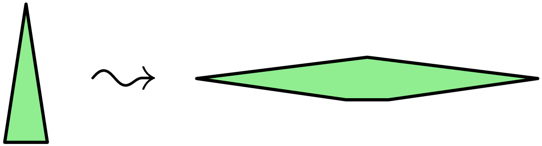
Theorem (Kabanets-Impagliazzo)

If $\mathrm{AFIT} \in \mathrm{NSUBEXP}$, *then either*

1. NEXP *does not have poly-size Boolean circuits, or*
2. Perm *does not have poly-size arithmetic formulae.*

Theorem (Agrawal-Vinay)

A polynomial-time identity test for depth-4 formula implies a subexponential-time identity test for arithmetic formula.



Deterministic Algorithms for AFIT

Deterministic algorithms for bounded-depth formulae:

Deterministic Algorithms for AFIT

Deterministic algorithms for bounded-depth formulae:

- Depth-2 [several]

Deterministic Algorithms for AFIT

Deterministic algorithms for bounded-depth formulae:

- Depth-2 [several]
- Constant-Top-Fanin Depth-3 [DS06,KS07,KS08,KS09,SS11]

Deterministic Algorithms for AFIT

Deterministic algorithms for bounded-depth formulae:

- Depth-2 [several]
- Constant-Top-Fanin Depth-3 [DS06,KS07,KS08,KS09,SS11]
- Multilinear Constant-Top-Fanin Depth-4 [KMSV10,SV11]

Deterministic Algorithms for AFIT

Deterministic algorithms for bounded-depth formulae:

- Depth-2 [several]
- Constant-Top-Fanin Depth-3 [DS06,KS07,KS08,KS09,SS11]
- Multilinear Constant-Top-Fanin Depth-4 [KMSV10,SV11]

Deterministic algorithms for bounded-read formulae:

Deterministic Algorithms for AFIT

Deterministic algorithms for bounded-depth formulae:

- Depth-2 [several]
- Constant-Top-Fanin Depth-3 [DS06,KS07,KS08,KS09,SS11]
- Multilinear Constant-Top-Fanin Depth-4 [KMSV10,SV11]

Deterministic algorithms for bounded-read formulae:

- Read-Once

Deterministic Algorithms for AFIT

Deterministic algorithms for bounded-depth formulae:

- Depth-2 [several]
- Constant-Top-Fanin Depth-3 [DS06,KS07,KS08,KS09,SS11]
- Multilinear Constant-Top-Fanin Depth-4 [KMSV10,SV11]

Deterministic algorithms for bounded-read formulae:

- Read-Once
- $\sum^k$-Read-Once [SV08,SV09]

Deterministic Algorithms for AFIT

Deterministic algorithms for bounded-depth formulae:

- Depth-2 [several]
- Constant-Top-Fanin Depth-3 [DS06,KS07,KS08,KS09,SS11]
- Multilinear Constant-Top-Fanin Depth-4 [KMSV10,SV11]

Deterministic algorithms for bounded-read formulae:

- Read-Once
- $\sum^k$-Read-Once [SV08,SV09]
- Multilinear Read-k [**we**]

Deterministic Algorithms for AFIT

Deterministic algorithms for bounded-depth formulae:

- Depth-2 [several]
- Constant-Top-Fanin Depth-3 [DS06,KS07,KS08,KS09,SS11]
- Multilinear Constant-Top-Fanin Depth-4 [KMSV10,SV11]

Deterministic algorithms for bounded-read formulae:

- Read-Once
- $\sum^k$-Read-Once [SV08,SV09]
- Multilinear Read-k [**we**]

Main Theorem

There is a $s^{O(1)} \cdot n^{k^{O(k)}}$ time deterministic identity test for size-s n-variable multilinear read-k formulae.

Outline

Weakened Main Theorem

There is a $s^{O(1)} \cdot n^{k^{O(k)} + O(k \log n)}$ time deterministic identity test for size-s n-variable multilinear read-k formulae.

Outline

Weakened Main Theorem

There is a $s^{O(1)} \cdot n^{k^{O(k)}+O(k \log n)}$ time deterministic identity test for size-s n-variable multilinear read-k formulae.

Techniques:

Outline

Weakened Main Theorem

There is a $s^{O(1)} \cdot n^{k^{O(k)}+O(k \log n)}$ time deterministic identity test for size-s n-variable multilinear read-k formulae.

Techniques:

1. **Fragmenting**

   Reduces multilinear read-$(k+1)$ to multilinear $\sum^2$-read-k.

Outline

Weakened Main Theorem

There is a $s^{O(1)} \cdot n^{k^{O(k)}+O(k \log n)}$ time deterministic identity test for size-s n-variable multilinear read-k formulae.

Techniques:

1. **Fragmenting**
   Reduces multilinear read-$(k+1)$ to multilinear $\sum^2$-read-k.
2. **Shattering**
   Reduces multilinear $\sum^2$-read-k to multilinear read-k.

Outline

Weakened Main Theorem

There is a $s^{O(1)} \cdot n^{k^{O(k)}+O(k \log n)}$ time deterministic identity test for size-s n-variable multilinear read-k formulae.

Techniques:

1. **Fragmenting**
   Reduces multilinear read-$(k+1)$ to multilinear $\sum^2$-read-k.
2. **Shattering**
   Reduces multilinear $\sum^2$-read-k to multilinear read-k.

Proof.

Combine and iterate the reductions. ∎

Outline

Weakened Main Theorem

There is a $s^{O(1)} \cdot n^{k^{O(k)}+O(k \log n)}$ *time deterministic identity test for size-*s n*-variable multilinear read-*k *formulae.*

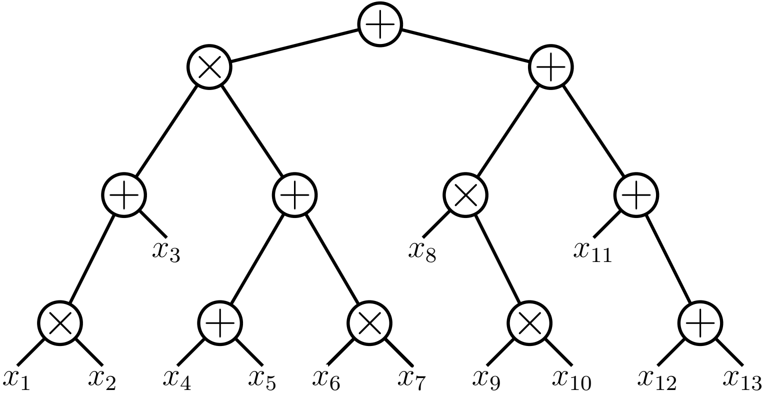
Techniques:

1. **Fragmenting**
   Reduces multilinear read-$(k+1)$ to multilinear $\sum^2$-read-k.
2. **Shattering**
   Reduces multilinear $\sum^2$-read-k to multilinear read-k.

Proof.

