Derandomizing Polynomial Identity Testing for Multilinear Constant-Read Formulae

Matthew Anderson  Dieter van Melkebeek
UW - Madison  UW - Madison

Ilya Volkovich
Technion

June 10th, 2011
Arithmetic Formula Identity Testing

Problem (AFIT)
Arithmetic Formula Identity Testing

**Problem (AFIT)**

**Input:** \( F \in \mathbb{F}[x_1, \ldots, x_n] \)
Arithmetic Formula Identity Testing

Problem (AFIT)

Input: $F \in \mathbb{F}[x_1, \ldots, x_n]$, given as an arithmetic formula.
Problem (AFIT)

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

*Question:* Is $F \equiv 0$?

Motivation: primality testing, circuit lower bounds, ...
Algorithms for AFIT

Randomized algorithm [DL78,Z79,S80,IM83]:
Algorithms for AFIT

Randomized algorithm [DL78, Z79, S80, IM83]:

- Pick $a_i \in S$ uniformly, accept iff $P(a_1, \ldots, a_n) = 0$
Algorithms for AFIT

Randomized algorithm [DL78, Z79, S80, IM83]:

- Pick $a_i \in S$ uniformly, accept iff $P(a_1, \ldots, a_n) = 0$
- Correctness: $\Pr_{a_i \in uS}[P(a_1, \ldots, a_n) = 0| P \neq 0] \leq \frac{d}{|S|}$
Algorithms for AFIT

Randomized algorithm [DL78,Z79,S80,IM83]:
- Pick $a_i \in S$ uniformly, accept iff $P(a_1, ..., a_n) = 0$
- Correctness: $\Pr_{a_i \in S}[P(a_1, ..., a_n) = 0 | P \neq 0] \leq \frac{d}{|S|}$

Deterministic algorithms for bounded-depth formulae:
Randomized algorithm [DL78,Z79,S80,IM83]:

- Pick $a_i \in S$ uniformly, accept iff $P(a_1, \ldots, a_n) = 0$
- **Correctness:** $\Pr_{a_i \in S}[P(a_1, \ldots, a_n) = 0 | P \not\equiv 0] \leq \frac{d}{|S|}$

Deterministic algorithms for bounded-depth formulae:

- Depth-2 [several]
Algorithms for AFIT

Randomized algorithm [DL78,Z79,S80,IM83]:
- Pick $a_i \in S$ uniformly, accept iff $P(a_1, \ldots, a_n) = 0$
- Correctness: $\Pr_{a_i \in u S}[P(a_1, \ldots, a_n) = 0|P \not\equiv 0] \leq \frac{d}{|S|}$

Deterministic algorithms for bounded-depth formulae:
- Depth-2 [several]
- Constant-Top-Fanin Depth-3 [DS06,KS07,KS08,KS09,SS11]
Algorithms for AFIT

Randomized algorithm [DL78,Z79,S80,IM83]:
- Pick $a_i \in S$ uniformly, accept iff $P(a_1, \ldots, a_n) = 0$
- Correctness: $\Pr_{a_i \in S}[P(a_1, \ldots, a_n) = 0 | P \neq 0] \leq \frac{d}{|S|}$

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]
Algorithms for AFIT

Randomized algorithm [DL78,Z79,S80,IM83]:

- Pick $a_i \in S$ uniformly, accept iff $P(a_1, \ldots, a_n) = 0$
- Correctness: $\Pr_{a_i \in S}[P(a_1, \ldots, a_n) = 0|P \not\equiv 0] \leq \frac{d}{|S|}$

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:
Algorithms for AFIT

Randomized algorithm [DL78,Z79,S80,IM83]:
- Pick $a_i \in S$ uniformly, accept iff $P(a_1, \ldots, a_n) = 0$
- Correctness: $\Pr_{a_i \in S}[P(a_1, \ldots, a_n) = 0 | P \not\equiv 0] \leq \frac{d}{|S|}$

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
Algorithms for AFIT

Randomized algorithm [DL78,Z79,S80,IM83]:
- Pick $a_i \in S$ uniformly, accept iff $P(a_1, \ldots, a_n) = 0$
- Correctness: $\Pr_{a_i \in uS}[P(a_1, \ldots, a_n) = 0 | P \neq 0] \leq \frac{d}{|S|}$

