Electronic Colloquium on Computational Complexity, Revision 2 of Report No. 120 (2026)

# Hitting point for sparse noncommutative polynomials

Foram Lakhani* Partha Mukhopadhyay†

July 2026

**Abstract**

For every $n, s \geq 1$, we construct a matrix tuple $(A_1, \ldots, A_n) \in \mathrm{M}_s(\mathbb{Z})^n$ in deterministic poly$(n, s)$ time such that every noncommutative polynomial

$$f \in \mathbb{C}\langle x_1, x_2, \ldots, x_n\rangle$$

of sparsity at most $s$ satisfies $f = 0$ if and only if $f(A_1, A_2, \ldots, A_n) = 0$. The bit complexity of the entries in the matrices $A_1, \ldots, A_n$ is $O(s \log n)$. This immediately gives a deterministic one-query black-box identity testing algorithm for these polynomials. This is done under the standard assumption that the black-box can be queried over matrix algebras.

In particular, a black-box randomized polynomial-time algorithm was known for sparse noncommutative polynomials of exponential degree [AJMR19], but (to the best of our knowledge) no deterministic polynomial-time algorithm was known. Interestingly, in the commutative case, deterministic polynomial-time black-box algorithm for sparse high-degree polynomials is well known [KS01, BHLV09, Sax09].

## 1 Introduction

Noncommutative computation was introduced in complexity theory by Hyafil [Hya77] and Nisan [Nis91]. Over the years, this has become a central topic of algebraic complexity theory.

In noncommutative computation, the main algebraic structure is the free noncommutative ring $\mathbb{F}\langle X\rangle$ over a field $\mathbb{F}$, where $X = \{x_1, x_2, \ldots, x_n\}$, and $x_i$ $(1 \leq i \leq n)$ are free noncommuting variables.

A fundamental problem in this area is designing efficient algorithms for noncommutative Polynomial Identity Testing (PIT). Just like in the usual commutative case, the problem can be stated as follows: Given $f \in \mathbb{F}\langle X\rangle$ by a noncommutative arithmetic circuit $C$ or by a black-box (which can be evaluated on matrices), the problem is to check if the polynomial is identically zero. Notice that if the size of the circuit $C$ is $s$, then the degree of the polynomial computed by $C$ can be $2^s$ and the sparsity of the polynomial can be as large as $2^{2^s}$. This can be observed from the standard example that $(x + y)^{2^s}$ has a circuit of size $O(s)$.

Bogdanov and Wee [BW05] have given a randomized polynomial-time algorithm when the degree of the noncommutative circuit $C$ is polynomially bounded in $s$ and $n$. The

*CSE, IIT Kanpur. Email: foramp23@iitk.ac.in.
†Chennai Mathematical Institute, Chennai, India. Email: partham@cmi.ac.in.



ISSN 1433-8092

algorithm is based on a fundamental result of Amitsur and Levitzki [AL50] which says that for any field $\mathbb{F}$, a nonzero noncommutative polynomial of degree $\leq 2d-1$ is not a polynomial identity for the matrix algebra $\mathrm{M}_d(\mathbb{F})$. The PIT algorithm in [BW05] essentially uses this result to prove that on random $d \times d$ matrix substitution, the polynomial evaluates to a nonzero matrix. Clearly, this result is not efficient once the degree of the polynomial is exponentially large. Over the years, solving the PIT question for noncommutative circuits that allow the degree to be exponentially large has become a central question.

In [AJMR19], an efficient *randomized* algorithm for the black-box setting is given based on a key result that says that a noncommutative nonzero polynomial of sparsity $s$ cannot be an identity for matrix algebras of dimension $O(\log s)$. Notice that this algorithm is efficient only when the sparsity of the polynomial is single-exponentially large. Additionally, they design a deterministic polynomial-time white-box algorithm for a special type of circuits (+-regular circuits) that allow the sparsity to be doubly-exponentially large. More recently, Bharadwaj and Raja [SBR25] have extended this result to solve the black-box identity testing problem for small-depth +-regular circuits in randomized polynomial time.

