Algorithms for structured approximation and interpolation

Internship report - MPRI

Sara KHICHANE
Université Paris Cité

Supervised by: Vincent NEIGER
At: LIP6, Sorbonne Université

1 Introduction

1.1 General context

Approximation problems are a central topic in computer algebra, with historical roots tracing back to the pioneering work of Hermite and Padé in the 19th century [23, 11]. Given a modulus M and a vector of elements $[f_1, \ldots, f_m]$, the goal is to find a vector of polynomials $[g_1, \ldots, g_m]$ that are a solution to $g_1 f_1 + \ldots + g_m f_m = 0 \bmod M$. This formulation encompasses several fundamental problems, in particular: Hermite-Padé approximation (taking $M = x^d$); Padé approximation (taking $M = x^d$ and $m = 2$), which applies to the search of relations for linearly recurrent sequences [5]; and rational interpolation (taking $M = (x - a_1) \cdots (x - a_d)$, where $a_1, \ldots, a_d$ are distinct known elements of the base field $\mathbb{K}$) [13].
Scientists have explored two main approaches to design efficient algorithms for this problem: one based on polynomial matrix computations, the other on structured linear algebra. Structured matrices are compactly represented by generators. A smaller displacement rank (the number of columns of the generators) yields a stronger structure, enabling faster algorithms for tasks like solving linear systems [3, 1].
In this work, the field $\mathbb{K}$ for coefficients of polynomials and matrices is left unspecified: it can be a prime field $\mathbb{Z}/p\mathbb{Z}$, the rational numbers $\mathbb{Q}$, extensions of them, etc. We focus on the complexity of algorithms performing exact algebraic computations, thus with no issues related to e.g. rounding errors or numerical stability.

1.2 Research problem

Let $d_1, \ldots, d_n, n, m$ be positive integers. We consider the following general approximation problem:

▶ **Problem 1.** ApproxProblem

> ***Input:*** *A modulus denoted by M, $\mathsf{F} = [f_{ij}] \in \mathbb{K}[x]^{n \times m}$, where f_{ij} is of degree strictly less than d_i for $i = 1, \ldots, n$ and $j = 1, \ldots, m$, and $\nu = [\nu_1, \ldots, \nu_m]$ a set of positive integers.*
> ***Output:*** *nonzero $\mathsf{g} = [g_1, \ldots, g_m]^\mathsf{T} \in \mathbb{K}[x]^{m \times 1}$ such that $\mathsf{F} \cdot \mathsf{g} = 0 \bmod M$ and $\deg(g_j) < \nu_j$ for $j = 1, \ldots, m$. If no such vector exists, return $\emptyset$.*

We study two versions of Problem 1, each arising from a different form of the modulus M:

1. ApproxMultMod$(\mathfrak{M}, \mathsf{F}, \nu)$: the modulus is a list of polynomials $\mathfrak{M} = [M_i]_{1 \leq i \leq n} \in \mathbb{K}[x]^n$ with $\deg(M_i) = d_i$.
2. ApproxMatrixMod$(\mathsf{M}, \mathsf{F}, \nu)$: the modulus is a polynomial matrix $\mathsf{M} = [M_{ij}] \in \mathbb{K}[x]^{n \times n}$, with $\deg(M_{ii}) = d_i$ and $\deg(M_{ij}) < d_i$ for all $i \neq j$.

The fastest known algorithms to solve these versions are based on polynomial matrix computations, with complexity $\tilde{O}(\mathsf{max}(n, m)^{\omega-1} \cdot D)$ in [20] and [22]. Here, $D = \sum_{i=1}^n d_i$, $\tilde{O}$ is the O-notation ignoring logarithmic factors and ω is the exponent of matrix multiplication such that $2 \leq \omega \leq 3$ (see Chapter 15 of [6]).
In [7], ApproxMultMod$(\mathfrak{M}, \mathsf{F}, \nu)$ is solved in $\tilde{O}(\mathsf{max}(n, m)^{\omega-1} \cdot D)$, which is the best known complexity. This is done via a reduction to a linear system with a quasi-Toeplitz structure, generalizing the reduction developed in [3] for the special case $n = 1$ (where the modulus is then a single polynomial $M \in \mathbb{K}[x]$).
We next formalize such problems of type $\mathsf{A} \cdot \mathsf{u} = \mathsf{v}$, where A has a quasi-Toeplitz structure. The generators of this structure are defined with respect to specific displacement operators (cyclic down and right shift matrices).

▶ **Problem 2.** LinearSystem

> ***Input:*** *$\mathsf{G} \in \mathbb{K}^{n \times \alpha}, \mathsf{H} \in \mathbb{K}^{m \times \alpha}, \mathsf{v} \in \mathbb{K}^{n \times 1}$ such that G and H are the generators of a quasi-Toeplitz matrix $\mathsf{A} \in \mathbb{K}^{n \times m}$, and $\mathsf{v} = [\mathsf{v}_1, \ldots, \mathsf{v}_m]^\mathsf{T} \in \mathbb{K}^{m \times 1}$ is a vector of coefficients.*
> ***Output:*** *nonzero $\mathsf{u} \in \mathbb{K}^{\alpha \times 1}$ such that $\mathsf{A} \cdot \mathsf{u} = \mathsf{v}$ if it exists, otherwise return $\emptyset$.*

The best known algorithms for solving Problem 2 are given in [3, 1] and run in $\tilde{O}(\alpha^{\omega-1} \cdot \mathsf{max}(n, m))$. These algorithms are randomized and thus require a large enough field $\mathbb{K}$ to ensure that they work with a positive probability of success.

1.3 Contribution

Let $C = \sum_{j=1}^m \nu_j$. We can classify our results into two main categories: the reduction of approximation problems to structured linear systems, and the design of a deterministic algorithm for solving such structured linear systems.

I. Reducing approximation problems to structured linear systems

We present a new reduction of $\mathsf{ApproxMatrixMod}(\mathsf{M}, \mathsf{F}, \nu)$ to a quasi-Toeplitz linear system (Algorithm 4), not previously done in the literature, built from an alternative reduction of $\mathsf{ApproxMultMod}(\mathfrak{M}, \mathsf{F}, \nu)$ (Algorithm 2). Both achieve the best known complexity $\tilde{O}(\mathsf{max}(n,m)^{\omega-1} \cdot D)$ for Problem 1.

The reduction for $\mathsf{ApproxMatrixMod}(\mathsf{M}, \mathsf{F}, \nu)$ is deterministic and relies on *the construction of a quasi-Toeplitz matrix* $\mathsf{A} = [\mathsf{A}_1, \ldots, \mathsf{A}_m] \in \mathbb{K}^{D \times C}$ where for each j, we have

$$\mathsf{A}_j = \begin{bmatrix} \overline{\mathsf{F}}_{*j} & \mathbb{X}(\mathsf{M}) \cdot \overline{\mathsf{F}}_{*j} & \cdots & \mathbb{X}(\mathsf{M})^{\nu_j - 1} \cdot \overline{\mathsf{F}}_{*j} \end{bmatrix} \in \mathbb{K}^{D \times \nu_j}.$$

Here $\overline{\mathsf{F}}_{*j}$ is the vector of coefficients of the j-th column of F and $\mathbb{X}(\mathsf{M})$ a generalization of companion matrices. We compute G, H, generators of A, with only $O(\mathsf{max}(n,m))$ columns, which is key to achieving the target complexity. Solving $\mathsf{LinearSystem}(\mathsf{G}, \mathsf{H}, [0, \ldots, 0]^\mathsf{T})$ yields $\mathsf{u} \in \mathbb{K}^{C \times 1}$ solution of $\mathsf{A} \cdot \mathsf{u} = [0, \ldots, 0]^\mathsf{T}$, from which we recover g, the solution to our approximation problem; u is the vector of coefficients of all polynomials in g.

A central step in the ApproxMatrixMod case is *the computation of the high-degree coefficients of polynomial matrix divisions* $x^k \cdot \mathsf{F}_{*j} \bmod \mathsf{M}$ *for* $k = 1, \ldots, \nu_j - 1$ *and* $j = 1, \ldots, m$. We provide an efficient algorithm for this task (see Theorem 36) that runs in less than $\tilde{O}(\mathsf{max}(n,m)^{\omega-1} \cdot D)$ operations, which is essential for the reduction and, to our knowledge, does not appear in the literature.

II. Solving structured linear systems deterministically

We eliminate the need for randomization and solve Problem 2 deterministically in $\tilde{O}(\alpha^{\omega-1} \cdot \mathsf{max}(n,m))$ (see Algorithm 5). Similarly to [24], which provides polynomial formulations for some linear systems such as Toeplitz (displacement rank 2) and block-Toeplitz, we find a polynomial formulation for any quasi-Toeplitz system, exploiting the known Σ-LU representation of quasi-Toeplitz matrices (that uses the generators). We manage to efficiently solve the resulting equation, with an unknown polynomial, via two extended approximation problems: computing an approximant basis [14] and performing a simultaneous extended Hermite-Padé approximation [25]. Solving them yields the unknown polynomial, from which we recover the solution to Problem 2.

1.4 Arguments supporting its validity

We provide detailed proofs of correctness and complexity for all our algorithms. Our first contribution is most importantly the reduction from ApproxMatrixMod to LinearSystem, achieving the same complexity as the best-known polynomial-approach based algorithms (see [20], [22] respectively). This shows that the structured approach can be as efficient as the polynomial one, even for a general modulus M, which provides more clarity on the relation between both frameworks. The reduction holds for any nonsingular M. Note that, the degree constraints on M can be lifted without loss of generality, as any instance with a nonsingular M can be reduced to one satisfying these constraints (see Appendix B).

Our second contribution, where we present a deterministic algorithm for solving quasi-Toeplitz linear systems, is a clear improvement as it eliminates the need for randomization and thus provides a robust solution over any field $\mathbb{K}$. We show that it runs in $\tilde{O}(\alpha^{\omega-1} \cdot \mathsf{max}(n,m))$, matching the best-known randomized bounds.

1.5 Summary and future work

We present a summary of our results in Figure 1. We got a better understanding of the links between the structured approach and the polynomial matrix computations one, and removed the need for randomization in solving quasi-Toeplitz linear systems. Our results open the door to further research in this area.

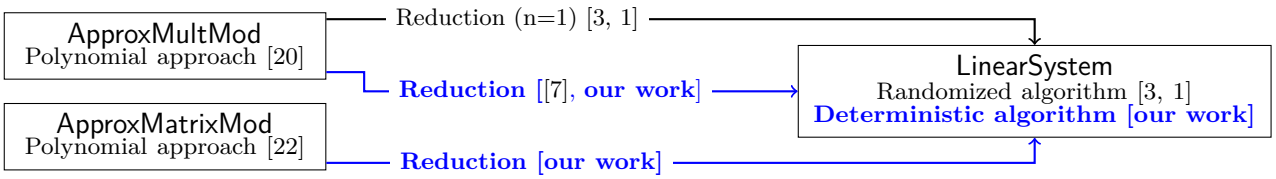

Figure 1 Summary of the results

In particular, our next step is to extend our last result to other structured linear systems, such as quasi-Vandermonde systems and other variations of the displacement operators introduced in [1]. We also aim to provide finer complexity bounds for our algorithms and thus avoid the $\tilde{O}$ notation. Longer-term goals include computing a basis of solutions, rather than a single solution, for our approximation problems, and studying cases with multiple nested structures, such as block-Hankel matrices, or approximations involving iterated powers or derivatives. For example, one can consider Problem 1 with $\mathsf{F} = [1, p, p^2, \ldots, p^{m-1}]$ for some $p \in \mathbb{K}[x]$.

2 Preliminaries

In this section, we introduce some notions and existing results that are the basis of our work. We classify them into two categories: structured matrices related notations and polynomial matrices ones. We consider for the rest of the report that n and m are positive integers. We start with some general notations.
We consider $\mathsf{coeff}(p, k)$ the coefficient of the polynomial $p \in \mathbb{K}[x]$ with respect to the monomial x^k and $\mathsf{coeff}(\mathsf{p}, i, k)$ as the coefficient of the polynomial, at the k-th column of the vector $\mathsf{p} \in \mathbb{K}[x]^{n\times 1}$, with respect to the monomial x^k.
For a given matrix M, we denote by M_{i*} the i-th row of the matrix M and by M_{*j} the j-th column of the matrix M. Moreover, we define $\mathsf{elem}(\mathsf{p}, i)$ as the element at row i of a vector p and $\mathsf{elem}(\mathsf{A}, i, j)$ as the element at row i and column j of a matrix A.
For a polynomial $p \in \mathbb{K}[x]$, we denote by $\mathsf{deg}(p)$ the degree of the polynomial p such that $\mathsf{deg}(p) = -\infty$ if $p = 0$. Moreover, we define $\mathsf{rev}_k(p)$ as the polynomial p reversed with respect to the monomial x^k, i.e., $\mathsf{rev}_k(p) = \sum_{i=0}^{\mathsf{deg}(p)} \mathsf{coeff}(p, i)x^{k-i}$. We extend this definition to a vector by reversing each polynomial in it.
The notation $P = \phi \bmod M$ means that there exists Q such that $P = M \cdot Q + \phi$, where P and M will be defined in the context (see Remark 18). If $\phi = 0$ then P is divisible by M.
For a polynomial matrix $\mathsf{P} \in \mathbb{K}[x]^{n\times m}$ such that $\mathsf{P} = [p_{ij}]_{j=1,\dots,m}^{i=1,\dots,n}$, we have $\mathsf{P}(x^{-1}) = [p_{ij}(x^{-1})]_{j=1,\dots,m}^{i=1,\dots,n}$.
We use $\mathsf{diag}(A_1, \dots, A_n)$ to denote the block diagonal matrix with blocks $A_1, \dots, A_n$ on the diagonal.
We denote by $|L|$ the number of elements of a list L and by $\sum(L)$ the sum of elements of L.
As the notion of probabilistic algorithms is mentioned, we define it as follows.