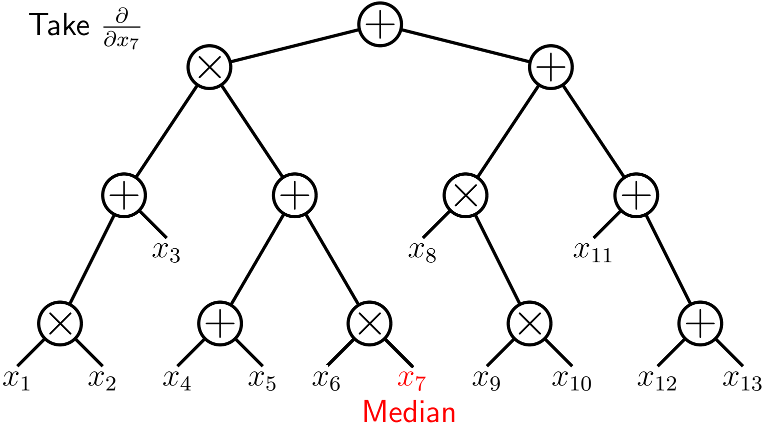
Combine and iterate the reductions. ∎

Fragmenting Read-1 Formulae



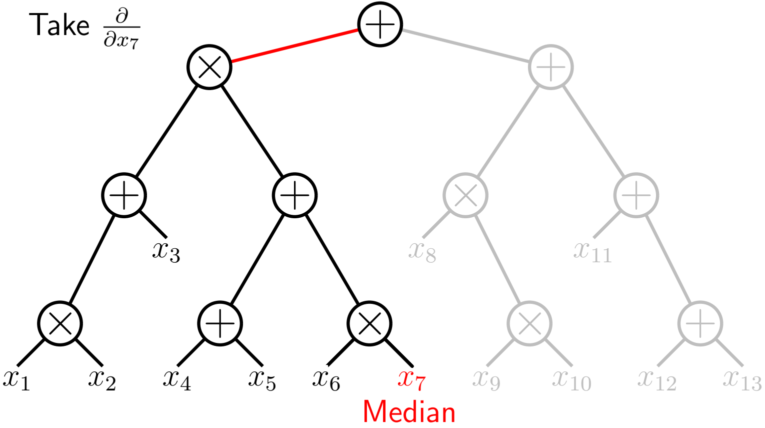
Fragmenting Read-1 Formulae

Take $\frac{\partial}{\partial x_7}$



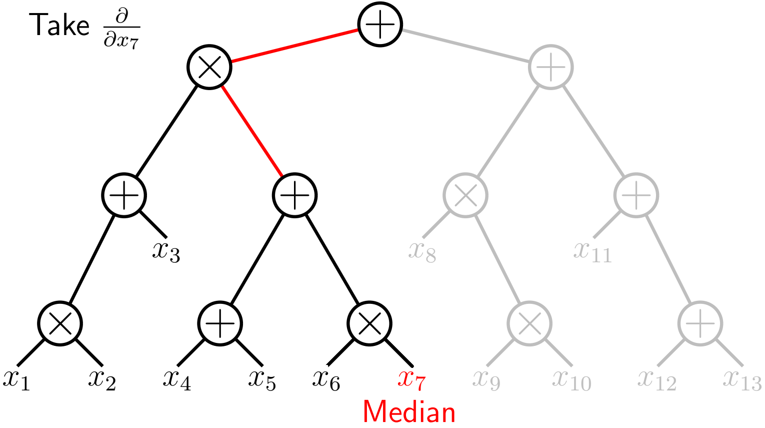
Fragmenting Read-1 Formulae

Take $\frac{\partial}{\partial x_7}$

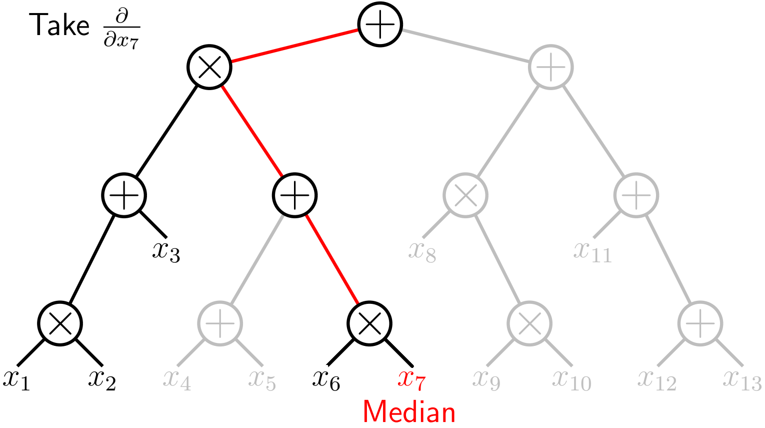


Fragmenting Read-1 Formulae

Take $\frac{\partial}{\partial x_7}$

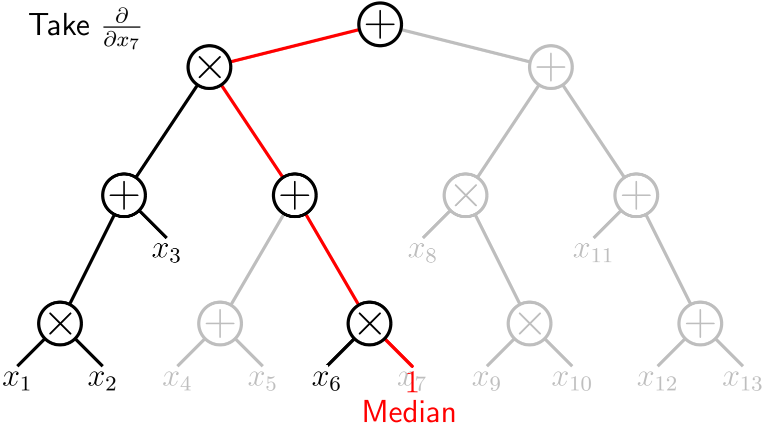


Fragmenting Read-1 Formulae



Fragmenting Read-1 Formulae

Take $\frac{\partial}{\partial x_7}$



A Fragmentation Lemma

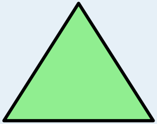
Lemma

Let F be a nonzero read-once formula.

A Fragmentation Lemma

Lemma

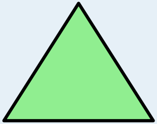
Let F be a nonzero read-once formula.



A Fragmentation Lemma

Lemma

Let F be a nonzero read-once formula.

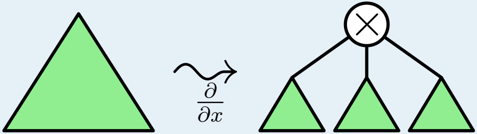


$$\underset{\frac{\partial}{\partial x}}{\rightsquigarrow}$$

A Fragmentation Lemma

Lemma

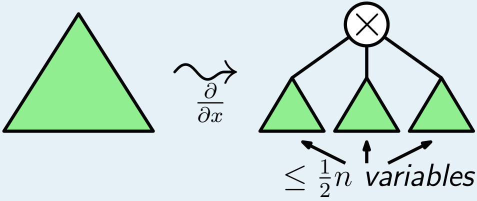
Let F be a nonzero read-once formula.



A Fragmentation Lemma

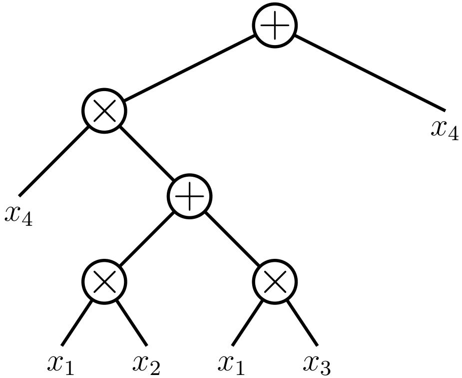
Lemma

Let F be a nonzero read-once formula.



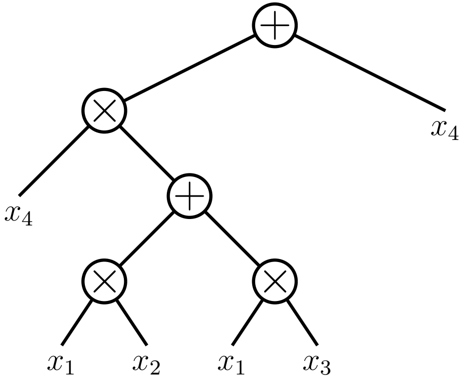
Fragmenting Read-$(k+1)$ Formulae

A read-2 formula:



Fragmenting Read-$(k+1)$ Formulae

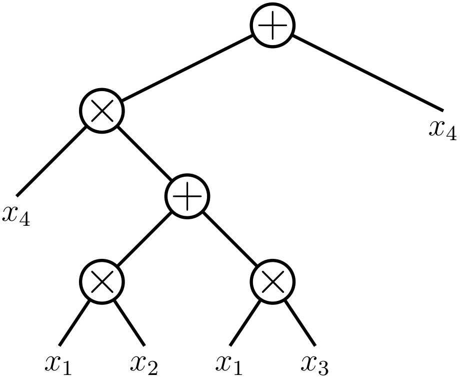
A read-2 formula:



Pick largest child which contains $k+1$ occurrences of some variable.

Fragmenting Read-$(k+1)$ Formulae

A read-2 formula:

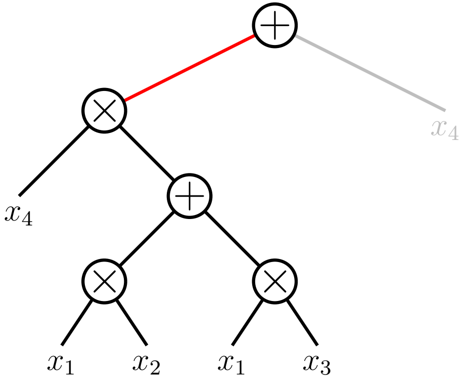


Pick largest child which contains $k+1$ occurrences of some variable.

"largest" $=$ most variables.

Fragmenting Read-$(k+1)$ Formulae

A read-2 formula:

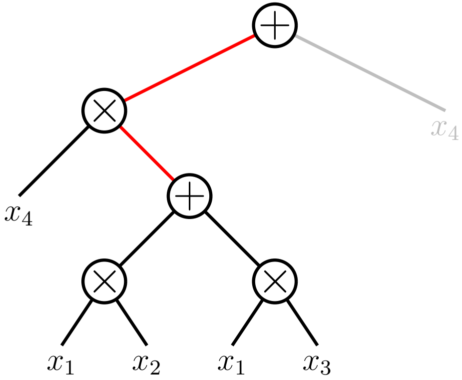


Pick largest child which contains $k+1$ occurrences of some variable.

"largest" $=$ most variables.

Fragmenting Read-$(k+1)$ Formulae

A read-2 formula:

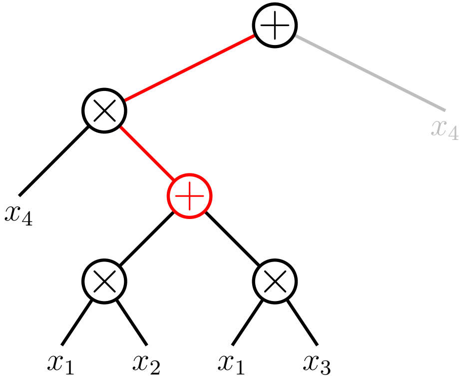


Pick largest child which contains $k+1$ occurrences of some variable.

"largest" $=$ most variables.

Fragmenting Read-$(k+1)$ Formulae

A read-2 formula:

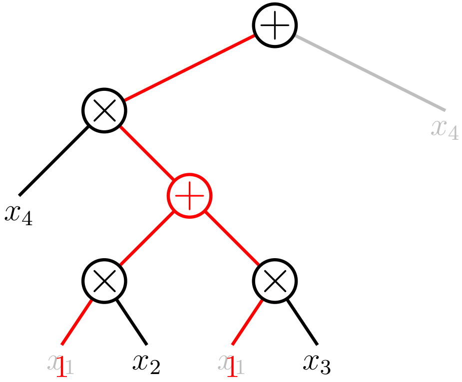


Pick largest child which contains $k+1$ occurrences of some variable.

"largest" $=$ most variables.

Fragmenting Read-$(k+1)$ Formulae

A read-2 formula:



Pick largest child which contains $k+1$ occurrences of some variable.

"largest" $=$ most variables.

The Fragmentation Lemma

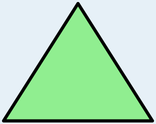
Fragmentation Lemma

Let F be a nonzero read-$(k+1)$ formula.

The Fragmentation Lemma

Fragmentation Lemma

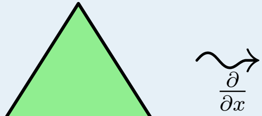
Let F be a nonzero read-$(k+1)$ formula.



The Fragmentation Lemma

Fragmentation Lemma

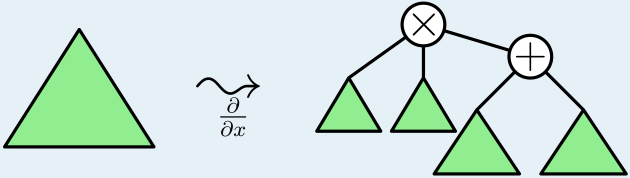
Let F be a nonzero read-$(k+1)$ formula.



The Fragmentation Lemma

Fragmentation Lemma

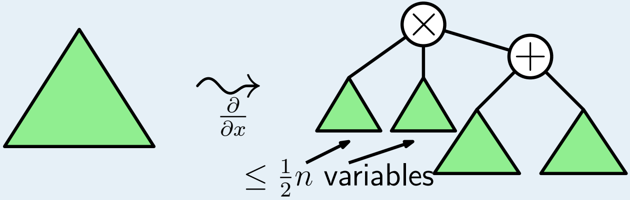
Let F be a nonzero read-$(k+1)$ formula.



The Fragmentation Lemma

Fragmentation Lemma

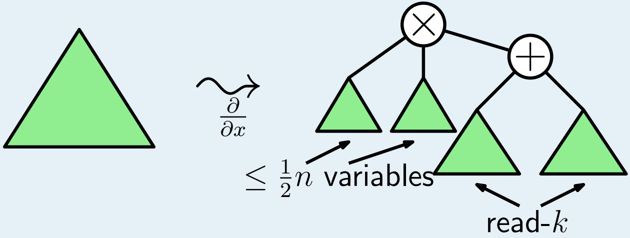
Let F be a nonzero read-$(k+1)$ formula.



The Fragmentation Lemma

Fragmentation Lemma

Let F be a nonzero read-$(k+1)$ formula.



The First Reduction

Read-$(k+1) \leq \sum^2$-Read-k

The First Reduction

Read-$(k+1) \leq \sum^2$-Read-k

While F has variables:

The First Reduction

Read-$(k+1) \leq \sum^2$-Read-k

While F has variables:

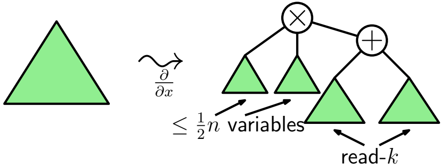
- Select x which fragments F.

The First Reduction

Read-$(k+1) \leq \sum^2$-Read-k

While F has variables:

- Select x which fragments F.