Despite all these results, some basic questions in the high-degree setting remain unresolved. The focus of our work is to give a *deterministic* polynomial-time black-box algorithm for sparse noncommutative polynomials which can potentially have exponentially large degrees. We note that when the degree is polynomially bounded, such a result was given in [AMS10]. Their algorithm is based on ideas from automata theory.

We note that in the commutative setting, for sparse polynomials of high degree, a black-box identity testing algorithm is well known [KS01, BHLV09, Sax09]. The algorithms are either based on the method of univariate substitution and Chinese remaindering, or uses deep results from number theory (Linnik's theorem). Such techniques are unlikely to be suitable for noncommutative polynomials.

**Organization.**

The paper is organized as follows. In Section 2, we prove the main result (over fields of characteristic zero) showing an efficient construction of a matrix tuple that hits noncommutative sparse polynomials. For simplicity, we describe it for polynomials defined over $\mathbb{C}$. The entries of the matrices are integers of polynomial size bit-complexity. In Section 3, we discuss the extension of this result over finite fields.

# 2 Main result

Let $f \in \mathbb{F}\langle x_1, \ldots, x_n \rangle$ be a polynomial in free noncommuting variables. Its monomials are words in the alphabet $\{x_1, \ldots, x_n\}$. We write $f = \sum_{w \in \mathrm{supp}(f)} c_w w$, where $c_w \in \mathbb{F} \setminus \{0\}$ are the coefficients. Define its sparsity by $\mathrm{sp}(f) := |\mathrm{supp}(f)|$.

Our main theorem is the following.

**Theorem 2.1.** *For every $n, s \geq 1$ there is an explicitly constructible tuple $(A_1, \ldots, A_n) \in \mathrm{M}_s(\mathbb{Z})^n$ such that, for every $f \in \mathbb{C}\langle x_1, \ldots, x_n \rangle$ with $\mathrm{sp}(f) \leq s$, $f = 0$ if and only if $f(A_1, A_2, \ldots, A_n) = 0$. Moreover, the bit-complexity of the entries of the matrices are $O(s \log n)$ and they are also computable in time $\mathrm{poly}(n, s)$.*

Choose the base integer $B = n + 1$ and for $x_1, x_2, \ldots, x_n$ fix $\delta_1 = 1, \ldots, \delta_n = n$

respectively. For a word $w = x_{i_1} x_{i_2} \cdots x_{i_\ell}$, define the encoding

$$\operatorname{enc}(w) := B^\ell + \sum_{t=1}^{\ell} \delta_{i_t} B^{\ell-t}. \tag{1}$$

For the empty word $\varepsilon$, set $\operatorname{enc}(\varepsilon) = 1$.

**Observation 2.2.** *The encoding is injective.*

Define the row vector

$$\mathbf{v}_s(z) := (1, z, z^2, \ldots, z^{s-1})$$

where $z$ is an indeterminate. For each $j \in \{1, \ldots, n\}$, define $A_j \in \mathrm{M}_s(\mathbb{Z})$, with rows and columns indexed by $0, 1, \ldots, s-1$, by

$$(A_j)_{a,b} := \begin{cases} \binom{b}{a} B^a \delta_j^{b-a}, & 0 \le a \le b \le s-1, \\ 0, & a > b. \end{cases} \tag{2}$$

These matrices are upper triangular, and $(A_j)_{a,b} \le 2^b B^b \le 2^s (n+1)^s$. So each entry can be represented using $O(s \log(n+1))$ bits.

**Lemma 2.3.** *For every $z$ and every $j$,*

$$\mathbf{v}_s(z) A_j = \mathbf{v}_s(Bz + \delta_j).$$

*Proof.* The $b$-th coordinate of $\mathbf{v}_s(z) A_j$ is

$$\sum_{a=0}^{b} z^a \binom{b}{a} B^a \delta_j^{b-a} = (Bz + \delta_j)^b$$

by the binomial theorem. □

For a word $w = x_{i_1} \cdots x_{i_\ell}$, the transition matrix is $A_w := A_{i_1} \cdots A_{i_\ell}$. For the word $\varepsilon$, the transition matrix is $I_s$.