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]
Algorithms for AFIT

Randomized algorithm [DL78,Z79,S80,IM83]:

- Pick $a_i \in S$ uniformly, accept iff $P(a_1, \ldots, a_n) = 0$
- Correctness: $\Pr_{a_i \in S}[P(a_1, \ldots, a_n) = 0 | P \neq 0] \leq \frac{d}{|S|}$

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]
Theorem (Main)

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

Theorem (Weakened Main)

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

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

Techniques:
Theorem (Weakened Main)

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

Techniques:

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

**Theorem (Weakened Main)**

There is a $s^{O(1)} \cdot n^{k^{O(k)} + O(k \log n)}$ time deterministic algorithm for identity testing $n$-variable size-$s$ 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

Theorem (Weakened Main)

There is a $s^{O(1)} \cdot n^{k^{O(k)} + O(k \log n)}$ time deterministic algorithm for identity testing $n$-variable size-$s$ 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.
Theorem (Weakened Main)

There is a \( s^{O(1)} \cdot n^{k^{O(k)} + O(k \log n)} \) time deterministic algorithm for identity testing \( n \)-variable size-\( s \) 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 \( \frac{\partial}{\partial x_7} \)

\[ x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6 \quad x_7 \]

\[ x_8 \quad x_9 \quad x_{10} \quad x_{11} \quad x_{12} \quad x_{13} \]

Median
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 $\frac{\partial}{\partial x_7}$

Median
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.
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:

\[
\begin{align*}
&x_4 \\
&\quad \times \quad + \\
&\quad x_1 \quad x_2 \quad x_1 \quad x_3 \\
&\quad + \\
&\quad x_4
\end{align*}
\]

Pick largest child which contains \(k+1\) occurrences of some variable.
Fragmenting Read-\((k + 1)\) Formulae

A read-2 formula:

\[ + \times x_1 \times x_2 \times \times x_3 \times x_4 \]

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.
A read-2 formula:

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

Lemma (Fragmentation Lemma)

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

Lemma (Fragmentation Lemma)

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

Lemma (Fragmentation Lemma)

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

**Lemma (Fragmentation Lemma)**

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

\[ \frac{\partial}{\partial x} \leq \frac{1}{2} n \text{ variables} \]
The Fragmentation Lemma

**Lemma (Fragmentation Lemma)**

*Let $F$ be a nonzero read-$(k + 1)$ formula.*
The Fragmentation Lemma

Lemma (Fragmentation Lemma)

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

\[
\begin{align*}
&\leq \frac{1}{2} n \text{ variables} \\
\end{align*}
\]
Lemma (Fragmentation Lemma)

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

- $\leq \frac{1}{2}n$ variables
- $\geq \frac{1}{2}n$ variables
- $\leq \frac{1}{2}n$ variables
The Fragmentation Lemma

**Lemma (Fragmentation Lemma)**

*Let $F$ be a nonzero read-$(k+1)$ formula.*
Theorem (Weakened Main)

There is a $s^{O(1)} \cdot n^{k^{O(k)} + O(k \log n)}$ time deterministic algorithm for identity testing $n$-variable size-$s$ 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-$k$. 
Theorem (Weakened Main)

There is a $s^{O(1)} \cdot n^{k^{O(k)}} + O(k \log n)$ time deterministic algorithm for identity testing $n$-variable size-$s$ 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-$k$. 
Theorem (Weakened Main)

There is a \( s^{O(1)} \cdot n^{k^{O(k)} + O(k \log n)} \) time deterministic algorithm for identity testing \( n \)-variable size-\( s \) 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

Theorem (Weakened Main)

There is a $s^{O(1)} \cdot n^{k^O(k) + O(k \log n)}$ time deterministic algorithm for identity testing $n$-variable size-$s$ 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 $\sum^2\text{-read-}k \leq \text{Testing read-}k$

**Fact (SV Hitting Set [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 $\sum^2\text{-read-}k \leq \text{Testing read-}k$

Fact (SV Hitting Set [SV09])

The set of binary strings $H_w$ with Hamming weight at most $w$ hits any class $F$ 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 $\sum^2\text{-read-}k$ formula.
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 [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 $\mathcal{F}$. 
Fact (SV Hitting Set [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

