# Testing Polynomial Equivalence by Scaling Matrices

Markus Bläser

Saarland University

Joint work with Raghavendra Rao and Jayalal Sarma (IIT Madras)

# Polynomial equivalence testing

**Input:** $f(X), g(X) \in K[x_1, \ldots, x_n]$ (by blackbox access)
**Question:** Is there an *invertible* matrix $A$ such that $f(X) = g(AX)$?

**Polynomial identity testing:**

- Given $f \in K[x_1, \ldots, x_n]$, is $f$ identically zero?
- If $f$ is given as a blackbox, then randomisation is inherently needed for achieving polynomial running time.
- If $f$ is given as a circuit, then it is a major open problem, whether there is a deterministic polynomial time algorithm.

Polynomial identity testing is a special case of equivalence testing.
$\longrightarrow$ look for randomized algorithms

# Previous results

Hardness:

- Ring isomorphism reduces to Equivalence testing (Agrawal–Saxena '06)

Randomized polynomial time algorithms when $g$ is a fixed polynomial:

- elementary symmetric polynomial (Kayal '11)
- power sum (Kayal '11)
- permanent (Kayal '12)
- determinant (Kayal '12)
- iterated matrix multiplication (Kayal et al. '17)

# Our results

We consider the case when $A$ is a diagonal matrix.

- Randomized polynomial time algorithm for this case.
- Randomized polynomial time algorithm that gives a maximum set of monomials such that their degree vectors are linearly independent.

Kayal's algorithms follow a general pattern:

- Reduction to permutation and scaling equivalence by studying the corresponding Lie algebra.
- Random polynomials have a trivial Lie algebra.
- Reduction to scaling equivalence.

# Related questions

What if $A$ is not invertible?

- Kayal'12: NP-hard
- Using a result by Shitov, this can be strengthened to complete for the existential theory over the underlying field.
- The question whether $X_{1,1}^{p(n)-n} \operatorname{per}_n(X) = \det_{p(n)}(AX)$ is equivalent to VP $=$ VNP.

# Isolating a monomial

**Lemma (Mulmuley–Vazirani–Vazirani)**

*Let* $F_1, \ldots, F_m \subseteq \{1, \ldots, n\}$. *If* $w_1, \ldots, w_n \in \{1, \ldots, 2n\}$ *are chosen uniformly at random, then there is a unique set of minimum weight with probability* $\geq 1/2$.

$(w(F) = \sum_{i \in F} w_i)$

- $f \in K[x_1, \ldots, x_n]$ multilinear
- monomials $\alpha \cdot x_1^{e_1} \cdots x_n^{e_n}$ with $e_i \in \{0, 1\}$
- $x_i \mapsto y^{w_i}$
- unique monomial of minimum degree

# Isolating a monomial (2)

**Theorem (Klivans and Spielman)**

*There is a randomized algorithm that given a non-zero* $f \in K[x_1, \ldots, x_n]$ *(by blackbox access) outputs a monomial* $m$ *of* $f$ *(degree vector* $\mathrm{Deg}(m)$ *and coefficient) with probability* $\geq 1 - \epsilon$ *in time polynomial in* $n$, $\Delta$, *and* $1/\epsilon$.

- Weighted sets $\longrightarrow$ larger weights.
- Since we only have blackbox access, substitutions are simulated when evaluating.

  Instead of substituting $x_i \to y^{w_i}$ and then evaluating at $\alpha$, we directly evaluate $f(\alpha^{w_1}, \ldots, \alpha^{w_n})$.

# Extracting a degree basis

- monomial $m = \alpha \cdot x_1^{d_1} \cdots x_n^{d_n}$
- degree vector $\mathrm{Deg}(m) := (d_1, \ldots, d_n)$

## Definition (Degree basis)

Let $f \in K[x_1, \ldots, x_n]$. Monomials $m_1, \ldots, m_t$ are a *degree basis* of $f$ if for any monomial $m$ of $f$,
$\mathrm{Deg}(m) \in \mathrm{lin}_{\mathbb{R}}\{\mathrm{Deg}(m_1), \ldots, \mathrm{Deg}(m_t)\}$.

## Theorem

*There is a randomized algorithm, that given* $f \in K[x_1, \ldots, x_n]$ *(by black box access) outputs a degree basis of* $f$. *The running time is polynomial in the degree* $\Delta$, $n$, *and the bit size* $L$ *of the coefficients.*

# Extracting a degree basis (2)

- Let $m_1, \ldots, m_t$ monomials of $f$ with
- linearly independent degree vectors $\nu_1, \ldots, \nu_t$, $t < n$.
- Extend $\nu_1, \ldots, \nu_t$ to an (unknown) degree basis $\nu_1, \ldots, \nu_{\hat{t}}$.
- Let $p$ be a prime such that $\nu_1, \ldots, \nu_{\hat{t}}$ stay linearly independent over $\mathbb{F}_p$
- By the Hadamard bound and the prime number theorem, $p$ is small.

# Extracting a degree basis (3)

- Let $u_1, \ldots, u_{n-t}$ be linearly such that $v_i u_j = 0$ over $\mathbb{F}_p$ for all $1 \leq i \leq t$, $1 \leq j \leq n - t$.
- If $w$ is a vector not contained in $\mathrm{lin}\{v_1, \ldots, v_t\}$, then there is a $j$ such that $w u_j \neq 0$ over $\mathbb{F}_p$.
- Substitute $x_i \to y^{u_{j,i}} x_i$, $1 \leq i \leq n$, where $u_{j,i}$ are the entries of $u_j$. Let $f_j$ be the resulting polynomial.
- This maps every monomial $m$ to $y^d m$ for some $d$.