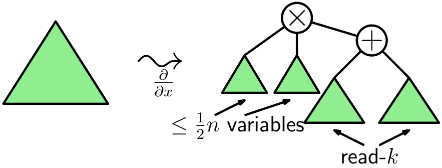


The First Reduction

Read-$(k+1) \leq \sum^2$-Read-k

While F has variables:

- Select x which fragments F.
- Test the factors of $\partial_x F$ recursively.

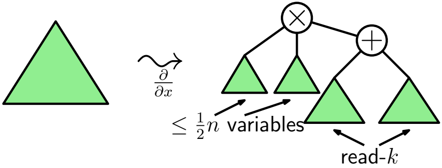


The First Reduction

Read-$(k+1) \leq \sum^2$-Read-k

While F has variables:

- Select x which fragments F.
- Test the factors of $\partial_x F$ recursively.
- If all factors of $\partial_x F$ are nonzero, REJECT.

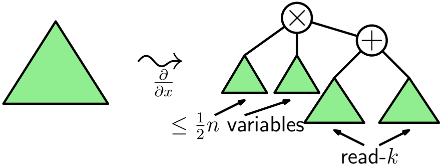


The First Reduction

Read-$(k+1) \leq \sum^2$-Read-k

While F has variables:

- Select x which fragments F.
- Test the factors of $\partial_x F$ recursively.
- If all factors of $\partial_x F$ are nonzero, REJECT.
- Set $F = F|_{x \leftarrow 0}$.



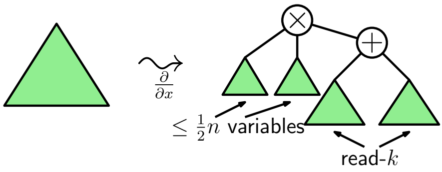
The First Reduction

Read-$(k+1) \leq \sum^2$-Read-k

While F has variables:

- Select x which fragments F.
- Test the factors of $\partial_x F$ recursively.
- If all factors of $\partial_x F$ are nonzero, REJECT.
- Set $F = F|_{x \leftarrow 0}$.

ACCEPT iff $F = 0$.



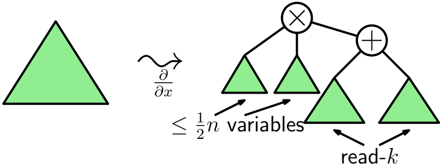
The First Reduction

Read-$(k+1) \leq \sum^2$-Read-k

While F has variables:

- Select x which fragments F.
- Test the factors of $\partial_x F$ recursively.
- If all factors of $\partial_x F$ are nonzero, REJECT.
- Set $F = F|_{x \leftarrow 0}$.

ACCEPT iff $F = 0$.



Makes $n^{O(\log n)}$ calls to the $\sum^2$-read-k identity test.

Outline

Weakened Main Theorem

There is a $s^{O(1)} \cdot n^{k^{O(k)}+O(k \log n)}$ time deterministic identity test for size-s n-variable multilinear read-k formulae.

Techniques:

1. **Fragmenting**

   Reduces multilinear read-$(k+1)$ to multilinear $\sum^2$-read-k.

2. **Shattering**

   Reduces multilinear $\sum^2$-read-k to multilinear read-k.

Outline

Weakened Main Theorem

There is a $s^{O(1)} \cdot n^{k^{O(k)}+O(k\log n)}$ *time deterministic identity test for size-*s n*-variable multilinear read-*k *formulae.*

Techniques:

1. **Fragmenting**
   Reduces multilinear read-$(k+1)$ to multilinear $\sum^2$-read-k.
2. **Shattering**
   Reduces multilinear $\sum^2$-read-k to multilinear read-k.

Outline

Weakened Main Theorem

There is a $s^{O(1)} \cdot n^{k^{O(k)} + O(k \log n)}$ time deterministic identity test for size-s n-variable multilinear read-k formulae.

Techniques:

1. **Fragmenting**

   Reduces multilinear read-$(k+1)$ to multilinear $\sum^2$-read-k.

2. **Shattering**

   Reduces multilinear $\sum^2$-read-k to multilinear read-k.

Outline

Weakened Main Theorem

There is a $s^{O(1)} \cdot n^{k^{O(k)}+O(k \log n)}$ time deterministic identity test for size-s n-variable multilinear read-k formulae.

Techniques:

1. **Fragmenting**

   Reduces multilinear read-$(k+1)$ to multilinear $\sum^2$-read-k.

2. **Shattering**

   Reduces multilinear $\sum^2$-read-k to multilinear read-k.

Testing $\sum^2$-read-k $\leq$ Testing read-k

Fact (SV Hitting Set, implicit in [SV09])

The set of binary strings H_w with Hamming weight at most w hits any class $\mathcal{F}$ of multilinear polynomials that:

1. *is closed under zero-substitutions, and*
2. *does not contain any monomial of degree $d \geq w$.*

Testing $\sum^2$-read-k $\leq$ Testing read-k

Fact (SV Hitting Set, implicit in [SV09])

The set of binary strings H_w with Hamming weight at most w hits any class $\mathcal{F}$ of multilinear polynomials that:

1. *is closed under zero-substitutions, and*
2. *does not contain any monomial of degree $d \geq w$.*

Let $F = F_1 + F_2$ be a nonzero multilinear $\sum^2$-read-k formula.

Testing $\sum^2$-read-k $\leq$ Testing read-k

Fact (SV Hitting Set, implicit in [SV09])

The set of binary strings H_w with Hamming weight at most w hits any class $\mathcal{F}$ of multilinear polynomials that:

1. *is closed under zero-substitutions, and*
2. *does not contain any monomial of degree $d \geq w$.*

Let $F = F_1 + F_2$ be a nonzero multilinear $\sum^2$-read-k formula.

- Let $\mathcal{F}$ consist of $F(\bar{x} + \bar{\sigma})$ and all its zero-substitutions.

Testing $\sum^2$-read-k $\leq$ Testing read-k

Fact (SV Hitting Set, implicit in [SV09])

The set of binary strings H_w with Hamming weight at most w hits any class $\mathcal{F}$ of multilinear polynomials that:

1. *is closed under zero-substitutions, and*
2. *does not contain any monomial of degree $d \geq w$.*

Let $F = F_1 + F_2$ be a nonzero multilinear $\sum^2$-read-k formula.

- Let $\mathcal{F}$ consist of $F(\bar{x} + \bar{\sigma})$ and all its zero-substitutions.
- Some simple conditions on $\bar{\sigma}$ give property 2 for $\mathcal{F}$.

Testing $\sum^2$-read-k $\leq$ Testing read-k

Fact (SV Hitting Set, implicit in [SV09])

The set of binary strings H_w with Hamming weight at most w hits any class $\mathcal{F}$ of multilinear polynomials that:

1. *is closed under zero-substitutions, and*
2. *does not contain any monomial of degree $d \geq w$.*

Let $F = F_1 + F_2$ be a nonzero multilinear $\sum^2$-read-k formula.

- Let $\mathcal{F}$ consist of $F(\bar{x} + \bar{\sigma})$ and all its zero-substitutions.
- Some simple conditions on $\bar{\sigma}$ give property 2 for $\mathcal{F}$.
- For such a $\bar{\sigma}$, $H_w + \bar{\sigma}$ hits F.

A Structural Witness Lemma

Witness Lemma

Let $F = \sum_{i=1}^{m} F_i$ *be a multilinear formula on* n*-variables,*

A Structural Witness Lemma

Witness Lemma

Let $F = \sum_{i=1}^{m} F_i$ *be a multilinear formula on* n*-variables, where*

1. *no variable divides any* F_i*,*

A Structural Witness Lemma

Witness Lemma

Let $F = \sum_{i=1}^{m} F_i$ *be a multilinear formula on* n*-variables, where*

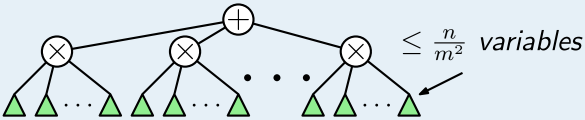
1. *no variable divides any* F_i*,*
2. *the factors of each* F_i *depend on at most* $\frac{n}{m^2}$ *variables:*

A Structural Witness Lemma

Witness Lemma

Let $F = \sum_{i=1}^{m} F_i$ *be a multilinear formula on* n*-variables, where*

1. *no variable divides any* F_i*,*
2. *the factors of each* F_i *depend on at most* $\frac{n}{m^2}$ *variables:*

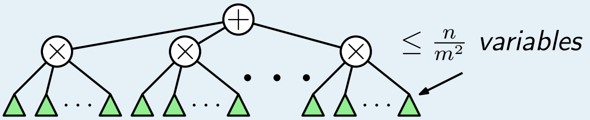


A Structural Witness Lemma

Witness Lemma

Let $F = \sum_{i=1}^{m} F_i$ be a multilinear formula on n-variables, where

1. *no variable divides any F_i,*
2. *the factors of each F_i depend on at most $\frac{n}{m^2}$ variables:*



then F does not compute a monomial of degree n.

The Shattering Lemma

Shattering Lemma

For any nonzero multilinear $\sum^2$*-read-*k *formula* F *on* n *variables, there exist sets of variables*

- P*, with* $|P| = \mathrm{poly}(k)$*, and*
- V*, with* $|V| = \frac{n}{k^{O(k)}}$

The Shattering Lemma

Shattering Lemma

For any nonzero multilinear $\sum^2$*-read-*k *formula* F *on* n *variables, there exist sets of variables*

- P*, with* $|P| = \mathrm{poly}(k)$*, and*
- V*, with* $|V| = \frac{n}{k^{O(k)}}$

such that $\frac{\partial F}{\partial P}$ *depends on at least the variables in* V*,*

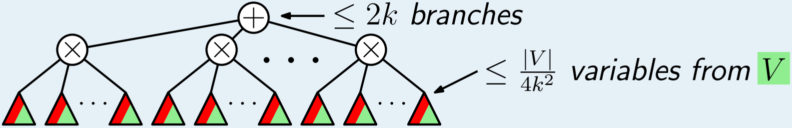
The Shattering Lemma

Shattering Lemma

For any nonzero multilinear $\sum^2$*-read-*k *formula* F *on* n *variables, there exist sets of variables*

- P, *with* $|P| = \mathrm{poly}(k)$, *and*
- V, *with* $|V| = \frac{n}{k^{O(k)}}$

such that $\frac{\partial F}{\partial P}$ *depends on at least the variables in* V, *and can be written as*



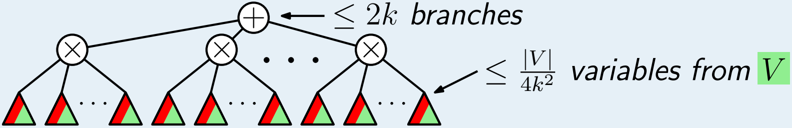
The Shattering Lemma

Shattering Lemma

For any nonzero multilinear $\sum^2$*-read-*k *formula* F *on* n *variables, there exist sets of variables*

- P, *with* $|P| = \mathrm{poly}(k)$, *and*
- V, *with* $|V| = \frac{n}{k^{O(k)}}$

such that $\frac{\partial F}{\partial P}$ *depends on at least the variables in* V, *and can be written as*



where each small subformula is the partial derivative of some subformula of F.