Now we state the main technical observation as a lemma.

**Lemma 2.4.** *For every word $w$ we have the following encoding,*

$$\mathbf{v}_s(1) A_w = \mathbf{v}_s(\operatorname{enc}(w)) = \big(1, \operatorname{enc}(w), \operatorname{enc}(w)^2, \ldots, \operatorname{enc}(w)^{s-1}\big).$$

*Proof.* We prove the lemma by an induction on the length of $w$. Suppose $\ell = 1$ and $w = x_{i_\ell}$. Then by Lemma 2.3, $\mathbf{v}_s(1) A_{x_{i_\ell}} = v_s(B + \delta_{i_\ell})$. This also holds for $\varepsilon$.

Next consider the word $w = w_{[\ell-1]} x_{i_\ell}$. Clearly, $A_w = A_{w_{[\ell-1]}} A_{i_\ell}$. Notice that

$$\mathbf{v}_s(1) A_w = (\mathbf{v}_s(1) A_{w_{[\ell-1]}}) A_{i_\ell}$$

By the induction hypothesis

$$\mathbf{v}_s(1) A_{w_{\ell-1}} = (1, \operatorname{enc}(w_{[\ell-1]}), \ldots, \operatorname{enc}^{s-1}(w_{[\ell-1]})).$$

Now the $b^{th}$ entry of the final vector is given by

$$\sum_{a=0}^{b} \operatorname{enc}^a(w_{[\ell-1]}) \binom{b}{a} B^a \delta_{i_\ell}^{b-a} = (B \operatorname{enc}(w_{[\ell-1]}) + \delta_{i_\ell})^b.$$

However $(B \operatorname{enc}(w_{[\ell-1]}) + \delta_{i_\ell})^b = \operatorname{enc}^b(w)$. □

## 2.1 The Vandermonde matrices

**Lemma 2.5.** *Let $\alpha_1, \dots, \alpha_r \in \mathbb{F}$ be distinct, where $r \le s$. Then the vectors*

$$\mathbf{v}_s(\alpha_i) = (1, \alpha_i, \dots, \alpha_i^{s-1}), \qquad i = 1, \dots, r,$$

*are linearly independent.*

*Proof.* It suffices to retain the first $r$ coordinates. The resulting $r \times r$ matrix is the Vandermonde matrix

$$V = \begin{pmatrix} 1 & \alpha_1 & \alpha_1^2 & \cdots & \alpha_1^{r-1} \\ 1 & \alpha_2 & \alpha_2^2 & \cdots & \alpha_2^{r-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \alpha_r & \alpha_r^2 & \cdots & \alpha_r^{r-1} \end{pmatrix}.$$

Its determinant is

$$\det V = \prod_{1 \le i < j \le r} (\alpha_j - \alpha_i),$$

which is nonzero because the $\alpha_i$ are distinct. □

Now we prove Theorem 2.1.

*Proof.* Write $f = \sum_{k=1}^{r} c_k w_k$ where $r \le s$. Suppose that $f(A_1, \dots, A_n) = 0$. Multiplying by $\mathbf{v}_s(1)$ and using Lemma 2.4, we obtain

$$0 = \mathbf{v}_s(1) f(A_1, \dots, A_n) = \sum_{k=1}^{r} c_k \mathbf{v}_s(\mathrm{enc}(w_k)).$$

By the injectivity of enc, the integers $\mathrm{enc}(w_1), \dots, \mathrm{enc}(w_r)$ are distinct. Since $\mathbb{F}$ has characteristic zero, distinct integers remain distinct as elements of $\mathbb{F}$. Hence Lemma 2.5 implies that $c_1 = \cdots = c_r = 0$, a contradiction. The reverse implication is immediate. □

The black-box algorithm follows immediately from Theorem 2.1.