**Lemma**

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

**Lemma**

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

1. no variable divides any $F_i$, 

\[ \sum^{k} \text{Read-}k \leq \text{Read-}k \]

\[ \sum^{2} \text{Read-}k \leq \text{Read-}k \]
A Structural Witness Lemma

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

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:

$\leq \frac{n}{m^2}$ variables
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:

\[ \cdot \cdot \cdot \times \cdot \cdot \cdot \times \cdot \cdot \cdot \times \leq \frac{n}{m^2} \] variables

$\Rightarrow$ $F$ does not compute a monomial of degree $n$. 
The Shattering Lemma

**Lemma (Shattering Lemma)**

For any nonzero multilinear $\sum^2$-read-$k$ formula $F$ on $n$ variables, there exist disjoint sets of variables $P$ and $V$, with $|P| = \text{poly}(k)$ and $|V| = \frac{n}{k^{O(k)}}$ such that $\frac{\partial F}{\partial P}$ is nonzero and can be written as

\[ \leq 2k \text{ branches} \]

\[ \leq \frac{|V|}{4k^2} \text{ variables in } V \]

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

1. $F(\bar{x} + \bar{\sigma})$ is not a monomial.
2. $\bar{\sigma}$ is easy to compute.
Theorem

1. \( F(\bar{x} + \bar{\sigma}) \) is not a monomial.
2. \( \bar{\sigma} \) is easy to compute.

Proof.

\[ \begin{align*}
\text{Suppose } F(\bar{x} + \bar{\sigma}) & \text{ is a monomial } M_n \text{ of degree } n. \\
\Rightarrow & \text{ If } n' \geq 1, \text{ by Lemma, some branch is divisible by a variable } x_j. \\
\Rightarrow & x_j = 0 \text{ is a root of that branch.} \\
\text{Pick } \bar{\sigma} \text{ to be a common nonzero of nonzero partial derivatives of all subformulae of the } F_i. \\
\text{Contradiction!} \\
F & \text{ is } \sum_{-\infty}^{2}, \text{ so } \bar{\sigma} \text{ can be computed efficiently using a read-}\k \text{ identity test.} \\
\end{align*} \]
Theorem

1. \( F(\bar{x} + \bar{\sigma}) \) is not a monomial.
2. \( \bar{\sigma} \) is easy to compute.

Proof.

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

1. $F(\bar{x} + \bar{\sigma})$ is not a monomial.
2. $\bar{\sigma}$ is easy to compute.

Proof.

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

$\Rightarrow$

\[
\begin{array}{c}
+ \\
F_1 \\
F_2 \\
\end{array} \quad \equiv M_n
\]
Theorem

1. \( F(\bar{x} + \bar{\sigma}) \) is not a monomial.
2. \( \bar{\sigma} \) is easy to compute.

Proof.

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

\[ \implies \text{Shatter}(\sum_{2}^{2} F_1 + F_2) \equiv M_n \]
Theorem

1. $F(\bar{x} + \bar{\sigma})$ is not a monomial.
2. $\bar{\sigma}$ is easy to compute.

Proof.

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

⇒ 

$\prod \ldots \prod \equiv M_{n'}$
Theorem

1. $F(\bar{x} + \bar{\sigma})$ is not a monomial.
2. $\bar{\sigma}$ is easy to compute.

Proof.

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

\[ \Rightarrow \]

\[ \equiv \]

\[ \Rightarrow \text{If } n' \geq 1 \]
Theorem

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

Proof.

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

$$\Rightarrow \quad \times \quad \times \quad + \quad \times \quad \times \quad \equiv M_{n'}$$

$$\Rightarrow \quad \text{If } n' \geq 1$$
Theorem

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

Proof.

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

$$\Rightarrow \quad \begin{array}{c}
\times \\
\cdot \\
\cdot \\
\cdot \\
\times \\
\times \\
\times \\
\times \\
\end{array} \quad \begin{array}{c}
\times \\
\cdot \\
\cdot \\
\cdot \\
\times \\
\times \\
\times \\
\times \\
\end{array} \quad \equiv M_{n'}
$$

$\Rightarrow$ If $n' \geq 1$, by Lemma, some branch is divisible by a variable $x_j$. 
Theorem

1. \( F(\bar{x} + \bar{\sigma}) \) is not a monomial of degree \( n \geq k^{O(k)} \).
2. \( \bar{\sigma} \) is easy to compute.

