Horner Systems: How to efficiently evaluate non-commutative polynomials (by matrices)

Konrad Schrempf

October 4, 2019

Abstract

By viewing non-commutative polynomials, that is, elements in free associative algebras, in terms of linear representations, we generalize Horner’s rule to the non-commutative (multivariate) setting. We introduce the concept of Horner systems (which has parallels to that of companion matrices), discuss their construction and show how they enable the efficient evaluation of non-commutative polynomials by matrices.

Keywords and 2010 Mathematics Subject Classification. Horner’s rule, free associative algebra, minimal linear representations, admissible linear systems, matrix polynomials, companion matrix, non-commutative factorization; Primary 68W30; Secondary 16Z05, 47A56

Introduction

When we talk about the evaluation of non-commutative (nc) polynomials by matrices, we actually take elements in the free associative algebra, aka “algebra of non-commutative polynomials”, (over a commutative field $\mathbb{K}$, e.g. $\mathbb{Q}$ or $\mathbb{C}$, and an alphabet $X$ with $d$ letters) and view them as functions on $d$-tuples of matrices (of appropriate sizes); the non-commuting letters $x_1, x_2, \ldots, x_d$ (or $x, y, z$ for $d = 3$) are “placeholders” where we plug in matrices $\bar{X}_1, \bar{X}_2, \ldots, \bar{X}_d$ (respectively $\bar{X}, \bar{Y}, \bar{Z}$).

Working symbolically with matrices (without inverse) just means that we add or multiply nc polynomials, that is, use the ring operations in free associative algebras (over an appropriate alphabet); usually in terms of (finite) formal sums of words with coefficients in a commutative field $\mathbb{K}$, for example $(x^2 + \frac{1}{2}xy) - (xy + 2y^2) = x^2 - \frac{1}{2}xy - 2y^2$ or $x \cdot (1 - xy) = x - xyx$.

*Contact: math@versibilitas.at (Konrad Schrempf), https://orcid.org/0000-0001-8509-009X, Universität Wien, Fakultät für Mathematik, Oskar-Morgenstern-Platz 1, 1090 Wien; FH Oberösterreich, Forschungsgruppe ASiC, Ringstraße 43a, 4600 Wels; Austria.
Another —at a first glance much more complicated— way to work with nc polynomials is in terms of linear representations in the sense of Cohn and Reutenauer [CR94]. Here a polynomial $p$ is written as $p = uA^{-1}v$ with $u^T, v \in \mathbb{K}^{n \times 1}$ and upper unitriangular (with ones in the diagonal) $n \times n$ matrix $A$ over linear nc polynomials, for example

$$p = x - xy_2 = \begin{bmatrix} 1 & \ldots & 0 \end{bmatrix} \begin{bmatrix} 1 & -x & -x \\ 1 & y & \\ \vdots & 1 & -x \\ \vdots & \vdots & 1 \end{bmatrix}^{-1} \begin{bmatrix} \vdots \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & \ldots & 0 \end{bmatrix} \begin{bmatrix} 1 & -xy & x - xy_2 \\ 1 & -y & -yx \\ \vdots & 1 & x \\ \vdots & \vdots & 1 \end{bmatrix} \begin{bmatrix} \vdots \\ 1 \end{bmatrix}$$

(zero entries are replaced by lower dots to emphasize the structure). The triple $\pi = (u, A, v)$ is called linear representation of $p$, the size of $A$ dimension. If the dimension is the smallest possible (for $p$), then $\pi$ is called minimal. Addition and multiplication can easily be formulated in terms of linear representations (discussed in detail in Section 1). Furthermore, minimal linear representations can be used to factorize nc polynomials (Section 3), and from another point of view they are the natural generalization of companion matrices (Section 2).

**Remark.** Here we restrict ourselves to the very special case of nc polynomials. Linear representations in the sense of Cohn and Reutenauer go far beyond, namely for elements in a free field [Ami66], that is, the universal field of fractions of a free associative algebra [Coh06, Chapter 7]. For a practical introduction see [Sch18b], for the computation of the left gcd of two nc polynomials [Sch18a, Example 5.4].

In other words: Linear representations are a powerful and universal language in the context of (symbolic) non-commutative rational expressions. For $u = [1, 0, \ldots, 0]$ we call $\pi = (u, A, v)$ an admissible linear system (ALS) for $p$ and write $A = \pi$ also as $As = v$. Then $p$ is the first component of the (unique) solution vector $s$. Evaluating $p$ in terms of an ALS by matrices is immediate: We start with $s_n = v_n$ and compute $s_k$ for $k = n - 1, \ldots, 1$. Thus we do not need to invert $A$ at all.

**Remark.** The term “admissible” means that the system matrix $A$ is invertible, that is, $As = v$ admits a unique solution. In our case $A$ is invertible over the free associative algebra. In general however, $A$ “just” needs to be invertible over the free field. Although this can be ensured by a rather simple algebraic property it goes deep into the heart of Cohn’s theory and is very subtle and difficult to understand. (This is the actual reason to restrict to the special case of nc polynomials.)

So, if we want to evaluate our polynomial $p = p(x, y)$ from before with $m \times m$ matrices $\bar{X}, \bar{Y}$ we have $s_4 = I_m$, $s_3 = \bar{X}s_4 = \bar{X}$, $s_2 = -\bar{Y}s_3 = -\bar{Y}\bar{X}$ and $s_1 = \bar{X}s_2 + \bar{X}s_4 = -\bar{X}\bar{Y}\bar{X} + \bar{X}$. Two matrix-matrix multiplications of complexity $O(m^3)$ are necessary, one to compute $s_2$ and one to compute $p = s_1$. The multiplication is
the dominating part since the addition of matrices has only complexity $O(m^2)$. In this case we did not gain anything by using linear representations since plugging in $\bar{X}, \bar{Y}$ directly into the words $x$ and $xyx$ from $p$ would also “cost” two multiplications.

Remark. We only assume that the multiplication is the dominating part, that is, its complexity is $O(m^{2+\varepsilon})$ for $\varepsilon > 0$; recall that Strassen’s algorithm has $O(m^{2.81})$ [Str69]. And since we are interested in practical applications, (numerical) stability is important. For details and references (including the complexity of the matrix multiplication) we refer to [DDHK07].

Now we take the polynomial $p = 3cyxb + 3xbyxb + 2cyxax + cyxb - cyaxb - 2xbyxax + 4xbyxb - 3xbyxb + 3xaxyxb - 3xbyxax + 6axbyxb + 2xaxyxax + xaxyxb - xaxyxb - 2xbyxax - bxbyxb + bxbyxb + 5axbyxb - 4xbyxb [CHS06, Section 8.2]. (Here $a$, $b$ and $c$ can be viewed as matrix-valued parameters.) If we want to evaluate $p = p(x, y, z; a, b, c)$ by matrices, 97 multiplications are necessary. By rewriting $p$ as “Sylvester mapping” with respect to $y$, that is $p = p_1yq_1 + p_2yq_2 + \ldots + p_kyq_k$, the number of multiplications can be reduced to 28 [CHS06]. However, only $6+2+7 = 15$ multiplications (left part, inner part, right part) are necessary using the “matrix-factorization”

$$\begin{pmatrix} \cdot & c \\ 1 + a & 1 + b & x \end{pmatrix} \begin{pmatrix} x & \cdot & \cdot \\ \cdot & x & \cdot \\ \cdot & \cdot & a \end{pmatrix} \begin{pmatrix} b & \cdot & \cdot \\ \cdot & -b & \cdot \\ \cdot & \cdot & x \end{pmatrix} \begin{pmatrix} y & \cdot & \cdot \\ \cdot & y & \cdot \\ \cdot & 3 + b - a & 2x \end{pmatrix} \begin{pmatrix} a & \cdot & \cdot \\ \cdot & a & \cdot \\ \cdot & \cdot & b \end{pmatrix}.$$

How to find such factorizations is discussed in Section 3 and summarized in Section 4. Factorizations are important steps towards “Horner systems” which are —roughly speaking— the most sparse admissible linear systems (for a given polynomial).

Remark. For Camino, Helton and Skelton [CHS06] the crucial point is to find the Sylvester index [KMP00], that is, the minimal number of “summands” (here it is $k = 2$ with respect to $y$), to solve the generalized Sylvester equation. Using three terms (instead of two) makes a significant difference since no $O(m^3)$ algorithm is known in the general case and the simple approach using tensor product requires $O(m^6)$ [Sim16]. See also [Hig08, Section 7.3].

In Section 1 we give a brief introduction to free associative algebras and set up the necessary formalism to work with linear representations. The main contribution is the concept of Horner systems (and bounds for the number of multiplications in Proposition 2.10) in Section 2. From a practical point of view the minimization of linear representations (which we recall at the end of Section 1) and the factorization into matrices in Section 3 are important since they are the major steps in the construction of Horner systems. And finally, in Section 4, we summarize how to construct Horner systems and state some related literature.

To get a first impression, one can start with Table 1 (page 11). While the number of words —and thus the number of multiplications— can grow exponentially, the number of multiplications using Horner systems is at most quadratic with respect to the rank (Definition 1.2), which is a good “measure” for the complexity of a nc
polynomial. (In the univariate case, the rank is just the degree plus one. Notice however, that the rank function is not a degree function.)

Notation. The set of the natural numbers is denoted by \( \mathbb{N} = \{1, 2, \ldots\} \). Zero entries in matrices are usually replaced by (lower) dots to emphasize the structure of the non-zero entries unless they result from transformations where there were possibly non-zero entries before. We denote by \( I_n \) the identity matrix of size \( n \) respectively \( I \) if the size is clear from the context.