▶ **Definition 3.** *An algorithm is* probabilistic of type $P(k, d)$ *if when choosing less than k elements in $\mathbb{K}$, the probability of success is at least $1 - \frac{d}{|\mathbb{K}|}$.*

In regards to ω, the exponent of matrix multiplication, i.e., the smallest real number such that two $n \times n$ matrices can be multiplied in $O(n^\omega)$ operations in $\mathbb{K}$, we have that the best known value is $\omega < 2.371552$ [29]. $\mathcal{M}(n)$ is the cost of multiplying two polynomials in $\mathbb{K}[x]_{\leq n}$. Note that $O(\mathcal{M}(n)) = \tilde{O}(n)$.

2.1 Structured matrices

Matrices with a specific structure play a crucial role in the design of efficient algorithms in computer algebra. For instance, a structured matrix with size $n \times n$ will be represented by $O(n)$ elements instead of n^2 elements in the general case. This reduction in the number of elements yields significant improvements in the complexity of algorithms that manipulate these matrices, such as solving a linear system. Some structured matrices are well-known, such as *Toeplitz matrices*, that are defined by their first row and first column, and are invariant along diagonals. However, the notion of structure is not limited to specific cases. In fact, any matrix that can be represented by a small number of elements can be considered structured. As introduced in [16], we define the notion of structure as the measure of to what extent a matrix can be represented by a small number of elements. We call this measure the *displacement rank*.

▶ **Definition 4.** *Let $\mathsf{P} \in \mathbb{K}^{n\times n}$ and $\mathsf{Q} \in \mathbb{K}^{m\times m}$ be two matrices. We define the* displacement operator $\Delta[\mathsf{P}, \mathsf{Q}]$ *as follows:*

$$\Delta[\mathsf{P}, \mathsf{Q}] : \begin{array}{ll} \mathbb{K}^{n\times m} & \longrightarrow \mathbb{K}^{n\times m} \\ \mathsf{A} & \longmapsto \mathsf{A} - \mathsf{P} \cdot \mathsf{A} \cdot \mathsf{Q} \end{array}$$

▶ **Definition 5.** *Let $\mathsf{P} \in \mathbb{K}^{n\times n}$ and $\mathsf{Q} \in \mathbb{K}^{m\times m}$ be two matrices. We define the* displacement rank *of a matrix $\mathsf{A} \in \mathbb{K}^{n\times m}$ with respect to the matrices P and Q as the rank of $\Delta[\mathsf{P}, \mathsf{Q}](\mathsf{A})$.*

▶ **Definition 6.** *Let $\mathsf{P} \in \mathbb{K}^{n\times n}$ and $\mathsf{Q} \in \mathbb{K}^{m\times m}$ be two matrices. We call two matrices $\mathsf{Y} \in \mathbb{K}^{n\times\alpha}$ and $\mathsf{Z} \in \mathbb{K}^{m\times\alpha}$* P, Q-generators of length α *for a matrix $\mathsf{A} \in \mathbb{K}^{n\times m}$, if they satisfy the following property:*

$$\Delta[\mathsf{P}, \mathsf{Q}](\mathsf{A}) = \mathsf{Y} \cdot \mathsf{Z}^\mathsf{T}$$

The goal of this notion of structure is to use these generators as entries for problems such as linear systems instead of the matrix A itself. Therefore, if the displacement rank α is smaller than the size of the matrix A, the number of elements to manipulate is reduced. Note that if $\alpha \in O(n, m)$, then A has no structure.

▶ **Definition 7.** *We define* the cyclic down-shift matrix $\mathbb{Z}_{n,\varphi} \in \mathbb{K}^{n\times n}$*, for a given* $\varphi \in \mathbb{K}$*, as follows:*

$$\mathbb{Z}_{n,\varphi} = \begin{bmatrix} 0 & 0 & \dots & 0 & \varphi \\ 1 & 0 & \dots & 0 & 0 \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & \cdots & 1 & 0 & 0 \end{bmatrix}$$

▶ **Definition 8.** *Let* $\mathsf{A} \in \mathbb{K}^{n\times m}$*. We define* the quasi-Toeplitz structure *as the structure that corresponds to the operators* $\mathbb{Z}_{n,0}$ *and* $\mathbb{Z}_{m,0}^{\mathsf{T}}$*.*
We denote by $\phi_+(\mathsf{A})$ the displacement operator associated with the quasi-Toeplitz structure*, i.e.* $\phi_+(\mathsf{A}) = \mathsf{A} - \mathbb{Z}_{n,0} \cdot \mathsf{A} \cdot \mathbb{Z}_{m,0}^{\mathsf{T}}$*, which corresponds to* $\mathsf{A} -$ (A *shifted down and right by one unit*)*.*
We call two matrices $\mathsf{Y} \in \mathbb{K}^{n\times\alpha}$ *and* $\mathsf{Z} \in \mathbb{K}^{m\times\alpha}$ ϕ_+-generators of length α *for* A*, if* $\phi_+(\mathsf{A}) = \mathsf{Y} \cdot \mathsf{Z}^{\mathsf{T}}$*.*

One can represent the classical structured matrices (Toeplitz, Hankel, Vandermonde, Sylvester, Cauchy, etc.) using the right displacement operators. For instance, a Toeplitz matrix is a quasi-Toeplitz matrix with a displacement rank of 2.

▶ **Definition 9.** *Let* $M \in \mathbb{K}[x]$ *be a polynomial of degree at most* n*. We define* $\mathbb{X}(M)$ *the* matrix of multiplication by x *in* $\mathbb{K}[x]/\langle M\rangle$*, commonly called companion matrix in the literature (see [15, 24]), as follows:*

$$\mathbb{X}(M) = \begin{bmatrix} 0 & \cdots & 0 & -m_0 \\ 1 & \cdots & 0 & -m_1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 1 & -m_{n-1} \end{bmatrix}$$

where $m_i = \mathsf{coeff}(M, i)$ *for* $i = 0, \dots, n-1$*.*

▶ **Definition 10.** *Let* $\mathsf{v} \in \mathbb{K}^{n\times 1}$ *a vector of elements* $v_1, \dots, v_n \in \mathbb{K}$*. We define* the lower Toeplitz matrix $\mathbb{L}(\mathsf{v}) \in \mathbb{K}^{n\times n}$ *as follows:*

$$\mathbb{L}(\mathsf{v}) = \begin{bmatrix} v_1 & 0 & \cdots & 0 \\ v_2 & v_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ v_n & v_{n-1} & \cdots & v_1 \end{bmatrix}$$

Likewise, we define the upper Toeplitz matrix $\mathbb{U}(\mathsf{v}) \in \mathbb{K}^{n\times n}$ *as* $\mathbb{U}(\mathsf{v}) = \mathbb{L}(\mathsf{v})^{\mathsf{T}}$*.*

For the rest of this subsection, we consider α a positive integer such that $\alpha \leq n$.

▶ **Definition 11.** *Let* $\mathsf{P} \in \mathbb{K}^{n\times n}$*,* $\mathsf{Q} \in \mathbb{K}^{m\times m}$ *and* $\mathsf{v} \in \mathbb{K}^{n\times 1}$*. Given* $\mathsf{Y} \in \mathbb{K}^{n\times\alpha}$ *and* $\mathsf{Z} \in \mathbb{K}^{m\times\alpha}$*;* P, Q*-generators of length* α *for a matrix* $\mathsf{A} \in \mathbb{K}^{n\times m}$*, we define the problem* $\mathsf{LinearSystem}[\mathsf{P}, \mathsf{Q}, \alpha](\mathsf{Y}, \mathsf{Z}, \mathsf{v})$ *as follows:*

$\mathsf{LinearSystem}[\mathsf{P}, \mathsf{Q}, \alpha]$	
Input:	$\mathsf{Y} \in \mathbb{K}^{n\times\alpha}, \mathsf{Z} \in \mathbb{K}^{m\times\alpha}, \mathsf{v} \in \mathbb{K}^{n\times 1}$.
Output:	*nonzero* $\mathsf{u} \in \mathbb{K}^{m\times 1}$ *such that* $\mathsf{A} \cdot \mathsf{u} = \mathsf{v}$ *if it exists, otherwise return* $\emptyset$*.*

For ease of notation, when v *is the zero vector, we denote this problem as* $\mathsf{LinearSystem}[\mathsf{P}, \mathsf{Q}, \alpha](\mathsf{Y}, \mathsf{Z})$*. Note that, the solution* u *has to be nonzero, otherwise the problem in this case becomes trivial.*
Furthermore, we only specify the parameters P*,* Q *and* α *when clarity is needed, otherwise we denote the problem as* $\mathsf{LinearSystem}(\mathsf{Y}, \mathsf{Z}, \mathsf{v})$*. The parameters* P*,* Q *and* α *can be retrieved from the inputs.*

Kailath, Kung and Morf managed to solve quasi-Toeplitz systems in $O(\alpha \cdot n^2)$ (see [16]). A more rapid version was given by Morf in [19] achieving a complexity of $O(\alpha^2 \cdot \mathcal{M}(n) \cdot \log(n))$, under some hypotheses that Kaltofen managed to remove in [17]. The complexity improved significantly in [3], where authors managed to introduce rapid matrix multiplication and show that the problem $\mathsf{LinearSystem}[\mathbb{Z}_{n,0}, \mathbb{Z}_{n,0}^{\mathsf{T}}, \alpha]$ for any $\mathbb{Z}_{n,0}, \mathbb{Z}_{n,0}^{\mathsf{T}}$-generators and any $\mathsf{v} \in \mathbb{K}^{n\times 1}$ can be solved in $O(\alpha^{\omega-1} \cdot \mathcal{M}(n) \cdot \log^2(n))$, by a probabilistic algorithm of type $P(3n-2, n^2+n)$. Later in [1], they generalized this result to the case of non-square systems and improved the complexity by a logarithmic factor, as follows.

▶ **Theorem 12** (Theorem 1.1 [1]). *The problem* LinearSystem$[\mathbb{Z}_{n,0}, \mathbb{Z}^{\mathsf{T}}_{m,0}, \alpha]$ *for any* $\mathbb{Z}_{n,0}, \mathbb{Z}^{\mathsf{T}}_{m,0}$*-generators and any* $\mathsf{v} \in \mathbb{K}^{n\times 1}$ *can be solved in* $O(\alpha^{\omega-1} \cdot \mathcal{M}(\mathsf{max}(n,m)) \cdot \log(\frac{\mathsf{max}(n,m)}{\alpha}))$*, by a probabilistic algorithm of type* $P(m+n-2, \mathsf{max}(m,n)^2 + \mathsf{max}(m,n))$.

A significant result of the previous theorem is the approximation problem defined in Problem 1 with $n = 1$.

▶ **Corollary 13** (Corollary 1 [3]). *Let* $M \in \mathbb{K}[x]$ *be a polynomial of degree* d*, let* $f_1, \ldots, f_m \in \mathbb{K}[x]$ *be polynomials of degrees strictly less than* d*, and let* $\nu_1, \ldots, \nu_m \in \mathbb{N}$ *be such that* $\sum_{i=1}^{m} \nu_i = d+1$*. One can find* $g_1, \ldots, g_m \in \mathbb{K}[x]$*, not all zero, of respective degrees strictly less than* $\nu_1, \ldots, \nu_m$*, such that*

$$g_1 f_1 + \cdots + g_m f_m = 0 \bmod M,$$

in time $O(m^{\omega-1} \cdot \mathcal{M}(d) \cdot \log^2(d))$*. The algorithm is probabilistic of type* $P(3d-2, d^2+d)$.

2.2 Polynomial matrices

Polynomial matrices are matrices whose entries are polynomials. Their theory was first introduced by Gantmacher in [9] and [18]. It has been, since, applied in the context of linear systems and approximation problems. In this section, we introduce some basic definitions and properties related to polynomial matrices, which will be useful in our study. We start with row reduced matrices (or their column reduced counterparts). The notion was introduced by Wolovich in [30] but the term was suggested by Heymann in [12].

