Electronic Colloquium on Computational Complexity, Report No. 25 (2023)

Multivariate to Bivariate Reduction for Noncommutative Polynomial Factorization

V. Arvind* Pushkar S Joglekar†

Abstract

Based on a theorem of Bergman [5] we show that multivariate noncommutative polynomial factorization is deterministic polynomial-time reducible to the factorization of bivariate noncommutative polynomials. More precisely, we show the following:

1. In the white-box setting, given an n-variate noncommutative polynomial $f \in \mathbb{F}\langle X \rangle$ over a field $\mathbb{F}$ (either a finite field or the rationals) as an arithmetic circuit (or algebraic branching program), computing a complete factorization of f is deterministic polynomial-time reducible to white-box factorization of a noncommutative bivariate polynomial $g \in \mathbb{F}\langle x, y \rangle$; the reduction transforms f into a circuit for g (resp. ABP for g), and given a complete factorization of g the reduction recovers a complete factorization of f in polynomial time.
We also obtain a similar deterministic polynomial-time reduction in the black-box setting.
2. Additionally, we show over the field of rationals that bivariate linear matrix factorization of 4×4 matrices is at least as hard as factoring square-free integers. This indicates that reducing noncommutative polynomial factorization to linear matrix factorization (as done in [1]) is unlikely to succeed over the field of rationals even in the bivariate case. In contrast, multivariate linear matrix factorization for 3×3 matrices over rationals is in polynomial time.

1 Introduction

The main aim of this paper is to show that multivariate polynomial factorization in the free noncommutative ring $\mathbb{F}\langle x_1, x_2, \ldots, x_n \rangle$ is polynomial-time reducible to *bivariate* noncommutative polynomial factorization in the bivariate ring $\mathbb{F}\langle x, y \rangle$. Such a result for commutative polynomial factorization is well-known due to Kaltofen's seminal work on multivariate polynomial factorization in the commutative polynomial ring $\mathbb{F}[y_1, y_2, \ldots, y_n]$ [9, 10]. However, this problem was open in the setting of noncommutative polynomials. Recently, [1] a randomized polynomial-time algorithm was obtained for the factorization of noncommutative polynomials over finite fields, where the input polynomial is given by a noncommutative formula.[1] Broadly speaking, the algorithm of [1] works via Higman linearization ([8] [6] [7]) and reduces the problem to linear matrix factorization which turns out to have a randomized polynomial-time algorithm over finite fields.

*Institute of Mathematical Sciences, Chennai, India and Chennai Mathematical Institute, Siruseri, Kelambakkam, India, email: arvind@imsc.res.in

†Vishwakarma Institute of Technology, Pune, India, email: joglekar.pushkar@gmail.com

[1]Factorization of *homogeneous* noncommutative polynomials is easier as it can be reduced to factorization of a special case of commutative polynomials. See [4] for details.

ISSN 1433-8092

Problem 1.1 (Linear Matrix Factorization Problem)**.** *The linear matrix factorization problem over a field $\mathbb{F}$ takes as input a linear matrix: $L = A_0 + \sum_{i=1}^{n} A_i x_i$, where the A_i are $d \times d$ scalar matrices (over $\mathbb{F}$), the $x_i, 1 \le i \le n$ are noncommuting variables, and A_0 is assumed invertible for technical reasons. The problem is to compute a factorization of L as a product of irreducible linear matrices.*

The study of matrix factorization (linear matrix factorization, in particular) is an important part of Cohn's factorization theory over general free ideal rings. [6] [5].

Coming back to the polynomial factorization algorithm described in [1], the algorithm reduces polynomial factorization to linear matrix factorization which is, in turn, reducible to the problem of computing a common invariant subspace for a collection of n matrices. The common invariant subspace problem can be efficiently solved in the case of finite fields using Ronyai's algorithm [12] which is based on the Artin-Wedderburn theorem for decomposition of algebras. This approach, however, runs into serious difficulties when $\mathbb{F}$ is the field of rationals. The main difficulty is that given a simple matrix algebra[2] $\mathcal{A}$ over rationals, we do not know an efficient algorithm for finding out if $\mathcal{A}$ is a division algebra or whether it has zero divisors. This is one of our motivations for obtaining a reduction from multivariate polynomial factorization to bivariate factorization. Because Higman Linearization of a bivariate noncommutative polynomial given by a formula will yield a bivariate linear matrix. One could hope that factorization of a bivariate linear matrix is computationally easier than factorization of an n-variate linear matrix. Unfortunately, this is not the case. As we will see, even for 4-dimensional bivariate linear matrices the problem of factorization is at least as hard as factoring square-free integers.

Multivariate to Bivariate We start with some formal preliminaries. Let $\mathbb{F}$ be any field and $X = \{x_1, x_2, \ldots, x_n\}$ be a set of n free noncommuting variables. Let X^* denote the set of all free words (which are monomials) over the alphabet X with concatenation of words as the monoid operation and the empty word ϵ as identity element.

The *free noncommutative ring* $\mathbb{F}\langle X\rangle$ consists of all finite $\mathbb{F}$-linear combinations of monomials in X^*, where the ring addition $+$ is coefficient-wise addition and the ring multiplication $*$ is the usual convolution product. More precisely, let $f, g \in \mathbb{F}\langle X\rangle$ and let $f(m) \in \mathbb{F}$ denote the coefficient of monomial m in polynomial f. Then we can write $f = \sum_m f(m)m$ and $g = \sum_m g(m)m$, and in the product polynomial fg for each monomial m we have

$$fg(m) = \sum_{m_1 m_2 = m} f(m_1)g(m_2).$$

The *degree* of a monomial $m \in X^*$ is the length of the monomial m, and the degree $\deg f$ of a polynomial $f \in \mathbb{F}\langle X\rangle$ is the degree of a largest degree monomial in f with nonzero coefficient. For polynomials $f, g \in \mathbb{F}\langle X\rangle$ we clearly have $\deg(fg) = \deg f + \deg g$.

A *nontrivial factorization* of a polynomial $f \in \mathbb{F}\langle X\rangle$ is an expression of f as a product $f = gh$ of polynomials $g, h \in \mathbb{F}\langle X\rangle$ such that $\deg g > 0$ and $\deg h > 0$. A polynomial $f \in \mathbb{F}\langle X\rangle$ is *irreducible* if it has no nontrivial factorization and is *reducible* otherwise. For instance, all degree 1 polynomials in $\mathbb{F}\langle X\rangle$ are irreducible. Clearly, by repeated factorization every polynomial in $\mathbb{F}\langle X\rangle$ can be expressed as a product of irreducibles.

The problem of noncommutative polynomial identity testing (PIT) for multivariate polynomials is known to easily reduce to noncommutative PIT for bivariate polynomials: the reduction is given

[2] i.e. the algebra has no nontrivial two-sided ideals.

by the substitution

$$x_i \to xy^i, 1 \le i \le n,$$

which transforms a given arithmetic circuit (or formula or algebraic branching program) computing a polynomial $f(x_1, x_2, \ldots, x_n)$ to the bivariate polynomial $g(x, y) = f(xy, xy^2, \ldots, xy^n)$. As this substitution map ensures that every monomial of f is mapped to a distinct monomial of $g(x, y)$, it easily follows that f is the zero polynomial if and only if $g(x, y)$ is the zero polynomial.

Indeed, it can be shown [6, Excercise 2.5, Problem 14] that this substitution map give an injective homomorphism from the ring $\mathbb{F}\langle x_1, x_2, \ldots, x_n\rangle$ to $\mathbb{F}\langle x, y\rangle$. Unfortunately, this map does not preserve factorizations. For example, the polynomial $f = x_3x_1 + x_4x_2 + x_4x_1 + x_5x_2 \in \mathbb{F}\langle X\rangle$ is clearly irreducible but image of f under the above map non trivially factorizes as $(xy^2 + xy^3)(yxy + y^2xy^2)$. Thus, we cannot use this substitution map to obtain a reduction from noncommutative multivariate polynomial factorization to bivariate polynomial factorization.

Bergman's 1-inert embedding However, based on a theorem of Bergman [5, Chapter 4], we can obtain a polynomial-time reduction from factorization of multivariate noncommutative polynomials in $\mathbb{F}\langle x_1, x_2, \ldots, x_n\rangle$ given by arithmetic circuits (respect. noncommutative algebraic branching programs(ABP)) to factorization of bivariate noncommutative polynomials in $\mathbb{F}\langle x, y\rangle$, again given by arithmetic circuit (respect. an ABP). This reduction is polynomial-time bounded for both finite fields and rationals. In the case of rationals we need to ensure that the bit complexities of all numbers involved are polynomially bounded. Furthermore, we show that essentially the same reduction works in the black-box setting as well.