## 1 Free Associative Algebras

After briefly introducing the “algebra of nc polynomials” and the notion of irreducible polynomials (needed for the factorization), we provide a detailed lead-in to the work with linear representations in the context of nc polynomials, accompanied by examples. At the end of this section we summarize the necessary setup and present the algorithm for the minimization. For the factorization we refer to Section 3.

Let \( \mathbb{K} \) be a commutative field (e.g. \( \mathbb{Q}, \mathbb{R} \) or \( \mathbb{C} \)) and \( \mathcal{X} = \{x_1, x_2, \ldots, x_d\} \) be a finite (non-empty) alphabet. The free monoid \( \mathcal{X}^* \) is the set of all finite words \( x_{i_1}x_{i_2} \cdots x_{i_n} \) with \( i_k \in \{1, 2, \ldots, d\} \), for example (for \( \mathcal{X} = \{x, y, z\} \)), \( \mathcal{X}^* = \{1, x, y, z, x^2, xy, xz, yx, y^2, yz, zx, zy, z^2, x^2y, \ldots\} \).

The multiplication on \( \mathcal{X}^* \) is the concatenation, that is, \( (x_{i_1} \cdots x_{i_m}) \cdot (x_{j_1} \cdots x_{j_n}) = x_{i_1} \cdots x_{i_m}x_{j_1} \cdots x_{j_n} \), with neutral element 1, the empty word. The length of a word \( w = x_{i_1}x_{i_2} \cdots x_{i_m} \) is (denoted by) \( |w| = m \). For an introduction see [BR11, Chapter 1].

By \( \mathbb{K}\langle \mathcal{X} \rangle \) we denote the free associative algebra or free \( \mathbb{K} \)-algebra (aka “algebra of nc polynomials”). Its elements can be uniquely expressed in the form \( \sum_{w \in \mathcal{X}^*} \kappa_w w \), \( \kappa_w \in \mathbb{K} \) (only finitely many \( \kappa_w \) are non-zero), that is, by finite formal sums. In the case of \( \mathcal{X} = \{x\} \), the free associative algebra is just the polynomial ring \( \mathbb{K}[x] \). Given two elements \( p = \sum \kappa_w w \) and \( q = \sum \lambda_w w \), the sum and the product are given by

\[
p + q = \sum_{w \in \mathcal{X}^*} (\kappa_w + \lambda_w) w \quad \text{resp.} \quad pq = \sum_{w \in \mathcal{X}^*} \left( \sum_{uv=w} \kappa_u \lambda_v \right) w.
\]

A very rich resource on free associative algebras is [Coh74]. For their role in the theory of formal languages we recommend [Coh75] and [BR11] or [SS78].

For detailed algebraic discussions a lot of definitions (and notations) are necessary. Therefore we formulate most as a special case and refer to [CR94, CR99] for linear representations and [BS15] for the factorization for further information and literature. The factorization in free associative algebras is a natural generalization of that in the (ring of) integers \( \mathbb{Z} \). However, in the non-commutative setting one needs to distinguish between prime elements (for divisibility) and irreducible elements or atoms (for factorization). (And the uniqueness of a factorization into atoms needs a generalization [Coh63].) The number of atoms is unique, for example \( x - xyx = x(1 - yx) = (1 - xy)x \).
**Definition 1.1** (Irreducible Polynomials). A (non-trivial) polynomial \( p \in \mathbb{K}\langle \mathcal{X} \rangle \setminus \mathbb{K} \), that is, a non-zero non-invertible element, is called an *atom* (or *irreducible*) if \( p = q_1 q_2 \) implies that either \( q_1 \in \mathbb{K} \) or \( q_2 \in \mathbb{K} \), that is, one of the factors is invertible. (Invertible elements are also called *units*.)

Now we go over to *linear representations* of elements in free associative algebras and formulate the ring operations (sum and product) and the factorization on that level. There are two main issues we need to take care of:

- Does every polynomial admit a linear representation?
- And, how can we construct *minimal* linear representations?

Both can be addressed in a constructive way. We start with “minimal monomials” (Proposition 1.8), add or multiply them (Proposition 1.9) and minimize (Algorithm 1.12). We illustrate these steps using \( p = x \) and \( q = 1 - xy \) with “manual” minimization to avoid a lot of technical details (necessary for an implementation in computer algebra systems).

**Definition 1.2** (Linear Representations, Dimension, Rank [CR94, CR99]). Let \( p \in \mathbb{K}\langle \mathcal{X} \rangle \). A *linear representation* of \( p \) is a triple \( \pi = (u, A, v) \) with \( u^\top, v \in \mathbb{K}^{n \times 1} \) (for some \( n \in \mathbb{N} \)) and an over \( \mathbb{K}\langle \mathcal{X} \rangle \) invertible \( n \times n \) matrix \( A = (a_{ij}) \) with entries \( a_{ij} = \kappa_{ij}^{(0)} + \kappa_{ij}^{(1)} x_1 + \ldots + \kappa_{ij}^{(d)} x_d \), \( \kappa_{ij}^{(d)} \in \mathbb{K} \), and \( p = u A^{-1} v \). The *dimension* of \( \pi \) is \( \dim (u, A, v) = n \). It is called *minimal* if \( A \) has the smallest possible dimension among all linear representations of \( p \). The “empty” representation \( \pi = (, ,) \) is the minimal one of \( 0 \in \mathbb{K}\langle \mathcal{X} \rangle \) with \( \dim \pi = 0 \). Let \( p \in \mathbb{K}\langle \mathcal{X} \rangle \) and \( \pi \) be a minimal linear representation of \( p \). Then the *rank* of \( p \) is defined as \( \text{rank} \ p = \dim \pi \).

**Definition 1.3** (Left and Right Families [CR94]). Let \( \pi = (u, A, v) \) be a linear representation of \( p \in \mathbb{K}\langle \mathcal{X} \rangle \) of dimension \( n \). The families \( \{s_1, s_2, \ldots, s_n\} \subseteq \mathbb{K}\langle \mathcal{X} \rangle \) with \( s_i = (A^{-1} v)_i \) and \( \{t_1, t_2, \ldots, t_n\} \subseteq \mathbb{K}\langle \mathcal{X} \rangle \) with \( t_j = (u A^{-1})_j \) are called *left family* and *right family* respectively. \( L(\pi) = \text{span}\{s_1, s_2, \ldots, s_n\} \) and \( R(\pi) = \text{span}\{t_1, t_2, \ldots, t_n\} \) denote their linear spans (over \( \mathbb{K} \)).

**Proposition 1.4** ([CR94, Proposition 4.7]). A representation \( \pi = (u, A, v) \) of an element \( p \in \mathbb{K}\langle \mathcal{X} \rangle \) is minimal if and only if both, the left family and the right family, are \( \mathbb{K} \)-linearly independent. In this case, \( L(\pi) \) and \( R(\pi) \) depend only on \( p \).

**Definition 1.5** (Admissible Linear Systems and Transformations [Sch18c]). A linear representation \( \mathcal{A} = (u, A, v) \) of \( p \in \mathbb{K}\langle \mathcal{X} \rangle \) is called *admissible linear system* (ALS) for \( p \), written also as \( As = v \), if \( u = e_1 = [1, 0, \ldots, 0] \). The element \( p \) is then the first component of the (unique) solution vector \( s \). Given a linear representation \( \mathcal{A} = (u, A, v) \) of dimension \( n \) of \( p \in \mathbb{K}\langle \mathcal{X} \rangle \) and invertible matrices \( P, Q \in \mathbb{K}^{n \times n} \), the transformed \( PAQ = (uQ, PAQ, Pv) \) is again a linear representation (of \( p \)). If \( \mathcal{A} \) is an ALS, the transformation \( (P, Q) \) is called *admissible* if the first row of \( Q \) is \( e_1 = [1, 0, \ldots, 0] \).
A polynomial ALS is also written as \( ALS = (\text{Minimal Monomial} \ P, Q \ \text{transformation}) \).

**Proposition 1.8.** Let \( k \in \mathbb{N} \) and \( p = x_{i_1}x_{i_2} \cdots x_{i_k} \) be a monomial in \( \mathbb{K}(\mathcal{X}) \). Then

\[
\mathcal{A} = \begin{pmatrix}
1 & 0 & \cdots & 0 \\
1 & -x_{i_1} & \cdots & -x_{i_2} \\
& \ddots & \ddots & \ddots \\
& & 1 & -x_{i_k} \\
& & & 1
\end{pmatrix}
\begin{pmatrix}
0 \\
0 \\
\vdots \\
0
\end{pmatrix}
\]

is a minimal polynomial ALS of dimension \( \dim \mathcal{A} = k + 1 \).
Proposition 1.9 (Rational Operations [CR99]). Let \( 0 \neq p, q \in \mathbb{K}\langle X \rangle \) be given by the admissible linear systems \( A_p = (u_p, A_p, v_p) \) and \( A_q = (u_q, A_q, v_q) \) respectively. Then an ALS for the sum \( p + q \) is given by

\[
A_p + A_q = \left( [u_p], \begin{bmatrix} A_p & -A_pu_p^\top u_q \\ A_q \\ v_p \end{bmatrix}, [v_q] \right).
\]

And an ALS for the product \( fg \) is given by

\[
A_p \cdot A_q = \left( [u_p], \begin{bmatrix} A_p & -v_pu_q \\ A_q \\ v_q \end{bmatrix}, [v_q] \right).
\]