Theorem (Correctness)

$F(\bar{x} + \bar{\sigma})$ *is not a monomial.*

Theorem (Correctness)

$F(\bar{x} + \bar{\sigma})$ *is not a monomial.*

Proof.

Theorem (Correctness)

$F(\bar{x} + \bar{\sigma})$ *is not a monomial.*

Proof.

Suppose $F(\bar{x} + \bar{\sigma})$ is a monomial M_n of degree n.

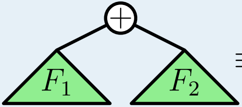
Theorem (Correctness)

$F(\bar{x} + \bar{\sigma})$ is not a monomial.

Proof.

Suppose $F(\bar{x} + \bar{\sigma})$ is a monomial M_n of degree n.

$\Rightarrow$



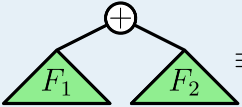
$\equiv M_n$

Theorem (Correctness)

$F(\bar{x} + \bar{\sigma})$ *is not a monomial.*

Proof.

Suppose $F(\bar{x} + \bar{\sigma})$ is a monomial M_n of degree n.

$\Rightarrow$ **Shatter(** 

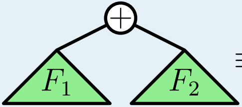
$\equiv M_n$)

Theorem (Correctness)

$F(\bar{x} + \bar{\sigma})$ *is not a monomial.*

Proof.

Suppose $F(\bar{x} + \bar{\sigma})$ is a monomial M_n of degree n.

$\Rightarrow$ **Shatter(** 

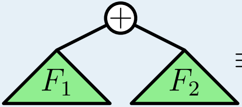
$\equiv M_n$), $\exists P, V$

Theorem (Correctness)

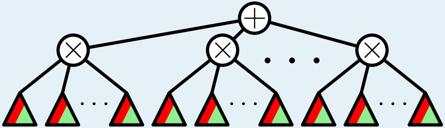
$F(\bar{x}+\bar{\sigma})$ *is not a monomial.*

Proof.

Suppose $F(\bar{x}+\bar{\sigma})$ is a monomial M_n of degree n.

$\Rightarrow$ **Shatter(** 

$\equiv M_n$), $\exists P, V$

$\overset{\partial_P}{\Rightarrow}$  $\equiv M_{n-|P|}$

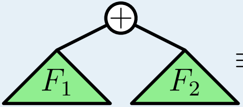
Theorem (Correctness)

$F(\bar{x} + \bar{\sigma})$ *is not a monomial.*

Proof.

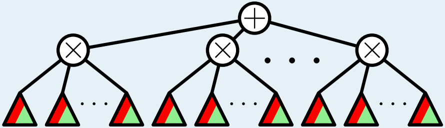
Suppose $F(\bar{x} + \bar{\sigma})$ is a monomial M_n of degree n.

$\Rightarrow$ **Shatter(**



$\equiv M_n$), $\exists P, V$

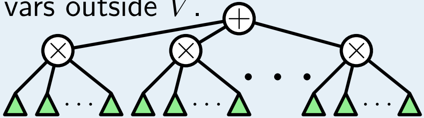
∂_P
$\Rightarrow$



$\equiv M_{n-|P|}$

Set vars outside V.

$\Rightarrow$



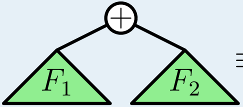
$\equiv M_{|V|}$

Theorem (Correctness)

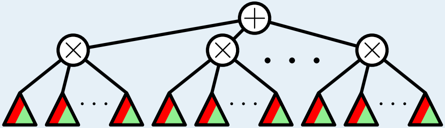
$F(\bar{x}+\bar{\sigma})$ *is not a monomial.*

Proof.

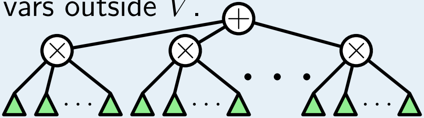
Suppose $F(\bar{x}+\bar{\sigma})$ is a monomial M_n of degree n.

$\Rightarrow$ **Shatter**(

$\equiv M_n$), $\exists P, V$

$\overset{\partial_P}{\Rightarrow}$  $\equiv M_{n-|P|}$

Set vars outside V.

$\Rightarrow$  $\equiv M_{|V|}, |V| \geq \frac{n}{k^{O(k)}} \geq 1$

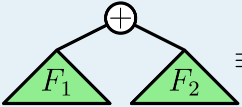
Theorem (Correctness)

$F(\bar{x} + \bar{\sigma})$ is not a monomial of degree $n \geq k^{O(k)}$.

Proof.

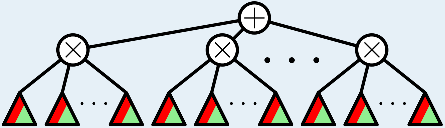
Suppose $F(\bar{x} + \bar{\sigma})$ is a monomial M_n of degree n.

$\Rightarrow$ **Shatter(**



$\equiv M_n$), $\exists P, V$

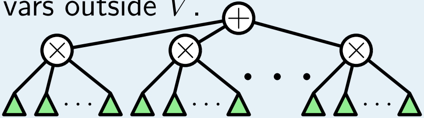
$\overset{\partial_P}{\Rightarrow}$



$\equiv M_{n-|P|}$

Set vars outside V.

$\Rightarrow$



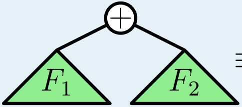
$\equiv M_{|V|}, |V| \geq \frac{n}{k^{O(k)}} \geq 1$

Theorem (Correctness)

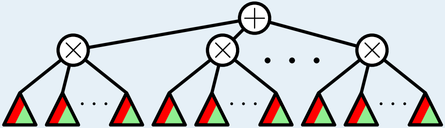
$F(\bar{x}+\bar{\sigma})$ is not a monomial of degree $n \geq k^{O(k)}$.

Proof.

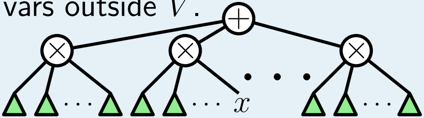
Suppose $F(\bar{x}+\bar{\sigma})$ is a monomial M_n of degree n.

$\Rightarrow$ **Shatter(** 

$\equiv M_n$), $\exists P, V$

∂_P
$\Rightarrow$ 
$\equiv M_{n-|P|}$

Set vars outside V.

$\Rightarrow$ 

$\equiv M_{|V|}, |V| \geq \frac{n}{k^{O(k)}} \geq 1$

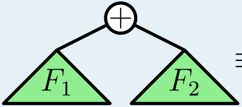
Theorem (Correctness)

$F(\bar{x}+\bar{\sigma})$ is not a monomial of degree $n \geq k^{O(k)}$.

Proof.

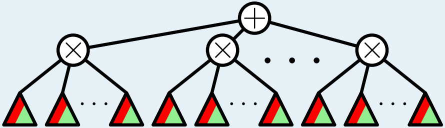
Suppose $F(\bar{x}+\bar{\sigma})$ is a monomial M_n of degree n.

$\Rightarrow$ **Shatter**(



$\equiv M_n$), $\exists P, V$

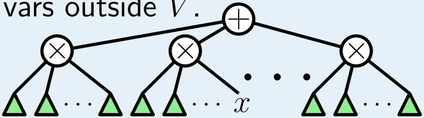
∂_P
$\Rightarrow$



$\equiv M_{n-|P|}$

Set vars outside V.

$\Rightarrow$



$\equiv M_{|V|}, |V| \geq \frac{n}{k^{O(k)}} \geq 1$

$\Rightarrow$ $\Delta|_{x\leftarrow 0} = 0$

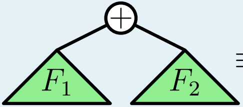
Theorem (Correctness)

$F(\bar{x} + \bar{\sigma})$ is not a monomial of degree $n \geq k^{O(k)}$.

Proof.

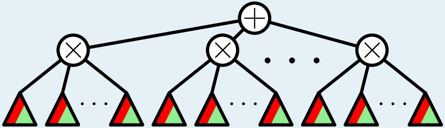
Suppose $F(\bar{x} + \bar{\sigma})$ is a monomial M_n of degree n.

$\Rightarrow$ **Shatter(**



$\equiv M_n$), $\exists P, V$

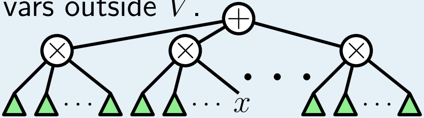
∂_P
$\Rightarrow$



$\equiv M_{n-|P|}$

Set vars outside V.

$\Rightarrow$



$\equiv M_{|V|}, |V| \geq \frac{n}{k^{O(k)}} \geq 1$

Unshifted:
$\Rightarrow \triangle\big|_{x \leftarrow \sigma_x} = 0$

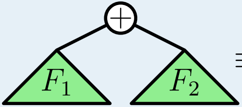
Theorem (Correctness)

$F(\bar{x}+\bar{\sigma})$ *is not a monomial of degree* $n \geq k^{O(k)}$.

Proof.

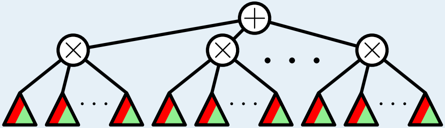
Suppose $F(\bar{x}+\bar{\sigma})$ is a monomial M_n of degree n.

$\Rightarrow$ **Shatter**(



$\equiv M_n$), $\exists P, V$

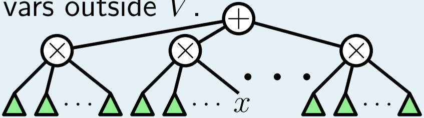
∂_P
$\Rightarrow$



$\equiv M_{n-|P|}$

Set vars outside V.

$\Rightarrow$



$\equiv M_{|V|}, |V| \geq \frac{n}{k^{O(k)}} \geq 1$

Unshifted:
$\Rightarrow \Delta|_{x \leftarrow \sigma_x} = 0$

Pick $\bar{\sigma}$ to be a common nonzero of ∂'s to order $|P|$ of the subformulae of F.

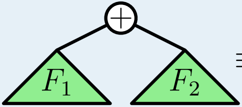
Theorem (Correctness)

$F(\bar{x}+\bar{\sigma})$ is not a monomial of degree $n \geq k^{O(k)}$.

Proof.

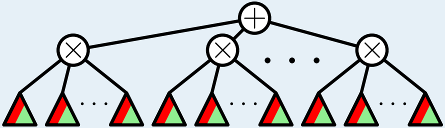
Suppose $F(\bar{x}+\bar{\sigma})$ is a monomial M_n of degree n.

$\Rightarrow$ **Shatter(**



$\equiv M_n$), $\exists P, V$

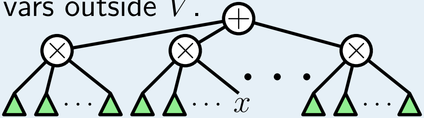
$\overset{\partial_P}{\Rightarrow}$



$\equiv M_{n-|P|}$

Set vars outside V.