More precisely, Bergman's theorem [5, Chapter 4, Theorem 5.2] shows a 1-inert embedding of free algebras of countable rank into free algebras of rank 2. The property of 1-inertness of the embedding map is defined below. We restrict the definition to free noncommutative polynomial rings.

Definition 1.2 (1-inert embedding)**.** [5] *Let $X = \{x_1, x_2, \ldots\}$ be a countably infinite set of free noncommuting variables and $Y = \{x, y\}$ be two free noncommuting variables. A* 1-inert embedding *of $\mathbb{F}\langle X\rangle$ into $\mathbb{F}\langle Y\rangle$ is an injective homomorphism $\varphi : \mathbb{F}\langle X\rangle \to \mathbb{F}\langle Y\rangle$ such that for each polynomial $f \in \mathbb{F}\langle X\rangle$, if its image $\varphi(f)$ factorizes nontrivially in $\mathbb{F}\langle Y\rangle$ as*

$$\varphi(f) = g_1 \cdot g_2,$$

then their preimages $\varphi^{-1}(g_1)$ and $\varphi^{-1}(g_2)$ exist and, since φ is a homomorphism, it gives a nontrivial factorization $f = \varphi^{-1}(g_1)\varphi^{-1}(g_2)$ of f in the ring $\mathbb{F}\langle X\rangle$.

Remark 1.3. *The above definition implies that for all factorizations $\varphi(f) = g_1g_2$, the polynomials g_1 and g_2 are in the range of φ. We note that Cohn's work [6, 5] treats 1-inert embeddings $\varphi : R_1 \to R_2$ for general noncommutative integral domains R_1 and R_2. In the general case, the definition only requires that there is some* unit *$u \in R_2$ such that g_1u and $u^{-1}g_2$ are in the range of the map φ which means that the factorization $\varphi(f) = (g_1u)(u^{-1}g_2)$ holds in the range of φ. As the only units in $\mathbb{F}\langle X\rangle$ are elements of $\mathbb{F}$, for a polynomial $f \in \mathbb{F}\langle X\rangle$, the factors of its image $\varphi(f)$ are all in the range $\varphi(\mathbb{F}\langle X\rangle)$ of the 1-inert embedding φ.*

Definition 1.4. *A* complete factorization *of noncommutative polynomial $f \in \mathbb{F}\langle X\rangle$ is a factorization $f = f_1 \cdot f_2 \cdots f_r$ into a product of irreducible polynomials $f_i \in \mathbb{F}\langle X\rangle$.*

Given an algebraic branching program (respec. Arithmetic Circuit) for f, we can efficiently obtain an algebraic branching program (respec. Arithmetic Circuit) for $\varphi(f)$ and then we use idea of running a substitution automata on ABPs or circuits (see e.g. [4], [2], [3]) to construct a complete factorization of f given a complete factorization of $\varphi(f)$. In the next section we will elaborate and expand upon Bergman's embedding theorem [5] and show how to get its effective algorithmic version which is useful for our purpose of reconstruction of factors of f from factors of $\varphi(f)$.

2 Bergman's embedding

We define a total ordering $\prec$ on monomials in $\{x, y\}^*$ as follows.

Definition 2.1. *For $m_1, m_2 \in \{x, y\}^*$, $m_1 \neq m_2$, we say $m_1 \prec m_2$ if one of the following holds:*

- $m_1 = m_2$.
- $\deg(m_1) < \deg(m_2)$.
- $\deg(m_1) = \deg(m_2)$ *and if $m_1 \neq m_2$ then the leftmost position i where they differ we have $m_1[i] = y$ and $m_2[i] = x$.*

The above ordering is just the usual lexicographic ordering on binary strings treating y as 0 and x as 1. For any polynomial g, let $\text{Mon}(g)$ denotes set of all monomials of g with non-zero coefficient.

When $m_1 \prec m_2$ we say that monomial m_1 is *smaller* than monomial m_2. Equivalently, m_2 is *larger* than m_1. The *leading monomial* of a polynomial $g \in \mathbb{F}\langle x, y\rangle$ is the monomial $m \in \text{Mon}(g)$ (denoted by $\text{lm}(g)$) such that $w \prec m$ for all $w \in \text{Mon}(g)$. That is, the leading monomial of g is the largest monomial in $\text{Mon}(g)$.

Definition 2.2. *For a monomial $m \in \{x, y\}^*$ let $d_x(m)$ (respectively, $d_y(m)$) denote the number of occurrences of x (respectively, y) in m. The* imbalance *$i(m)$ of the monomial m is defined as*

$$i(m) = d_x(m) - d_y(m).$$

Let B be the algebra of all polynomials such that every monomial of the polynomials has imbalance 0.

Let T be set of all monomials m such that either $m = \epsilon$ or $i(m) = 0$ and for any prefix m' of m with $m' \neq \epsilon$, $m' \neq m$, $i(m') > 0$. We call these monomials as *minimally balanced monomials*. Clearly in all the non-empty monomials in T the leftmost symbol is x. We arrange the non-empty monomials in T in lexicographic ordering $\prec$ and the i^{th} monomial in the sequence is denoted by u_i. Let $\overline{u_i}$ is a monomial obtained by replacing every occurrence of x by y and y by x in u_i. Let $\overline{T} = \{\overline{u_i} \mid i \geq 1\}$. It is clear that the monomials in T and $\overline{T}$ together generate the algebra B.

Let C be an algebra generated by $\{u_i + \overline{u_i} \mid i \geq 1\}$. Clearly C is a subalgebra of B.

As each u_i and $\overline{u_i}$ is balanced, clearly the imbalance of any monomial of a polynomial in C is 0. Let $X = \{x_1, x_2, \ldots, x_n\}$ and $X_\infty = \{x_1, x_2, \ldots\}$ be a countably infinite set of indeterminates.

We observe a crucial property of polynomials in $B \setminus C$.

Lemma 2.3.

- *The leading monomial m of any polynomial in C has the form $m = u_{i_1} u_{i_2} \cdots u_{i_\ell}$, where each $u_{i_j} \in T$. That is, m does not have as subword any $\overline{u} \in \overline{T}$.*

- *Every polynomial $f \in B \setminus C$ can be expressed as $f = g + h$ for $g \in C$ and $h \in B$. Moreover, if $h \neq 0$ then the leading monomial of h has some $\overline{u} \in \overline{T}$ as subword.*

Proof. By definition of algebra B (and hence C), for $g \in B$ and any monomial $m \in \mathrm{supp}(g)$ we have $m \in (T \cup \overline{T})^*$. Moreover, each $g \in C$ is an linear combination of products of the form $\prod_{k=1}^{\ell}(u_{i_k} + \overline{u_{i_k}})$. Hence, if $\deg(g) = d$ and $\mathrm{supp}(g)$ contains a degree-d monomial $g_{j_1} g_{j_2} \dots g_{j_\ell}$, where $g_{j_k} \in \{u_{j_k}, \overline{u_{j_k}}\}$ for $k \in [\ell]$, then $\mathrm{supp}(g)$ also contains the degree-d monomial $u_{j_1} u_{j_2} \dots u_{j_\ell}$. By the definition $\prec$, the monomial $u_{j_1} u_{j_2} \dots u_{j_\ell}$ is larger than (with respect to ordering $\prec$) all the monomials with some $\overline{u} \in \overline{T}$ as a subword. Therefore, the leading monomial of any polynomial $g \in C$ has the form claimed.