▶ **Definition 14.** *Let* $\mathsf{M} = [M_{ij}]_{i,j=1,\ldots,n} \in \mathbb{K}[x]^{n\times m}$ *be a polynomial matrix.*
Given $s = [s_1, \ldots, s_m]$ *a list of non-negative integers, we define* $\mathsf{rdeg}_s(M)$ *as the list* $[d_1, \ldots, d_n]$ *where* $d_i = \mathsf{max}_{j=1,\ldots,m}(\mathsf{deg}(M_{ij}) + s_j)$ *for* $i = 1, \ldots, n$.
Given $s = [s_1, \ldots, s_n]$ *a list of non-negative integers, we define* $\mathsf{cdeg}_s(M)$ *as the list* $[d_1, \ldots, d_m]$ *where* $d_j = \mathsf{max}_{i=1,\ldots,n}(\mathsf{deg}(M_{ij}) + s_i)$ *for* $j = 1, \ldots, m$.
We define $\mathsf{deg}(M)$ *as the maximum degree of the entries of the polynomial matrix* M*, i.e.,* $\mathsf{deg}(M) = \mathsf{max}^{i=1,\ldots,n}_{j=1,\ldots,m} \mathsf{deg}(M_{ij})$.

For ease of notation, if $s = [0, \ldots, 0]$, we write $\mathsf{rdeg}(M)$ for $\mathsf{rdeg}_0(M)$ and $\mathsf{cdeg}(M)$ for $\mathsf{cdeg}_0(M)$.

▶ **Definition 15.** *Let* $\mathsf{M} = [M_{ij}]_{i,j=1,\ldots,n} \in \mathbb{K}[x]^{n\times n}$ *be a polynomial matrix,* $s = [s_1, \ldots, s_m]$ *be a list of non-negative integers and* $\mathsf{rdeg}_s(M) = [d_1, \ldots, d_n]$*. We define* the row leading coefficients matrix of M with respect to s *as the matrix* $[\mathsf{coeff}(M_{ij}, d_i - s_j)]_{i,j=1,\ldots,n} \in \mathbb{K}^{n\times n}$*. We denote this matrix by* $\mathsf{rlm}_s(\mathsf{M})$.

▶ **Definition 16.** *Let* $s = [s_1, \ldots, s_n]$ *be a list of non-negative integers. A nonsingular matrix* $\mathsf{M} = [M_{ij}]_{i,j=1,\ldots,n} \in \mathbb{K}[x]^{n\times n}$*, such that* $\mathsf{rdeg}_s(M) = [d_1, \ldots, d_n]$ *is called* s-row reduced *if the matrix* rlm_s *is invertible. We say that* M *is* row reduced *if* $s = [0, \ldots, 0]$.

For instance, the matrix $\mathsf{M} = \begin{bmatrix} x & x+1 \\ x & x^2 \end{bmatrix}$ is row reduced, while the matrix $\mathsf{M} = \begin{bmatrix} x & x^2 \\ x & x^3 \end{bmatrix}$ is not row reduced.
The following theorem is the counterpart of Theorem 6.3.15 in [15], where the division is done on the right and the matrix we divide by is column reduced.

▶ **Theorem 17.** *For any row reduced* $\mathsf{M} = [M_{ij}]_{i,j=1\ldots n} \in \mathbb{K}[x]^{n\times n}$*, and any* $\mathsf{F} = [f_{ij}]^{i=1,\ldots,n}_{j=1,\ldots,m} \in \mathbb{K}[x]^{n\times m}$*, there exist unique matrices* $\mathsf{Q} = [q_{ij}]^{i=1,\ldots,n}_{j=1,\ldots,m} \in \mathbb{K}[x]^{n\times m}$ *and* $\mathsf{R} = [r_{ij}]^{i=1,\ldots,n}_{j=1,\ldots,m} \in \mathbb{K}[x]^{n\times m}$ *such that* $\mathsf{rdeg}(\mathsf{R}) < \mathsf{rdeg}(\mathsf{M})$ *and*

$$\mathsf{F} = \mathsf{M} \cdot \mathsf{Q} + \mathsf{R},$$

where R *is the left remainder matrix and* Q *is the left quotient matrix.*
We consider the left euclidean division of a polynomial matrix F *by a polynomial matrix* M *to be the default division. Thus, we denote by* F quo M *the matrix quotient* Q *and by* F rem M *the matrix remainder* R *for the left euclidean division of the polynomial matrix* F *by the polynomial matrix* M.

Similarly, we denote by F rrem M and F rquo M the right remainder matrix R and the right quotient matrix Q respectively, for the right division of the polynomial matrix F by the polynomial matrix M. We refer to Appendix A for the formal definition of a column reduced matrix and the right euclidean division theorem.

▶ Remark 18. We use the notation R mod M to denote the class of equivalence, i.e. all F such that $\mathsf{F} = \mathsf{M}\cdot\mathsf{Q}+\mathsf{R}$ for some Q. We use, however, rem to denote the remainder of a division.

▶ **Lemma 19.** *Let* $\mathsf{M} = [M_{ij}]_{i,j=1,\dots,n} \in \mathbb{K}[x]^{n\times n}$ *a row reduced polynomial matrix such that* $\mathsf{rdeg}(\mathsf{M}) = [d_1,\dots,d_n]$. *Let* $\mathsf{F} = [f_{ij}]_{i=1,\dots,n}^{j=1,\dots,m} \in \mathbb{K}[x]^{n\times m}$, *such that* $\mathsf{rdeg}(\mathsf{F}) < [d_1+k,\dots,d_n+k]$ *where* k *is a positive integer. Let* $\mathsf{Q} = [q_{ij}]_{j=1,\dots,m}^{i=1,\dots,n} \in \mathbb{K}[x]^{n\times m}$ *be* F quo M. *Then, we have* $\mathsf{deg}(\mathsf{Q}) < k$.

Proof. We start this proof by proving the following property: $\mathsf{rlm}_0(\mathsf{M}) = \mathsf{clm}_{-\mathsf{rdeg}(\mathsf{M})}(\mathsf{M})$.
We have on one side, $\forall i,j \in \{1,\dots,n\}$:

$$\begin{aligned}\mathsf{elem}(\mathsf{rlm}_0(\mathsf{M}),i,j) &= \mathsf{elem}(\mathsf{clm}_0(\mathsf{M}^\mathsf{T})^\mathsf{T},i,j) = \mathsf{elem}(\mathsf{clm}_0(\mathsf{M}^\mathsf{T}),j,i)\\ &= \mathsf{coeff}(M_{ij},\mathsf{elem}(\mathsf{cdeg}(\mathsf{M}^\mathsf{T}),j)) = \mathsf{coeff}(M_{ij},\mathsf{max}_{k=1,\dots,n}(\mathsf{deg}(M_{kj}^\mathsf{T})+0))\\ &= \mathsf{coeff}(M_{ij},\mathsf{max}_{k=1,\dots,n}(\mathsf{deg}(M_{jk}))) = \mathsf{coeff}(M_{ij},\mathsf{elem}(\mathsf{rdeg}(\mathsf{M}),j))\end{aligned}$$

On the other side, we have $\forall i,j \in \{1,\dots,n\}$:

$$\begin{aligned}\mathsf{elem}(\mathsf{clm}_{-\mathsf{rdeg}(\mathsf{M})}(\mathsf{M}),i,j) &= \mathsf{coeff}(M_{ij},\mathsf{elem}(\mathsf{cdeg}_{-\mathsf{rdeg}(\mathsf{M})}(\mathsf{M}),i)+\mathsf{elem}(\mathsf{rdeg}(\mathsf{M}),j)) \text{ (by Definition 15)}\\ &= \mathsf{coeff}(M_{ij},\mathsf{max}_{k=1,\dots,n}(\mathsf{deg}(M_{ki})-\mathsf{elem}(\mathsf{rdeg}(\mathsf{M}),k))+\mathsf{elem}(\mathsf{rdeg}(\mathsf{M}),j))\\ &= \mathsf{coeff}(M_{ij},\mathsf{max}_{k=1,\dots,n}(\mathsf{deg}(M_{ki})-\mathsf{max}_{\ell=1,\dots,n}(\mathsf{deg}(M_{k\ell})+0))+\mathsf{elem}(\mathsf{rdeg}(\mathsf{M}),j))\\ &= \mathsf{coeff}(M_{ij},\mathsf{elem}(\mathsf{rdeg}(\mathsf{M}),j)) \text{ (since } \mathsf{deg}(M_{ki}) \le \mathsf{max}_{\ell=1,\dots,n}(\mathsf{deg}(M_{k\ell})) \text{ for all } i,k)\end{aligned}$$

Thus, we conclude that $\mathsf{clm}_0(\mathsf{M}) = \mathsf{rlm}_{-\mathsf{cdeg}(\mathsf{M})}(\mathsf{M})$.
Now, we show that $\mathsf{cdeg}_{-\mathsf{rdeg}(\mathsf{M})}(\mathsf{M}) = [0,\dots,0]$. Since we shift the degrees of each row of M by its leading coefficient degree, we have that all degrees after the shifts are either 0 or negative. We consider that an entry with negative degree is equivalent to a zero entry and thus has a degree of $-\infty$. Therefore, the only possible values for the degrees of the entries of $\mathsf{cdeg}_{-\mathsf{rdeg}(\mathsf{M})}(\mathsf{M})$ are 0 or $-\infty$. Moreover, if we have $-\infty$ in the degree of an entry, then all degrees of that column in the shifted matrix are $-\infty$, which means that the column is entirely zero. If we assume this, then $\mathsf{clm}_{\mathsf{rdeg}(\mathsf{M})}(\mathsf{M})$ is a matrix with a column of zeros, which implies that the matrix is not invertible. This is a contradiction since we know that $\mathsf{rlm}_0(\mathsf{M}) = \mathsf{clm}_{-\mathsf{rdeg}(\mathsf{M})}(\mathsf{M})$ by the above and $\mathsf{rlm}_0(\mathsf{M})$ is invertible by Definition 15. Thus, we conclude that $\mathsf{cdeg}_{-\mathsf{rdeg}(\mathsf{M})}(\mathsf{M}) = [0,\dots,0]$.
Let $\mathsf{R} = \mathsf{F}$ rem M. By Theorem 17, we have $\mathsf{F} = \mathsf{M}\cdot\mathsf{Q}+\mathsf{R}$ and $\mathsf{rdeg}(\mathsf{R}) < [d_1,\dots,d_n]$. Since $\mathsf{F}-\mathsf{R} = \mathsf{M}\cdot\mathsf{Q}$, we have $\mathsf{rdeg}(\mathsf{F}-\mathsf{R}) = \mathsf{rdeg}(\mathsf{M}\cdot\mathsf{Q})$. We use the fact that $\mathsf{rdeg}(\mathsf{F}) < [d_1+k,\dots,d_n+k]$ and $\mathsf{rdeg}(\mathsf{R}) < [d_1,\dots,d_n]$ to deduce $\mathsf{rdeg}(\mathsf{F}-\mathsf{R}) < [d_1+k,\dots,d_n+k]$ and thus $\mathsf{rdeg}(\mathsf{M}\cdot\mathsf{Q}) < [d_1+k,\dots,d_n+k]$. By the same argument as above, we have $\mathsf{cdeg}_{-\mathsf{rdeg}(\mathsf{M})}(\mathsf{M}\cdot\mathsf{Q}) < [k,\dots,k]$.
Since M is $-\mathsf{rdeg}(\mathsf{M})$-column reduced (as it is row reduced) and $\mathsf{cdeg}_{-\mathsf{rdeg}(\mathsf{M})}(\mathsf{M}) = [0,\dots,0]$, we can apply the predictable degree property (see Theorem 6.3.13 in [15]) on each column Q_{*j} of Q for $j = 1,\dots,m$ to get:

$$\begin{aligned}\mathsf{cdeg}_{-\mathsf{rdeg}(\mathsf{M})}(\mathsf{M}\cdot Q_{*j}) &= \mathsf{max}_{i=1,\dots,n}(\mathsf{deg}(Q_{ij})+\mathsf{cdeg}_{-\mathsf{rdeg}(\mathsf{M})}(M_{*j}))\\ &= \mathsf{max}_{i=1,\dots,n}(\mathsf{deg}(Q_{ij})+0) = \mathsf{cdeg}(Q_{*j}).\end{aligned}$$

Thus, we have $\mathsf{cdeg}(Q_{*j}) < k$ for $j = 1,\dots,n$, which means that $\mathsf{cdeg}(\mathsf{Q}) < [k,\dots,k]$. Finally, we conclude with $\mathsf{deg}(\mathsf{Q}) = \mathsf{max}(\mathsf{cdeg}(\mathsf{Q})) < k$. ◀

Let $\mathbb{I}_n$ be the identity matrix of size n.

▶ **Definition 20.** *A square polynomial matrix* $\mathsf{U} \in K[x]^{n\times n}$ *is called* unimodular *if it is invertible over* $\mathbb{K}[x]$, *i.e., there exists a polynomial matrix* $\mathsf{V} \in K[x]^{n\times n}$ *such that* $\mathsf{U}\cdot\mathsf{V} = \mathbb{I}_n$. *Equivalently, the determinant of* U *is a nonzero element of* $\mathbb{K}$.

▶ **Definition 21.** *Let* $s = [s_1,\dots,s_n]$ *be a list of non-negative integers. A polynomial matrix* $\mathsf{M} \in K[x]^{n\times n}$ *is in* s-Popov form *if* $\mathsf{clm}_0(\mathsf{M}) = \mathbb{I}_n$ *and* $\mathsf{rlm}_s(\mathsf{M})$ *is a unit lower triangular matrix (lower triangular matrix with all diagonal entries equal to* 1*). We say that* M *is in* Popov form *if* $s = [0,\dots,0]$.