$\Rightarrow$



$\equiv M_{|V|}, |V| \geq \frac{n}{k^{O(k)}} \geq 1$

Unshifted:

$\Rightarrow \Delta|_{x \leftarrow \sigma_x} = 0$

Pick $\bar{\sigma}$ to be a common nonzero of ∂'s to order $|P|$ of the subformulae of F.

! ∎

Theorem (Efficiency)

$\bar{\sigma}$ is easy to compute.

Proof.

Pick $\bar{\sigma}$ to be a common nonzero of partial derivatives of order up to $|P|$ of all subformulae of F.

Theorem (Efficiency)

$\bar{\sigma}$ is easy to compute.

Proof.

Pick $\bar{\sigma}$ to be a common nonzero of partial derivatives of order up to $|P|$ of all subformulae of F.

- Each such formula is read-k, since F is $\sum^2$-read-k.

Theorem (Efficiency)

$\bar{\sigma}$ is easy to compute.

Proof.

Pick $\bar{\sigma}$ to be a common nonzero of partial derivatives of order up to $|P|$ of all subformulae of F.

- Each such formula is read-k, since F is $\sum^2$-read-k.
- There are $O(kn^{|P|+1})$ such formulae.

Theorem (Efficiency)

$\bar{\sigma}$ is easy to compute.

Proof.

Pick $\bar{\sigma}$ to be a common nonzero of partial derivatives of order up to $|P|$ of all subformulae of F.

- Each such formula is read-k, since F is $\sum^2$-read-k.
- There are $O(kn^{|P|+1})$ such formulae.

Determine $\bar{\sigma}$ using trial substitution and read-k identity test. ∎

Theorem (Efficiency)

$\bar{\sigma}$ is easy to compute.

Proof.

Pick $\bar{\sigma}$ to be a common nonzero of partial derivatives of order up to $|P|$ of all subformulae of F.

- Each such formula is read-k, since F is $\sum^2$-read-k.
- There are $O(kn^{|P|+1})$ such formulae.

Determine $\bar{\sigma}$ using trial substitution and read-k identity test. ∎

Overall reduction:

Theorem (Efficiency)

$\bar{\sigma}$ *is easy to compute.*

Proof.

Pick $\bar{\sigma}$ to be a common nonzero of partial derivatives of order up to $|P|$ of all subformulae of F.

- Each such formula is read-k, since F is $\sum^2$-read-k.
- There are $O(kn^{|P|+1})$ such formulae.

Determine $\bar{\sigma}$ using trial substitution and read-k identity test. ∎

Overall reduction:

- Makes $n^{\mathrm{poly}(k)}$ calls to the read-k identity test.

Theorem (Efficiency)

$\bar{\sigma}$ is easy to compute.

Proof.

Pick $\bar{\sigma}$ to be a common nonzero of partial derivatives of order up to $|P|$ of all subformulae of F.

- Each such formula is read-k, since F is $\sum^2$-read-k.
- There are $O(kn^{|P|+1})$ such formulae.

Determine $\bar{\sigma}$ using trial substitution and read-k identity test. ∎

Overall reduction:

- Makes $n^{\mathrm{poly}(k)}$ calls to the read-k identity test.
- Does $n^{k^{O(k)}}$ work evaluating the formula on $H_w + \bar{\sigma}$.

Main Theorem

1. **Fragmenting**

   Reduces multilinear read-$(k+1)$ to multilinear $\sum^2$-read-k.

$$T(k+1) = n^{\log n} T_2(k)$$

2. **Shattering**

   Reduces multilinear $\sum^2$-read-k to multilinear read-k.

$$T_2(k) = n^{\mathrm{poly}(k)} T(k) + n^{k^{O(k)}}$$

Main Theorem

1. **Fragmenting**

   Reduces multilinear read-$(k+1)$ to multilinear $\sum^2$-read-k.

   $$T(k+1) = n^{\log n} T_2(k)$$

2. **Shattering**

   Reduces multilinear $\sum^2$-read-k to multilinear read-k.

   $$T_2(k) = n^{\mathrm{poly}(k)} T(k) + n^{k^{O(k)}}$$

Weakened Main Theorem

There is a $s^{O(1)} \cdot n^{k^{O(k)} + O(k \log n)}$ time deterministic identity test for n-variable size-s multilinear read-k formulae.

Main Theorem

1. **Fragmenting**

   Reduces multilinear read-$(k+1)$ to multilinear $\sum^2$-read-k.

   $$T(k+1) = n^{\log n} T_2(k)$$

2. **Shattering**

   Reduces multilinear $\sum^2$-read-k to multilinear read-k.

   $$T_2(k) = n^{\mathrm{poly}(k)} T(k) + n^{k^{O(k)}}$$

Weakened Main Theorem

There is a $s^{O(1)} \cdot n^{k^{O(k)} + O(k \log n)}$ time deterministic identity test for n-variable size-s multilinear read-k formulae.

Main Theorem

1. **Fragmenting**

   Reduces multilinear read-$(k+1)$ to multilinear $\sum^2$-read-k.

$$T(k+1) = n^{\log n} T_2(k)$$

2. **Shattering**

   Reduces multilinear $\sum^2$-read-k to multilinear read-k.

$$T_2(k) = n^{\mathrm{poly}(k)} T(k) + n^{k^{O(k)}}$$

Main Theorem

There is a $s^{O(1)} \cdot n^{k^{O(k)}}$ time deterministic identity test for n-variable size-s multilinear read-k formulae.

Main Theorem

1. **Fragmenting**

   Reduces multilinear read-$(k+1)$ to multilinear $\sum^2$-read-k.

$$T(k+1) = n^{\log n} T_2(k)$$

2. **Shattering**

   Reduces multilinear $\sum^2$-read-k to multilinear read-k.

$$T_2(k) = n^{\mathrm{poly}(k)} T(k) + n^{k^{O(k)}}$$

Main Theorem

There is a $s^{O(1)} \cdot n^{k^{O(k)}}$ time deterministic identity test for n-variable size-s multilinear read-k formulae.

Corollary

There is a polynomial-time deterministic identity test for multilinear constant-read formulae.

Main Theorem

1. **Fragmenting**

   Reduces multilinear read-$(k+1)$ to multilinear $\sum^2$-read-k.

$$T(k+1) = n^{\log n} T_2(k)$$

2. **Shattering**

   Reduces multilinear $\sum^2$-read-k to multilinear read-k.

$$T_2(k) = n^{\mathrm{poly}(k)} T(k) + n^{k^{O(k)}}$$

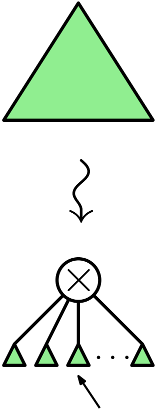
Main Theorem

There is a $s^{O(1)} \cdot n^{k^{O(k)}}$ time deterministic identity test for n-variable size-s multilinear read-k formulae.

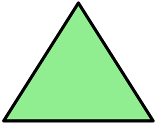
Corollary

There is a polynomial-time deterministic identity test for multilinear constant-read formulae.

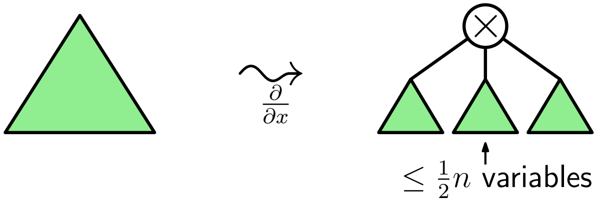
Shattering Read-once Formulae



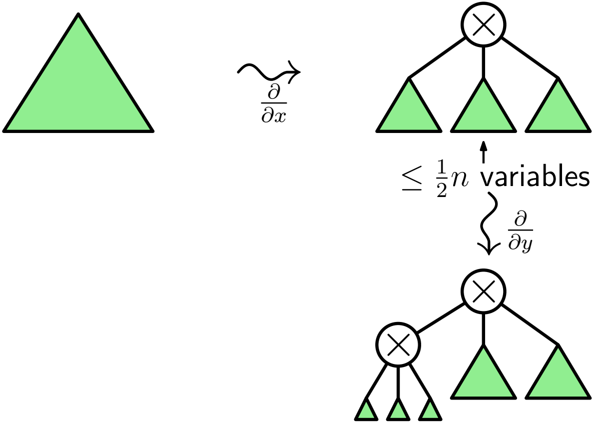
$\leq \alpha n$ variables



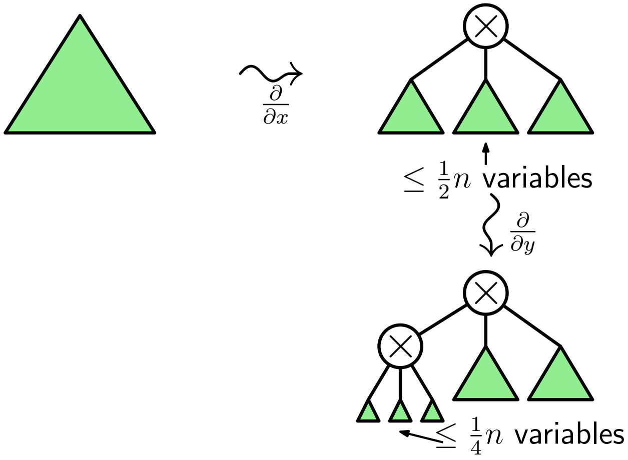
Shattering Read-once Formulae

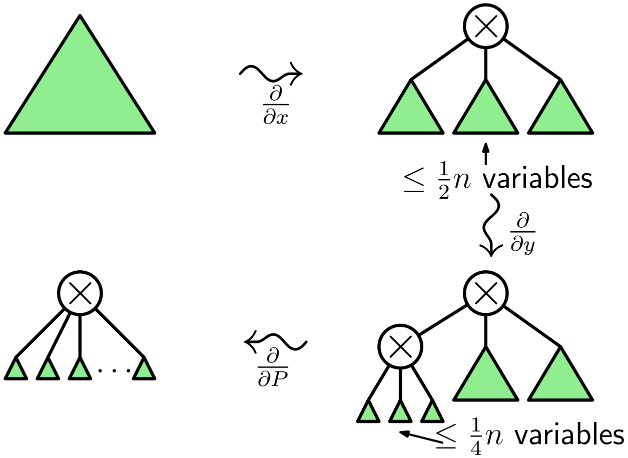


Shattering Read-once Formulae

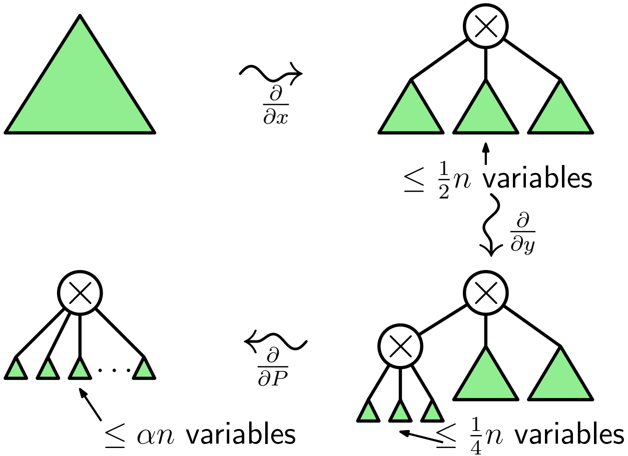


Shattering Read-once Formulae




$\frac{\partial}{\partial x}$
$\leq \frac{1}{2}n$ variables
$\frac{\partial}{\partial y}$
$\frac{\partial}{\partial P}$
$\leq \frac{1}{4}n$ variables