*Remark* 2.6 (Extension to arbitrary characteristic-zero fields). Although Theorem 2.1 is stated over $\mathbb{C}$, the same construction and proof work over every field $\mathbb{F}$ of characteristic zero. Indeed, the matrices $A_1, \dots, A_n$ constructed above have integer entries and therefore can be viewed as matrices over $\mathbb{F}$ via the canonical embedding $\mathbb{Z} \hookrightarrow \mathbb{F}$.

## 2.2 Optimality of the matrix dimension

In this section, we argue that the $s \times s$ dimension of the matrices is optimal among all *single* matrix hitting points. This lower bound follows easily from the Cayley-Hamilton Theorem for univariate polynomials.

**Theorem 2.7.** *Let $\mathbb{F}$ be any field, let $n \ge 1$, and let $\mathbf{A} = (A_1, \dots, A_n) \in \mathrm{M}_k(\mathbb{F})^n$. If $\mathbf{A}$ hits every noncommutative polynomial in $\mathbb{F}\langle x_1, \dots, x_n \rangle$ of sparsity at most $s$, then $k \ge s$.*

*Proof.* Assume for contradiction that $k < s$. Let

$$\mu_{A_1}(T) = T^t + a_{t-1} T^{t-1} + \cdots + a_1 T + a_0$$

be the minimal polynomial of $A_1$. By the Cayley-Hamilton theorem, its degree satisfies $t \leq k$. Therefore the univariate polynomial $g(x_1) = \mu_{A_1}(x_1)$ is nonzero and has at most $t+1$ monomials. Since $t \leq k < s$, the sparsity of $g$ is at most $s$. On the other hand, by the definition of the minimal polynomial $g(A_1) = \mu_{A_1}(A_1) = 0$, contradicting the hitting property of $\mathbf{A}$. Thus $k \geq s$. □

**Corollary 2.8.** *Let $\mathbb{F}$ be any field, let $n \geq 1$, let $\mathbf{A_i} = (A_{i,1}, \ldots, A_{i,n}) \in \mathrm{M}_c(\mathbb{F})^n$ for some parameter $c$, and let $\mathcal{H} = \{\mathbf{A_i} \mid i \in [m]\}$ be a hitting set for every noncommutative polynomial in $\mathbb{F}\langle x_1, \ldots, x_n\rangle$ of sparsity at most $s$, then $cm = \Omega(s)$.*

*Proof.* Consider a $cm \times cm$ dimensional matrix $B_j$, as a block matrix where the $(i,i)$-th block is $A_{i,j}$ and is zero everywhere else. Then $\mathbf{B} = (B_1, \ldots, B_n)$ is a single matrix hitting point for all polynomials of sparsity $s$. By Theorem 2.7, we have $cm \geq s$. □

# 3 Over fields of positive characteristic

Our first observation is that over finite fields of polynomial size, we can not get a small size hitting set of matrices when the degrees of the polynomials are exponentially large. This follows by a simple counting argument. Next, we show that over sufficiently large finite fields, the techniques of Theorem 2.1 can be lifted to obtain a similar result in the algebraic setting.

## 3.1 Polynomially large finite fields

Let $\mathbb{F}$ be a finite field of size $q$, and

$$\mathcal{H} = \{\bar{A}_1, \ldots, \bar{A}_N\}$$

be a hitting set of size $N$ for polynomials $f \in \mathbb{F}\langle x_1, x_2, \ldots, x_n\rangle$ of degree $\leq D$. The $\bar{A}_i$ are matrix $n$-tuple of dimension $k$. Then the number of words of length $D$ is $n^D$. For any two words $w_1, w_2$, it must be the case that

$$\left(w_1(\bar{A}_1), \ldots, w_1(\bar{A}_N)\right) \neq \left(w_2(\bar{A}_1), \ldots, w_2(\bar{A}_N)\right),$$

otherwise, the polynomial $f = w_1 - w_2$ will be zero on $\mathcal{H}$. However, the total number of such $N$ tuples is $q^{k^2 N}$. Then we need

$$n^D \leq q^{k^2 N}.$$

In other words, $N \geq \frac{D \log n}{k^2 \log q}$. Consequently,

**Observation 3.1.** *If the matrix dimension $k$ is polynomially bounded in the input parameters, then an exponentially large degree $D$ forces the number $N$ of hitting points to be exponentially large (over polynomially large fields).*