We introduce polynomial matrices with a specific structure, which will be useful in the design of our algorithms. We call these matrices *modulus polynomial matrices*.

▶ **Definition 22.** *Let* $\mathsf{M} = [M_{ij}]_{i,j=1,\ldots,n} \in \mathbb{K}[x]^{n\times n}$ *be a polynomial matrix, such that* $\mathsf{rdeg}(\mathsf{M}) = [d_1, \ldots, d_n]$. *We call* M *a* modulus polynomial matrix *if for* $i, j = 1, \ldots, n$, *we have:*

- M_{ii} *is a monic polynomial of degree* d_i,
- M_{ij} *is a polynomial of degree strictly less than* d_i *for all* $j \neq i$.

▶ **Lemma 23.** *Given a modulus polynomial matrix* M, M *is a row reduced polynomial matrix.*

Proof. $\mathsf{rlm}_0(\mathsf{M}) = \mathbb{I}_n$, as the diagonal entries are monic polynomials of degree d_i and the off-diagonal entries at row i are polynomials of degree strictly less than d_i. Thus, $\mathsf{rlm}_0(\mathsf{M})$ is nonsingular and by Definition 16, M is row reduced. ◀

3 Multiple modular approximations

In this section, we solve efficiently multiple simultaneous approximation problems, i.e. where different moduli intervene. We use structured linear algebra techniques to achieve the same bound, that polynomial computations managed to get (see [20]). Note that a reduction to a structured linear system was done [7] in the same complexity bound. Here, we provide an alternative reduction to a quasi-Toeplitz system. We consider the approximation problem defined in Definition 24.

▶ **Definition 24.** *Given*

- $\mathfrak{M} = M_1, \ldots, M_n$ *of degrees* $d_1, \ldots, d_n$ *respectively, such that* $M_i \in \mathbb{K}[x]$ *for* $i = 1, \ldots, n$ *and* $d_1, \ldots, d_n$ *positive integers.*
- $\mathsf{F} = [f_{ij}]_{j=1,\ldots,m}^{i=1,\ldots,n} \in \mathbb{K}[x]^{n\times m}$, *such that* $\mathsf{rdeg}(\mathsf{F}) < [d_1, \ldots, d_n]$.
- $\nu = \{\nu_1, \ldots, \nu_m\}$ *be a set of positive integers.*

We define the set of solutions

$$S = \left\{ \mathsf{g} \in \mathbb{K}[x]^{m\times 1} \mid \mathsf{F}\cdot\mathsf{g} = 0 \bmod \mathfrak{M} \right\}, \tag{1}$$

where $\mathsf{F}\cdot\mathsf{g} = 0 \bmod \mathfrak{M}$ *means that* $\mathsf{F}_{i*}\cdot\mathsf{g} = 0 \bmod M_i$ *(i.e.,* $\exists q \in \mathbb{K}[x]$ *such that* $\mathsf{F}_{i*}\cdot\mathsf{g} = q\cdot M_i$*) for* $i = 1, \ldots, n$. *We define the problem* $\mathsf{ApproxMultMod}(\mathfrak{M}, \mathsf{F}, \nu)$, *as follows:*

ApproxMultMod	
Input:	$\mathfrak{M} \in \mathbb{K}[x]^n$, $\mathsf{F} \in \mathbb{K}[x]^{n\times m}$, $\nu = \{\nu_1, \ldots, \nu_m\}$.
Output:	$\mathsf{g} \in S$ *such that* g *is not the zero vector and* $\mathsf{deg}(g_j) < \nu_j$ *for* $j = 1, \ldots, m$.

We denote by $C = \sum_{j=1}^{m} \nu_j$ *the total number of unknown coefficients of the unknown polynomial vector* g *and by* $D = \sum_{i=1}^{n} d_i$ *the total degree of the moduli polynomials* M_i *for* $i = 1, \ldots, n$.

3.1 Intermediate results: modular polynomial multiplication

The results presented in this section were used in structured linear systems related works, such as [3] and [1]. We present them here for completeness and to provide detailed proofs.

▶ **Lemma 25.** *Let* $M \in \mathbb{K}[x]$ *of degree* $d > 0$, *and* $P \in \mathbb{K}[x]$ *of degree less than* d. *Let* $\nu > 0$. *One can compute the coefficients of* $x^\nu P \mathsf{\ rem\ } M$ *in* $O(\mathcal{M}(d))$ *if* $\nu < d$ *and more generally in* $O(\mathcal{M}(d)\cdot\log(\nu))$ *even if* $\nu \gg d$.

Proof. We distinguish two cases:

- If $\nu < d$, we can compute $x^\nu P$ and then reduce it modulo M using the fast polynomial division algorithm [27]. The cost is $O(\mathcal{M}(d))$.
- If $\nu \gg d$, we use the fact that $x^\nu P \mathsf{\ rem\ } M = ((x^\nu \mathsf{\ rem\ } M)\cdot(P \mathsf{\ rem\ } M)) \mathsf{\ rem\ } M$. We can compute $x^\nu \mathsf{\ rem\ } M$ in $O(\mathcal{M}(d)\cdot\log(\nu))$ using a fast modular polynomial exponentiation algorithm. The fastest known algorithm for this is presented in [4], and has a cost of $2(\mathcal{M}(d)\cdot\log(\nu))$. Since P is of degree less than d, we have $P \mathsf{\ rem\ } M = P$. Moreover, $x^\nu \mathsf{\ rem\ } M$ is also a polynomial of degree less than d, thus we can compute the product $(x^\nu \mathsf{\ rem\ } M)\cdot P$ in $O(\mathcal{M}(d))$ and get a polynomial of degree less than $2d$. Finally, we reduce the result modulo M using the fast polynomial division algorithm [27] in $O(\mathcal{M}(d))$. Therefore, the total cost is bounded by $O(\mathcal{M}(d)\cdot\log(\nu))$. ◀

Algorithm 1 Coefficient $d-1$ of each of the polynomials $x^k P \bmod M$ for $k = 0, \dots, \nu$

Input: $M = m_0 + m_1 x + \cdots + m_d x^d$, $P = p_0 + p_1 x + \cdots + p_{d-1} x^{d-1}$, ν
Output: $[\mathsf{coeff}(P \bmod M, d-1), \mathsf{coeff}(xP \bmod M, d-1), \dots, \mathsf{coeff}(x^{\nu-1}P \bmod M, d-1)]$

$g = \mathsf{rev}_d(M) \bmod x^\nu$;
$h = \mathsf{rev}_{d-1}(P) \bmod x^\nu$;
$f = g^{-1} \cdot h \bmod x^\nu$;
for $i = 0$ **to** $\nu - 1$ **do**
  $a_i = \mathsf{coeff}(f, i)$;
return $[a_0, \dots, a_{\nu-1}]$;

▶ **Lemma 26.** *Let $M \in \mathbb{K}[x]$ of degree $d > 0$, and $P \in \mathbb{K}[x]$ of degree less than d. Let $\nu > 0$. One can compute $\mathsf{coeff}(x^k P \bmod M, d-1)$, for $k = 0, \dots, \nu-1$ in $O(\mathcal{M}(\nu))$ operations in $\mathbb{K}$ using Algorithm 1.*

Proof. We denote by $\mathsf{a}^\mathsf{T} = [a_0, \dots, a_{\nu-1}] \in \mathbb{K}^{1\times\nu}$ the vector where $a_k = \mathsf{coeff}(x^k P \bmod M, d-1)$ for $k = 0, \dots, \nu-1$. The goal is to compute the vector a.

Correctness

For M, P and k, There exist unique Q_k, R_k such that $x^k \cdot P = M \cdot Q_k + R_k$ where $\mathsf{deg}(R_k) < d$ and $\mathsf{deg}(Q_k) < k$ (see Theorem 17 and Lemma 19). We replace x by x^{-1} and multiply the identity by $x^{d+(k-1)}$ to get:

$$\begin{aligned} & x^{d+(k-1)} \cdot (x^{-k} \cdot P(x^{-1})) = x^{d+(k-1)} \cdot M(x^{-1}) \cdot Q_k(x^{-1}) + x^{d+(k-1)} \cdot R_k(x^{-1}) \\ \Leftrightarrow \quad & x^{d-1} \cdot P(x^{-1}) = x^d \cdot M(x^{-1}) \cdot x^{k-1} \cdot Q_k(x^{-1}) + x^{d+(k-1)} \cdot R_k(x^{-1}) \\ \Leftrightarrow \quad & \mathsf{rev}_{d-1}(P) = \mathsf{rev}_d(M) \cdot \mathsf{rev}_{k-1}(Q_k) + x^k \cdot \mathsf{rev}_{d-1}(R_k) \end{aligned}$$

We reduce the last equation modulo x^k to get $\mathsf{rev}_{d-1}(P) \bmod x^k = \mathsf{rev}_d(M) \cdot \mathsf{rev}_{k-1}(Q_k) \bmod x^k$. Thus, we get that $\mathsf{rev}_{k-1}(Q_k) = \mathsf{rev}_d(M)^{-1} \cdot \mathsf{rev}_{d-1}(P) \bmod x^k$, which is what we compute in Algorithm 1 for $k = \nu$.
We show in what follows that $\mathsf{rev}_\nu(Q_\nu) = \sum_{k=0}^{\nu-1} a_k \cdot x^{k+1}$. We can write $\mathbb{R}_k = a_k \cdot x^{d-1} + S_k$, where S_k is a polynomial of degree strictly less than $d-1$. Moreover, we can write $M = x^d - M'$ where M' is a polynomial of degree $d-1$. We multiply the identity $x^k \cdot P = M \cdot Q_k + R_k$ by x and we get:

$$\begin{aligned} x^{k+1} \cdot P &= x \cdot M \cdot Q_k + x \cdot \mathbb{R}_k = x \cdot M \cdot Q_k + a_k \cdot x^d + x \cdot S_k \\ &= x \cdot M \cdot Q_k + a_k \cdot (M + M') + x \cdot S_k = (x \cdot Q_k + a_k) \cdot M + a_k \cdot M' + x \cdot S_k \end{aligned}$$

Moreover, we have $x^{k+1} \cdot P = M \cdot Q_{k+1} + R_{k+1}$, and by the uniqueness of the quotient and remainder (Theorem 17), we have $Q_{k+1} = x \cdot Q_k + a_k$ and $R_{k+1} = a_k \cdot M' + x \cdot S_k$. We replace x by x^{-1} in the first equation, then multiply it by x^{k+1} to get:

$$\begin{aligned} & x^{k+1} \cdot Q_{k+1}(x^{-1}) = x^{k+1} \cdot (x^{-1} \cdot Q_k(x^{-1}) + a_k) \\ \Leftrightarrow \quad & \mathsf{rev}_{k+1}(Q_{k+1}) = \mathsf{rev}_k(Q_k) + x^{k+1} \cdot a_k \end{aligned}$$

We can deduce that $Q_0 = 0$ since Q_0 has degree less than 0. Thus, we unroll the recurrence relation to get:

$$\begin{aligned} & \mathsf{rev}_0(Q_0) = 0, \\ & \mathsf{rev}_1(Q_1) = \mathsf{rev}_0(Q_0) + x^1 \cdot a_0 = a_0 \cdot x \\ & \qquad \vdots \\ & \mathsf{rev}_\nu(Q_\nu) = \mathsf{rev}_{\nu-1}(Q_{\nu-1}) + x^\nu \cdot a_{\nu-1} = a_0 \cdot x + a_1 \cdot x^2 + \cdots + a_{\nu-1} \cdot x^\nu \end{aligned}$$

We conclude that $\mathsf{rev}_\nu(Q_\nu) = \sum_{k=0}^{\nu-1} a_k \cdot x^{k+1}$.

Complexity

We need to compute the coefficients of $f = g^{-1} \cdot h \bmod x^\nu$. The computation of $g^{-1} \bmod x^\nu$ can be done using the Newton's polynomial inversion method [10] in $O(\mathcal{M}(\nu))$. The product $g^{-1} \cdot h$ can be computed in $O(\mathcal{M}(\nu))$ as well. Finally, we reduce the result modulo x^ν and get the coefficients of f and thus a in $O(\mathcal{M}(\nu))$. ◀

3.2 Main result: multiple modular approximations

▶ **Theorem 27.** *Let $\mathfrak{M} = M_1, \ldots, M_n$ monic polynomials of degrees $d_1, \ldots, d_n$ respectively, $\mathsf{F} = [f_{ij}]_{j=1,\ldots,m}^{i=1,\ldots,n} \in \mathbb{K}[x]^{n\times m}$, such that $\mathsf{rdeg}(\mathsf{F}) < [d_1, \ldots, d_n]$ and $\nu = \{\nu_1, \ldots, \nu_m\}$ a set of positive integers. We define $C = \sum_{j=1}^{m} \nu_j$ and $D = \sum_{i=1}^{n} d_i$.*
The problem $\mathsf{ApproxMultMod}(\mathfrak{M}, \mathsf{F}, \nu)$ can be solved in $\tilde{O}(\mathsf{max}(m,n)^{\omega-1} \cdot \mathsf{max}(D,C))$ operations in $\mathbb{K}$ by Algorithm 2.