Shattering Read-once Formulae



A Shattering Lemma

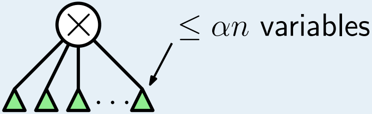
Lemma

For any read-once formula F on n variables and $\alpha \in [0,1]$ there exists a sets of variables P, with $|P| = O(\frac{1}{\alpha})$, such that $\frac{\partial F}{\partial P}$ can be written as

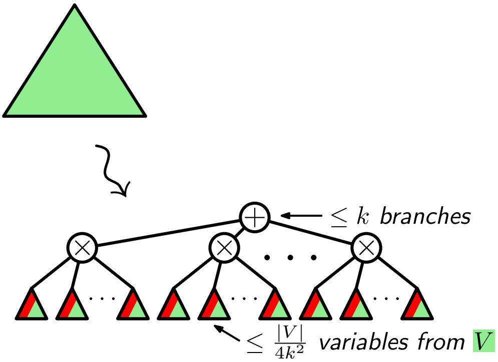
A Shattering Lemma

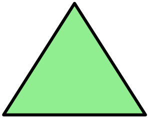
Lemma

For any read-once formula F on n variables and $\alpha \in [0,1]$ there exists a sets of variables P, with $|P| = O(\frac{1}{\alpha})$, such that $\frac{\partial F}{\partial P}$ can be written as

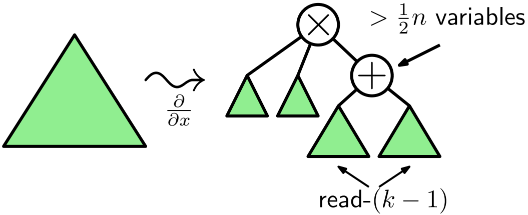


Shattering Read-k Formulae

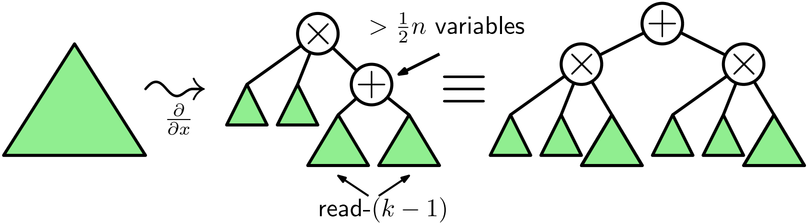




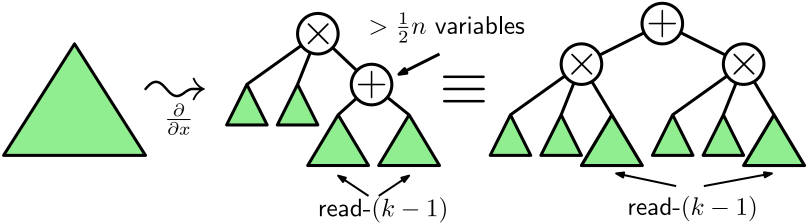
Shattering Read-k Formulae



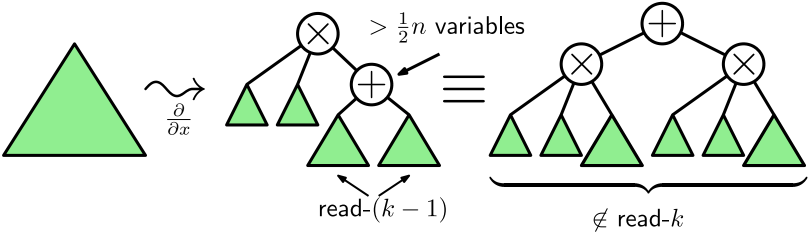
Shattering Read-k Formulae



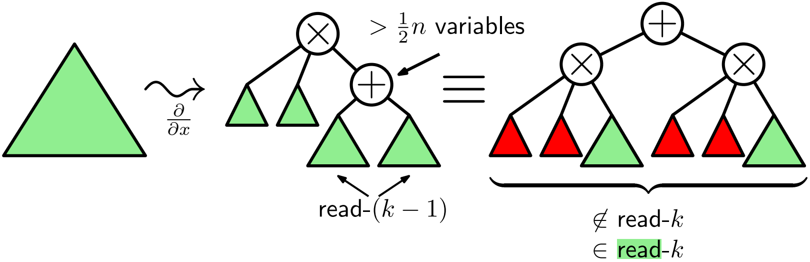
Shattering Read-k Formulae



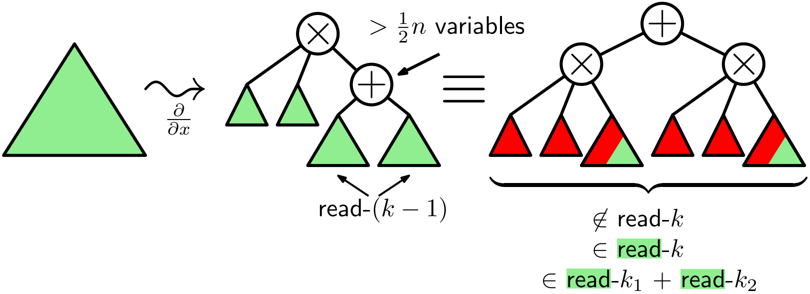
Shattering Read-k Formulae



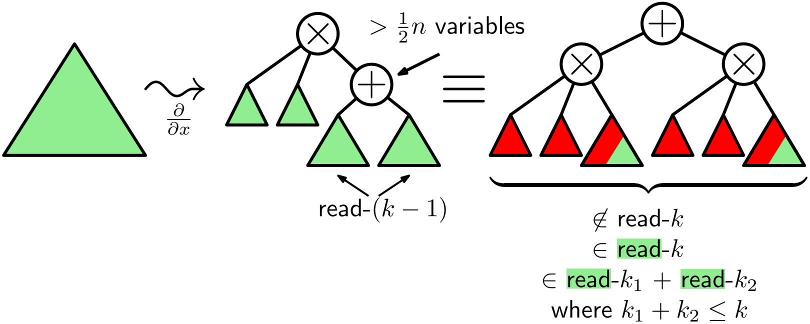
Shattering Read-k Formulae



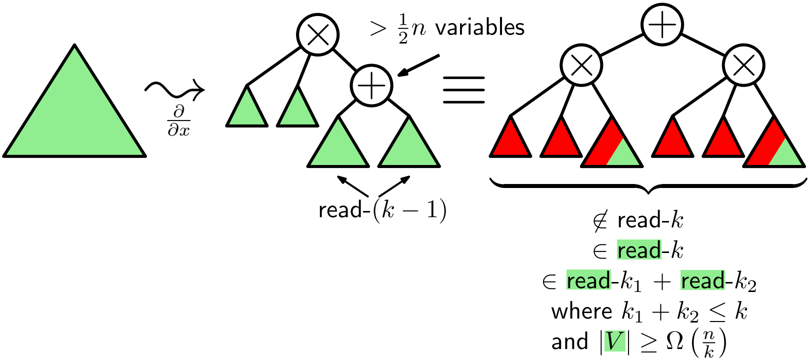
Shattering Read-k Formulae



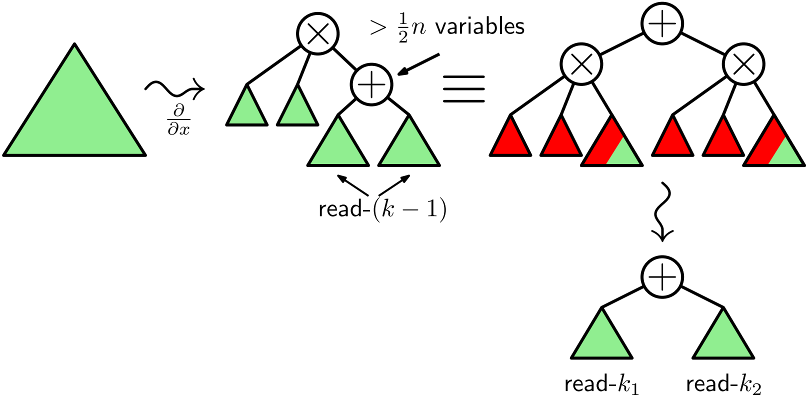
Shattering Read-k Formulae



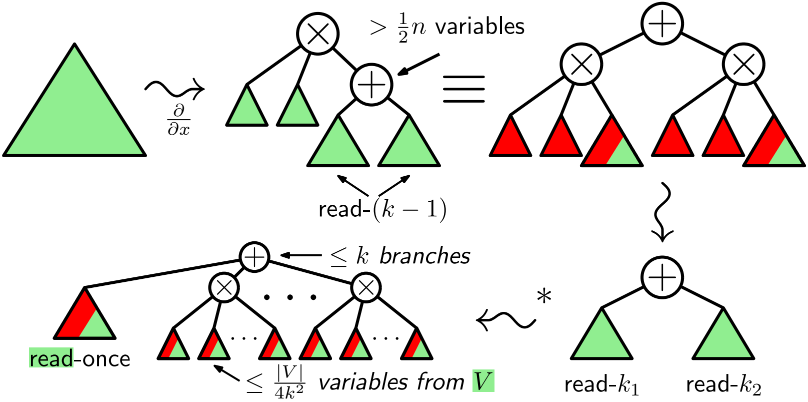
Shattering Read-k Formulae



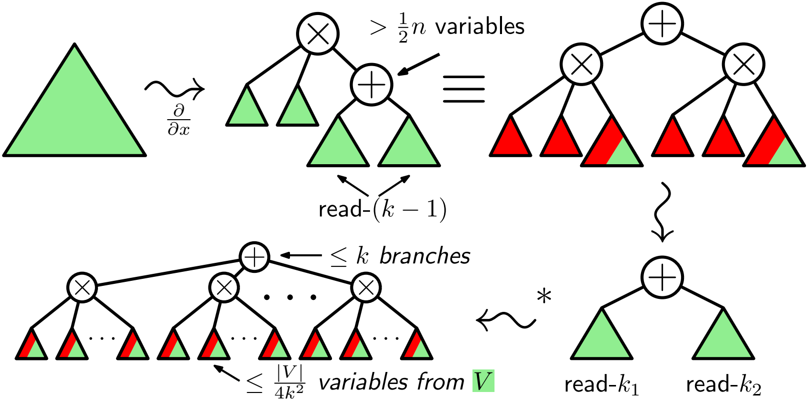
Shattering Read-k Formulae



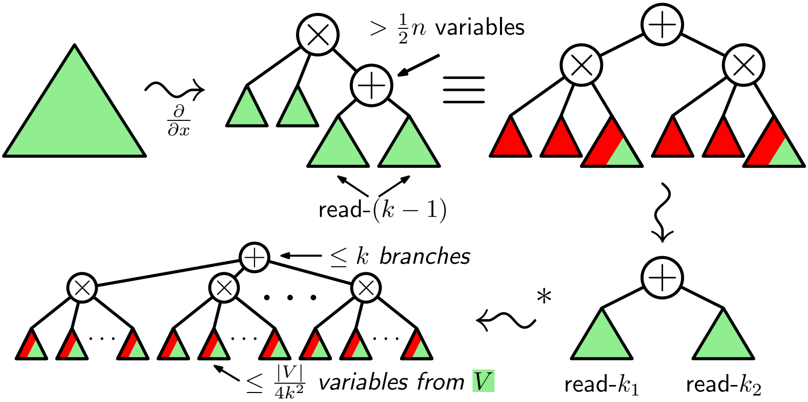
Shattering Read-k Formulae



Shattering Read-k Formulae



Shattering Read-k Formulae



At most k iterations are required to successfully shatter a read-k formula.