Example 1.10. An ALS for \( h_1 = p + q \) from Example 1.6 is

\[
\begin{bmatrix}
1 & -x & -1 & . \\
. & 1 & . & . \\
. & . & 1 & y & -1 \\
. & . & . & 1 & -x \\
. & . & . & . & 1
\end{bmatrix}
\begin{bmatrix}
s \\
1 + x - yx \\
1 \\
1 - yx \\
x \\
1
\end{bmatrix}.
\]

If we add row 3 to row 1 we get

\[
\begin{bmatrix}
1 & -x & 0 & y & -1 \\
. & 1 & . & . & . \\
. & . & 1 & y & -1 \\
. & . & . & 1 & -x \\
. & . & . & . & 1
\end{bmatrix}
\begin{bmatrix}
s \\
1 + x - yx \\
1 \\
1 - yx \\
x \\
1
\end{bmatrix}.
\]

and can remove row/column 3 because the corresponding column equation reads \( t_3 = 0 \) (recall that \( u_j = 0 \) for \( j \geq 2 \) in an ALS):

\[
\begin{bmatrix}
1 & -x & y & -1 \\
. & 1 & . & . \\
. & . & 1 & -x \\
. & . & . & 1
\end{bmatrix}
\begin{bmatrix}
s \\
1 + x - yx \\
1 \\
x \\
1
\end{bmatrix}.
\]

Now we can subtract row 4 from row 2 and add column 2 to column 4 (which results in subtracting \( s_4 \) from \( s_2 \)):

\[
\begin{bmatrix}
1 & -x & y & -1-x \\
. & 1 & . & 0 \\
. & . & 1 & -x \\
. & . & . & 1
\end{bmatrix}
\begin{bmatrix}
s \\
1 + x - yx \\
0 \\
x \\
1
\end{bmatrix}.
\]
Removing row/column 2 yields a minimal ALS (of dimension 3). Thus \( \text{rank}(p+q) = 3 \).

An ALS for \( h_2 = pq \) is

\[
\begin{bmatrix}
1 & -x & . & . \\
. & 1 & -1 & . \\
. & . & 1 & y & -1 \\
. & . & . & 1 & -x \\
. & . & . & . & 1
\end{bmatrix}
\begin{bmatrix}
s_1 \\
s_2 \\
s_3
\end{bmatrix}
= \begin{bmatrix}
x(1-yx) \\
1 - yx \\
x
\end{bmatrix}.
\]

Since \( p \) and \( q \) are given by minimal admissible linear systems, there is exactly one minimization step possible. Here we add row 3 to row 2 and remove row/column 3:

\[
\begin{bmatrix}
1 & -x & 0 & 0 \\
. & 1 & y & -1 \\
. & . & 1 & -x \\
. & . & . & 1
\end{bmatrix}
\begin{bmatrix}
s_1 \\
s_2 \\
s_3
\end{bmatrix}
= \begin{bmatrix}
x(1-yx) \\
1 - yx \\
x
\end{bmatrix}.
\]

Notice the upper right block of zeros of size \( 1 \times 2 \) in the system matrix. This is what we need for the factorization later (Theorem 3.2).

The “left” (row) and “right” (column) minimization steps are rather simple. However, to ensure minimality, we need to do that systematically. For further details we refer to [Sch19, Section 2]. To formulate the algorithm we need to decompose the polynomial ALS \( A = (u, A, v) \) of dimension \( n \geq 2 \) with respect to some row/column \( k \):

\[
A[k] = \left( \begin{bmatrix}
u_1 & . & . \\
A_{1,1} & A_{1,2} & A_{1,3} \\
. & 1 & A_{2,3} \\
. & . & A_{3,3}
\end{bmatrix}, \begin{bmatrix}
v_1 \\
v_2 \\
v_3
\end{bmatrix} \right).
\]

(By avoiding confusion, we use underlined subscripts to denote blocks in vectors.)

\[
A[-k] = \left( \begin{bmatrix}
u_1 & . \\
A_{1,1} & A_{1,3} \\
. & A_{3,3}
\end{bmatrix}, \begin{bmatrix}
v_1 \\
v_2 \\
v_3
\end{bmatrix} \right).
\]

Removing row/column \( k \) is only “admissible” if either \( A_{1,2} = 0 \) or \( A_{2,3} = 0 \) (and \( v_2 = 0 \)). For row minimization steps we use the transformation

\[
(P(T), Q(U)) = \begin{bmatrix}
I_{k-1} & . & . \\
. & 1 & T \\
. & . & I_{n-k}
\end{bmatrix}, \begin{bmatrix}
I_{k-1} & . & . \\
. & 1 & U \\
. & . & I_{n-k}
\end{bmatrix},
\]

for column minimization steps we use

\[
(P(T), Q(U)) = \begin{bmatrix}
I_{k-1} & T & . \\
. & 1 & . \\
. & . & I_{n-k}
\end{bmatrix}, \begin{bmatrix}
I_{k-1} & U & . \\
. & 1 & . \\
. & . & I_{n-k}
\end{bmatrix}.
\]
Definition 1.11 (Minimization Equations, Transformations [Sch19, Definition 31]). Let \( A = (u, A, v) \) be a polynomial ALS of dimension \( n \geq 2 \). For \( k = \{1, 2, \ldots, n-1\} \) the equations \( U + A_{2,3} + TA_{3,3} = 0 \) and \( v_2 + T v_3 = 0 \), with respect to the block decomposition \( A^{[k]} \) are called left minimization equations, denoted by \( \mathcal{L}_k = \mathcal{L}_k(A) \). A solution by the row block pair \((T, U)\) is denoted by \( \mathcal{L}_k(T, U) = 0 \), the corresponding transformation \((P, Q) = (P(T), Q(U))\) is called left minimization transformation. For \( k = \{2, 3, \ldots, n\} \) the equations \( A_{1,1}U + A_{1,2} + T = 0 \), with respect to the block decomposition \( A^{[k]} \) are called right minimization equations, denoted by \( \mathcal{R}_k = \mathcal{R}_k(A) \). A solution by the column block pair \((T, U)\) is denoted by \( \mathcal{R}_k(T, U) = 0 \), the corresponding transformation is called right minimization transformation.

Algorithm 1.12 (Minimizing a polynomial ALS [Sch19, Algorithm 32]).

Input: \( A = (u, A, v) \) polynomial ALS of dimension \( n \geq 2 \) (for some polynomial \( p \)).
Output: \( A' = (\ldots) \) if \( p = 0 \) or a minimal polynomial ALS \( A' = (u', A', v') \) if \( p \neq 0 \).

1: \( k := 2 \)
2: \( \textbf{while } k \leq \dim A \textbf{ do} \)
3: \( n := \dim (A) \)
4: \( k' := n + 1 - k \)
5: \( \text{Is the left subfamily } (s_k, s_{k+1}, \ldots, s_n) \text{ } \mathbb{K} \text{-linearly dependent?} \)
6: \( \text{if } \exists T, U \in \mathbb{K}^{1 \times (k-1)} \text{ admissible : } \mathcal{L}_k(A) = \mathcal{L}_k(T, U) = 0 \text{ then} \)
7: \( \text{if } k' = 1 \text{ then} \)
8: \( \text{return } (\ldots) \)
9: \( \text{if } k > \max\{2, \frac{n+1}{2}\} \text{ then} \)
10: \( \quad k := k - 1 \)
11: \( \text{endif} \)
12–17: \( \text{(for alignment)} \)
18: \( \text{Is the right subfamily } (t_1, \ldots, t_{k-1}, t_k) \text{ } \mathbb{K} \text{-linearly dependent?} \)
19: \( \text{if } \exists T, U \in \mathbb{K}^{(k-1) \times 1} \text{ admissible : } \mathcal{R}_k(A) = \mathcal{R}_k(T, U) = 0 \text{ then} \)
20: \( \quad A := (P(T)A Q(U))[-k] \)
21: \( \quad \text{if } k > \max\{2, \frac{n+1}{2}\} \text{ then} \)
22: \( \quad \quad k := k - 1 \)
23: \( \text{endif} \)
24: \( \text{continue} \)
25: \( \text{done} \)
26: \( \text{return } PA, \text{ with } P, \text{ such that } P v = [0, \ldots, 0, \lambda]^T \)

Remark. The line numbering is with respect to the general algorithm [Sch18a, Algorithm 4.14]. Polynomial admissible linear systems, called “pre-standard” in [Sch19];
are a special case of refined admissible linear systems because their diagonal blocks are as small as possible, namely $1 \times 1$. The lines 12–15 in [Sch19, Algorithm 32] are not even necessary since this special case is detected in the following part of the algorithm (the right family is linearly dependent).

Remark 1.13. If a linear representation $\pi = (u, A, v)$, say of dimension $n$, of some element $p \in K\langle X \rangle$ is not in the form of a polynomial ALS, there exists invertible matrices $P, Q \in K^{n \times n}$ such that $A = P\pi Q$ has this form. In general, $\pi$ needs to be minimal. Then [CR99, Proposition 2.1] ensures the existence of an upper unitriangular linear representation and [CR99, Theorem 1.4] implies the existence of such $P$ and $Q$, which are usually difficult to find. In our situation it is much simpler: We get the existence of a minimal polynomial ALS (for each element in the free associative algebra) directly by construction.

2 Horner Systems

Given $p \in K\langle X \rangle \setminus K$ by a polynomial ALS $A = (1, A, \lambda)$, say of dimension $n$, it is almost straightforward to generalize the idea of Horner’s rule to the non-commutative setting once we recall how to evaluate $p$ by a $d$-tuple of $m \times m$ matrices $\bar{X}_1, \bar{X}_2, \ldots, \bar{X}_d$ for the letters $x_i \in X$ (abusing the notation for the left and the right family):

- Starting with $s_n = I_m$, we compute (rowwise) $s_{n-1}$ to $s_1 = p$.
- Or, starting with $t_1 = I_m$, we compute (columnwise) $t_2$ to $t_n = \frac{1}{\lambda}p$.