Proof. The idea is to reduce the given problem to the problem of solving a quasi-Toeplitz linear system. We can use known algorithms [3, 1] to solve such a linear system in the desired complexity (see Theorem 12). Note that such algorithms are randomized. To adress this limitation, we present a deterministic alternative in Section 5 that maintains the same complexity (see Algorithm 5). Formally, we reduce $\mathsf{ApproxMultMod}(\mathfrak{M}, \mathsf{F}, \nu)$ to a problem of type $\mathsf{LinearSystem}[\mathbb{Z}_{D,0}, \mathbb{Z}_{C,0}^{\mathsf{T}}, O(\mathsf{max}(m,n))]$ as presented in Algorithm 2.

Algorithm 2 Simultaneous modular approximations

Input: $\mathfrak{M} \in \mathbb{K}[x]^n$, $\mathsf{F} \in \mathbb{K}[x]^{n\times m}$, $\nu = \{\nu_1, \ldots, \nu_m\}$
Output: $\mathsf{ApproxMultMod}(\mathfrak{M}, \mathsf{F}, \nu)$

```
A ∈ K^{D×C};
for i = 1 to n do
    for j = 1 to m do
        f_ij ∈ K^{d_i} is the vector of coefficients of the polynomial f_ij;
        A_ij = [f_ij  X(M_i)f_ij  ···  X(M_i)^{ν_j−1} f_ij];
Compute G, H ∈ K^{*×O(max(m,n))} such that φ_+(A) = G · H^T as in the proof of Lemma 28;
return LinearSystem[Z_{D,0}, Z_{C,0}^T, O(max(m,n))](G, H);
```

Thus, we construct $\mathbb{Z}_{D,0}, \mathbb{Z}_{C,0}^{\mathsf{T}}$-generators of length in $O(\mathsf{max}(m,n))$ for a given matrix $\mathsf{A} \in \mathbb{K}^{D\times C}$, where A has a quasi-Toeplitz structure and is such that solving the linear system $\mathsf{A} \cdot \mathsf{u} = \begin{bmatrix} 0 & \cdots & 0 \end{bmatrix}^{\mathsf{T}}$, where $\mathsf{u} \in \mathbb{K}^{C\times 1}$, is equivalent to solving the problem $\mathsf{ApproxMultMod}(\mathfrak{M}, \mathsf{F}, \nu)$.
For this, we have defined the matrix A as the matrix $\mathsf{A} = [\mathsf{A}_{ij}]_{j=1,\ldots,m}^{i=1,\ldots,n} \in \mathbb{K}^{D\times C}$, such that $\mathsf{A}_{ij} \in \mathbb{K}^{d_i \times \nu_j}$, for $i = 1, \ldots, n$ and $j = 1, \ldots, m$, corresponds to the Krylov matrix

$$\mathsf{A}_{ij} = \begin{bmatrix} \mathsf{f}_{ij} & \mathbb{X}(M_i)\mathsf{f}_{ij} & \cdots & \mathbb{X}(M_i)^{\nu_j-1}\mathsf{f}_{ij} \end{bmatrix} \tag{2}$$

where f_{ij} is the vector of coefficients of the polynomial f_{ij}.
The matrix A is quasi-Toeplitz with a displacement rank in $O(\mathsf{max}(m,n))$, and to show this, we compute two matrices G and H such that G has $O(\mathsf{max}(m,n))$ columns and $\phi_+(\mathsf{A}) = \mathsf{G} \cdot \mathsf{H}^{\mathsf{T}}$, as done in Lemma 28.

▶ **Lemma 28.** *Let A be the matrix defined in (2). We construct ϕ_+-generators of length $O(\mathsf{max}(m,n))$ for A in $O(\mathsf{max}(m,n)^{\omega-1} \cdot \mathcal{M}(\mathsf{max}(D,C)))$ operations in $\mathbb{K}$.*

Proof. We know by Definition 8 that, if A is a quasi-Toeplitz matrix, then $\phi_+(\mathsf{A}) = \mathsf{A} - \mathbb{Z}_{D,0} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}}$.
We define $\mathbb{X} \in \mathbb{K}^{D\times D}$ as the matrix $\mathsf{diag}(\mathbb{X}(M_1), \ldots, \mathbb{X}(M_n))$, i.e.,

$$\mathbb{X} = \begin{bmatrix} \mathbb{X}(M_1) & 0 & \cdots & 0 \\ 0 & \mathbb{X}(M_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbb{X}(M_n) \end{bmatrix}.$$

We define $\delta \in \mathbb{K}^{D\times D}$ as the matrix $[\delta_{ab}]_{a,b=1,\ldots,D}$ where $\delta_{ab} = 1$ if $a = \sum_{k=1}^{i} d_k + 1$ and $b = \sum_{k=1}^{i} d_k$ for $i = 1, \ldots, n$, and $\delta_{ab} = 0$ otherwise, for $a, b = 1, \ldots, D$.
Let $\mathbb{X}' \in \mathbb{K}^{D\times D}$ be the matrix $\mathbb{X} + \delta$. We represent this matrix in Figure 2 for a better understanding of its structure.

$$\mathbb{X}' = \begin{bmatrix} \mathbb{X}(M_1) & 0\cdots00 & \cdots & 0 \\ 0\cdots01 & \mathbb{X}(M_2) & \cdots & 0 \\ 0\cdots00 & 0\cdots01 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0\cdots00 & 0\cdots00 & \cdots & \mathbb{X}(M_n) \end{bmatrix} = \left[\begin{array}{cccc} \begin{bmatrix} 0 & \cdots & 0 & -m_{0,1} \\ 1 & \cdots & 0 & -m_{1,1} \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 1 & -m_{d_1-1,1} \end{bmatrix} & \mathbf{0} & \cdots & \mathbf{0} \\ \begin{bmatrix} 0 & \cdots & 0 & 1 \\ 0 & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & \cdots & 0 & -m_{0,2} \\ 1 & \cdots & 0 & -m_{1,2} \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 1 & -m_{d_2-1,2} \end{bmatrix} & \cdots & \mathbf{0} \\ \mathbf{0} & \begin{bmatrix} 0 & \cdots & 0 & 1 \\ 0 & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 \end{bmatrix} & \ddots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \begin{bmatrix} 0 & \cdots & 0 & 1 \\ 0 & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & \cdots & 0 & -m_{0,n} \\ 1 & \cdots & 0 & -m_{1,n} \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 1 & -m_{d_n-1,n} \end{bmatrix} \end{array}\right].$$

Figure 2 Matrix $\mathbb{X}' = \mathbb{X} + \delta$.

We notice that $\mathbb{Z}_{D,0} = \mathbb{X}' + \mathsf{M} \cdot \mathsf{E}$, where $\mathsf{M} \in \mathbb{K}^{D\times n}$ and $\mathsf{E} \in \mathbb{K}^{n\times D}$ are block diagonal matrices such that $\mathsf{M} = \mathsf{diag}(\mathsf{m}_1, \ldots, \mathsf{m}_n)$ with $\mathsf{m}_i \in \mathbb{K}^{d_i\times 1}$; the vector of coefficients of the polynomial M_i, and $\mathsf{E} = \mathsf{diag}(\mathsf{e}_1, \ldots, \mathsf{e}_n)$ with $\mathsf{e}_i \in \mathbb{K}^{1\times d_i}$; the vector $[0, \ldots, 0, 1]$ for $i = 1, \ldots, n$.
We get $\mathbb{Z}_{D,0} = \mathbb{X} + \delta + \mathsf{M} \cdot \mathsf{E}$ and can now rewrite $\phi_+(\mathsf{A})$ as follows:

$$\phi_+(\mathsf{A}) = \mathsf{A} - \mathbb{X} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} - \delta \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} - \mathsf{M} \cdot \mathsf{E} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}}. \tag{3}$$

In order to construct the generators, we proceed as follows:

- For $\mathsf{A} - \mathbb{X} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}}$:
  Let $\mathsf{B} = \mathsf{A} - \mathbb{X} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} \in \mathbb{K}^{D\times C}$, such that $\mathsf{B} = [\mathsf{B}_{ij}]_{j=1,\ldots,m}^{i=1,\ldots,n}$ where $\mathsf{B}_{ij} \in \mathbb{K}^{d_i\times\nu_j}$, for $i = 1, \ldots, n$ and $j = 1, \ldots, m$, corresponds the matrix

$$\mathsf{B}_{ij} = \begin{cases} \begin{bmatrix} \mathsf{f}_{ij} & 0 & \cdots & 0 \end{bmatrix} & \text{if } j = 1 \\ \begin{bmatrix} \mathsf{f}_{ij} - \mathbb{X}(M_i)^{\nu_{j-1}} \cdot \mathsf{f}_{i(j-1)} & 0 & \cdots & 0 \end{bmatrix} & \text{if } j > 1 \end{cases}$$

  We can write $\mathsf{B} = \mathsf{Y} \cdot \mathsf{Z}^{\mathsf{T}}$, with Y and Z defined below.
  Let $\mathsf{Y} \in \mathbb{K}[x]^{D\times m}$ as $\mathsf{Y} = [\mathsf{Y}_{ij}]_{j=1,\ldots,m}^{i=1,\ldots,n}$ with $\mathsf{Y}_{ij} \in \mathbb{K}^{d_i\times 1}$ such that

$$\mathsf{Y}_{ij} = \begin{cases} \mathsf{f}_{i1} & \text{for } i = 1, \ldots, n \text{ and } j = 1 \\ \mathsf{f}_{ij} - \mathbb{X}(M_i)^{\nu_{j-1}} \cdot \mathsf{f}_{i(j-1)} & \text{for } i = 1, \ldots, n \text{ and } j = 2, \ldots, m \end{cases}$$

  We define $\mathsf{Z} \in \mathbb{K}[x]^{C\times m}$ as $\mathsf{Z} = [\mathsf{Z}_j]_{j=1,\ldots,m}$ with $\mathsf{Z}_j \in \mathbb{K}^{C\times 1}$ for $j = 1, \ldots, m$ such that Z_1 is the vector of all zeros except for the first position, which is 1, and Z_j, for $j = 2, \ldots, m$, is the vector of all zeros except for the first position and the $\sum_{k=1}^{j-1} \nu_k + 1$-th position, which are 1.
  The computation of $\mathbb{X}(M_i)^{\nu_{j-1}} \cdot \mathsf{f}_{i(j-1)}$; the coefficient vector of the polynomial $x^{\nu_{j-1}} \cdot f_{i(j-1)}$ rem M_i, can be done using Lemma 25 in $O(\mathcal{M}(d_i) \cdot \log(\nu_{j-1}))$ operations in $\mathbb{K}$. Thus, the total cost of computing Y is:

$$\begin{aligned} O(\textstyle\sum_{j=1}^{m}\sum_{i=1}^{n} \mathcal{M}(d_i) \cdot \log(\nu_{j-1})) &= O(\textstyle\sum_{j=1}^{m} \mathcal{M}(D) \cdot \log(\nu_{j-1})) \text{ (by the superlinearity property of } \mathcal{M}) \\ &= O(\mathcal{M}(D) \cdot \log(\textstyle\prod_{j=1}^{m} \nu_j)) \\ &= O(\mathcal{M}(D) \cdot \log((\mathsf{max}_{j=1,\ldots,m}\nu_j)^m)) \\ &= O(m \cdot \mathcal{M}(D) \cdot \log(\mathsf{max}_{j=1,\ldots,m}\nu_j)). \end{aligned}$$

- For $\mathsf{M}\cdot\mathsf{E}\cdot\mathsf{A}\cdot\mathbb{Z}_{C,0}^{\mathsf{T}}$:
  Let $\mathsf{P}\in\mathbb{K}^{C\times n}$ such that $\mathsf{P}=\mathsf{A}^{\mathsf{T}}\cdot\mathsf{E}^{\mathsf{T}}$ and $\mathsf{Q}\in\mathbb{K}^{C\times n}$ such that $\mathsf{Q}=\mathbb{Z}_{C,0}\cdot\mathsf{P}$. We can write $\mathsf{M}\cdot\mathsf{E}\cdot\mathsf{A}\cdot\mathbb{Z}_{C,0}^{\mathsf{T}}=\mathsf{M}\cdot\mathsf{Q}^{\mathsf{T}}$. In order to compute Q, we need to compute P first. We have $\mathsf{P}=[\mathsf{P}_{ij}]_{j=1,\ldots,m}^{i=1,\ldots,n}$ where $\mathsf{P}_{ij}\in\mathbb{K}^{1\times\nu_j}$, and

  $$\mathsf{P}_i^{\mathsf{T}}=\Big[\mathsf{coeff}(f_{ij},d_i-1),\ldots,\mathsf{coeff}(f_{ij}\cdot x^{\nu_j-1}\ \mathsf{rem}\ M_i,d_i-1)\Big]$$

  for $i=1,\ldots,n$ and $j=1,\ldots,m$.
  We use Lemma 26 to compute P_{ij} in $O(\mathcal{M}(\nu_j))$ operations in $\mathbb{K}$. Thus, the total cost of computing P is:

  $$O(\textstyle\sum_{i=1}^{n}\sum_{j=1}^{m}\mathcal{M}(\nu_j))=O(\sum_{i=1}^{n}\mathcal{M}(C))=O(n\cdot\mathcal{M}(C)).$$