Next, let $f \in B \setminus C$. If the leading monomial of f has a subword $\overline{u} \in \overline{T}$ then the claim follows as $f = 0 + f$ and $0 \in C$. Suppose the leading monomial of f is $m = u_{j_1} u_{j_2} \cdots u_{j_\ell}$, $u_{j_k} \in T$ for all k. If coefficient of m in f is α, Let

$$f_1 = f - \alpha(u_{j_1} + \overline{u_{j_1}})(u_{j_2} + \overline{u_{j_2}}) \dots (u_{j_\ell} + \overline{u_{j_\ell}}). \tag{1}$$

If m_1 is the leading monomial of f_1 then clearly $m_1 \prec m$. Furthermore, $f_1 \in B \setminus C$ as $f - f_1 \in C$. Hence, it suffices to show $f_1 = g_1 + h_1$ for some $g_1 \in C$ and $h_1 \in B$ with the claimed property. We can apply the subtraction step of Equation 1 to f_1 to obtain f_2 and so on, where in the i^{th} step we obtain a new polynomial f_i whose leading monomial is smaller than the leading monomials of f and each $f_j, j < i$. Since the $\prec$-ordering is a well-ordering on monomials, this process will terminate giving us the desired expression of f as $f = g + h$ where $g \in C$ and the leading term of h has a subword $\overline{u}$ for some $u \in T$. This proves the second part. □

Lemma 2.4. *There is an injective homomorphism (i.e. a homomorphic embedding) from the ring $\mathbb{F}\langle X_\infty \rangle$ to $\mathbb{F}\langle x, y \rangle$.*

Proof. Consider the function $\varphi : \mathbb{F}\langle X_\infty \rangle \mapsto \mathbb{F}\langle x, y \rangle$ defined as follows:

- Let $\varphi(x_i) = u_i + \overline{u_i}$ for all $x_i \in X_\infty$.
- Extend φ to all monomials by multiplication. That is, $\varphi(x_{i_1} x_{i_2} \dots x_{i_k}) = \prod_{j=1}^{k} \varphi(x_{i_j})$.
- Further, extend φ to the ring $\mathbb{F}\langle X_\infty \rangle$ by linearity: $\varphi(\sum_{i=1}^{t} \alpha_i m_i) = \sum_{i=1}^{t} \alpha_i \varphi(m_i)$, for monomials $m_i \in X_\infty^*$ and scalars $\alpha_i \in \mathbb{F}$ for $i = 1$ to t.

To see that φ is a homomorphism, we first note that, by linearity, we have $\varphi(f+g) = \varphi(f) + \varphi(g)$ for $f, g \in \mathbb{F}\langle X_\infty \rangle$. To verify that $\varphi(fg) = \varphi(f)\varphi(g)$, let $f = \sum_m f_m m$ and $g = \sum_m g_m m$ where $f_m, g_m \in \mathbb{F}$ are the coefficients of monomial m in f and g, respectively. Then

$$\begin{aligned}
\varphi(fg) &= \varphi\left((\sum_m f_m m)(\sum_w g_w w) \right) \\
&= \varphi\left(\sum_{m,w} f_m g_w m w \right) \\
&= \sum_{m,w} f_m g_w \varphi(mw) \quad \text{(by linearity of } \varphi) \\
&= \left(\sum_m f_m \varphi(m) \right) \left(\sum_w f_w \varphi(w) \right) \\
&= \varphi(f)\varphi(g).
\end{aligned}$$

In order to show φ is injective, it suffices to show $\varphi(f) \neq 0$ for $f \neq 0$. Suppose $m \in \mathrm{supp}(f)$. Then we note that $\varphi(m) \neq 0$ by the definition of φ. Hence, if m is the only monomial in $\mathrm{supp}(f)$ it follows that $\varphi(f) \neq 0$.

Otherwise, let $m' \in \mathrm{supp}(f)$ and $m' \neq m$. Let u be largest common prefix of m and m'. Then

$$m = ux_iv \text{ and } m' = ux_jw,$$

for monomials $u, v, w \in X_\infty^*$ and $x_i \neq x_j$. Noting that $\varphi(x_i) = u_i + \overline{u}_i$ and $\varphi(x_j) = u_j + \overline{u}_j$ we have

$$\varphi(m) = \varphi(u)(u_i + \overline{u}_i)\varphi(v) \text{ and } \varphi(m') = \varphi(u)(u_j + \overline{u}_j)\varphi(w).$$

From the definition of φ, clearly $\varphi(u)$ is a homogeneous polynomial in $\mathbb{F}\langle x, y\rangle$. Let $\deg(\varphi(u)) = D$. Suppose $\ell = |u_i| = |\overline{u}_i|$ and $\ell' = |u_j| = |\overline{u}_j|$. We can assume without loss of generality that $u_i \prec u_j$. Hence $\ell \leq \ell'$. As u_i and u_j are minimally balanced, u_i cannot be a prefix of u_j. Therefore, for any monomials w_1, w_2 in $\mathrm{supp}(\varphi(m_1))$ and $\mathrm{supp}(\varphi(m_2))$, respectively, w_1 and w_2 will differ in the length ℓ subword starting at location $D + 1$. It follows that $\mathrm{supp}(\varphi(m)) \cap (\varphi(m')) = \emptyset$. Hence, $\varphi(f) \neq 0$ implying that φ is injective. □

We next have an important property about factorization of polynomials in the algebra C. In order to keep our presentation self-contained we give a complete proof with more details than in Cohn's book [5].

Theorem 2.5 (Bergman). *[5, Chapter 4, Theorem 5.2] Let $f \in C$. For any factorization $f = g \cdot h$ the polynomials g and h are in C.*

Proof. First we show that all monomials of g have the same imbalance. Likewise, all monomials of h have the same imbalance. Suppose a_{min} and a_{max} are the minimum and the maximum imbalances of monomials of g. Let b_{min} and b_{max} be the minimum and the maximum imbalance of monomials of h. Let m_{min} be a smallest monomial (with respect to $\prec$) among all monomials of g with imbalance a_{min}, and m_{max} be a largest monomial (with respect to $\prec$) among all the monomials of g with imbalance a_{max}. Let w_{min}, w_{max} be monomials similarly defined for polynomial h corresponding to b_{min} and b_{max}. Now consider the monomial $u = m_{max}w_{max}$. It is non-zero in $f = g.h$ and has imbalance $a_{max} + b_{max}$. Similarly, monomial $v = m_{min}w_{min}$ is non-zero in f and has imbalance $a_{min} + b_{min}$. As $f \in C \subset B$, each monomial of f has imbalance 0. Hence, $a_{max} + b_{max} = 0$ and $a_{min} + b_{min} = 0$. So $a_{max} = -b_{max} \leq -b_{min} = a_{min}$, implying $a_{min} = a_{max} = a$ and $b_{min} = b_{max} = -a$. Thus, all monomials of g have imbalance a and all monomials of h have imbalance $-a$.

Let m be the leading monomial of f. Clearly, m is a maximum degree monomial of f. Moreover, m is largest among the max-degree monomials of f. Let $m = m_1m_2$ with $m_1 \in \mathrm{supp}(g)$ and $m_2 \in \mathrm{supp}(h)$. We have $i(m_1) = a$, $i(m_2) = -a$. As $f \in C$, the monomial $\bar{m}$ obtained by replacing every occurrence of x by y, and y by x in m is also in $\mathrm{supp}(f)$. Moreover, $\bar{m}$ is the smallest monomial among the max-degree monomials of f. This forces that the monomial $\bar{m}_1$ (obtained by interchanging x, y in m_1) is in $\mathrm{supp}(g)$. Similarly, monomial $\bar{m}_2$ (obtained by swapping x, y in m_2) is in $\mathrm{supp}(h)$. We have $i(\bar{m}_1) = -a$ and $i(\bar{m}_2) = a$. Now, all the monomials of g have the same imbalance, and $m_1, \bar{m}_1 \in \mathrm{supp}(g)$. This forces $a = -a = 0$. Consequently, all monomials in $\mathrm{supp}(g) \cup \mathrm{supp}(h)$ have imbalance zero which implies $g, h \in B$.

By Lemma 2.3 applied to g and h we have

1. $g = g_1 + g_2$, $h = h_1 + h_2$, $g_1, h_1 \in C$, $\mathrm{lm}(g_2)$ contains $\bar{u} \in \bar{T}$, and $\mathrm{lm}(h_2)$ contains $\bar{v} \in \bar{T}$,

2. Consequently, the $\deg(g_2)$ prefix of $\mathrm{lm}(g_2 h_1)$ contains the subword $\bar{u}$ and the $\deg(h_2)$ suffix of $\mathrm{lm}(g_1 h_2)$ contains the subword $\bar{v}$.

3. Finally, the $\deg(g_2)$ prefix and the $\deg(h_2)$ suffix of $\mathrm{lm}(g_2 \cdot h_2)$ contains both subwords $\bar{u}$ and $\bar{v}$.