Proof.

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

\[
\Rightarrow \quad \begin{array}{c}
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \\
\times \
\end{array} 
\equiv \ M_{n'}
\]

\( \Rightarrow \) If \( n' \geq 1 \), by Lemma, some branch is divisible by a variable \( x_j \).
\( \Rightarrow \) \( x_j = 0 \) is a root of that branch.
Theorem

1. \( F(\bar{x} + \bar{\sigma}) \) is not a monomial of degree \( n \geq k^{O(k)} \).
2. \( \bar{\sigma} \) is easy to compute.

Proof.

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

\[
\Rightarrow \quad \times \quad \times \quad \ldots \quad \times \quad \ldots \quad \times \quad \ldots \quad \times \\
\Rightarrow \quad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \\
\Rightarrow \quad \text{If } n' \geq 1, \text{ by Lemma, some branch is divisible by a variable } x_j. \\
\Rightarrow \quad x_j = 0 \text{ is a root of that branch.}

Pick \( \bar{\sigma} \) to be a common nonzero of nonzero partial derivatives of all subformulae of the \( F_i \).
Theorem

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

Proof.

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

$\Rightarrow$ 

\[
\begin{array}{c}
\times \\
\times \\
\times \\
\vdots \\
\vdots \\
\vdots \\
\times \\
\times \\
\times \\
\end{array} 
\begin{array}{c}
\times \\
\times \\
\times \\
\vdots \\
\vdots \\
\vdots \\
\times \\
\times \\
\times \\
\end{array} 
\begin{array}{c}
\vdots \\
\vdots \\
\vdots \\
\times \\
\times \\
\times \\
\vdots \\
\vdots \\
\vdots \\
\end{array} 
\begin{array}{c}
\times \\
\times \\
\times \\
\vdots \\
\vdots \\
\vdots \\
\times \\
\times \\
\times \\
\end{array} 
\begin{array}{c}
\vdots \\
\vdots \\
\vdots \\
\times \\
\times \\
\times \\
\vdots \\
\vdots \\
\vdots \\
\end{array} 
\end{array} 
\equiv M_{n'}

\Rightarrow$ If $n' \geq 1$, by Lemma, some branch is divisible by a variable $x_j$.

\Rightarrow x_j = 0$ is a root of that branch.

Pick $\bar{\sigma}$ to be a common nonzero of nonzero partial derivatives of all subformulæ of the $F_i$.  Contradiction!
Theorem

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

Proof.

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

$\Rightarrow$ If $n' \geq 1$, by Lemma, some branch is divisible by a variable $x_j$.
$\Rightarrow x_j = 0$ is a root of that branch.

Pick $\bar{\sigma}$ to be a common nonzero of nonzero partial derivatives of all subformulae of the $F_i$. Contradiction!

$F$ is $\sum^2$-read-$k$, so $\bar{\sigma}$ can be computed efficiently using a read-$k$ identity test.
Outline

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-\(k\).
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-$k$. 

Theorem ()

Corollary

There is a polynomial-time deterministic algorithm for identity testing multilinear constant-read formulae.
Outline

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-$k$.

**Theorem (Weakened Main)**

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

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-$k$.

**Theorem (Main)**

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

Techniques:

1. **Fragmenting**
   Reduces multilinear read- \((k + 1)\) to multilinear \(\text{read-}^2\text{-}k\).

2. **Shattering**
   Reduces multilinear \(\text{read-}^2\text{-}k\) to multilinear read-\(k\).

**Theorem (Main)**

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

**Corollary**

There is a polynomial-time deterministic algorithm for identity testing multilinear constant-read formulae.
Conclusion

Extensions
Conclusion

Extensions

Conclusion

Extensions

   - Constant-depth formulae: poly-time.
Conclusion

Extensions

   - Constant-depth formulae: poly-time.

Conclusion

Extensions

   - Constant-depth formulae: poly-time.

   - Encompasses depth-four multilinear formulae [KMSV10], and pre-processed $\sum^k$-read-once formulae [SV09].
Questions?

Thanks!

The full version of our paper may be found on ECCC.