**Lemma**

1. *The degree of* $f_j$ *is bounded by* $O(\Delta n \mathit{polylog}(\Delta n))$ *for all* $j$.
2. *If* $\mathrm{Deg}(m) \in \mathrm{lin}\{v_1, \ldots, v_t\}$, *then for every* $j$, $p | d$.
3. *If* $\mathrm{Deg}(m) \notin \mathrm{lin}\{v_1, \ldots, v_t\}$, *then there is a* $j$ *such that* $p \not| d$.

# Extracting a degree basis (4)

Let

$$f_j = \sum_{d=0}^{\Delta_j} g_d \cdot y^d.$$

Recall: $m \mapsto y^d m$.

- View $f_j$ as a polynomial in $y$ with coefficients from $K[x_1, \ldots, x_n]$.
- Extract a monomial from the coefficient polynomial of a power $y^d$ with $p \not| d$ using Klivans–Spielman.
- If we find a monomial, then we set $\nu_{t+1}$ to be its degree vector.
- If we do not find such a monomial, then $\nu_1, \ldots, \nu_t$ is a degree basis.

# Simulating blackbox access to $g_d$

$$f_j = \sum_{d=0}^{\Delta_j} g_d \cdot y^d$$

We have to provide blackbox access to the $g_d$'s:

- Given blackbox access to $f$, it is easy to simulate blackbox access to $f_j$.
- Now assume we want to evaluate $g_d$ at a point $\xi \in K^n$.
- We evaluate $f_j$ at the points $(\xi, \alpha_i) \in K^{n+1}$, $0 \leq i \leq \Delta_j$, where the $\alpha_i$ are pairwise distinct, that is, we compute values $f_j(\xi, \alpha_i) = \sum_{d=0}^{\Delta_j} g_d(\xi)\alpha_i^d$.
- From these values, we interpolate the coefficients of $f_j$, viewed as a univariate polynomial in $y$. The coefficient of $y^d$ is $g_d(\xi)$.

# Polynomial equivalence

- Let $f(X), g(X) \in K[x_1, \ldots, x_n]$ (given by black box access).
- Degree of $f$ and $g$ is bounded by $\Delta$ and all coefficients of $f$ and $g$ have bit length $\leq L$.
- Assume there is an invertible diagonal matrix $A$ such that $f(X) = g(AX)$.

**Observation**

*If* $f(X) = g(AX)$, *then* $f$ *and* $g$ *have the same set of monomials.*

$$x_1^{d_1} \cdots x_n^{d_n} \mapsto a_1^{d_1} \cdots a_n^{d_n} \cdot x_1^{d_1} \cdots x_n^{d_n}$$

# Polynomial equivalence (2)

## Lemma

*Let* $S = \{(m_i, \alpha_i) \mid 1 \leq i \leq n\}$ *be a degree base of* $f$*. If* $f(X) = g(AX)$ *for a non-singular diagonal matrix* $A$*, then such an* $A$ *can be computed deterministically in time polynomial in* $n$*,* $\Delta$ *and* $L$*. (*$a_i$ *given by polynomial size expressions with roots.)*

- Let $\alpha_i \neq 0$ and $\beta_i \neq 0$ be the coefficient of $m_i$ in $f$ and $g$.
- We get $n$ equations

$$\alpha_i = \beta_i \prod_{j=1}^{n} a_j^{d_{i,j}}$$

  where $\nu_i = \mathrm{Deg}(m_i) =: (d_{i,1}, \ldots, d_{i,n})$.
- Unique solution:

$$a_i = \prod_{j=1}^{n} (\alpha_j/\beta_j)^{\bar{d}_{i,j}}$$

  where $D = (d_{i,j})$ and $D^{-1} = (\bar{d}_{i,j})$.

# Polynomial equivalence (3)

What if a degree basis of $f$ has size $t < n$?

## Lemma

*Let* $a_1, \ldots, a_n$ *be any solution to*

$$\log \alpha_i = \log \beta_i + \sum_{j=1}^{n} d_{i,j} \log a_j, \qquad 1 \le i \le t,$$

*and let* $A$ *be the corresponding diagonal matrix. Let* $r(x)$ *be a monomial with coefficient* $\delta$ *and degree vector* $u = (e_1, \ldots, e_n)$ *contained in the linear span of* $v_1, \ldots, v_t$*, i.e.,* $u = \lambda_1 v_1 + \cdots + \lambda_t v_t$*. Then the coefficient of* $r(Ax)$ *is*

$$\delta \cdot \left(\frac{\alpha_1}{\beta_1}\right)^{\lambda_1} \cdots \left(\frac{\alpha_t}{\beta_t}\right)^{\lambda_t},$$

*in particular, it is independent of the chosen solution for* $a_1, \ldots, a_n$*.*

# Algorithm

**Scaling equivalence test**

**Input:** Black box access to polynomials $f, g \in K[x_1, \ldots, x_n]$

**Output:** Nonsingular diagonal matrix $A$ with $f(x) = g(Ax)$ if such an $A$ exists

1: Apply Gen-Mon with polynomial $f$ to get a set $S$.
2: Apply Gen-Mon to $g$ using the *same* random bits as above to get a set $S'$.
3: If $S$ and $S'$ do not contain the same degree vectors, then REJECT.
4: Solve for the entries of $A$ using Lemma 6.
5: ACCEPT if and only if $f(x) - g(Ax)$ is identically zero.

**Theorem**

*The Algorithm returns correct the correct answer with high probability. It runs in time polynomial in* $\Delta$, $n$ *and* $L$.

# Discussion

- Can our result extended to permutation and scaling equivalence?
- Are the other polynomials $g$ for which equivalence testing can be done in polynomial time?