Hence the leading monomials $\mathrm{lm}(g_2 \cdot h_1), \mathrm{lm}(g_1 \cdot h_2)$, and $\mathrm{lm}(g_2 \cdot h_2)$ cannot cancel with each other. As a consequence, the leading monomial of $g_2 \cdot h_1 + g_1 \cdot h_2 + g_2 \cdot h_2$ contains a sub-word from $\bar{T}$ unless both $g_2 = 0$ and $h_2 = 0$. Hence,

$$\begin{aligned} g_2 \cdot h_1 + g_1 \cdot h_2 + g_2 \cdot h_2 &= gh - g_1 \cdot h_1 \in C \text{ and} \\ f &= g \cdot h, g_1, h_1 \in C. \end{aligned}$$

By Lemma 2.3, for any polynomial $\mathfrak{f} \in C$ its leading monomial $\mathrm{lm}(\mathfrak{f})$ cannot have a subword from $\bar{T}$. It forces $g_2 = 0$ and $h_2 = 0$ which implies $g, h \in C$. □

The following theorem, which is a consequence of Theorem 2.5 shows that the embedding φ is a 1-inert embedding (see Definition 1.2). That is, it preserves factorizations.

Theorem 2.6. *Let $f \in \mathbb{F}\langle X \rangle$, where $X = \{x_1, \ldots, x_n\}$. Suppose $f' = \varphi(f) = g' \cdot h'$ is a non-trivial factorization of $\varphi(f)$ in the ring $\mathbb{F}\langle x, y \rangle$. Then there exist polynomials $g, h \in \mathbb{F}\langle X \rangle$, $g, h \notin \mathbb{F}$ such that $g' = \varphi(g)$, $h' = \varphi(h)$ and $f = g \cdot h$.*

Proof. By construction, the homomorphism φ injectively maps $\mathbb{F}\langle X_\infty \rangle$ into $\mathbb{F}\langle x, y \rangle$. As $\mathbb{F}\langle X \rangle \subset \mathbb{F}\langle X_\infty \rangle$, φ maps $f \in \mathbb{F}\langle X \rangle$ to some $f' = \varphi(f) \in C$. Suppose $f' = g' \cdot h'$ is a nontrivial factorization of f' in $\mathbb{F}\langle x, y \rangle$. By Theorem 2.5, as $f' \in C$ both the factors $g', h' \in C$. Since $g' \in C$, it is an $\mathbb{F}$-linear combination of products of the form $(u_{t_1} + \overline{u_{t_1}})(u_{t_2} + \overline{u_{t_2}}) \ldots (u_{t_\ell} + \overline{u_{t_\ell}})$. By definition of φ,

$$(u_{t_1} + \overline{u_{t_1}})(u_{t_2} + \overline{u_{t_2}}) \ldots (u_{t_\ell} + \overline{u_{t_\ell}}) = \varphi(x_{t_1} x_{t_2} \ldots x_{t_\ell}).$$

Hence, by linearity, it follows that $g' = \varphi(g)$ for some nontrivial polynomial $g \in \mathbb{F}\langle X_\infty \rangle$, similarly there is a nontrivial polynomial $h \in \mathbb{F}\langle X_\infty \rangle$ such that $h' = \varphi(h)$. Since φ is a homomorphism, we have

$$\varphi(f) = f' = g' \cdot h' = \varphi(g) \cdot \varphi(h) = \varphi(g \cdot h).$$

As φ is injective, we have $f = g \cdot h$. To complete the proof we need to argue that $g, h \in \mathbb{F}\langle X \rangle$. Let $\mathrm{Var}(g)$ denotes set of variables x_i which appears in some non-zero monomial of g. We want to show that $\mathrm{Var}(g) \subseteq X$. Suppose $\mathrm{Var}(g)$ contains some $x_i \notin X$. Among all monomials of g containing x_i, let m be the largest monomial (under $\prec$-ordering). Then the monomial $m \cdot \mathrm{lm}(h)$ contains the variable x_i and has a non-zero coefficient in $f = gh$. This is a contradiction as $f \in \mathbb{F}\langle X \rangle$ and X does not contain x_i. Hence $\mathrm{Var}(g) \subseteq X$. Similarly, $\mathrm{Var}(h) \subseteq X$. □

3 Multivariate to Bivariate reduction

In this section we will apply Bergman's theorem to show that multivariate noncommutative polynomial factorization is reducible to bivariate noncommutative polynomial factorization in both white-box and black-box.

We first describe some simple tools using which we can obtain an efficient reduction from Bergman's theorem (Theorem 2.5).

Let $X = \{x_1, x_2, \ldots, x_n\}$, and $v_1, v_2, \ldots, v_n$ be any n distinct and minimally balanced monomials in $\{x, y\}^*$. We define function $\varphi : \mathbb{F}\langle X\rangle \to \mathbb{F}\langle x, y\rangle$:

- $\varphi(x_i) = v_i + \overline{v_i}$ for all i.
- φ is extended to monomials by multiplication, i.e. $\varphi(x_{i_1} x_{i_2} \ldots x_{i_k}) = \prod_{j=1}^{k} \varphi(x_{i_j})$.
- φ is extended to $\mathbb{F}\langle X\rangle$ by linearity.

Remark 3.1. *The above definition is essentially like in the proof of Bergman's theorem, except that here X is a finite set of variables and the v_i, $1 \le i \le n$ are any n distinct minimally balanced monomials.*

We can show the following along the same lines as Theorem 2.5 and Theorem 2.6. The straightforward proof is by a suitable renaming of the variables $x_1, \ldots, x_n$ before and after application of Theorem 2.5 in the proof of the Theorem 2.6.

Lemma 3.2. *Let $X = \{x_1, \ldots, x_n\}$, $f \in \mathbb{F}\langle X\rangle$. Suppose $v_1, v_2, \ldots, v_n$ are any distinct minimally balanced monomials in $\{x, y\}^*$. If $f' = \varphi(f) = g' \cdot h'$ is a non-trivial factorization of f' in $\mathbb{F}\langle x, y\rangle$ then there are polynomials $g, h \in \mathbb{F}\langle X\rangle$ such that $g' = \varphi(g)$, $h' = \varphi(h)$ and $f = g \cdot h$.*

In order to obtain polynomial-time computable reduction it is convenient to choose $v_1, v_2, \ldots, v_n$ such that each v_i has the same length. The next lemma ensures that $\ell = O(\log n)$ suffices.

Lemma 3.3. *There are at least n minimally balanced monomials of length 2ℓ in $\{x, y\}^*$ for $\ell \ge max(\lceil \log 4n \rceil, 7)$. Furthermore, the lexicographically first n minimally balanced monomials of length 2ℓ can be computed in time polynomial in n.*

Proof. First we consider the number of minimally balanced monomials of length 2ℓ for $\ell \ge 2$. The first symbol of any minimally balanced monomial is x. If it is more than 2, the second symbol is also x (if it was y, then the balanced monomial xy would be a strict prefix of the minimally balanced monomial, which is a contradiction.) We consider monomials of the form

$$v = xx \cdot w \cdot yy,$$

where w is a Dyck monomial[3]. That is, w is a balanced monomial such that every prefix of w has at most as many y's as x's. Notice that $w \in \{x, y\}^{2\ell-4}$. It follows that any nontrivial prefix of v has strictly more x than y. So any such monomial is minimally balanced of length 2ℓ. The number of Dyck monomials of length $2\ell - 4$ is $C_{\ell-2}$ (the $(\ell-2)^{th}$ Catalan number). A standard estimate yields

$$C_k \sim \frac{4^k}{k^{3/2}\sqrt{\pi}},$$

which implies that C_k is $2^{\Omega(k)}$. Specifically, $C_k > 2^k$ for $k \ge 5$. If $n < 2^{\ell-2}$ and $\ell \ge 7$ then there are at least n minimally balanced monomials of length 2ℓ, for $\ell = \max(\lceil \log 4n \rceil, 7)$.

Clearly, we can compute the v_i, $1 \le i \le n$ by enumeration in poly(n) time. □

[3]Essentially a balanced parenthesis string with x as left and y as right parenthesis, respectively

3.1 White-box reduction

We first describe the reduction in the white-box case for input polynomial $f \in \mathbb{F}\langle X\rangle$ given by a noncommutative arithmetic circuit.