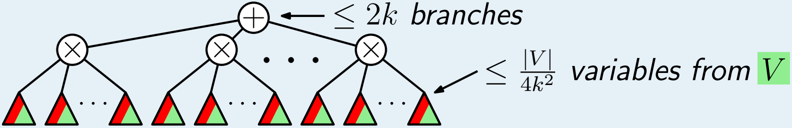
The Shattering Lemma

Shattering Lemma

For any nonzero multilinear $\sum^2$*-read-*k *formula* F *on* n *variables, there exist sets of variables*

- P, *with* $|P| = \mathrm{poly}(k)$, *and*
- V, *with* $|V| = \frac{n}{k^{O(k)}}$

such that $\frac{\partial F}{\partial P}$ *depends on at least the variables in* V, *and can be written as*



where each small subformula is the partial derivative of some subformula of F.

Extension: Blackbox

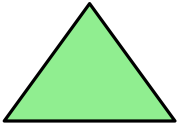
- Our algorithm uses the structure of the formula.

Extension: Blackbox

- Our algorithm uses the structure of the formula.
- A **blackbox** algorithm may only evaluate the formula.

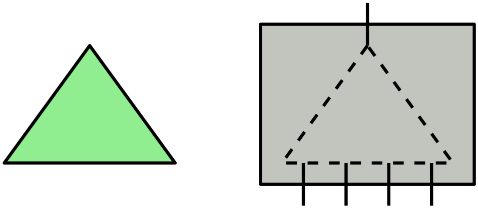
Extension: Blackbox

- Our algorithm uses the structure of the formula.
- A **blackbox** algorithm may only evaluate the formula.



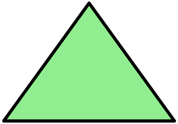
Extension: Blackbox

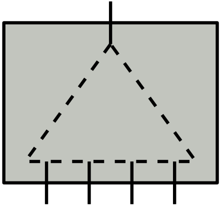
- Our algorithm uses the structure of the formula.
- A **blackbox** algorithm may only evaluate the formula.



Extension: Blackbox

- Our algorithm uses the structure of the formula.
- A **blackbox** algorithm may only evaluate the formula.





Theorem (Agrawal-Vinay)

*A **blackbox** poly-time identity test for depth-4 formula implies a blackbox subexp-time identity test for arithmetic formula.*

Extension: Blackbox - Outline

- Hitting Set Generators
- SV Generator
- Making our algorithm blackbox

Extension: Blackbox - Hitting Set Generators

Definition (Hitting Set Generator)

A polynomial map G,

Extension: Blackbox - Hitting Set Generators

Definition (Hitting Set Generator)

A polynomial map G,

$$G : \mathbb{F}^m \to \mathbb{F}^n,\ G = (G_1, \ldots, G_n),\ G_i \in \mathbb{F}[y_1, \ldots, y_m]$$

Extension: Blackbox - Hitting Set Generators

Definition (Hitting Set Generator)

A polynomial map G,

$$G : \mathbb{F}^m \to \mathbb{F}^n,\ G = (G_1, \ldots, G_n),\ G_i \in \mathbb{F}[y_1, \ldots, y_m]$$

is a hitting set generator (HSG) for a set of formulae $\mathcal{F}$, if

$$\forall F \in \mathcal{F},\ F \circ G \not\equiv 0 \text{ iff } F \not\equiv 0.$$

Extension: Blackbox - Hitting Set Generators

Definition (Hitting Set Generator)

A polynomial map G,

$$G : \mathbb{F}^m \to \mathbb{F}^n,\ G = (G_1, \ldots, G_n),\ G_i \in \mathbb{F}[y_1, \ldots, y_m]$$

is a hitting set generator (HSG) for a set of formulae $\mathcal{F}$, if

$$\forall F \in \mathcal{F},\ F \circ G \not\equiv 0 \text{ iff } F \not\equiv 0.$$

HSGs induce blackbox identity tests:

Extension: Blackbox - Hitting Set Generators

Definition (Hitting Set Generator)

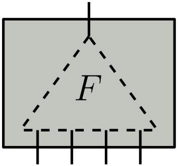
A polynomial map G,

$$G : \mathbb{F}^m \to \mathbb{F}^n,\ G = (G_1, \ldots, G_n),\ G_i \in \mathbb{F}[y_1, \ldots, y_m]$$

is a hitting set generator (HSG) for a set of formulae $\mathcal{F}$, if

$$\forall F \in \mathcal{F},\ F \circ G \not\equiv 0 \text{ iff } F \not\equiv 0.$$

HSGs induce blackbox identity tests:



Extension: Blackbox - Hitting Set Generators

Definition (Hitting Set Generator)

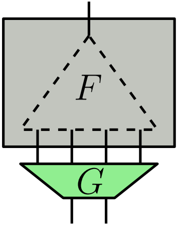
A polynomial map G,

$$G : \mathbb{F}^m \to \mathbb{F}^n,\ G = (G_1, \ldots, G_n),\ G_i \in \mathbb{F}[y_1, \ldots, y_m]$$

is a hitting set generator (HSG) for a set of formulae $\mathcal{F}$, if

$$\forall F \in \mathcal{F},\ F \circ G \not\equiv 0 \text{ iff } F \not\equiv 0.$$

HSGs induce blackbox identity tests:



Extension: Blackbox - Hitting Set Generators

Definition (Hitting Set Generator)

A polynomial map G,

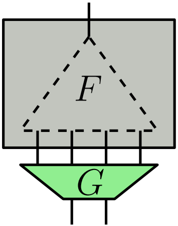
$$G : \mathbb{F}^m \to \mathbb{F}^n,\ G = (G_1, \dots, G_n),\ G_i \in \mathbb{F}[y_1, \dots, y_m]$$

is a hitting set generator (HSG) for a set of formulae $\mathcal{F}$, if

$$\forall F \in \mathcal{F},\ F \circ G \not\equiv 0 \text{ iff } F \not\equiv 0.$$

HSGs induce blackbox identity tests:

- Apply the Schwartz-Zippel Lemma to $F \circ G$.



Extension: Blackbox - Hitting Set Generators

Definition (Hitting Set Generator)

A polynomial map G,

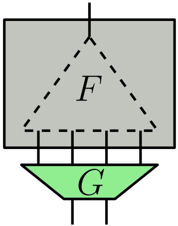
$$G : \mathbb{F}^m \to \mathbb{F}^n,\ G = (G_1, \ldots, G_n),\ G_i \in \mathbb{F}[y_1, \ldots, y_m]$$

is a hitting set generator (HSG) for a set of formulae $\mathcal{F}$, if

$$\forall F \in \mathcal{F},\ F \circ G \not\equiv 0 \text{ iff } F \not\equiv 0.$$

HSGs induce blackbox identity tests:

- Apply the Schwartz-Zippel Lemma to $F \circ G$.
- The test queries $O((d_F \cdot d_G)^m)$ inputs.



Extension: Blackbox - Hitting Set Generators

Definition (Hitting Set Generator)

A polynomial map G,

$$G : \mathbb{F}^m \to \mathbb{F}^n,\ G = (G_1, \ldots, G_n),\ G_i \in \mathbb{F}[y_1, \ldots, y_m]$$

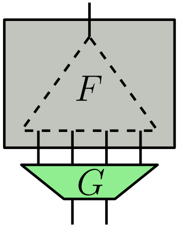
is a hitting set generator (HSG) for a set of formulae $\mathcal{F}$, if

$$\forall F \in \mathcal{F},\ F \circ G \not\equiv 0 \text{ iff } F \not\equiv 0.$$

HSGs induce blackbox identity tests:

- Apply the Schwartz-Zippel Lemma to $F \circ G$.
- The test queries $O((d_F \cdot d_G)^m)$ inputs.

Efficient HSGs: low degree d_G and seed length m.



Extension: Blackbox - SV Generator

We will use the generator G_{SV} from [SV09]:

Extension: Blackbox - SV Generator

We will use the generator G_{SV} from [SV09]:

- H_1 is in the image of G_{SV}.

Extension: Blackbox - SV Generator

We will use the generator G_{SV} from [SV09]:

- H_1 is in the image of G_{SV}.
- G_{SV} has degree $n+1$ and seed length 2.

Extension: Blackbox - SV Generator

We will use the generator G_{SV} from [SV09]:

- H_1 is in the image of G_{SV}.
- G_{SV} has degree $n+1$ and seed length 2.
- Let G_{SV}^w be the sum of w copies of G_{SV} over new variables.

Extension: Blackbox - SV Generator

We will use the generator G_{SV} from [SV09]:

- H_1 is in the image of G_{SV}.
- G_{SV} has degree $n+1$ and seed length 2.
- Let G_{SV}^w be the sum of w copies of G_{SV} over new variables.
- H_w is in the image of G_{SV}^w.

Extension: Blackbox - SV Generator

We will use the generator G_{SV} from [SV09]:

- H_1 is in the image of G_{SV}.
- G_{SV} has degree $n+1$ and seed length 2.
- Let G_{SV}^w be the sum of w copies of G_{SV} over new variables.
- H_w is in the image of G_{SV}^w.

Lemma

If G is a HSG for $\partial_x F$, then $G + G_{SV}$ is a HSG for F.

Extension: Blackbox - SV Generator

We will use the generator G_{SV} from [SV09]:

- H_1 is in the image of G_{SV}.
- G_{SV} has degree $n+1$ and seed length 2.
- Let G_{SV}^w be the sum of w copies of G_{SV} over new variables.
- H_w is in the image of G_{SV}^w.

Lemma

If G is a HSG for $\partial_x F$, then $G + G_{SV}$ is a HSG for F.

Fact

If G is a HSG for $\mathcal{F}$, then G is a HSG for products over $\mathcal{F}$.

Extension: Blackbox - Read-$(k+1) \leq \sum^2$-Read-k

Lemma

If G is a HSG for $\sum^2$-read-k formulae, then $G + G_{SV}^{\log n}$ is a HSG read-$(k+1)$ formulae.