- For $\delta\cdot\mathsf{A}\cdot\mathbb{Z}_{C,0}^{\mathsf{T}}$:
  We can write $\delta\cdot\mathsf{A}\cdot\mathbb{Z}_{C,0}^{\mathsf{T}}=\mathsf{R}\cdot\mathsf{S}^{\mathsf{T}}$ where $\mathsf{R}\in\mathbb{K}^{D\times n}$ and $\mathsf{S}\in\mathbb{K}^{C\times n}$ are the matrices such that $\mathsf{R}=[\mathsf{R}_{ij}]_{j=1,\ldots,m}^{i=0,\ldots,n-1}$ and $\mathsf{S}=[\mathsf{S}_{ji}]_{j=1,\ldots,m}^{i=0,\ldots,n-1}$, where $\mathsf{R}_{ij}\in\mathbb{K}^{d_i\times 1}$ and $\mathsf{S}_{ji}\in\mathbb{K}^{\nu_j\times 1}$ correspond to the vectors:

  $$\mathsf{R}_{ij}=\begin{bmatrix}0\\\vdots\\0\end{bmatrix}\ \text{for}\ \begin{cases}j=1\\ i=0,\ldots,(n-1)\end{cases}\qquad \mathsf{R}_{ij}=\begin{bmatrix}1\\0\\\vdots\\0\end{bmatrix}\ \text{for}\ \begin{cases}j=2,\ldots,n\\ i=0,\ldots,(n-1)\end{cases}$$

  $$\mathsf{S}_{ji}=\begin{bmatrix}0\\\vdots\\0\end{bmatrix}\ \text{for}\ \begin{cases}j=1\\ i=0,\ldots,(n-1)\end{cases}\qquad \mathsf{S}_{ji}=\begin{bmatrix}0\\\mathsf{elem}(\mathsf{f}_{ij},d_i-1)\\\mathsf{elem}(\mathsf{f}_{ij}\cdot\mathbb{X}(M_i),d_i-1)\\\vdots\\\mathsf{elem}(\mathsf{f}_{ij}\cdot\mathbb{X}(M_i)^{\nu_j-2},d_i-1)\end{bmatrix}\ \text{for}\ \begin{cases}j=2,\ldots,m\\ i=0\end{cases}$$

  $$\mathsf{S}_{ji}=\begin{bmatrix}\mathsf{elem}(\mathsf{f}_{(i-1)j}\cdot\mathbb{X}(M_{(i-1)})^{\nu_j-1},d_{(i-1)}-1)\\\mathsf{elem}(\mathsf{f}_{ij},d_i-1)\\\mathsf{elem}(\mathsf{f}_{ij}\cdot\mathbb{X}(M_i),d_i-1)\\\vdots\\\mathsf{elem}(\mathsf{f}_{ij}\cdot\mathbb{X}(M_i)^{\nu_j-2},d_i-1)\end{bmatrix}\ \text{for}\ \begin{cases}j=2,\ldots,m\\ i=1,\ldots,(n-1)\end{cases}$$

  The computation of one S_{ji} is again done using Lemma 26 and thus can be done in $O(\mathcal{M}(\nu_j))$. Therefore, the total cost of computing S is bounded by:

  $$O(\textstyle\sum_{i=1}^{n}\sum_{j=1}^{m}\mathcal{M}(\nu_j))=O(\sum_{i=1}^{n}\mathcal{M}(C))=O(n\cdot\mathcal{M}(C)).$$

Therefore, we get

$$\phi_+(\mathsf{A})=\mathsf{Y}\cdot\mathsf{Z}^{\mathsf{T}}-\mathsf{R}\cdot\mathsf{S}^{\mathsf{T}}-\mathsf{M}\cdot\mathsf{Q}^{\mathsf{T}}\tag{4}$$

We define $\mathsf{G}=\begin{bmatrix}\mathsf{Y} & -\mathsf{R} & -\mathsf{M}\end{bmatrix}\in\mathbb{K}^{D\times(m+2n)}$ and $\mathsf{H}=\begin{bmatrix}\mathsf{Z} & \mathsf{S} & \mathsf{Q}\end{bmatrix}\in\mathbb{K}^{C\times(m+2n)}$ such that $\phi_+(\mathsf{A})=\mathsf{G}\cdot\mathsf{H}^{\mathsf{T}}$. Therefore, we were able to compute G and H in $O(m\cdot\mathcal{M}(D)+n\cdot\mathcal{M}(C))$ operations in $\mathbb{K}$. We, thus, obtain ϕ_+-generators of length $O(\mathsf{max}(m,n))$ for the matrix A in $O(\mathsf{max}(m,n)^{\omega-1}\cdot\mathcal{M}(\mathsf{max}(D,C)))$. ◀

After computing the ϕ_+-generator of length $O(\mathsf{max}(m,n))$ for the matrix A, we have formally reduced the problem $\mathsf{ApproxMultMod}(\mathfrak{M},\mathsf{F},\nu)$ to the problem of solving a linear system of equations with a quasi-Toeplitz matrix A, i.e., $\mathsf{LinearSystem}[\mathbb{Z}_{D,0},\mathbb{Z}_{C,0}^{\mathsf{T}},\mathsf{max}(m,n)](\mathsf{G},\mathsf{H})$, where G and H are the matrices defined in the proof of Lemma 28, which can be solved in $\tilde{O}(\mathsf{max}(m,n)^{\omega-1}\cdot\mathsf{max}(D,C))$ operations in $\mathbb{K}$ (see Theorem 12).
Finally, we recover the solution g of the problem $\mathsf{ApproxMultMod}(\mathfrak{M},\mathsf{F},\nu)$ from the solution u of the linear system $\mathsf{A}\cdot\mathsf{u}=\begin{bmatrix}0 & \cdots & 0\end{bmatrix}^{\mathsf{T}}$ by taking the coefficients of the polynomials g_j from the vector u. More formally, we have $g_j=\sum_{k=0}^{\nu_j-1}x^k\cdot\mathsf{elem}(\mathsf{u},k+\sum_{\ell=1}^{j-1}\nu_\ell-1)$, for $j=1,\ldots,m$.
We conclude that the problem $\mathsf{ApproxMultMod}(\mathfrak{M},\mathsf{F},\nu)$ can be solved in $\tilde{O}(\mathsf{max}(m,n)^{\omega-1}\cdot\mathsf{max}(D,C))$. ◀

4 Matrix modular approximation

After efficiently solving simultaneous modular polynomial approximations, we tackle a more general problem, which is the matrix modular approximation problem, where we work modulo a matrix M. We can suppose without loss of generality here that M is a modulus polynomial matrix. The key idea is that we can always reduce the problem given a modulo that is simply a nonsingular matrix M to a problem with a modulus polynomial matrix. We formalize this notion of no loss of generality in Lemma 52, which can be found in Appendix B, along with its proof.

▶ **Definition 29.** *Given*

- $\mathsf{M} = [M_{ij}]_{i,j=1,\dots,n} \in \mathbb{K}[x]^{n\times n}$, *where* M *is a modulus polynomial matrix and* $\mathsf{rdeg}(M) = [d_1, \dots, d_n]$.
- $\mathsf{F} = [f_{ij}]_{j=1,\dots,m}^{i=1,\dots,n} \in \mathbb{K}[x]^{n\times m}$, *such that* $\mathsf{rdeg}(\mathsf{F}) < [d_1, \dots, d_n]$,
- $\nu = \{\nu_1, \dots, \nu_m\}$ *a set of positive integers.*

We define the set of solutions

$$S = \left\{ \mathsf{g} \in \mathbb{K}[x]^{m\times 1} \mid \mathsf{F}\cdot\mathsf{g} = 0 \bmod \mathsf{M} \right\}, \tag{5}$$

where $\mathsf{F}\cdot\mathsf{g} = 0 \bmod \mathsf{M}$ *means that* $\mathsf{F}\cdot\mathsf{g} = \mathsf{M}\cdot\mathsf{q}$ *for some* $\mathsf{q} \in \mathbb{K}[x]^{n\times 1}$*; or in other words, the vector* $\mathsf{F}\cdot\mathsf{g}$ *lies in the* $\mathbb{K}[x]$*-module generated by the columns of* M.

We define the problem $\mathsf{ApproxMatrixMod}(\mathsf{M}, \mathsf{F}, \nu)$*, as follows:*

ApproxMatrixMod	
Input:	$\mathsf{M} \in \mathbb{K}[x]^{n\times n}$, $\mathsf{F} \in \mathbb{K}[x]^{n\times m}$, $\nu = \{\nu_1, \dots, \nu_m\}$.
Output:	$\mathsf{g} \in S$ *such that* g *is not the zero vector and* $\mathsf{deg}(g_j) < \nu_j$ *for* $j = 1, \dots, m$.

We denote by $C = \sum_{j=1}^{m} \nu_j$ *the total number of unknown coefficients of the unknown polynomial vector* g *and by* $D = \sum_{i=1}^{n} d_i$ *the sum of diagonal degrees of the modulus polynomial matrix* M.

We note that this version is the most general version of our approximation problems; it includes the polynomial case as a particular instance where $n = 1$ and the simultaneous multiple polynomial approximations case where we consider a diagonal matrix M, with the diagonal entries being the polynomials of the family of moduli.

We introduce in what follows the matrix $\mathbb{X}(\mathsf{M})$, which is an attempt to generalize the notion of the multiplication matrix for a single polynomial, defined in Definition 9. This matrix can be found in Section 9 of [26]. Here, we manage to use it to define a matrix that will help us reduce the problem of matrix modular approximation to a linear system of equations with a quasi-Toeplitz structure.

▶ **Definition 30.** *Given a modulus polynomial matrix* $\mathsf{M} = [M_{ij}]_{i,j=1,\dots,n} \in \mathbb{K}[x]^{n\times n}$*, where* M *is a modulus polynomial matrix and* $\mathsf{rdeg}(M) = [d_1, \dots, d_n]$*. Let* $D = \sum_{i=1}^{n} d_i$*. We define the matrix* $\mathbb{X}(\mathsf{M}) \in \mathbb{K}^{D\times D}$ *as follows*

$$\mathbb{X}(\mathsf{M}) = \begin{bmatrix} \mathbb{X}(M_{11}) & \mathsf{C}_{12} & \cdots & \mathsf{C}_{1n} \\ \mathsf{C}_{21} & \mathbb{X}(M_{22}) & \cdots & \mathsf{C}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathsf{C}_{n1} & \mathsf{C}_{n2} & \cdots & \mathbb{X}(M_{nn}) \end{bmatrix}, \tag{6}$$

where $\mathbb{X}(M_{ii})$ *is the multiplication matrix of* M_{ii} *(defined in Definition 9) and* $\mathsf{C}_{ij} \in \mathbb{K}^{d_i\times d_j}$ *is the matrix where the* d_j*-th column is the vector of coefficients of* M_{ij} *and zero elsewhere, for* $i, j = 1, \dots, n$*, i.e.,*

$$\mathsf{C}_{ij} = \begin{bmatrix} 0 & \cdots & 0 & -\mathsf{coeff}(M_{ij}, 0) \\ 0 & \cdots & 0 & -\mathsf{coeff}(M_{ij}, 1) \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & -\mathsf{coeff}(M_{ij}, d_i - 1) \end{bmatrix}.$$

Let $\mathsf{F} = [f_{ij}]_{j=1,\dots,m}^{i=1,\dots,n} \in \mathbb{K}[x]^{n\times m}$ *such that* $\mathsf{rdeg}(\mathsf{F}) < [d_1, \dots, d_n]$ *and* k *a non-negative integer.* $\mathbb{X}(M)^k \cdot \bar{\mathsf{F}}$*, where* $\bar{\mathsf{F}} = [\mathsf{f}_{ij}]_{j=1,\dots,m}^{i=1,\dots,n} \in \mathbb{K}[x]^{D\times m}$ *such that* $\mathsf{f}_{ij} \in \mathbb{K}^{d_j\times 1}$ *is the vector of coefficients of the polynomial* $f_{ij}(x)$ *for* $i = 1, \dots, n$ *and* $j = 1, \dots, m$*, is the matrix whose block vectors are the vectors of coefficients of polynomials in* $\mathsf{diag}(x^k, \dots, x^k) \cdot \mathsf{F} \bmod \mathsf{M}$.