Lemma 3.4. *Let $X = \{x_1, \ldots, x_n\}$ and $f \in \mathbb{F}\langle X\rangle$ be a noncommutative polynomial given by arithmetic circuit C of size s. Then there is a deterministic polynomial time algorithm that outputs an arithmetic circuit computing the polynomial $\varphi(f) \in \mathbb{F}\langle x, y\rangle$, where the minimally balanced monomials $v_i, 1 \le i \le n$ defining the map φ are as described by Lemma 3.3.*

Proof. For $1 \le i \le n$, we note that the sum of two monomials $v_i + \overline{v_i}$ can be computed by a noncommutative arithmetic formula F_i of size $O(\log n)$. Let C' be the arithmetic circuit obtained from circuit C by replacing input variable x_i with the formula F_i. Clearly, C' computes $\varphi(f)$ and its size is polynomially bounded. □

Lemma 3.5. *For $f \in \mathbb{F}\langle X\rangle$ suppose $\varphi(f) = f'_1 \cdot f'_2 \cdots f'_r$ is a complete factorization of $\varphi(f)$ in $\mathbb{F}\langle x, y\rangle$ into irreducible factors $f'_i \in \mathbb{F}\langle x, y\rangle$. Then there are irreducible polynomials $f_1, f_2, \ldots, f_r \in \mathbb{F}\langle X\rangle$ such that $f = f_1 f_2 \ldots f_r$ and $\varphi(f_i) = f'_i$ for each i.*

Proof. It follows by repeated application of Lemma 3.2 that if

$$\varphi(f) = f'_1 \cdot f'_2 \cdots f'_r,$$

is a factorization into irreducible factors $f'_i \in \mathbb{F}\langle x, y\rangle$, then there are polynomials $f_1, f_2, \ldots, f_r \in \mathbb{F}\langle X\rangle$ such that $f = f_1 f_2 \ldots f_r$ and $\varphi(f_i) = f'_i$ for each i. We claim each f_i is irreducible. For, if $f_i = g \cdot h$ is a nontrivial factorization of f_i in $\mathbb{F}\langle X\rangle$ then clearly $f'_i = \varphi(f_i) = \varphi(g)\varphi(h)$ is a nontrivial factorization of f'_i, which contradicts its irreducibility. □

Suppose C'_i is an arithmetic circuit of size s'_i for f'_i for $i \in [r]$. We will construct a circuit of size $poly(s'_i, n)$ for f_i efficiently for each $i \in [r]$, which is the crucial part of our multivariate to bivariate reduction.

The next lemma describes the algorithm crucial to the white-box reduction.

Lemma 3.6. *Given as input a noncommutative arithmetic circuit C for the polynomial $\varphi(g) \in \mathbb{F}\langle x, y\rangle$, where $g \in \mathbb{F}\langle X\rangle$ is a degree d polynomial, $X = \{x_1, x_2, \ldots, x_n\}$, there is a deterministic polynomial-time algorithm, running in time $\mathrm{poly}(d, \mathrm{size}(C), n)$ that computes a noncommutative arithmetic circuit C' for the polynomial g. Furthermore, if $\varphi(g)$ is given by an algebraic branching program then the algorithm computes an algebraic branching program for g.*

Proof. The proof is based on the idea of evaluating a noncommutative arithmetic circuit on an automaton (specifically, a substitution automaton) described in [4] (see e.g., for related applications [2],[3]).

Let $g' = \varphi(g)$. Let $g = \sum_m \alpha_m m$ where $m \in X^*$ and α_m is the coefficient of m in g. As noted before, the map φ has the property that $\mathrm{Mon}(\varphi(m)) \cap \mathrm{Mon}(\varphi(m') =$ for monomials $m \ne m'$ in X^*. Moreover if $m = x_{i_1} x_{i_2} \ldots x_{i_\ell}$ has nonzero coefficient α_m in g then g' has a monomial $m' = v_{i_1} v_{i_2} \ldots v_{i_\ell}$ with coefficient α_m. Hence, to retrieve an arithmetic circuit for g from the given circuit C' for g' our aim is to carry out the following transformation of the polynomial g' given by the circuit C':

- Get rid of the monomials of g' containing of all $\overline{v_j} \in \overline{T}$ for $j \in [n]$.
- For each remaining monomial m' of g' substitute x_i wherever the monomial v_i occurs as substring in m' for $i \in [n]$.

We will accomplish this transformation by evaluating the circuit C' at suitably chosen matrix substitutions $x \leftarrow M_x$ and $y \leftarrow M_y$, where M_x and M_y will be $N \times N$ matrices for polynomially bounded N. The resulting evaluation $C'(M_x, M_y)$ will be be an $N \times N$ matrix. A designated entry of this matrix will contain the polynomial g. Clearly, if we can efficiently compute the claimed matrices M_x and M_y it will yield an arithmetic circuit C for the polynomial g. These matrices M_x and M_y will be obtained as transition matrices of a substitution automaton that will carry out the above transformation steps on the polynomial g'.

We recall substitution automata in the current context. A finite substitution automaton $\mathcal{A}$ is a deterministic finite automata $\mathcal{A}$ along with a substitution map

$$\delta : Q \times \{x, y\} \to Q \times (X \cup \mathbb{F})$$

where Q is a set of states and $X = \{x_1, x_2, \dots, x_n\}$ are noncommuting variables. For $i, j \in Q$, $a \in \{x, y\}$, $u \in X \cup \mathbb{F}$, if $\delta(i, a) = (j, u)$, it means that when automata $\mathcal{A}$ in state i reads a, it replaces a by u and transitions to state j. For each $a \in \{x, y\}$ we can define $|Q| \times |Q|$ transition matrix M_a such that $M_a(i, j) = u$ if $\delta(i, a) = (j, u)$ and 0 otherwise.

With δ we associate projections $\delta_1 : Q \times \{x, y\} \to Q$ and $\delta_2 : Q \times \{x, y\} \to X \cup \mathbb{F}$ defined as $\delta_1(i, a) = j$ and $\delta_2(i, a) = u$ if $\delta(i, a) = (j, u)$. The functions δ_1 and δ_2 extend naturally to monomials: For $w \in \{x, y\}^*$, $\delta_1(i, w) = j$ means the automaton $\mathcal{A}$ goes from state i to j on reading w. Let $\tilde{w}_\ell$ denotes length ℓ prefix of w and w_ℓ denotes ℓ^{th} symbol of w from left. $\delta_2(i, w) = p$ means

$$p = \prod_{\ell=0}^{|w|-1} \delta_2(\delta_1(i, \tilde{w}_\ell), w_{\ell+1}).$$

Note that $\delta_2(i, w)$ has the form $\beta \cdot w'$ where $\beta \in \mathbb{F}, w' \in X^*$. For $\alpha \in \mathbb{F}$ define $\delta_2(i, \alpha \cdot w)$ as $\alpha \cdot \delta_2(i, w)$.

Let $g'(x, y) = \sum_m \alpha_m m \in \mathbb{F}\langle x, y \rangle$. Then, the $(s, t)^{th}$ entry of the $|Q| \times |Q|$ matrix $g'(M_x, M_y)$ is a polynomial $g \in \mathbb{F}\langle X \rangle$ such that

$$g = \sum_{m \in W_t} \alpha_m \delta_2(s, m),$$

where W_t is the set of all monomials that take the automaton $\mathcal{A}$ from state s to state t.

Clearly, if g' has an arithmetic circuit of size s then we can construct an arithmetic circuit of size poly$(s, n, |Q|)$ for g in deterministic time poly$(s, n, |Q|)$.

Turning back to the reduction, consider the input circuit C for $g' = \varphi(g) \in \mathbb{F}\langle x, y \rangle$. We will construct a substitution automaton $\mathcal{A}$ such that the polynomial g is the $(s, t)^{th}$ entry of the matrix $g'(M_x, M_y)$.