Although we will see later (in Remark 2.9) that minimal admissible linear systems are not necessarily optimal with respect to the number of multiplications (for the evaluation), minimization (Algorithm 1.12) is the major step towards Horner systems (Definition 2.8). This becomes visible in particular in Table 1. Minimality plays also a crucial role for the factorization of a polynomial into a product of atoms (irreducible elements), or an atom into a product of matrices, and thus for creating (upper right) blocks of zeros in the system matrix $A$ (if possible). For details we refer to Section 3.

Remark 2.1. Finding the “most sparse” polynomial ALS can be very difficult in general because non-linear systems of equations need to be solved, similarly to [Sch19, Proposition 42]. So the minimization is rather cheap since it can be done with complexity $O(dn^4)$. For details we refer to [Sch19, Remark 33]. Fortunately one can also try linear (algebraic) techniques to “break” huge polynomials into smaller factors [Sch18a, Remark 5.8].

Since there are close connections to companion matrices (for the univariate case) we recall some basics and start with companion systems (Definition 2.2) to construct minimal polynomial admissible linear systems.

A univariate polynomial $p(x) = a_0 + a_1 x + \ldots + a_{n-1} x^{n-1} + x^n \in K[x] = K\langle \{x\} \rangle$ can be expressed as the characteristic polynomial of its companion matrix $L = L(p)$,
Table 1: Number of multiplications for the evaluation of $p_k = (x + y + z)^k$ (column 4 resp. 5) and $q_k = (x_1 + y_1 + z_1)q_{k-1} + \ldots + (x_k + y_k + z_k)q_1$ (column 7 resp. 8) as (finite) formal sum respectively minimal polynomial ALS. See also Remark 2.11.

that is, $p(x) = \det(xI - L)$ [Gan66, Section VI.6],

$$p(x) = \det(xI - L) = \det \begin{bmatrix} x & 0 & \ldots & 0 & a_0 \\ -1 & x & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots \\ -1 & x & \ldots & a_{n-2} \\ -1 & x + a_{n-1} \end{bmatrix}.$$ 

In [Coh95, Section 8.1], $\hat{C}(p) = xI - L(p)^\top$ is also called companion matrix. Viewing $C(p)$ as linear matrix pencil $C(p) = C_0 \otimes 1 + C_x \otimes x$ generalizes nicely to nc polynomials: $C(p)$ is —modulo sign— just the upper right $(n-1) \times (n-1)$ block of the system matrix of the (minimal) right companion system $C_p = (u, A, v) = (1, A, 1)$ of dimension $n$ (Definition 2.2). Evaluating $p$ in the special case of $q_i = x$ starting from the bottom right in this minimal ALS yields directly Horner’s rule. Notice that here $a_n = 1$, thus $n - 1 = \text{rank}(p) - 2$ multiplications are needed.

Remark. Notice that the system matrix $A$ in Definition 1.2 could be also written using the tensor product $A = A_0 \otimes 1 + A_1 \otimes x_1 + \ldots + A_d \otimes x_d$, $A_i \in \mathbb{K}^{n \times n}$, which reduces to the Kronecker tensor product when we plug in $m \times m$ matrices $\bar{X}_1, \ldots, \bar{X}_d$: $\bar{A} = A_0 \otimes I_m + A_1 \otimes \bar{X}_1 + \ldots + A_d \otimes \bar{X}_d \in \mathbb{K}^{mn \times mn}$.

Remark. In [BGKR08, Section 11.1] the companion matrix $L(p)$ is called second companion, its transpose $L(p)^\top$ first companion (matrix).

**Definition 2.2** (Companion Systems [Sch19, Definition 46]). For $i = 1, 2, \ldots, m$ let $q_i \in \mathbb{K}\langle\mathcal{X}\rangle$ with rank $q_i = 2$ and $a_i \in \mathbb{K}$. For a polynomial $p \in \mathbb{K}\langle\mathcal{X}\rangle$ of the form
\[ p = q_m q_{m-1} \cdots q_1 + a_{m-1} q_{m-1} \cdots q_1 + \ldots + a_2 q_2 q_1 + a_1 q_1 + a_0 \] the polynomial ALS

\[
\begin{bmatrix}
1 & -q_m & -a_{m-1} & \cdots & -a_2 & -a_1 & -a_0 \\
1 & -q_m & 0 & \cdots & 0 \\
\ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\
1 & -q_2 & 0 & \cdots & 0 \\
1 & -q_1 & 0 & \cdots & 0 \\
1 & 0 & \cdots & \cdots & \cdots & \cdots & 1
\end{bmatrix}
= \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 1 \end{bmatrix}
\]

is called left companion system. And for a polynomial \( p \in \mathbb{K}\langle X \rangle \) of the form

\[ p = a_0 + a_1 q_1 + a_2 q_1 q_2 + \ldots + a_{m-1} q_1 q_2 \cdots q_{m-1} + q_1 q_2 \cdots q_m \] the polynomial ALS

\[
\begin{bmatrix}
1 & -q_1 & 0 & \cdots & 0 & -a_0 \\
1 & -q_2 & \ddots & \ddots & \ddots \ddots \\
\ddots & \ddots & \ddots & \ddots & \ddots & 0 \\
1 & -q_{m-1} & -a_{m-2} & \cdots & 0 \\
1 & -q_{m-2} & -a_{m-3} & \ddots & \ddots & \ddots \\
1 & 0 & 0 & \ddots & \ddots & \cdots & 1
\end{bmatrix}
= \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 1 \end{bmatrix}
\]

is called right companion system.

**Example 2.5** ([Sch19, Example 50]). The left companion system of \( p(x) = x^3 - 10 x^2 + 31 x - 30 \) is

\[
\begin{bmatrix}
1 & -x + 10 & -31 & 30 \\
1 & -x & . & . \\
. & 1 & -x & . \\
. & . & 1 & -x \\
. & . & . & 1
\end{bmatrix}
\begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \end{bmatrix}
= \begin{bmatrix} p(x) \\ x^2 q_2 \\ x q_1 \\ q_2 \\ 1 \end{bmatrix}.
\]

**Remark 2.6.** One can view the algorithm (in the univariate case) in [TOT14] as taking the left companion system (2.3) and evaluate the matrix powers in the left family (Definition 1.3) \( s = (p, x^{n-1}, \ldots, x^2, x, 1) \) efficiently, for example, \( x^4 = x^2 \cdot x^2 \). Notice, that in this case the coefficients \( a_i \in \mathbb{K} \) are assumed to be scalar. Matrix valued coefficients (or parameters) can easily be treated by an augmented alphabet, here \( X = \mathcal{X} \cup \{a_0, a_1, \ldots, a_{n-1}\} \).

Here we consider the general case and assume that our alphabet \( \mathcal{X} \) contains the matrix valued parameters (mainly \( a, b \) and \( c \)). Although we can evaluate a polynomial with matrices of appropriate sizes, we typically plug in \( m \times m \) matrices and measure the “evaluation complexity” as the minimal number of matrix-matrix multiplications with respect to a polynomial ALS.

**Remark.** Recall that we assume only that the multiplication is the dominating part, that is, its complexity is \( \mathcal{O}(m^{2+\varepsilon}) \) for \( \varepsilon > 0 \).
Definition 2.7 (Left/Right/Minimal Number of Multiplications). Let \( p \in \mathbb{K}\langle \mathcal{X} \rangle \) be given by the polynomial ALS \( \mathcal{A} = (1, A, \lambda) \) of dimension \( n \geq 2 \). The minimal number of non-scalar entries in the upper left (respectively lower right) \((n - 1) \times (n - 1)\) block of \( A \) is called left (respectively right) number of multiplications, written as \( \mathcal{N}_s(\mathcal{A}) \) (respectively \( \mathcal{N}_t(\mathcal{A}) \)). The number of multiplications (of a polynomial ALS) is denoted by \( N(\mathcal{A}) = \min\{\mathcal{N}_s(\mathcal{A}), \mathcal{N}_t(\mathcal{A})\} \). If \( N(\mathcal{A}) \leq N(\mathcal{B}) \) for all polynomial admissible linear systems \( \mathcal{B} \) for \( p \), we write \( N(p) = N(\mathcal{A}) \).

Definition 2.8 (Horner System). Let \( p \in \mathbb{K}\langle \mathcal{X} \rangle \). A polynomial ALS \( \mathcal{A} \) for \( p \) is called Horner system if \( N(\mathcal{A}) = N(p) \).