## 3.2 Exponentially large finite fields

Let the field size be $q = p^m$ for a prime number $p$. More precisely, $\mathbb{F}_q = \mathbb{F}_p(\theta)$ where the minimal polynomial of $\theta$ over $\mathbb{F}_p$ has degree $m$. Set $a = \lceil \log_p(n+1) \rceil$, which implies that $p^a \geq n+1$. We also choose $m > aD$ where $D$ is the degree bounds for the noncommutative

polynomials. Assume that $m$ and $p$ are given as part of inputs using $O(\log m)$ and $O(\log p)$ bits respectively.

In the polynomial ring $\mathbb{F}_p[T]$, choose distinct polynomials $P_1(T), P_2(T), \ldots, P_n(T)$ such that the degree of $P_i : 1 \leq i \leq n$ is less than $a$. Such a collection of polynomials exists since $p^a \geq n+1$.

For a word $w = x_{i_1} x_{i_2} \ldots x_{i_\ell}$, define the polynomial,

$$P_w(T) = T^{a\ell} + \sum_{t=1}^{\ell} P_t(T) T^{a(\ell-t)}$$

and the encoding $\mathrm{enc}(w) = P_w(\theta)$. We claim the following.

**Claim 3.2.** *The encoding map* enc *is injective for all words of length* $\leq D$.

*Proof.* First, we prove that if $w \neq w'$ then $P_w(T) \neq P_{w'}(T)$. Observe the following pattern in the polynomials $P_w(T)$ for any word $w$: For $t_1 < t_2$, the polynomials $P_{t_1} T^{a(\ell-t_1)}$ cannot contribute a term generated in $P_{t_2} T^{a(\ell-t_2)}$.

Otherwise, there will be two monomials $m_{t_1}$ and $m_{t_2}$ from $P_{t_1}(T)$ and $P_{t_2}(T)$ such that

$$m_{t_1} T^{a(\ell-t_1)} = m_{t_2} T^{a(\ell-t_2)}.$$

This would imply that $\deg(m_{t_2}) = \deg(m_{t_1}) + a(t_2 - t_1)$ which is greater than $a$. This shows that from $P_w(T)$ one can *uniquely* recover $w$.

Hence, for $w \neq w'$, we get $P_w(T) \neq P_{w'}(T)$. If $\mathrm{enc}(w) = \mathrm{enc}(w')$, it would imply that $(P_w - P_{w'})(\theta) = 0$. But we just proved that $P_w - P_{w'}$ is a nonzero polynomial and its degree $\leq aD$ is less than $m$. Since the degree of the minimal polynomial of $\theta$ is $m$, we arrive at a contradiction.

Hence, the enc map is injective.

□

Next, to define the matrices, set $\lambda = \theta^a$ and $\delta_j = P_j(\theta)$. Also, for $1 \leq j \leq n$ define the matrices

$$(A_j)_{u,v} = \begin{cases} \binom{v}{u} \lambda^u \delta_j^{v-u}, & 0 \leq u \leq v \leq s-1, \\ 0, & u > v. \end{cases} \tag{3}$$

Next we prove the main result of this section which is an extension of Theorem 2.1.

**Theorem 3.3.** *For a sufficiently large field* $\mathbb{F}_{p^m}$ *together with an element* $\theta$ *of order* $m \geq D \lceil \log_p(n+1) \rceil$, *The* $n$*-tuple of matrices* $(A_1, \ldots, A_n)$ *is a hitting point for all nonzero noncommutative polynomials* $f(x_1, \ldots, x_n)$ *of degree* $\leq D$ *and sparsity* $\leq s$.