Description of the Substitution Automata As already observed each v_i is of the form xxw_iyy, where w_i is a Dyck monomial. Let $v'_i = xw_iy$ for $i \in [n]$. We can easily design a deterministic finite automaton A' with $O(mn)$ states such that the language accepted by A' is precisely the finite set $\{v'_1, v'_2, \dots, v'_n\}$, where m is the length of v_i for $i \in [n]$. Let δ' denote the transition function and Q' be the set of states of A', where q_1 is the initial state and q_{f_i} is the final state associated with acceptance of string v_i' for $i \in [n]$. A' has a tree structure with root q_1 and leaves q_{f_i} for $i \in [n]$, and any root to leaf path has length exactly $2\ell - 2$. We now define the substitution automaton $\mathcal{A}$. Its state set is $Q = Q' \cup \{q_0, q_f, q_r\}$. The transition function $\delta : Q \times \{x, y\} \to Q \times (X \cup \mathbb{F})$ is defined as follows:

1. $\delta(q_0, x) = (q_1, 1); \delta(q_0, y) = (q_r, 0)$.
2. for $q \in Q' \setminus \{q_{f_i} | 1 \le i \le n\}$. and $a \in \{x, y\}$, let $\delta(q, a) = (\delta'(q, a), 1)$.
3. $\delta(q_{f_i}, x) = (q_r, 0)$; $\delta(q_{f_i}, y) = (q_f, x_i)$ for each $i \in [n]$.
4. $\delta(q_f, x) = (q_1, 1)$ and $\delta(q_f, y) = (q_r, 0)$.
5. $\delta(q_r, a) = (q_r, 0)$ for $a \in \{x, y\}$.

The final state of $\mathcal{A}$ is q_f. For a monomial $w \in \{x, y\}^*$, starting at state q_0 the automaton $\mathcal{A}$ substitutes all the variables with 1 as long as it matches with a prefix of v_i for $i \in [n]$ (given by transitions in 1,2 above). When the monomial matches with v_i for some i (which will happen while reading symbol y as each string v_i ends with y), $\mathcal{A}$ substitutes y by x_i and moves to state q_f. If it reads x instead of y then $\mathcal{A}$ enters a rejecting state q_r (given by transition in 3 above). Hence, if $\mathcal{A}$ finds substring v_i in w it replaces it with x_i. Whenever $\mathcal{A}$ is in state q_f, it means the monomial read so far is of the form $v_{i_1} v_{i_2} \dots v_{i_t}$, and it has replaced it with $x_{i_1} x_{i_2} \dots x_{i_t}$. If in the state q_f symbol y is encountered, it means the next substring cannot match with a minimally balanced monomial (as these start with x) and the automaton goes to the rejecting state q_r. If in state q_f variable x is read the automaton goes to state q_1 and restarts the search for a new substring that matches with some v_i (transition in 4 above).

In conclusion $\mathcal{A}$ replaces all the monomials of the form $v_{i_1} v_{i_2} \dots v_{i_t}$ by $x_{i_1} x_{i_2} \dots x_{i_t}$. If the monomial contains an occurrence of $\overline{v_i}$, or it is not of the form $v_{i_1} v_{i_2} \dots v_{i_t}$, then $\mathcal{A}$ zeros out that monomial by suitably setting an occurrence of y to zero or enters the reject state q_r.[4]

It follows that the $(q_0, q_f)^{th}$ entry of the $|Q| \times |Q|$ matrix $g'(M_x, M_y)$ is the polynomial g, where $g' = \varphi(g)$, and M_x, M_y are the transition matrices for the substitution automaton $\mathcal{A}$. This completes the proof.

Finally, if $\varphi(g)$ is given by an algebraic branching program P then it is easy to see that the above construction with the substitution automaton $\mathcal{A}$ yields $P(M_x, M_y)$ which is an algebraic branching program. ◻

The main theorem of this section, stated below, summarizes the discussion in this section.

Theorem 3.7. *In the white-box setting, factorization of multivariate noncommutative polynomials into irreducible factors is deterministic polynomial-time reducible to factorization of bivariate noncommutative polynomials into irreducible factors. More precisely, given as input $f \in \mathbb{F}\langle X \rangle$ by an arithmetic circuit (respectively, algebraic branching program), the problem of computing a complete factorization $f = f_1 \cdot f_2 \cdots f_r$ where each f_i is output as an arithmetic circuit (resp. algebraic branching program) is deterministic polynomial-time reducible to the same problem for bivariate polynomials in $\mathbb{F}\langle x, y \rangle$.*

Proof. We describe the reduction:

[4] We can dispense with the reject state q_r, as suitably setting an occurrence of y to 0 would also suffice. We have transitions to the reject state q_r for exposition.

1. Input $f \in \mathbb{F}\langle X \rangle$ (as a circuit or ABP).
2. Transform f to $f' = \varphi(f) \in \mathbb{F}\langle x, y \rangle$ as a circuit (resp. ABP) by the algorithm of Lemma 3.3.
3. Compute a complete factorization of $f' = f'_1 \cdot f'_2 \cdots f'_r$, where each $f'_i \in \mathbb{F}\langle x, y \rangle$ is irreducible and is computed as a circuit (resp. ABP).
4. Apply the algorithm of Lemma 3.6 to obtain a complete factorization of $f = f_1 \cdot f_2 \cdots f_r$, where each f_i is irreducible and is output as circuit (resp. ABP).

The correctness of the reduction and its polynomial time bound follow from Lemmas 3.2, 3.3 and 3.6. □

Remark 3.8. *We note that in the case $\mathbb{F}$ is the field $\mathbb{Q}$ (of rationals), we need to take into account the bit complexity of the rational numbers involved and argue that the reduction is still polynomial time computable. The main point to note here is that the reduction guarantees the size of the factor f_i is polynomially bounded in the size of $g_i, 1 \le i \le r$, where the size of g_i includes the sizes of any rational numbers that might be involved in the description of the arithmetic circuit (or ABP) for g_i.*

Remark 3.9. *We note here that the ring $\mathbb{F}\langle X \rangle$ is not a unique factorization domain. That is, a polynomial $f \in \mathbb{F}\langle X \rangle$ may have, in general, multiple factorizations into irreducibles [6]. A standard example is the polynomial $x + xyx$ which factorizes as $x(1 + yx)$ as well as $(1 + xy)x$, where $x, y, 1 + yx, 1 + xy$ are irreducible. As the map φ is an injective homomorphism, there is a 1-1 correspondence between factorizations of $\varphi(f)$ and factorizations of f. More specifically, our reduction takes as input any complete factorization $\varphi(f) = f'_1 f'_2 \ldots f'_r$ and computes the corresponding complete factorization $f = f_1 f_2 \ldots f_r$ of f.*

Remark 3.10. *We note that the embedding φ does not preserve sparsity[5] of the polynomial f. More precisely, if the sparsity of the n-variate degree d polynomial f is s then the sparsity of the bivariate polynomial $\varphi(f)$ is $O(2^d s)$. Thus, using this embedding map we do not get a reduction from sparse n-variate degree d polynomial factorization to sparse bivariate polynomial factorization, where s, d are allowed to be part of the running time. This problem remains unanswered.*

3.2 Black-box reduction

The reduction in the black-box case is essentially identical. The only point to note, which is easy to see, is the that analogue of Lemma 3.6 holds in the black-box setting. We state that below. We recall what a black-box means in the noncommutative setting.

Definition 3.11. *A noncommutative polynomial $f \in \mathbb{F}\langle X \rangle$ given by* black-box *essentially means we can evaluate f at any matrix substitution $x_i \leftarrow M_i$, $M_i \in \mathbb{F}^{N \times N}$, where the cost of each evaluation is the matrix dimension N.*

In the black-box setting, suppose we have an efficient algorithm for bivariate noncommutative polynomial factorization of degree D polynomials $g \in \mathbb{F}\langle x, y \rangle$, where the algorithm takes a black-box for g and outputs black-boxes for the irreducible factors of some factorization of g in time poly(D). Then, given a black-box for a degree D n-variate polynomial $f \in \mathbb{F}\langle X \rangle$ as input, we require that the reduction transforms it into a black-box of a bivariate polynomial $g \in \mathbb{F}\langle x, y \rangle$, and from the output black-boxes of g's irreducible factors, the reduction has to efficiently recover black-boxes for the corresponding irreducible factors of f.

[5]The sparsity of a polynomial f is the number of monomials in Mon(f).

Lemma 3.12. *Given as input a black-box for the polynomial $\varphi(g) \in \mathbb{F}\{x, y\}$, where $g \in \mathbb{F}\langle X\rangle$ is a degree d polynomial, $X = \{x_1, x_2, \ldots, x_n\}$, with matrix substitutions for x and y computed in deterministic polynomial-time time we can obtain a black-box for the polynomial $g \in \mathbb{F}\langle X\rangle$.*

Proof. The proof of Lemma 3.6 already implies this because the matrices M_x and M_y described there do not require $\varphi(g)$ to be given in white-box as circuit or ABP. Thus, the black-box for $\varphi(g)$ yields a black-box for g by accessing the $(q_0, q_f)^{th}$ entry of the matrix output $\varphi(g)(M_x, M_y)$. □

As a consequence we obtain the claimed reduction from multivariate factorization to bivariate factorization in the black-box setting as well.