Remark 2.9. It is clear that a Horner System (for a given polynomial) is not unique. Less obvious is the fact that a Horner system is not necessarily a minimal ALS. This is shown in the following example: Let \( p = ab(xy + yz + z + 1) + acxyz \). A Horner system for \( p \) is given by the ALS \( \mathcal{A} \),

\[
\begin{bmatrix}
1 & -a & \cdots & \cdots & \cdots & \cdots & \cdots \\
. & 1 & -b & -c & \cdots & \cdots & \cdots \\
. & . & 1 & \cdots & \cdots & \cdots & \cdots \\
. & . & . & 1 & -x & \cdots & \cdots \\
. & . & . & . & 1 & -y & \cdots \\
. & . & . & . & . & 1 & -z \\
. & . & . & . & . & . & 1
\end{bmatrix}
\begin{bmatrix}
s \\
. \\
. \\
. \\
. \\
. \\
. \\
1
\end{bmatrix}
\]

with \( N(\mathcal{A}) = \mathcal{N}_s(\mathcal{A}) = \mathcal{N}_t(\mathcal{A}) = 5 \). Adding column 3 to columns 4–7 and removing row 3 and column 3 yields the minimal ALS \( \mathcal{A}' \),

\[
\begin{bmatrix}
1 & -a & \cdots & \cdots & \cdots & \cdots & \cdots \\
. & 1 & -b & -c & -b & -b & -b \\
. & . & 1 & \cdots & \cdots & \cdots & \cdots \\
. & . & . & 1 & -x & \cdots & \cdots \\
. & . & . & . & 1 & -y & \cdots \\
. & . & . & . & . & 1 & -z \\
. & . & . & . & . & . & 1
\end{bmatrix}
\begin{bmatrix}
s \\
. \\
. \\
. \\
. \\
. \\
. \\
1
\end{bmatrix}
\]

with \( N(\mathcal{A}') = \mathcal{N}_s(\mathcal{A}') = 6 \) and \( \mathcal{N}_t(\mathcal{A}') = 7 \). However, evaluating \( p \) using the representation as (finite) formal sum—in a naive way—needs \( 13 \) multiplications. And since the worst case is of exponential complexity, the restriction to minimal admissible linear systems will suffice in practice for a first “evaluation simplification”. See Table 1.

Proposition 2.10 (Evaluation Complexity of Polynomials). Let \( p \in \mathbb{K}\langle \mathcal{X} \rangle \) of rank \( n \geq 2 \). Then \( n - 2 \leq N(p) \leq \frac{1}{2}(n - 1)(n - 2) \).

Proof. Since a polynomial of rank \( n \) admits a polynomial ALS of dimension \( n \), the upper bound follows directly from the upper unitriangular system matrix. For the lower bound we can assume without loss of generality that \( N(p) = \mathcal{N}_s(\mathcal{A}) \) for some polynomial ALS \( \mathcal{A} = (1, A, \lambda) \) of dimension \( \dim \mathcal{A} = m \geq n \). However, \( \mathcal{N}_s(\mathcal{A}) \leq n - 3 \)
would imply that there are at least $m - n + 1$ scalar columns in the system matrix $A$ (except column 1 which must not be touched and column $m$ which is irrelevant for $N_s$) which could be removed after appropriate row operations, contradicting that $n$ is the rank of $p$. 

**Remark 2.11.** The polynomial ALS from Table 1 for $p_k$ is

$$
\begin{bmatrix}
1 & -(x + y + z) \\
1 & -(x + y + z) \\
\ddots & \ddots \\
1 & -(x + y + z) \\
1 & -(x + y + z)
\end{bmatrix}
\begin{bmatrix}
s \\
s \\
\vdots \\
s \\
1
\end{bmatrix} =
\begin{bmatrix}
0 \\
0 \\
\vdots \\
0 \\
1
\end{bmatrix},
$$

that for $q_k$ is

$$
\begin{bmatrix}
1 & -(x_1 + y_1 + z_1) & -(x_2 + y_2 + z_2) & \ldots & -(x_k + y_k + z_k) \\
1 & -(x_1 + y_1 + z_1) & -(x_2 + y_2 + z_2) & \ldots & -(x_{k-1} + y_{k-1} + z_{k-1}) \\
\ddots & \ddots & \ddots & \ddots & \ddots \\
1 & -(x_1 + y_1 + z_1) & -(x_2 + y_2 + z_2) & \ldots & -(x_{k-1} + y_{k-1} + z_{k-1})
\end{bmatrix}
\begin{bmatrix}
s \\
s \\
\vdots \\
s \\
1
\end{bmatrix} =
\begin{bmatrix}
0 \\
0 \\
\vdots \\
0 \\
1
\end{bmatrix}.
$$

Let $\ell$ be the number of letters in each entry of the system matrix (here $\ell = 3$) and $n$ the dimension of the (polynomial) admissible linear system. Then there are $\ell$ words of length 1 in $s_{n-1}$ which we denote by “$\ell \cdot 1$”. In $s_{n-2}$ the words are $\ell \cdot 1$ and $\ell$-times the words of $s_{n-1}$ with one additional letter, that is, $\ell \cdot 1 + \ell^2 \cdot (1 + 1)$. In $s_{n-3}$ the words are $\ell \cdot 1 + \ell^2 \cdot 2 + \ell (\ell \cdot (1 + 1) + \ell^2 \cdot (2 + 1)) = \ell \cdot 1 + 2 \ell^2 \cdot 2 + \ell^3 \cdot 3$. In $s_{n-4}$ and $s_{n-5}$ the words are

$$
\begin{array}{c}
1 \\
\ell \cdot 1 + 3 \ell^2 \cdot 2 + 3 \ell^3 \cdot 3 + 1 \ell^4 \cdot 4 \\
\ell \cdot 1 + 4 \ell^2 \cdot 2 + 6 \ell^3 \cdot 3 + 4 \ell^4 \cdot 4 + 1 \ell^5 \cdot 5,
\end{array}
$$

revealing that the coefficients are the entries in the respective row of the Pascal triangle

$$
\begin{array}{cccccccccccc}
1 & & & & & & & & & & & & \\
1 & 1 & & & & & & & & & & \\
1 & 2 & 1 & & & & & & & & & \\
1 & 3 & 3 & 1 & & & & & & & & \\
1 & 4 & 6 & 4 & 1 & & & & & & & \\
1 & 5 & 10 & 10 & 5 & 1 & & & & & & \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots
\end{array}
$$

Now both, the number of terms and the number of multiplications, are immediate.
3 Matrix Factorization

Before we introduce the concept of the factorization of polynomials into a product of matrices (aka “matrix factorization”) in Definition 3.6 we recall the basics from the “minimal” multiplication of polynomials and the opposite point of view, namely the polynomial factorization (Theorem 3.2).

The factorization of polynomials into atoms, that is, irreducible elements (Definition 1.1) corresponds to the transformation of a minimal polynomial admissible linear system to one with a system matrix having the “finest” possible upper right “staircase” of zeros, for example \( p = xyz \) given by the ALS (Definition 1.7)

\[
\begin{pmatrix}
1 & -x & 0 & 0 \\
. & 1 & -y & 0 \\
. & . & 1 & -z \\
. & . & . & 1
\end{pmatrix}
= \begin{pmatrix}
\cdot \\
\cdot \\
\cdot \\
1
\end{pmatrix},
\begin{pmatrix}
xyz \\
yz \\
z
\end{pmatrix}.
\]

Those upper right blocks of zeros come directly from the “minimal” polynomial multiplication [Sch19, Proposition 28], illustrated in the following example. Notice however, that this (upper right) form is not unique in general.

**Example 3.1.** Let \( p = xy + 1 \) and \( q = zx - 3 \) be given by the minimal ALS

\[
\begin{pmatrix}
1 & -x & -1 \\
. & 1 & -y \\
. & . & 1
\end{pmatrix}
= \begin{pmatrix}
\cdot \\
\cdot \\
1
\end{pmatrix},
\begin{pmatrix}
1 & -z & 3 \\
. & 1 & -x \\
. & . & 1
\end{pmatrix}
= \begin{pmatrix}
\cdot \\
\cdot \\
1
\end{pmatrix}
\]

respectively. Recall that the symbol \( s \) for the solution vector is used in a generic way. By Proposition 1.9, an ALS for the product \( pq = (xy + 1)(zx - 3) \) is given by

\[
\begin{pmatrix}
1 & -x & -1 & . & . \\
. & 1 & -y & . & . \\
. & . & 1 & -1 & . \\
. & . & . & 1 & -z & 3 \\
. & . & . & . & 1 & -x \\
. & . & . & . & . & 1
\end{pmatrix}
= \begin{pmatrix}
\cdot \\
\cdot \\
\cdot \\
\cdot \\
\cdot \\
1
\end{pmatrix}.
\]

Now, if we add column 3 to column 4 we get

\[
\begin{pmatrix}
1 & -x & -1 & -1 & . & . \\
. & 1 & -y & -y & . & . \\
. & . & 1 & 0 & . & . \\
. & . & . & 1 & -z & 3 \\
. & . & . & . & 1 & -x \\
. & . & . & . & . & 1
\end{pmatrix}
= \begin{pmatrix}
\cdot \\
\cdot \\
\cdot \\
\cdot \\
\cdot \\
0
\end{pmatrix}
\]

where the third row equation reads \( s_3 = 0 \) and hence we can remove row 3 and column 3 since there is no contribution to the first component \( s_1 = pq \) in the solution.
vector \( s \). Thus a *minimal* ALS for \( pq \) is given by

\[
\begin{bmatrix}
1 & -x & -1 & 0 & 0 \\
1 & -y & 0 & 0 \\
. & 1 & -z & 3 \\
. & . & 1 & -x \\
. & . & . & . & 1
\end{bmatrix}
\begin{bmatrix}
s \\
0 \\
0 \\
\end{bmatrix} =
\begin{bmatrix}
0 \\
. \\
. \\
. \\
1
\end{bmatrix}
\]

For concrete examples minimality can be checked easily by using Proposition 1.4. In the general case a systematic application of left and right minimization steps (as in Algorithm 1.12) ensures minimality. Notice the upper right 2 \( \times \) 2 block of zeros in the system matrix (and the upper zeros in the right hand side).