Extension: Blackbox - Read-$(k+1) \leq \sum^2$-Read-k

Lemma

If G is a HSG for $\sum^2$-read-k formulae, then $G + G_{SV}^{\log n}$ is a HSG read-$(k+1)$ formulae.

Proof.

By induction on n.

Extension: Blackbox - Read-$(k+1) \leq \sum^2$-Read-k

Lemma

If G is a HSG for $\sum^2$-read-k formulae, then $G + G_{SV}^{\log n}$ is a HSG read-$(k+1)$ formulae.

Proof.

By induction on n.

- Suppose x fragments F.

Extension: Blackbox - Read-$(k+1) \leq \sum^2$-Read-k

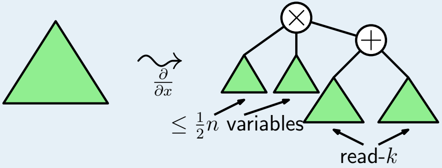
Lemma

If G is a HSG for $\sum^2$-read-k formulae, then $G + G_{SV}^{\log n}$ is a HSG read-$(k+1)$ formulae.

Proof.

By induction on n.

- Suppose x fragments F.



Extension: Blackbox - Read-$(k+1) \leq \sum^2$-Read-k

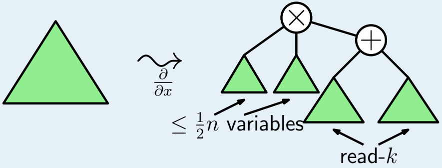
Lemma

If G is a HSG for $\sum^2$-read-k formulae, then $G + G_{SV}^{\log n}$ is a HSG read-$(k+1)$ formulae.

Proof.

By induction on n.

- Suppose x fragments F.



- By IH, $G + G_{SV}^{(\log n)-1}$ is a HSG for $\partial_x F$.

Extension: Blackbox - Read-$(k+1) \leq \sum^2$-Read-k

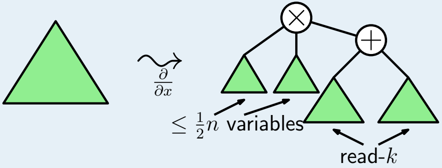
Lemma

If G is a HSG for $\sum^2$-read-k formulae, then $G + G_{SV}^{\log n}$ is a HSG read-$(k+1)$ formulae.

Proof.

By induction on n.

- Suppose x fragments F.



- By IH, $G + G_{SV}^{(\log n)-1}$ is a HSG for $\partial_x F$.
- Therefore, $G + G_{SV}^{\log n}$ is a HSG for F. ∎

Extension: Blackbox - $\sum^2$-Read-$k \leq$ Read-k

Lemma

If G is a HSG for read-k formulae, then $G + G_{SV}^{k^{O(k)}}$ is a HSG for $\sum^2$-read-k formulae.

Extension: Blackbox - $\sum^2$-Read-k $\leq$ Read-k

Lemma

If G is a HSG for read-k formulae, then $G + G_{SV}^{k^{O(k)}}$ is a HSG for $\sum^2$-read-k formulae.

Proof.

Extension: Blackbox - $\sum^2$-Read-k $\leq$ Read-k

Lemma

If G is a HSG for read-k formulae, then $G + G_{SV}^{k^{O(k)}}$ is a HSG for $\sum^2$-read-k formulae.

Proof.

- Since G is a HSG for read-k formulae, G is a HSG for all the formulae in the condition on $\bar{\sigma}$.

Extension: Blackbox - $\sum^2$-Read-$k \leq$ Read-k

Lemma

If G is a HSG for read-k formulae, then $G + G_{SV}^{k^{O(k)}}$ is a HSG for $\sum^2$-read-k formulae.

Proof.

- Since G is a HSG for read-k formulae, G is a HSG for all the formulae in the condition on $\bar{\sigma}$.
- Select an appropriate point in the image of G to be $\bar{\sigma}$.

Extension: Blackbox - $\sum^2$-Read-k $\leq$ Read-k

Lemma

If G is a HSG for read-k formulae, then $G + G_{SV}^{k^{O(k)}}$ is a HSG for $\sum^2$-read-k formulae.

Proof.

- Since G is a HSG for read-k formulae, G is a HSG for all the formulae in the condition on $\bar{\sigma}$.
- Select an appropriate point in the image of G to be $\bar{\sigma}$.
- $H_{k^{O(k)}}$ is in the image of $G_{SV}^{k^{O(k)}}$.

Extension: Blackbox - $\sum^2$-Read-k $\leq$ Read-k

Lemma

If G is a HSG for read-k formulae, then $G + G_{SV}^{k^{O(k)}}$ is a HSG for $\sum^2$-read-k formulae.

Proof.

- Since G is a HSG for read-k formulae, G is a HSG for all the formulae in the condition on $\bar{\sigma}$.
- Select an appropriate point in the image of G to be $\bar{\sigma}$.
- $H_{k^{O(k)}}$ is in the image of $G_{SV}^{k^{O(k)}}$.
- $\bar{\sigma} + H_{k^{O(k)}}$ is in the image of $G + G_{SV}^{k^{O(k)}}$.

Extension: Blackbox - $\sum^2$-Read-k $\leq$ Read-k

Lemma

If G is a HSG for read-k formulae, then $G + G_{SV}^{k^{O(k)}}$ is a HSG for $\sum^2$-read-k formulae.

Proof.

- Since G is a HSG for read-k formulae, G is a HSG for all the formulae in the condition on $\bar{\sigma}$.
- Select an appropriate point in the image of G to be $\bar{\sigma}$.
- $H_{k^{O(k)}}$ is in the image of $G_{SV}^{k^{O(k)}}$.
- $\bar{\sigma} + H_{k^{O(k)}}$ is in the image of $G + G_{SV}^{k^{O(k)}}$.
- Apply the non-blackbox analysis. ∎

Extension: Blackbox - The Final HSG

Theorem

$G_{SV}^{k^{O(k)}+O(k\log n)}$ *is a HSG for multilinear read-k formula.*

Extension: Blackbox - The Final HSG

Theorem

$G_{SV}^{k^{O(k)}+O(k \log n)}$ *is a HSG for multilinear read-k formula.*

Corollary

There is a quasi-polynomial-time blackbox identity test for multilinear constant-read formulae.

Extension: Blackbox - The Final HSG

Theorem

$G_{SV}^{k^{O(k)}+O(k\log n)}$ *is a HSG for multilinear read-k formula.*

Corollary

There is a quasi-polynomial-time blackbox identity test for multilinear constant-read formulae.

Corollary

There is a polynomial-time blackbox identity test for multilinear constant-read constant-depth formulae.

Extension: Blackbox - The Final HSG

Theorem

$G_{SV}^{k^{O(k)}+O(k\log n)}$ *is a HSG for multilinear read-k formula.*

Corollary

There is a quasi-polynomial-time blackbox identity test for multilinear constant-read formulae.

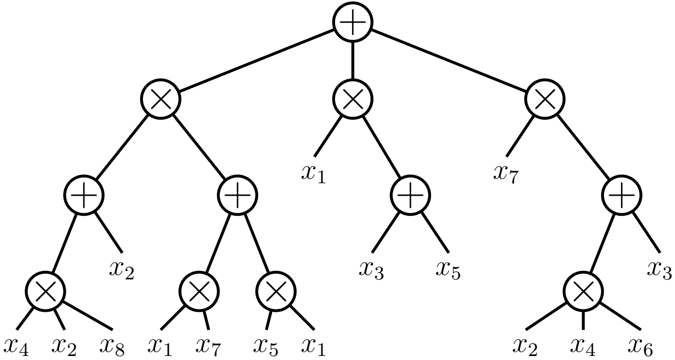
Corollary

There is a polynomial-time blackbox identity test for multilinear constant-read constant-depth formulae.

Idea: *Analyze the depth parameter in the Fragmentation Lemma.*

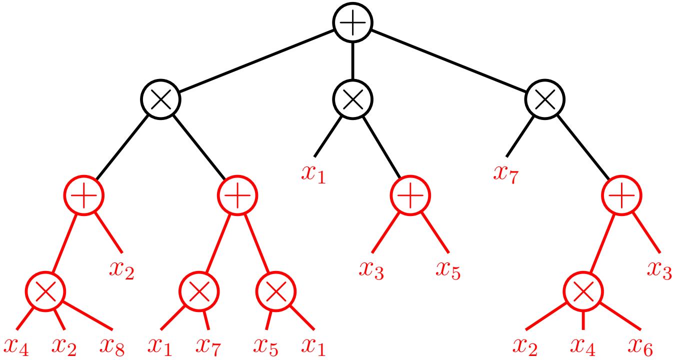
Extension: Sparse-Substituted

Read-3 depth-4



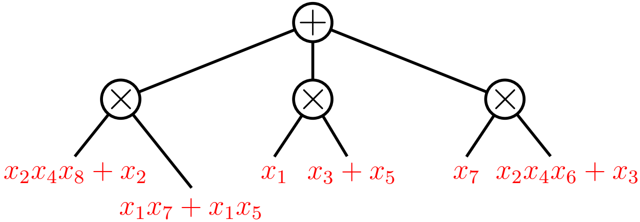
Extension: Sparse-Substituted

Read-3 depth-4



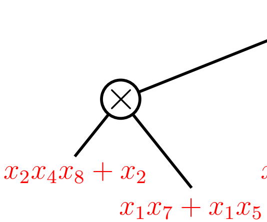
Extension: Sparse-Substituted

Read-3 depth-4 (and read-2 depth-2 sparse-substituted)



Extension: Sparse-Substituted

Read-3 depth-4 (and read-2 depth-2 sparse-substituted)

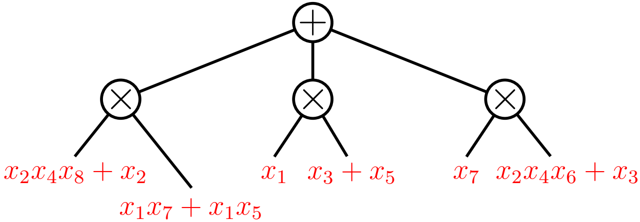


- Our tests extend to this model at quasi-polynomial cost.

  Idea: *Fragment sparse polys by also using substitutions.*

Extension: Sparse-Substituted

Read-3 depth-4 (and read-2 depth-2 sparse-substituted)



- Our tests extend to this model at quasi-polynomial cost.

  Idea: *Fragment sparse polys by also using substitutions.*
- Encompasses tests for
  - Multilinear Constant-Top-Fanin Depth-4 [KMSV10],
  - A generalized version of $\sum^k$-Read-Once [SV09].

Summary

Main Theorem

There is a polynomial-time deterministic identity test for multilinear constant-read formulae.

Summary

Main Theorem

There is a polynomial-time deterministic identity test for multilinear constant-read formulae.

Extensions:

1. Blackbox: quasi-poly-time.
2. Sparse substituted: quasi-poly-time.

Open Questions

- Is there a poly-time blackbox test for multilinear constant-read formulae?
- Can we drop the multilinearity requirement?
- For these types of formulae can we get
  - interesting lower bounds?
  - reconstruction algorithms?
- Is AFIT in P?
- Can any randomized algorithm be efficiently derandomized?

Questions?

Thanks!