Theorem 3.13. *The problem of computing a complete factorization of $f \in \mathbb{F}\langle X\rangle$ given by black-box is deterministic polynomial-time reducible to the problem of black-box computation of a complete factorization of polynomials in $\mathbb{F}\langle x, y\rangle$.*

Proof. Given a black-box for f we obtain a black-box for $\varphi(f)$ applying Lemma 3.3. Then, given a complete factorization

$$\varphi(f) = f_1' \cdot f_2' \cdots f_r',$$

where each factor f_i' is output by a black-box for it, by Lemma 3.12 we can obtain black-boxes for each f_i. This yields a complete factorization $f = f_1 \cdot f_2 \cdots f_r$ of f where the factors are given by black-box. □

4 Factorizing 4×4 linear matrices over $\mathbb{Q}$

We have shown in Section 3 that multivariate noncommutative polynomial factorization is efficiently reducible to the bivariate case. Suppose $f \in \mathbb{F}\langle x, y\rangle$ is a bivariate polynomial given by a formula of size s. Applying Higman linearization [6], as done in [1], we can transform the problem to factorization of bivariate linear matrices $A_0 + A_1x + A_2y$, where the matrices have size bounded by $2s$. In [1] the problem of factorizing an n-variate polynomial $f \in \mathbb{F}\langle X\rangle$ given by a formula was solved in two steps when $\mathbb{F}$ is a finite field: (i) Transform f to a linear matrix L and factorize L into irreducible factors by reducing it to the common invariant subspace problem, and (ii) extract the factors of f from the factors of L. This approach does not work for $\mathbb{F} = \mathbb{Q}$ because the common invariant subspace problem for matrices over $\mathbb{Q}$ is shown by Ronyai [12] to be at least as hard as factoring square-free integers.

In this section we show that even for 4×4 bivariate linear matrices factorization remains at least as hard as factoring square-free integers. Thus, efficient polynomial factorization over $\mathbb{Q}$ remains elusive even for bivariate polynomials. The proof is based on Ronyai's aforementioned result.

Definition 4.1 (generalized quaternion algebra)**.** *Let $\alpha, \beta \in Q$ be nonzero rationals. The generalized quaternion algebra $H(\alpha, \beta)$ is the 4-dimensional algebra over $\mathbb{Q}$ generated by elements $1, u, v, uv$ where the rules for multiplication in $H(\alpha, \beta)$ are given by $u^2 = \alpha$, $v^2 = \beta$, and $uv = -vu$.*

A *simple* algebra $\mathcal{A}$ over a field $\mathbb{F}$ is an algebra that has no nontrivial two-sided ideal. The *center* C of algebra $\mathcal{A}$ is the subalgebra consisting of all elements of $\mathcal{A}$ that commute with every element of $\mathcal{A}$.

Fact 4.2. *For any nonzero $\alpha, \beta \in \mathbb{Q}$, the algebra $H(\alpha, \beta)$ is a simple algebra with center $\mathbb{Q}$.*

Furthermore, it follows from general theory [11, Chapter 1.6] that

Fact 4.3. *The algebra $H(\alpha,\beta)$ is either a division algebra (which means no zero divisors in it) or is isomorphic to the algebra of 2×2 matrices over $\mathbb{Q}$ (which means it has zero divisors).*

The 4-dimensional algebra $H(\alpha,\beta)$ can be represented as an algebra of 4×4 matrices over $\mathbb{Q}$, which is the *regular representation*. It is easy to see that the matrix corresponding to 1 is I_4, and the matrices M_u and M_v corresponding to u and v are

$$M_u = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \alpha & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \alpha & 0 \end{bmatrix} \tag{2}$$

$$M_v = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \\ \beta & 0 & 0 & 0 \\ 0 & -\beta & 0 & 0 \end{bmatrix} \tag{3}$$

We next observe that factorizing 4×4 bivariate linear matrices is at least as hard as finding zero divisors in generalized quaternion algebras.

Theorem 4.4. *Finding zero divisors in an input quaternion algebra $H(\alpha,\beta)$ is polynomial-time reducible to factorizing 4×4 bivariate linear matrices $A_0 + A_1x + A_2y$, where each scalar matrix A_i is in $\mathcal{M}_4(\mathbb{Q})$.*

Proof. Let $H(\alpha,\beta)$ be the given generalized quaternion algebra. Then

$$H(\alpha,\beta) = \{a_o + a_1u + a_2v + a_3uv \mid a_i \in \mathbb{Q}\},$$

where $u^2 = \alpha$, $v^2 = \beta$, and $uv = -vu$ defines the algebra multiplication.

It is well-known (see e.g. Pierce's book [11, Chapter 1.6]) that the algebra $H(\alpha,\beta)$ is *simple* (that is, it has no nontrivial 2-sided ideals) with center $\mathbb{Q}$. Furthermore, it is either a *division algebra* (which means there are no zero divisors in it) or it is isomorphic to the algebra $\mathcal{M}_2(\mathbb{Q})$ of 2×2 matrices over $\mathbb{Q}$ (which has zero divisors).

We now consider factorizations of the 4×4 linear matrix $I_4 + M_ux + M_vy$, where matrices M_u and M_v are defined in Equations 2 and 3.

Claim. *The linear matrix $I_4 + M_ux + M_vy$ is irreducible if and only if the quaternion algebra is a division algebra.*

Proof of Claim. Suppose the linear matrix $L = I_4 + M_ux + M_vy$ has a nontrivial factorization

$$L = I_4 + M_ux + M_vy = FG.$$

That means neither F nor G is a scalar matrix. By a theorem of Cohn [6, Theorem 5.8.8], there are invertible scalar matrices P and Q in $\mathcal{M}_4(\mathbb{Q})$ such that

$$PLQ = \begin{bmatrix} A & 0 \\ D & B \end{bmatrix}. \tag{4}$$

Remark 4.5. *To apply Cohn's theorem we need to have matrix L to be monic (that is the matrix* $[M_u \mid M_v]$ *has full row rank and the matrix* $[M_u^T \mid M_v^T]^T$ *has full column rank). The monicity is ensured for L as matrices* M_u *and* M_v *are full rank matrices.*

Putting $x = y = 0$ we observe that

$$PQ = \begin{bmatrix} A_0 & 0 \\ D_0 & B_0 \end{bmatrix},$$

where A_0, B_0 and D_0 are scalar matrices. As P and Q are invertible, it following that both A_0 and B_0 are invertible matrices. Hence we have

$$PLP^{-1} = \begin{bmatrix} A & 0 \\ D & B \end{bmatrix} \cdot \begin{bmatrix} A_0 & 0 \\ D_0 & B_0 \end{bmatrix}^{-1} = \begin{bmatrix} A' & 0 \\ D' & B' \end{bmatrix},$$

where A', B' and D' are also linear matrices. We now recall that the matrices I_4, M_u and M_v are the matrix representation of the elements $1, u, v \in H(\alpha, \beta)$ w.r.t. the basis $\{1, u, v, uv\}$ is the basis of $H(\alpha, \beta)$. Treating P as a basis change matrix, the above equation yields a new basis $\{w_1, w_2, w_3, w_4\}$ of $H(\alpha, \beta)$. Let $\dim(A') = k$. Then $1 \leq \dim(A') \leq 3$ and the vectors $w_1, \ldots, w_k$ spans a k-dimensional subspace $W \subset H(\alpha, \beta)$ that is a common invariant subspace for the matrices I_4, M_u, M_v and M_{uv}. In other words, the subspace W is preserved under left multiplication by u and v. We can assume, without loss of generality, that $w_1 \neq 1$: if $k > 1$ then clearly we can assume this. If $k = 1$ notice that $w_1 = 1$ is impossible because the subspace W is not preserved under left multiplication by u or v. Then the four elements w_1, uw_1, vw_1, uvw_1 are all in W and hence linearly dependent. Thus for some nontrivial linear combination

$$\gamma_0 w_1 + \gamma_1 u w_1 + \gamma_2 v w_1 + \gamma_3 u v w_1 = 0.$$

which means $(\gamma_0 + \gamma_1 u + \gamma_2 v + \gamma_3 uv) \times w_1 = 0$. Hence w_1 is a zero divisor in $H(\alpha, \beta)$.