**Theorem 3.2** (Polynomial Factorization \[\text{Sch19, Theorem 40}\]). Let \( p \in \mathbb{K}\langle X \rangle \) be given by the minimal polynomial ALS \( A = (1, A, \lambda) \) of dimension \( n = \text{rank} \ p \geq 3 \). Then \( p \) has a factorization into \( p = q_1q_2 \) with \( \text{rank}(q_i) = n_i \geq 2 \) if and only if there exists a polynomial transformation \( (P, Q) \) such that \( PAQ \) has an upper right block of zeros of size \( (n_1 - 1) \times (n_2 - 1) \).

**Example 3.3.** Let \( p = 2aexc + 2bxc - aexd - bxd \) \[\text{dO12}\] given by the minimal ALS \( A = (u, A, v) \),

\[
\begin{bmatrix}
1 & -a & -b & -a \\
1 & -e & . & 2c - d \\
. & 1 & -x \\
. & . & 1 & d - 2c \\
. & . & . & 1
\end{bmatrix}
\begin{bmatrix}
s \\
. \\
. \\
. \\
1
\end{bmatrix} = \begin{bmatrix}
. \\
. \\
. \\
. \\
1
\end{bmatrix}
\]

To find a non-trivial factor of \( p \) we need to find a transformation \( (P, Q) \) of the form

\[
(P, Q) = \begin{pmatrix}
1 & \alpha_{1,2} & \alpha_{1,3} & \alpha_{1,4} & 0 \\
. & 1 & \alpha_{2,3} & \alpha_{2,4} & 0 \\
. & . & 1 & \alpha_{3,4} & 0 \\
. & . & . & 1 & 0 \\
. & . & . & . & 1
\end{pmatrix}
\begin{pmatrix}
1 & 0 & 0 & 0 & 0 \\
. & 1 & \beta_{2,3} & \beta_{2,4} & \beta_{2,5} \\
. & . & 1 & \beta_{3,4} & \beta_{3,5} \\
. & . & . & 1 & \beta_{3,5} \\
. & . & . & . & 1
\end{pmatrix}
\]

(see Definition 1.7) such that \( PAQ \) has an upper right block of zeros of size 1 \( \times \) 3, 2 \( \times \) 2 or 3 \( \times \) 1. In this case it is (almost) immediate that we need to add row 4 to row 2 and subtract column 2 from column 4, that is, \( \alpha_{2,4} = 1 \) and \( \beta_{2,4} = -1 \),

\[
(P, Q) = \begin{pmatrix}
1 & . & . & . & 0 \\
. & 1 & 1 & 0 & . \\
. & 1 & . & 0 & . \\
. & . & 1 & 0 & . \\
. & . & . & 1 & .
\end{pmatrix}
\begin{pmatrix}
1 & 0 & 0 & 0 & 0 \\
. & 1 & . & -1 & . \\
. & . & 1 & . & . \\
. & . & . & 1 & . \\
. & . & . & . & 1
\end{pmatrix}
\]
and thus $P.AQ = (uQ, PAQ, Pv)$,

$$
\begin{bmatrix}
1 & -a & -b & 0 & 0 \\
. & 1 & -c & 0 & 0 \\
. & . & 1 & -x & 0 \\
. & . & . & 1 & d - 2c \\
\end{bmatrix}
\begin{bmatrix}
s \\
. \\
. \\
. \\
\end{bmatrix}
= 
\begin{bmatrix}
. \\
. \\
. \\
1 \\
\end{bmatrix}
$$

For the evaluation of $p$ (in the “expanded” form) with $m \times m$ matrices, $10 \mathcal{O}(m^3)$ operations are necessary while only $3 \mathcal{O}(m^3)$ suffice for the factorized version $p = 2ae xc + 2bxc - aexd - bxd = (ae + b)x(2c - d)$.

**Remark.** In general it can be difficult to find these (invertible) transformation matrices (if they exist), in particular, if the base field $\mathbb{K}$ is not algebraically closed, that is, $\mathbb{K} \subsetneq \overline{\mathbb{K}}$. Testing (ir)reducibility works practically for rank $\leq 12$, in some cases up to rank $\leq 17$ [Jan18, Chapter 2]. In the previous example it was easy because we can solve a *linear* system of equations for “non-overlapping” row and column transformations, that is, if we use column 3 to create an upper right block of zeros of size $2 \times 2$, we are not allowed to use row 3 (and vice versa). See also [Sch18a, Remark 5.8].

Before we formalize the factorization of a polynomial into matrices we show the idea in an example. A comprehensive theory for the work with matrices (from an algebraic perspective including the general factorization theory [Sch17]) is considered in future work. Here we need only the fact that we can *admissibly* transform a (polynomial) ALS. If we find a certain pattern of zeros, we can read off the matrices —more or less— directly and their product yields the polynomial. In the case of a polynomial matrix (not to be confused with matrix polynomial), additional letters can be used to view it as a “classical” nc polynomial (Example 3.8).

**Example 3.4** ("Matrix factorization" of the Antikommutator). Let $p = xy + yx$ given by the minimal polynomial ALS $A = (u, A, v)$,

$$
\begin{bmatrix}
1 & -x & -y & 0 \\
. & 1 & 0 & -y \\
. & . & 1 & -x \\
. & . & . & 1 \\
\end{bmatrix}
\begin{bmatrix}
s \\
. \\
. \\
1 \\
\end{bmatrix}
= 
\begin{bmatrix}
. \\
. \\
. \\
1 \\
\end{bmatrix}
$$

In this case only 2 multiplications are necessary. Notice the zeros in the system matrix. In this case we can write $p$ as a product of two matrices:

$$
p = [x \ y] \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}^{-1} \begin{bmatrix} y \\ x \end{bmatrix} = [x \ y] \begin{bmatrix} y \\ x \end{bmatrix}.
$$

If $p$ is given by any other *minimal* polynomial ALS we can look for an admissible
transformation \((P, Q)\) of the form

\[
(P, Q) = \begin{pmatrix}
1 & \alpha_{1,2} & \alpha_{1,3} & 0 \\
. & 1 & \alpha_{2,3} & 0 \\
. & . & 1 & 0 \\
. & . & . & 1
\end{pmatrix}
\begin{pmatrix}
1 & 0 & 0 & 0 \\
1 & \beta_{2,3} & \beta_{2,4} & . \\
. & 1 & \beta_{3,4} & . \\
. & . & . & 1
\end{pmatrix}
\]

such that \(PAQ\) has the form ("\(*\)" denotes some non-zero entry)

\[
\begin{pmatrix}
1 & * & * & 0 \\
. & 1 & 0 & * \\
. & . & 1 & * \\
. & . & . & 1
\end{pmatrix}
\]

This yields a non-linear (polynomial) system of equations. For details and how to solve such a systems we refer to [Sch18b, Section 4.4]. Notice that these transformation matrices do not suffice in general because permutations of rows/columns are excluded. Thus we need (admissible) transformations of the form

\[
(P, Q) = \begin{pmatrix}
\alpha_{1,1} & \ldots & \alpha_{1,n-1} & 0 \\
. & \ddots & \vdots & \vdots \\
. & . & \alpha_{n-1,1} & \alpha_{n-1,n-1} \\
\alpha_{n,1} & \ldots & \alpha_{n,n-1} & 1
\end{pmatrix}
\begin{pmatrix}
1 & 0 & \ldots & 0 \\
\beta_{2,1} & \beta_{2,2} & \ldots & \beta_{2,n} \\
. & . & \ddots & \vdots \\
\beta_{n,1} & \beta_{n,2} & \ldots & \beta_{n,n}
\end{pmatrix}
\]

(3.5)

and invertibility conditions \(\det P \neq 0\) and \(\det Q \neq 0\). In such a case we call \((P, Q)\) (admissible) factorization transformation.

**Definition 3.6** (Matrix Reducibility). Let \(p \in \mathbb{K}\langle \mathcal{X} \rangle\) of rank \(n \geq 3\) given by the minimal polynomial ALS \(A = (u, A, v)\) and \(k \in \{1, 2, \ldots, n - 2\}\). If there exists an \(i \in \{1, 2, \ldots, n - k - 1\}\) and a factorization transformation \((P, Q)\) such that \(PAQ\) is again a polynomial ALS, \(PAQ\) has an upper right block of zeros of size \(i \times (n - i - k)\) and an identity diagonal \(k \times k\) block in rows \(i + 1\) to \(i + k\), that is, \(PAQ\) has the form

\[
\begin{pmatrix}
i \text{ rows}
* & * & 0 \\
k \text{ rows}
0 & I_k & * \\
n - i - k \text{ rows}
0 & 0 & *
\end{pmatrix}
\]

then \(p\) is called \(k\)-reducible. If there is no such \(i\), it is called \(k\)-irreducible.

**Remark.** 1-irreducibility is just the “classical” irreducibility. The anticommutator (Example 3.4) is (1-)irreducible but 2-reducible.