*Proof.* The proof is very similar to the proof of Theorem 2.1. Similar to Lemma 2.3, we notice that

$$v_s(z) A_j = v_s(\lambda z + \delta_j) = v_s(\theta^a z + \delta_j).$$

Moreover, using the same proof of Lemma 2.4,

$$v_s(1) A_w = v_s(\mathrm{enc}(w)).$$

The rest of the argument is exactly similar and follows from the full rank of Vandermonde matrix over $\mathbb{F}_q$. □

*Remark* 3.4. In contrast to Theorem 2.1, whose hitting matrices have integer entries of small bit complexity, the computational interpretation of the finite-field theorem depends on the representation of elements of $\mathbb{F}_{p^m}$. If the black-box model requires every field element to be given explicitly by its coordinates with respect to the basis

$$1, \theta, \ldots, \theta^{m-1},$$

then a general matrix entry may require $\Theta(m \log p)$ bits. Thus, when $m$ is large, the theorem should primarily be viewed as an algebraic statement. On the other hand, the entries of the matrices $A_j$ admit succinct descriptions as sparse polynomials or small arithmetic expressions in $\theta$. Consequently, in a black-box model that allows such succinctly represented field elements as inputs, the entire hitting point has a description of size polynomial in $n$, $s$, $\lceil \log_p(n+1) \rceil$, $\log p$, and $\log m$.


## Acknowledgements

The authors thank the organizers of WACT 2026, where the discussions started. The second author thanks Nutan Limaye and Srikanth Srinivasan for hosting him at the IT University of Copenhagen and the University of Copenhagen. The authors also thank Abhranil Chatterjee, Pranjal Dutta, Mrinal Kumar, and Amit Sinhababu for their comments.


# References


[AJMR19] Vikraman Arvind, Pushkar S. Joglekar, Partha Mukhopadhyay, and S. Raja. Randomized polynomial-time identity testing for noncommutative circuits. *Theory of Computing*, 15(7):1–36, 2019.

[AL50] Avraham Shimshon Amitsur and Jakob Levitzki. Minimal identities for algebras. *Proceedings of the American Mathematical Society*, 1(4):449–463, 1950.

[AMS10] Vikraman Arvind, Partha Mukhopadhyay, and Srikanth Srinivasan. New results on noncommutative and commutative polynomial identity testing. *Comput. Complex.*, 19(4):521–558, 2010.

[BHLV09] Markus Bläser, Moritz Hardt, Richard J. Lipton, and Nisheeth K. Vishnoi. Deterministically testing sparse polynomial identities of unbounded degree. *Information Processing Letters*, 109(3):187–192, 2009.

[BW05] A. Bogdanov and H. Wee. More on noncommutative polynomial identity testing. In *20th Annual IEEE Conference on Computational Complexity (CCC'05)*, pages 92–99, 2005.

[Hya77] Laurent Hyafil. The power of commutativity. In *Proceedings of the 18th Annual IEEE Symposium on Foundations of Computer Science (FOCS 1977)*, pages 171–174. IEEE, 1977.

[KS01] Adam R. Klivans and Daniel A. Spielman. Randomness efficient identity testing of multivariate polynomials. In Jeffrey Scott Vitter, Paul G. Spirakis, and

Mihalis Yannakakis, editors, *Proceedings on 33rd Annual ACM Symposium on Theory of Computing, July 6-8, 2001, Heraklion, Crete, Greece*, pages 216–223. ACM, 2001.

[Nis91] N. Nisan. Lower bounds for non-commutative computation. In *Proc. 23rd ACM Symp. on Theory of Computing*, pages 410–418, 1991.

[Sax09] Nitin Saxena. Progress on polynomial identity testing. *Bull. EATCS*, 99:49–79, 2009.

[SBR25] G. V. Sumukha Bharadwaj and S. Raja. Randomized Black-Box PIT for Small Depth +-Regular Non-Commutative Circuits. In *45th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2025)*, volume 360 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 51:1–51:16, 2025.



ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il