4.1 Intermediate results: matrix modular computations

▶ **Lemma 31.** *Let* $\mathsf{M} = [M_{ij}]_{i,j=1,\dots,n} \in \mathbb{K}[x]^{n\times n}$*, where* M *is a modulus polynomial matrix with* $\mathsf{rdeg}(M) = [d_1,\dots,d_n]$*. Let* $\mathsf{F} = [f_{ij}]_{j=1,\dots,m}^{i=1,\dots,n} \in \mathbb{K}[x]^{n\times m}$*, such that* $\mathsf{rdeg}(\mathsf{F}) < [d_1,\dots,d_n]$ *and* $\nu = \{\nu_1,\dots,\nu_m\}$ *a set of positive integers. We define* $C = \sum_{j=1}^{m}\nu_j$ *and* $D = \sum_{i=1}^{n} d_i$*. We suppose that* $C \in O(D)$*. We can deduce that* $n, m \in O(D)$ *as well. One can compute the coefficients of the polynomial* $x^{\nu_j}\cdot \mathsf{F}_{*j}$ rem M*, for* $j = 1,\dots,m$*, in* $\tilde{O}(n^{\omega-1}\cdot D + m\cdot n^{\omega-2}\cdot D)$ *operations in* $\mathbb{K}$*.*

Proof. We define the matrix $\mathsf{P} \in \mathbb{K}[x]^{m\times m}$ as $\mathsf{diag}(x^{\nu_1}, x^{\nu_2},\dots,x^{\nu_m})$.
Since $\mathsf{F}\cdot\mathsf{P} = \begin{bmatrix} x^{\nu_1}\cdot\mathsf{F}_{*1} & x^{\nu_2}\cdot\mathsf{F}_{*2} & \cdots & x^{\nu_m}\cdot\mathsf{F}_{*m}\end{bmatrix}$, we can compute the coefficients of the polynomial $x^{\nu_j}\cdot\mathsf{F}_{*j}$ rem M for $j = 1,\dots,m$ by computing $\mathsf{F}\cdot\mathsf{P}$ rem M. Since we have $C, m, n \in O(D)$ and M is a modulus polynomial matrix and thus a row reduced matrix (see Lemma 23), we can apply Algorithm 3 from [22] to compute the coefficients of the polynomial $\mathsf{F}\cdot\mathsf{P}$ rem M. With the entries $\mathsf{F} \in \mathbb{K}[x]^{n\times m}$ and $\mathsf{P} \in \mathbb{K}[x]^{m\times m}$, and the modulus polynomial matrix $\mathsf{M} \in \mathbb{K}[x]^{n\times n}$, the algorithm can compute the coefficients of the polynomial $\mathsf{F}\cdot\mathsf{P}$ rem M in $\tilde{O}(\mathsf{max}(m,n)^{\omega-1}\cdot D)$ operations in $\mathbb{K}$ (see Proposition 3.6 in [22]). We distinguish two cases:

- If $n \geq m$, we achieve a complexity of $\tilde{O}(n^{\omega-1}\cdot D)$ by simply applying the algorithm as is.
- If $n < m$, we can exploit the diagonal structure of P to achieve a better complexity. We suppose without loss of generality that n divides m. We split the matrices F, P and M into blocks of size $n\times n$, such that $\mathsf{F} = \begin{bmatrix}\mathsf{F}_1 & \mathsf{F}_2 & \cdots & \mathsf{F}_{\frac{m}{n}}\end{bmatrix}$, where $\mathsf{F}_i \in \mathbb{K}[x]^{n\times n}$ for $i = 1,\dots,\frac{m}{n}$ and $\mathsf{P} = \mathsf{diag}(\mathsf{P}_1,\mathsf{P}_2,\dots,\mathsf{P}_{\frac{m}{n}})$, where $\mathsf{P}_i \in \mathbb{K}[x]^{n\times n}$ for $i = 1,\dots,\frac{m}{n}$. Thus, we have $\mathsf{F}\cdot\mathsf{P}$ rem $\mathsf{M} = \begin{bmatrix}\mathsf{F}_1\cdot\mathsf{P}_1 \text{ rem } \mathsf{M} & \mathsf{F}_2\cdot\mathsf{P}_2 \text{ rem } \mathsf{M} & \cdots & \mathsf{F}_{\frac{m}{n}}\cdot\mathsf{P}_{\frac{m}{n}} \text{ rem } \mathsf{M}\end{bmatrix}$. The total cost is the sum of the costs of computing each block product $\mathsf{F}_i\cdot\mathsf{P}_i$ rem M for $i = 1,\dots,\frac{m}{n}$, which is $\tilde{O}(\frac{m}{n}\cdot n^{\omega-1}\cdot D) = \tilde{O}(m\cdot n^{\omega-2}\cdot D)$.

Thus, we are able to get an overall slightly better complexity than $\tilde{O}(\mathsf{max}(m,n)^{\omega-1}\cdot D)$, which is $\tilde{O}(n^{\omega-1}\cdot D + m\cdot n^{\omega-2}\cdot D)$. ◀

▶ **Definition 32.** *Let* $\mathsf{G} \in \mathbb{K}[x]^{n\times n}$*, where* G *is a row reduced nonsingular matrix and* $\mathsf{rdeg}(\mathsf{G}) = [d_1,\dots,d_n]$*. Let* $\mathsf{P} \in \mathbb{K}[x]^{n\times m}$ *and* $\nu = \{\nu_1,\dots,\nu_m\}$ *a set of positive integers.*
We define the problem $\mathsf{TruncatedInverseProduct}(\mathsf{G},\mathsf{P},\nu)$ *as follows:*

TruncatedInverseProduct	
Input:	$\mathsf{G} \in \mathbb{K}[x]^{n\times n}$, $\mathsf{P} \in \mathbb{K}[x]^{n\times m}$, $\nu = \{\nu_1,\dots,\nu_m\}$.
Output:	$\mathsf{G}^{-1}\cdot\mathsf{P}$ rrem $\mathbb{X}^{\nu}$ *where* $\mathbb{X}^{\nu} = \mathsf{diag}(x^{\nu_1}, x^{\nu_2},\dots,x^{\nu_m})$.

A variation of this problem is solved in [22], where the maximum degree is at most twice the average degree in ν. In other words, the degrees in ν have to be balanced, to avoid the case where one degree is much larger than the others. Authors manage to solve the problem in $\tilde{O}(n^{\omega-1}\cdot C)$ operations in $\mathbb{K}$, where C is the total number of coefficients in ν. In the following lemma, we show that we can lift this restriction and solve the problem for any set of degrees ν, while keeping the same complexity, up to a logarithmic factor in the number of coefficients in ν. We note that the problem is naively solved in $\tilde{O}(n^{\omega}\cdot C)$ using Newton iteration (see [10]).

▶ **Lemma 33.** *Let* $\mathsf{G} \in \mathbb{K}[x]^{n\times n}$*, where* G *is row reduced,* $\mathsf{P} \in \mathbb{K}[x]^{n\times m}$*,* $\nu = \{\nu_1,\dots,\nu_m\}$ *a set of positive integers and* $C = \sum_{j=1}^{m}\nu_j$*. We suppose that* $n \in O(m)$*.*
The problem $\mathsf{TruncatedInverseProduct}(\mathsf{G},\mathsf{P},\nu)$ *can be solved in* $\tilde{O}(n^{\omega-1}\cdot C)$ *operations in* $\mathbb{K}$*, using Algorithm 3.*

The idea is that Algorithm 3 selects at each step columns that are reduced by a power of x, such that their degrees are small enough to be able to compute the truncated inverse product efficiently.

Proof. For a given step ℓ of the algorithm, we have:

- $L_\ell = \{j \in \{1,\dots,m\} \mid \nu_j \leq 2^{\ell}\cdot\frac{C}{m}\}$ and $\Gamma_\ell = \{j \in \{1,\dots,m\} \mid \nu_j > 2^{\ell}\cdot\frac{C}{m}\}$.
- $S_\ell = \{j \in S \mid 2^{\ell-1}\cdot\frac{C}{m} < \nu_j \leq 2^{\ell}\cdot\frac{C}{m}\}$ if $\ell \geq 1$ and $S_1 = \{j \in S \mid \nu_j \leq 2\cdot\frac{C}{m}\}$.
- $\mathsf{P}' \in \mathbb{K}[x]^{n\times|S_\ell|}$, which is the matrix P with only the columns $j \in S_\ell$.
- $\nu' = (\nu_j)_{j\in S_\ell}$.

Algorithm 3 Truncated Inverse Product

Input: $\mathsf{G} \in \mathbb{K}[x]^{n\times n}$ such that G is row reduced, $\mathsf{P} \in \mathbb{K}[x]^{n\times m}$ and $\nu = \{\nu_1, \ldots, \nu_m\}$.
Output: $\mathsf{G}^{-1} \cdot \mathsf{P} \text{ rem } \mathbb{X}^\nu$, where $\mathbb{X}^\nu = \mathsf{diag}(x^{\nu_1}, x^{\nu_2}, \ldots, x^{\nu_m})$.
Initialization: $\mathsf{P}' = \mathsf{P}$, $\nu' = \nu$, $S = \{1, \ldots, m\}$, $\ell = 1$, $C = \sum_{j=1}^m \nu_j$, $\mathsf{R} \in \mathbb{K}[x]^{n\times m}$ with $\mathsf{R} = [\mathsf{R}_{*j}]_{j=1,\ldots,m}$.

$S_\ell = \{j \in S \mid \nu_j \leq 2 \cdot \frac{C}{m}\}$; // S_1
for $\ell = 1$ **to** $\lceil \log_2(C/m) \rceil$ **do**
  $\mathsf{P}' \in \mathbb{K}[x]^{n\times|S_\ell|}$, where $\mathsf{P}' = [\mathsf{P}_{*j}]_{j\in S_\ell}$;
  $\nu' = (\nu_j)_{j\in S_\ell}$;
  $\mathbb{X}^{\nu'} \in \mathbb{K}[x]^{|S_\ell|\times|S_\ell|}$, where $\mathbb{X}^{\nu'} = \mathsf{diag}(x^{\nu_j})_{j\in S_\ell}$;
  Computing $\mathsf{B} = \mathsf{G}^{-1} \cdot \mathsf{P}' \text{ rem } \mathbb{X}^{\nu'}$;
  $i = 1$;
  for $j \in S_\ell$ **do**
    $\mathsf{R}_{*j} = \mathsf{B}_{*i}$;
    $i = i + 1$;
  $S_{\ell+1} = \{j \in S \mid 2^\ell \cdot \frac{C}{m} < \nu_j \leq 2^{\ell+1} \cdot \frac{C}{m}\}$;
return R;

Number of steps

Suppose by contradiction that we need more than $\lceil \log_2(m) \rceil$ steps of the algorithm, i.e., $\ell > \lceil \log_2(m) \rceil$. Then, we have $\nu_j > 2^\ell \cdot \frac{C}{m} > 2^{\lceil \log_2(m) \rceil} \cdot \frac{C}{m} > C$, which is a contradiction since ν_j is a positive integer and $C = \sum_{j=1}^m \nu_j$. Thus, we have that $S_\ell = \{j \in S \mid 2^{\ell-1} \cdot \frac{C}{m} < \nu_j \leq 2^\ell \cdot \frac{C}{m}\} = \emptyset$ and therefore Algorithm 3 terminates after at most $\lceil \log_2(m) \rceil$ steps.

Complexity of one step

The main argument is that, at each step, we treat more than $\frac{m}{2^\ell}$ columns, i.e., $|L_\ell| \geq \frac{m}{2^\ell}$. To prove this, we can proceed by contradiction. Suppose that $|L_\ell| < \frac{m}{2^\ell}$ or equivalently $|\Gamma_\ell| \geq \frac{m}{2^\ell}$.

We have $L_\ell \cup \Gamma_\ell = \{1, \ldots, m\}$ and $|L_\ell| + |\Gamma_\ell| = m$. Thus, we have $|\Gamma_\ell| = m - |L_\ell| > m - \frac{m}{2^\ell} = m(\frac{(2^\ell - 1)}{2^\ell})$. Since the minimum degree of the polynomials in P_{*j} such that $j \in \Gamma_\ell$ is greater than $2^\ell \cdot \frac{C}{m}$, we have that

$$\sum_{j\in\Gamma_\ell} \nu_j \geq \sum_{j\in\Gamma_\ell} 2^\ell \cdot \frac{C}{m} \geq |\Gamma_\ell| \cdot 2^\ell \cdot \frac{C}{m} > m(\frac{(2^\ell - 1)}{2^\ell}) \cdot 2^\ell \cdot \frac{C}{m} = C \cdot (2^\ell - 1). \tag{7}$$

Moreover, we have that $C = \sum_{j\in L_\ell} \nu_j + \sum_{j\in\Gamma_\ell} \nu_j \geq \sum_{j\in\Gamma_\ell} \nu_j$. We use the inequality in (7) to get:

$$C \geq \sum_{j\in\Gamma_\ell} \nu_j \geq C \cdot (2^\ell - 1),$$

which gives us $1 \geq (2^\ell - 1)$, since $C \neq 0$. This implies that $\ell < 1$. Therefore, we have a contradiction since ℓ is a positive integer. Thus, we have that $|L_\ell| \geq \frac{m}{2^\ell}$ and equivalently$|\Gamma_\ell| < \frac{m}{2^\ell}$ for all $\ell \geq 1$.