Conversely, if $z \in H(\alpha, \beta)$ is a zero divisor then the we can see that the left ideal

$$J = \{xz \mid x \in H(\alpha, \beta)\}$$

is a proper subspace of $H(\alpha, \beta)$ that is invariant under M_u and M_v. Then, applying Cohn's theorem [6, Theorem 5.8.8], we can obtain invertible scalar matrices P and Q such that Equation 4 holds which yields the factorization

$$PLQ = \begin{bmatrix} A & 0 \\ D & B \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & I \end{bmatrix} \cdot \begin{bmatrix} I & 0 \\ D & I \end{bmatrix} \cdot \begin{bmatrix} I & 0 \\ 0 & B \end{bmatrix}.$$

■

To complete the reduction, notice that if $I_4 + M_u x + M_v y$ is irreducible then $H(\alpha, \beta)$ is a division algebra. On the other hand, if we are given a nontrivial factorization $I_4 + M_u x + M_v y = FG$ then, analyzing the proof of Cohn's theorem [6, Theorem 5.8.8] (also see [1] for details), by suitable row and column operations we can compute in polynomial time the invertible scalar matrices P and Q from the factors F and G. Hence, by the proof of the above claim, we can efficiently compute a zero divisor w_1 in $H(\alpha, \beta)$. □

As finding zero-divisors in the quaternion algebra $H(\alpha, \beta)$ is known to be at least as hard as square-free integer factorization [12] we have the following.

Corollary 4.6. *Factorizing* 4×4 *bivariate linear matrices over* $\mathbb{Q}$ *is at least as hard as factorizing square-free integers.*

5 Factorizing 3×3 linear matrices over $\mathbb{Q}$

In this section we present a deterministic polynomial-time algorithm for factorization of 3×3 multivariate linear matrices over $\mathbb{Q}$. We start with a simple observation about linear matrix factorization in general.

Lemma 5.1. *Suppose $L = I_d + \sum_{i=1}^{n} A_i x_i$ is a linear matrix where each $A_i, 0 \leq i \leq d$ is a $d \times d$ matrix over $\mathbb{Q}$. Then L is irreducible if the characteristic polynomial of A_i is irreducible over $\mathbb{Q}$ for any i.*

Proof. For if L is reducible then there is an invertible scalar matrix P such that

$$PLP^{-1} = \begin{bmatrix} A & 0 \\ D & B \end{bmatrix},$$

which implies that

$$PA_iP^{-1} = \begin{bmatrix} A_i' & 0 \\ D_i' & B_i' \end{bmatrix},$$

for scalar matrices A_i', B_i', and D_i'. Thus, the characteristic polynomial of A_i is the product of the characteristic polynomials of A_i' and B_i' which is a nontrivial factorization. □

Theorem 5.2. *There is a deterministic polynomial-time algorithm for factorization of* 3×3 *multivariate linear matrices over* $\mathbb{Q}$.

Proof. We will first consider linear matrices of the form $L = I_3 + \sum_{i=1}^{n} A_i x_i$, where each $A_i \in \mathcal{M}_3(\mathbb{Q})$ and the x_i are noncommuting variables. The algorithm computes a complete factorization of L into (at most three) irreducible linear matrix factors. By Cohn's theorem [6, Theorem 5.8.8], either L is irreducible or there is an invertible scalar matrix P such that

$$PLP^{-1} = \begin{bmatrix} A & 0 \\ D & B \end{bmatrix}.$$

Either A or B is a 1×1 matrix. If A is a 1×1 matrix then corresponding to it there is a 1-dimensional common invariant subspace spanned by a vector, say v, for the matrices $A_i, 1 \leq i \leq n$. More precisely, the row vector v^T is an eigenvector for each matrix A_i, and $v^T A_i = \lambda_i v^T$ where $\lambda_i \in \mathbb{Q}$ is the corresponding eigenvalue of matrix A_i for each i. Likewise, if B is a 1×1 matrix then there is a corresponding 1-dimensional common invariant subspace spanned by a (column) vector u such that $A_i u = \mu_i u$ for eigenvalues μ_i of A_i. In either case, the common eigenspace is easy to compute from the characteristic polynomial of say A_1 and then verifying that it is an eigenspace for the remaining A_i as well. This will yield the factorization

$$PLP^{-1} = \begin{bmatrix} A & 0 \\ 0 & I \end{bmatrix} \cdot \begin{bmatrix} I & 0 \\ D & I \end{bmatrix} \cdot \begin{bmatrix} I & 0 \\ 0 & B \end{bmatrix},$$

where B is a 2×2 linear matrix. The problem now reduces to factorizing the linear matrix $B = I_2 + \sum_{i=1}^{n} B_i x_i$, where $B_i \in \mathcal{M}_2(\mathbb{Q})$. A simple case analysis discussed below yields a polynomial-time algorithm for factorization of B.

1. If the characteristic polynomial of any B_i is irreducible over $\mathbb{Q}$ then the linear matrix B is clearly irreducible.

2. Some B_i has two distinct eigenvalues $\lambda \neq \lambda' \in \mathbb{Q}$ then the corresponding eigenspaces are 1-dimensional, spanned by their eigenvectors $u \neq u'$. Then either u or u' has to be an eigenvector for every B_j (otherwise B is irreducible), in which case we have a factorization of B.

3. Suppose each B_i has only one eigenvalue λ_i. Then, by linear algebra, after a basis change B_i is either of the form
$$\begin{bmatrix} \lambda_i & 1 \\ 0 & \lambda_i \end{bmatrix}$$
in which case the eigenspace is 1-dimensional with eigenvector $(10)^T$. We can check if this eigenspace is invariant for each B_j or not as before. Otherwise, after basis change each
$$B_i = \begin{bmatrix} \lambda_i & \\ 0 & \lambda_i \end{bmatrix}$$
which means $B_i = \lambda_i I_2$ for each i and the factorization of B is given by
$$B = \begin{bmatrix} 1 + \sum_{i=1}^{n} \lambda_i x_i & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ 0 & 1 + \sum_{i=1}^{n} \lambda_i x_i \end{bmatrix}$$

□

References

[1] Vikraman Arvind and Pushkar S. Joglekar. On efficient noncommutative polynomial factorization via higman linearization. In Shachar Lovett, editor, *37th Computational Complexity Conference, CCC 2022, July 20-23, 2022, Philadelphia, PA, USA*, volume 234 of *LIPIcs*, pages 12:1–12:22. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022.

[2] Vikraman Arvind, Pushkar S. Joglekar, Partha Mukhopadhyay, and S. Raja. Randomized polynomial time identity testing for noncommutative circuits. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 831–841, 2017.

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

[4] Vikraman Arvind, Gaurav Rattan, and Pushkar S. Joglekar. On the complexity of noncommutative polynomial factorization. In *Mathematical Foundations of Computer Science 2015 - 40th International Symposium, MFCS 2015, Milan, Italy, August 24-28, 2015, Proceedings, Part II*, pages 38–49, 2015.

[5] P. M. Cohn. *Free Rings and their Relations*. London Mathematical Society Monographs. Academic Press, 1985.

[6] P. M. Cohn. *Free Ideal Rings and Localization in General Rings*. New Mathematical Monographs. Cambridge University Press, 2006.

[7] Ankit Garg, Leonid Gurvits, Rafael Mendes de Oliveira, and Avi Wigderson. Operator scaling: Theory and applications. *Found. Comput. Math.*, 20(2):223–290, 2020.

[8] Graham Higman. The units of group-rings. *Proceedings of the London Mathematical Society*, s2-46(1):231–248, 1940.

[9] Erich Kaltofen. Factorization of polynomials given by straight-line programs. *Adv. Comput. Res.*, 5:375–412, 1989.

[10] Erich Kaltofen and Barry M. Trager. Computing with polynomials given by black boxes for their evaluations: Greatest common divisors, factorization, separation of numerators and denominators. *J. Symb. Comput.*, 9(3):301–320, 1990.

[11] Richard S. Pierce. *Associative Algebras*. Graduate Texts in Mathematics. Springer, 1982.

[12] Lajos Rónyai. Simple algebras are difficult. In Alfred V. Aho, editor, *Proceedings of the 19th Annual ACM Symposium on Theory of Computing, 1987, New York, New York, USA*, pages 398–408. ACM, 1987.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il