**Example 3.7.** Let \(p = 3cyxb + 3xyxb + 2cyxax + cytxb - cyx - 2xbyxax + 4xybxb - 3xyaxb + 3xaxyxb - 3bxbyxb + 6xbyx + 2xaxybx + xaxyxh - xaxyxb - 2bxbyxax - bxybxb + bxbyxb + 5xbyxb - 4axbyxb\) [CHS06, Section 8.2]. The rank of \(p\) is 16, that is, the system matrix of a minimal ALS has dimension 16. The polynomial \(p\) has
19 terms (monomials). A *minimal* (polynomial) ALS $A = (1, A, 3)$ constructed iteratively starting with zero and adding monomial by monomial (including minimization by Algorithm 1.12) using the computer algebra system [Fri19] and the (experimental) implementation of the free field FDALG “Free Division ALGebra” (available in Release 1.3.5) results already in a very sparse ALS showing that $p$ is 2-reducible for $i = 7$ (in particular that $A$ has a upper right block of zeros of size $7 \times 7$). The left number of multiplications is $N_s(A) = 24$, the right number $N_t(A) = 22$. For simplicity we show only the lower right “subsystem” of $A$ of size $9 \times 9$ and hide the upper left in the polynomials $p'_1$ and $p'_2$:

$$
\begin{bmatrix}
1 & -p'_1 & -p'_2 & 0 & 0 & 0 & 0 & 0 & 0 \\
8 & 1 & 0 & -3y & -4y & . & . & -y & . \\
9 & 0 & 1 & -y & -y & . & . & -y & . \\
10 & . & . & 1 & . & -5x & . & \frac{1}{3}a & . \\
11 & . & . & 1 & . & 4x & . & \frac{1}{3}b & . \\
12 & . & . & . & 1 & -a & . & . & . \\
13 & . & . & . & . & 1 & . & \frac{1}{3}x & . \\
14 & . & . & . & . & 1 & -x & . & . \\
15 & . & . & . & . & . & 1 & -b & . \\
16 & . & . & . & . & . & . & 1 & . \\
\end{bmatrix} \begin{bmatrix}
 & \\
 & \\
 & \\
 & \\
 & \\
 & \\
 & \\
 & \\
 & \\
\end{bmatrix}
$$

$s = \begin{bmatrix}
 & \\
 & \\
 & \\
 & \\
 & \\
 & \\
 & \\
 & \\
 & \\
\end{bmatrix}$

Only two elementary operations (subtracting 3-times column 10 from column 14 and adding 2-times column 11 to column 14) yield $A'$,

$$
\begin{bmatrix}
1 & -p'_1 & -p'_2 & . & . & . & . & . & . \\
8 & 1 & . & -3y & -4y & 0 & 0 & 0 & 0 \\
9 & 0 & 1 & -y & -y & 0 & 0 & 0 & 0 \\
10 & . & . & 1 & 0 & -5x & . & \frac{1}{3}a & . \\
11 & . & . & 0 & 1 & 4x & . & \frac{1}{3}b & . \\
12 & . & . & . & 1 & -a & . & . & . \\
13 & . & . & . & . & 1 & . & \frac{1}{3}x & . \\
14 & . & . & . & . & 1 & -x & . & . \\
15 & . & . & . & . & . & 1 & -b & . \\
16 & . & . & . & . & . & . & 1 & . \\
\end{bmatrix} \begin{bmatrix}
 & \\
 & \\
 & \\
 & \\
 & \\
 & \\
 & \\
 & \\
 & \\
\end{bmatrix}
$$

revealing that $p$ is also 2-reducible for $i = 9$ and (by Proposition 2.10) $14 \leq N(p) \leq 20 = N_t(A')$. Recursively, by using an ALS as a “workbench”, one can find the matrix factorization $p = (X_1X_2X_3 + X_4)YZ_1Z_2Z_3$ with

$$
X_1 = \begin{bmatrix} 1 + a & 1 + b & x \end{bmatrix}, \quad X_2 = \begin{bmatrix} x & . & . \\
. & x & . \\
. & . & a \end{bmatrix}, \quad X_3 = \begin{bmatrix} b & . \\
. & -b \\
. & x \end{bmatrix}, \quad X_4 = \begin{bmatrix} . & c \end{bmatrix},
$$

$$
Y = \begin{bmatrix} y & . \\
. & y \end{bmatrix}, \quad Z_1 = \begin{bmatrix} 6 + 5b - 4a & 3 + b - a & 2x \end{bmatrix}, \quad Z_2 = \begin{bmatrix} . & x \\
. & a \end{bmatrix}, \quad \text{and} \quad Z_3 = \begin{bmatrix} x \\
. & b \end{bmatrix},
$$
showing that \( N(p) = 15 \). A “block” polynomial ALS for \( p \) is

\[
\begin{bmatrix}
1 & -X_1 & -X_4 & . & . & . \\
I_3 & -X_2 & . & . & . \\
. & I_3 & -X_3 & . & . & . \\
. & . & I_2 & -Y & . & . \\
. & . & . & I_2 & -Z_1 & . \\
. & . & . & . & I_2 & -Z_2 \\
. & . & . & . & . & I_2 & -Z_3 \\
. & . & . & . & . & . & 1
\end{bmatrix}
= \begin{bmatrix}
. \\
. \\
. \\
. \\
. \\
. \\
. \\
1
\end{bmatrix}
\]

Notice that \( p \) is (1-)irreducible because it is not possible to (admissibly) transform a minimal ALS (for \( p \)) into one with an upper right block of zeros of size \( 1 \times 14, 2 \times 13, \ldots, 13 \times 2 \) or \( 14 \times 1 \).

**Remark.** To evaluate \( p \) as (finite) formal sum, 97 matrix-matrix multiplications are necessary. On the other hand \( N(p) = 15 \), that is, only 15 multiplications (starting from the top left) are necessary using a Horner system.

**Remark.** The matrix factorization \( p = XYZ \) shows immediately that the Sylvester index [KMP00] (with respect to \( y \)) is 2, that is, \( p = p_1 y q_1 + p_2 y q_2 \).

**Example 3.8.** Taking the polynomial matrix

\[
X = \begin{bmatrix}
(ae + b)x_{11}(2c - d) & (ae + b)x_{12}(c + d) - (ae + b)x_{11}d \\
(bx_{21} - (ae + b)x_{11})(2c - d) & (ae + b)(x_{11}d - x_{12}(c + d)) - bx_{21}d + bx_{22}(c + d)
\end{bmatrix}
\]

from [dO12], by multiplying a row vector from the left respectively a column vector from the right (both with generic variables), we can consider it as a polynomial:

\[
p = \begin{bmatrix}
y_1 \\
y_2
\end{bmatrix}
X
\begin{bmatrix}
z_1 \\
z_2
\end{bmatrix}
\]

A minimal ALS for \( p \) is

\[
\begin{bmatrix}
1 & -y_1 & -y_2 & . & . & . & . & . & . \\
1 & . & -a & -b & 0 & . & . & . & . \\
1 & a & b & -b & . & . & . & . & . \\
1 & . & -a & 0 & . & . & . & . & . \\
1 & . & -x_{11} & -x_{12} & . & . & . & . & . \\
1 & . & -x_{21} & -x_{22} & . & . & . & . & . \\
1 & . & d - 2c & d & . & . & . & . & . \\
1 & 0 & -d - c & . & . & . & . & . & . \\
1 & . & -z_1 & . & . & . & . & . & . \\
1 & . & -z_2 & . & . & . & . & . & .
\end{bmatrix}
= \begin{bmatrix}
. \\
. \\
. \\
. \\
. \\
. \\
. \\
1
\end{bmatrix}
\]
which translates directly—the second matrix appears in a linearized form in the system matrix—into the matrix factorization

\[
p = \begin{bmatrix} y_1 & y_2 \\ \end{bmatrix} \begin{bmatrix} ae + b & 0 \\ -ae - b & b \\ \end{bmatrix} \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \\ \end{bmatrix} \begin{bmatrix} 2c - d & -d \\ 0 & c + d \\ \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ \end{bmatrix}.
\]

Notice that we already know that there must exist an upper right block of zeros of size \(1 \times 8\) and one of size \(8 \times 1\). The tricky part here is that one “matrix-factor” of \(X\) has non-linear entries. In this case one could substitute \(ae\) by a new symbol/letter. Then the ALS would have dimension 10 (recall that rank \(p = 11\)).

4 Epilogue

Given a (non-trivial) nc polynomial \(p\) with rank \(p = n \geq 2\), a Horner system is the most sparse polynomial admissible linear system for \(p\) with respect to matrix-matrix multiplications. Unfortunately, finding Horner systems in general is very difficult since one needs to solve non-linear (polynomial) systems of equations. However, in concrete situations, one can get good “approximations” quite easily by starting with a minimal polynomial ALS \(A\) (constructed by Algorithm 1.12) and trying to find non-trivial factorizations by linear techniques [Sch18a, Remark 5.8], that is, using “non-overlapping” row and column transformations, yielding some ALS \(A'\). From Proposition 2.10 we have bounds for the minimal number of multiplications (for the evaluation of \(p\)), namely,

\[
n - 2 \leq N(p) \leq N(A') \leq \frac{1}{2}(n - 1)(n - 2).
\]

If \(n - 2 \ll N(A') \leq \frac{1}{2}(n - 1)(n - 2)\) it might be worth to check systematically for \(k\)-reducibility of \(p\) for \(k = 1, 2, \ldots, n - 2\) (Definition 3.6). This can be done recursively, using already known factorizations in \(A'\), yielding some \(A''\). And finally one can minimize the number of non-scalar entries in the “matrix factors” of \(A''\) by looking for appropriate (scalar) invertible matrices, for example, \(P \in \mathbb{K}^{k_1 \times k_1}\) and \(Q \in \mathbb{K}^{k_2 \times k_2}\).

\[
p = XP P^{-1}YQ Q^{-1}Z.
\]

In general, this is very difficult, since already for a special case, namely pivot block refinement [Sch18a, Section 3], one needs to solve non-linear (polynomial) systems of equations. (See also [Sch18b, Section 4.4].) If there is no additional structure one can use, this is comparable to testing “fullness” of matrices [Jan18, Chapter 3], so one cannot expect a brute-force approach to work practically for \(k_1 = k_2 = k > 5\).

For “totally” irreducible elements of rank \(n\) one would need to check for all “sparsity patterns” with respect to evaluation by the left and by the right family. For an
ALS \( \mathcal{A} = (1, A, \lambda) \) of dimension \( n \) there are \( \bar{n} := (n - 2)(n - 1)/2 - 1 \) entries to test for \( n - 3, n - 2, \ldots, \bar{n} - 1 \) non-scalar entries. For illustration we take \( n = 5 \):

\[
\begin{bmatrix}
\alpha_{1,1} & \alpha_{1,2} & \alpha_{1,3} & \alpha_{1,4} & 0 \\
\alpha_{2,1} & \alpha_{2,2} & \alpha_{2,3} & \alpha_{2,4} & 0 \\
\alpha_{3,1} & \alpha_{3,2} & \alpha_{3,3} & \alpha_{3,4} & 0 \\
\alpha_{4,1} & \alpha_{4,2} & \alpha_{4,3} & \alpha_{4,4} & 0 \\
\alpha_{5,1} & \alpha_{5,2} & \alpha_{5,3} & \alpha_{5,4} & 1 \\
\end{bmatrix}
\]

\[= P \in \mathbb{K}^{n \times n}, \det P \neq 0\]

\[
\begin{bmatrix}
1 & * & ? & ? & * \\
1 & 1 & ? & ? & * \\
. & . & 1 & ? & * \\
. & . & . & 1 & * \\
. & . & . & . & 1 \\
\end{bmatrix}
\]

\[= A\]

\[
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
\beta_{2,1} & \beta_{2,2} & \beta_{2,3} & \beta_{2,4} & \beta_{2,5} \\
\beta_{3,1} & \beta_{3,2} & \beta_{3,3} & \beta_{3,4} & \beta_{3,5} \\
\beta_{4,1} & \beta_{4,2} & \beta_{4,3} & \beta_{4,4} & \beta_{4,5} \\
\beta_{5,1} & \beta_{5,2} & \beta_{5,3} & \beta_{5,4} & \beta_{5,5} \\
\end{bmatrix}
\]

\[= Q \in \mathbb{K}^{n \times n}, \det Q \neq 0\]

This would yield \( 10 + 10 + 5 = 25 \) possibilities already for \( n = 5 \), each inducing a non-linear (polynomial) system of equations with \( 2n(n - 1) \) commuting unknowns. So in general this will not be very useful, in particular because — compared to the factorization — one does not get any “structural” insight. However, heuristic approaches for increasing (non-scalar) sparsity, that is, “approximating” Horner systems, by “local” row and column transformations depending on the existing structure in the system matrix (respectively the coefficient matrices \( A_1, \ldots, A_d \) of the linear matrix pencil \( A \)) might be possible and could be very helpful.

**Remark 4.1.** The evaluation of a polynomial \( p \) given by the minimal polynomial ALS \( \mathcal{A} = (1, A, \lambda) \) of dimension \( n \) by non-square matrices (of appropriate size) yields a natural block structure, entry \((i, j)\) in the system matrix \( A \) has size \( m_i \times m_j \) with \( m_1 = m_n = 1 \). In this case one can get a priority for checking in particular 1-reducibility to avoid huge inner dimensions, for example in \( p = x_1 x_2 x_3 x_4 \) with row vectors \( x_1, x_3 \) and column vectors \( x_2, x_4 \). Here the factorization \( p = (x_1 x_2) (x_3 x_4) \) is of higher importance with respect to the evaluation.

For further references with respect to the application of nc polynomials (and appropriate software for symbolic computations) we refer to [CHS06] and [dO12]. There is a close connection to optimization respectively *semidefinite programming* (SDP) [BPT13], in particular visible in [CKP11]. As a starting point for the evaluation of commutative multivariate polynomials one could take [CS15]. If one has huge arrays of commutative polynomials to evaluate it might be possible to use non-commutativity (in terms of matrix-matrix multiplication) like in [DHM13].

**Acknowledgement**

This work has been partially supported by research subsidies granted by the government of Upper Austria (research project “Methodenentwicklung für Energieflussoptimierung”).

**References**

[Ami66] S. A. Amitsur. Rational identities and applications to algebra and geometry. *J. Algebra*, 3:304–359, 1966.
[BGKR08] H. Bart, I. Gohberg, M. A. Kaashoek, and A. C. M. Ran. *Factorization of matrix and operator functions: the state space method*, volume 178 of *Operator Theory: Advances and Applications*. Birkhäuser Verlag, Basel, 2008. Linear Operators and Linear Systems.

[BPT13] G. Blekherman, P. A. Parrilo, and R. R. Thomas, editors. *Semidefinite optimization and convex algebraic geometry*, volume 13 of *MOS-SIAM Series on Optimization*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA; Mathematical Optimization Society, Philadelphia, PA, 2013.

[BR11] J. Berstel and C. Reutenauer. *Noncommutative rational series with applications*, volume 137 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, 2011.

[BS15] N. R. Baeth and D. Smertnig. Factorization theory: from commutative to noncommutative settings. *J. Algebra*, 441:475–551, 2015.

[CHS06] J. F. Camino, J. W. Helton, and R. E. Skelton. Solving matrix inequalities whose unknowns are matrices. *SIAM J. Optim.*, 17(1):1–36, 2006.

[CKP11] K. Cafuta, I. Klep, and J. Povh. NCSOStools: a computer algebra system for symbolic and numerical computation with noncommutative polynomials. *Optim. Methods Softw.*, 26(3):363–380, 2011.

[Coh63] P. M. Cohn. Noncommutative unique factorization domains. *Trans. Amer. Math. Soc.*, 109:313–331, 1963.

[Coh74] P. M. Cohn. Progress in free associative algebras. *Israel J. Math.*, 19:109–151, 1974.

[Coh75] P. M. Cohn. Algebra and language theory. *Bull. London Math. Soc.*, 7:1–29, 1975.

[Coh95] P. M. Cohn. *Skew fields*, volume 57 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, 1995. Theory of general division rings.

[Coh06] P. M. Cohn. *Free ideal rings and localization in general rings*, volume 3 of *New Mathematical Monographs*. Cambridge University Press, Cambridge, 2006.

[CR94] P. M. Cohn and C. Reutenauer. A normal form in free fields. *Canad. J. Math.*, 46(3):517–531, 1994.

[CR99] P. M. Cohn and C. Reutenauer. On the construction of the free field. *Internat. J. Algebra Comput.*, 9(3-4):307–323, 1999. Dedicated to the memory of Marcel-Paul Schützenberger.
[CS15] J. Czekansky and T. Sauer. The multivariate Horner scheme revisited. *BIT*, 55(4):1043–1056, 2015.

[DDHK07] J. Demmel, I. Dumitriu, O. Holtz, and R. Kleinberg. Fast matrix multiplication is stable. *Numer. Math.*, 106(2):199–224, 2007.

[DHM13] H. Dym, J. W. Helton, and C. Meier. Non-commutative representations of families of $k^2$ commutative polynomials in $2k^2$ commuting variables. *Internat. J. Algebra Comput.*, 23(7):1685–1753, 2013.

[dO12] M. de Oliveira. Simplification of symbolic polynomials on non-commutative variables. *Linear Algebra Appl.*, 437(7):1734–1748, 2012.

[Fri19] FriCAS Computer Algebra System, 2019. W. Hebisch, http://axiom-wiki.newsynthesis.org/FrontPage.

[Gan66] F. R. Gantmacher. *Theorie der Matrizen*. 2., erg. Aufl. Moskau: Verlag 'Nauka'. Hauptredaktion für physikalisch-mathematische Literatur. 576 S., 1966.

[Hig08] N. J. Higham. *Functions of matrices*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2008. Theory and computation.

[Jan18] B. Janko. Factorization of non-commutative Polynomials and Testing Fullness of Matrices. Diplomarbeit, TU Graz, 2018.

[KMP00] M. Konstantinov, V. Mehrmann, and P. Petkov. On properties of Sylvester and Lyapunov operators. *Linear Algebra Appl.*, 312(1-3):35–71, 2000.

[Sch17] K. Schrempf. A factorization theory for some free fields. *arXiv e-prints*, December 2017. Version 2, March 2019, http://arxiv.org/pdf/1712.09102.

[Sch18a] K. Schrempf. A Standard Form in (some) Free Fields: How to construct Minimal Linear Representations. *arXiv e-prints*, March 2018. Version 2, March 2019, http://arxiv.org/pdf/1803.10627.

[Sch18b] K. Schrempf. Free fractions: An invitation to (applied) free fields. *ArXiv e-prints*, September 2018.

[Sch18c] K. Schrempf. Linearizing the word problem in (some) free fields. *Internat. J. Algebra Comput.*, 28(7):1209–1230, 2018.

[Sch19] K. Schrempf. On the factorization of non-commutative polynomials (in free associative algebras). *Journal of Symbolic Computation*, 94:126–148, 2019.

[Sim16] V. Simoncini. Computational methods for linear matrix equations. *SIAM Rev.*, 58(3):377–441, 2016.
[SS78] A. Salomaa and M. Soittola. *Automata-theoretic aspects of formal power series*. Springer-Verlag, New York-Heidelberg, 1978. Texts and Monographs in Computer Science.

[Str69] V. Strassen. Gaussian elimination is not optimal. *Numer. Math.*, 13:354–356, 1969.

[TOT14] S. Tajima, K. Ohara, and A. Terui. An extension and efficient calculation of the Horner’s rule for matrices. In *Mathematical software—ICMS 2014*, volume 8592 of *Lecture Notes in Comput. Sci.*, pages 346–351. Springer, Heidelberg, 2014.