Note that $|S_\ell| < \frac{m}{2^\ell}$ and $\mathbb{X}^{\nu'} = \mathsf{diag}(x^{k_j})_{j\in S_\ell} \in \mathbb{K}[x]^{\frac{m}{2^\ell}\times\frac{m}{2^\ell}}$, since we have $|\Gamma_\ell| < \frac{m}{2^\ell}$. This allows us to compute the truncated product $\mathsf{G}^{-1} \cdot \mathsf{P}' \text{ rem } \mathbb{X}^{\nu'}$ in $\tilde{O}(n^{\omega-1} \cdot C)$ using Lemma 3.3 in [22]. The lemma can be applied since the matrix G is invertible and $\nu_j \leq 2^\ell \cdot \frac{C}{m}$ for $j \in L_\ell$. Thus, we get a complexity of :

$$\begin{aligned}
\tilde{O}(\lceil \frac{\frac{m}{2^\ell} \cdot 2^\ell}{n} \rceil \cdot n^\omega \cdot \lceil \frac{C}{m} \rceil) = \tilde{O}(\lceil \frac{m}{n} \rceil \cdot n^\omega \cdot \lceil \frac{C}{m} \rceil) &= \tilde{O}(m \cdot n^{\omega-1} \cdot \lceil \frac{C}{m} \rceil) && \text{since } n \in O(m) \\
&= \tilde{O}(n^{\omega-1} \cdot C) && \text{since } m \in O(C)
\end{aligned}$$

Final complexity

Since we have $\ell \leq \log_2(m)$, we can compute $\mathsf{G}^{-1} \cdot \mathsf{P} \text{ rem } \mathbb{X}^\nu$ in $\tilde{O}(\log_2(m) \cdot n^{\omega-1} \cdot C)$ operations in $\mathbb{K}$. ◀

▶ Notation 34. We consider in what follows that x^k, where k is a positive integer, is the matrix in $\mathsf{diag}(x^k, x^k, \ldots, x^k) \in \mathbb{K}[x]^{n\times n}$, when multiplied by a polynomial matrix $\mathsf{F} \in \mathbb{K}[x]^{n\times m}$.

▶ **Definition 35.** *Let* $\mathsf{M} = [M_{ij}]_{i,j=1,\ldots,n} \in \mathbb{K}[x]^{n\times n}$*, where* M *is a modulus polynomial matrix and* $\mathsf{rdeg}(M) = [d_1,\ldots,d_n]$*. Let* $\mathsf{F} = [f_{ij}]_{j=1,\ldots,m}^{i=1,\ldots,n} \in \mathbb{K}[x]^{n\times m}$*, such that* $\mathsf{rdeg}(\mathsf{F}) < [d_1,\ldots,d_n]$*. Let* $\nu = \{\nu_1,\ldots,\nu_m\}$ *be a set of positive integers.*
We define the problem $\mathsf{LastCoeffModMat}(\mathsf{M},\mathsf{F},\nu)$ *as follows:*

LastCoeffModMat	
Input:	$\mathsf{M} \in \mathbb{K}[x]^{n\times n}$, $\mathsf{F} \in \mathbb{K}[x]^{n\times m}$, $\nu = \{\nu_1,\ldots,\nu_m\}$.
Output:	$\mathsf{coeff}(\mathsf{elem}(x^{k_j}\cdot\mathsf{F} \text{ rem } \mathsf{M},i,j),d_i-1)$ *for* $k_j = 0,\ldots,\nu_j-1$, $i = 1,\ldots,n$ *and* $j = 1,\ldots,m$.

The following theorem is the main ingredient for solving the matrix modular approximation problem in the most optimal complexity, currently achieved using polynomial computations in [22]. We show that computing the coefficients described in Definition 29 is equivalent to computing the quotient of an euclidean division in the proof of the following theorem.

▶ **Theorem 36.** *Let* $\mathsf{M} = [M_{ij}]_{i,j=1,\ldots,n} \in \mathbb{K}[x]^{n\times n}$*, where* M *is a modulus polynomial matrix and* $\mathsf{rdeg}(M) = [d_1,\ldots,d_n]$*. Let* $\mathsf{F} = [f_{ij}]_{j=1,\ldots,m}^{i=1,\ldots,n} \in \mathbb{K}[x]^{n\times m}$*, such that* $\mathsf{rdeg}(\mathsf{F}) < [d_1,\ldots,d_n]$*. Let* $\nu = \{\nu_1,\ldots,\nu_m\}$ *be a list of positive integers.*
The problem $\mathsf{LastCoeffModMat}(\mathsf{M},\mathsf{F},\nu)$ *can be solved in* $\tilde{O}(n^{\omega-1}\cdot D + m\cdot n^{\omega-2}\cdot D + n^{\omega-1}\cdot C)$*, where* $D = \sum_{i=1}^{n} d_i$ *and* $C = \sum_{j=1}^{m}\nu_j$.

Proof. Let $k = 1,\ldots,\nu$ where ν is a positive integer. For M,F and k, there exist unique matrices Q_k and R_k such that $x^k\cdot\mathsf{F} = \mathsf{M}\cdot\mathsf{Q}_k + \mathsf{R}_k$ where $\mathsf{R}_k \in \mathbb{K}[x]^{n\times m}$ such that $\mathsf{rdeg}(\mathsf{R}_k) < [d_1,\ldots,d_n]$ and $\mathsf{Q}_k \in \mathbb{K}[x]^{n\times m}$ (see Theorem 17). Since $\mathsf{rdeg}(x^k\cdot\mathsf{F}) < [d_1+k,\ldots,d_n+k]$, we have that $\mathsf{deg}(\mathsf{Q}_k) < k$ by Lemma 19.
We define the matrix $\mathbb{X}^{d+\ell} \in \mathbb{K}[x]^{n\times n}$ for any integer ℓ as $\mathsf{diag}(x^{d_1+\ell},x^{d_2+\ell},\ldots,x^{d_n+\ell})$.
We extend the notion of reversal to polynomial matrices and define it for $\mathsf{M},\mathsf{F},\mathsf{Q}_k$ and R_k as follows:

$$\overline{\mathsf{M}} = \mathbb{X}^d\cdot\mathsf{M}(x^{-1}),$$

$$\overline{\mathsf{F}} = \mathbb{X}^{d-1}\cdot\mathsf{F}(x^{-1}),$$

$$\overline{\mathsf{Q}_k} = x^{k-1}\cdot\mathsf{Q}_k(x^{-1}),$$

$$\overline{\mathsf{R}_k} = \mathbb{X}^{d-1}\cdot\mathsf{R}_k(x^{-1}).$$

Using the previous euclidean division identity, where we replace x by x^{-1}, we have $x^{-k}\cdot\mathsf{F}(x^{-1}) = \mathsf{M}(x^{-1})\cdot\mathsf{Q}_k(x^{-1}) + \mathsf{R}_k(x^{-1})$. Then, we multiply on the left by $\mathbb{X}^{d+(k-1)}$ the previous identity and we get:

$$\mathbb{X}^{d+(k-1)}\cdot x^{-k}\cdot\mathsf{F}(x^{-1}) = \mathbb{X}^{d+(k-1)}\cdot\mathsf{M}(x^{-1})\cdot\mathsf{Q}_k(x^{-1}) + \mathbb{X}^{d+(k-1)}\cdot\mathsf{R}_k(x^{-1})$$
$$\Leftrightarrow (\mathbb{X}^{d-1}\cdot\mathsf{F}(x^{-1})) = (\mathbb{X}^d\cdot\mathsf{M}(x^{-1}))\cdot(x^{(k-1)}\cdot\mathsf{Q}_k(x^{-1})) + x^k\cdot(\mathbb{X}^{d-1}\cdot\mathsf{R}_k(x^{-1}))$$

We obtain the following identity:

$$\overline{\mathsf{F}} = \overline{\mathsf{M}}\cdot\overline{\mathsf{Q}_k} + x^k\cdot\overline{\mathsf{R}_k}. \tag{9}$$

Consider $k_j = 0,\ldots,\nu_j-1$ for $j = 1,\ldots,m$. We rewrite Equation (9) with $k = k_j$ for each column $j = 1,\ldots,m$ as follows:

$$\overline{\mathsf{F}_{*j}} = \overline{\mathsf{M}}\cdot\overline{\mathsf{Q}_{k_j,j}} + x^{k_j}\cdot\overline{\mathsf{R}_{k_j,j}}, \tag{10}$$

where $\overline{\mathsf{F}_{*j}} \in \mathbb{K}[x]^{n\times 1}$ is the j-th column of the matrix $\overline{\mathsf{F}}$, $\overline{\mathsf{Q}_{k_j,j}} \in \mathbb{K}[x]^{n\times 1}$ is the j-th column of the matrix $\overline{\mathsf{Q}_{k_j}}$ and $\overline{\mathsf{R}_{k_j,j}} \in \mathbb{K}[x]^{n\times 1}$ is the j-th column of the matrix $\overline{\mathsf{R}_{k_j}}$.
We reduce the identity (10) by x^{k_j}, which gives us $\overline{\mathsf{F}_{*j}} \text{ rem } x^{k_j} = \overline{\mathsf{M}}\cdot\overline{\mathsf{Q}_{k_j,j}} \text{ rem } x^{k_j}$.
Since M is a modulus polynomial matrix, we have that $\mathsf{rlm}_0(\mathsf{M}) = \mathbb{I}_n$ and this is invertible. The matrix, in $\mathbb{K}^{n\times n}$, of the constant coefficients of the polynomials in $\overline{\mathsf{M}}$ is an invertible matrix, as it is equal to $\mathsf{rlm}_0(\mathsf{M})$. Thus, the matrix $\overline{\mathsf{M}}$ is invertible in $\mathbb{K}[\![x]\!]$ and we denote by $\overline{\mathsf{M}}^{-1}$ the inverse. In what follows, we only consider the truncated version of the matrix $\overline{\mathsf{M}}^{-1}$ to work over the field $\mathbb{K}[x]$. We obtain the following identity:

$$\overline{\mathsf{Q}_{k_j,j}} = \overline{\mathsf{M}}^{-1}\cdot\overline{\mathsf{F}_{*j}} \text{ rem } x^{k_j}. \tag{11}$$

We denote by $\mathsf{a}_{k_j} \in \mathbb{K}^{n\times 1}$ the vector where at the i-th row we have the $d_i - 1$-th coefficient of the polynomial $x^{k_j} \cdot F_{*j} \bmod \mathsf{M}$, for $i = 1, \ldots, n$, $k_j = 0, \ldots, \nu_j - 1$ and $j = 1, \ldots, m$.
The idea is that computing, for each $j = 1, \ldots, m$, the vector of polynomials $\overline{\mathsf{Q}_{\nu_j,j}}$ (see (11) with $k_j = \nu_j$), i.e.,

$$\overline{\mathsf{Q}_{\nu_j,j}} = \overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}_{*j}} \bmod x^{\nu_j} \tag{12}$$

allows us to compute the vectors of coefficients $\mathsf{a}_{k_j,j}$ for $k_j = 0, \ldots, \nu_j - 1$ and $j = 1, \ldots, m$.
We show that for one column j in Lemma 37. We have $\overline{\mathsf{Q}_{\nu_j,j}} = \sum_{k_j=0}^{\nu_j-1} x^{k_j+1} \cdot \mathsf{elem}(\mathsf{a}_{k_j}, j)$. We apply this lemma to each column $j = 1, \ldots, m$.
We now focus on computing the vectors $\overline{\mathsf{Q}_{\nu_j,j}}$ for $j = 1, \ldots, m$. For this, we define $\overline{\mathsf{Q}_\nu} \in \mathbb{K}[x]^{n\times m}$ as $\begin{bmatrix}\overline{\mathsf{Q}_{\nu_1,1}} & \overline{\mathsf{Q}_{\nu_2,2}} & \cdots & \overline{\mathsf{Q}_{\nu_m,m}}\end{bmatrix}$. Moreover, we define $\mathbb{X}^\nu$ as the matrix $\mathsf{diag}(x^{\nu_1}, x^{\nu_2}, \ldots, x^{\nu_m})$.
Since we can verify that

$$\begin{aligned}\overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}} \operatorname{rrem} \mathbb{X}^\nu &= \begin{bmatrix}\overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}_{*1}} \bmod x^{\nu_1} & \overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}_{*2}} \bmod x^{\nu_2} & \cdots & \overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}_{*m}} \bmod x^{\nu_m}\end{bmatrix} \\ &= \begin{bmatrix}\overline{\mathsf{Q}_{\nu_1,1}} & \overline{\mathsf{Q}_{\nu_2,2}} & \cdots & \overline{\mathsf{Q}_{\nu_m,m}}\end{bmatrix},\end{aligned}$$

it is equivalent to compute $\overline{\mathsf{Q}_\nu} = \overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}} \bmod \mathbb{X}^\nu$.
To compute $\overline{\mathsf{Q}_\nu}$, we use Algorithm 3. We apply the algorithm with the inputs $\mathsf{G} = \overline{\mathsf{M}} \in \mathbb{K}[x]^{n\times n}$, $\mathsf{P} = \overline{\mathsf{F}} \in \mathbb{K}[x]^{n\times m}$ and $\nu = \{\nu_1, \ldots, \nu_m\}$. By Lemma 33, Algorithm 3 computes $\overline{\mathsf{Q}_\nu}$ in $O(n^{\omega-1} \cdot C)$.
Therefore, we get a_{k_j} for $k = 0, \ldots, \nu_j - 1$, for $j = 1, \ldots, m$, in $O(n^{\omega-1} \cdot C)$, which concludes the proof. ◀

▶ **Lemma 37.** *Let $\mathsf{M} = [M_{ij}]_{i,j=1,\ldots,n} \in \mathbb{K}[x]^{n\times n}$, where M is a modulus polynomial matrix with $\mathsf{rdeg}(M) = [d_1, \ldots, d_n]$ and $D = \sum_{i=1}^n d_i$. Let $\mathsf{f} = [f_i]_{i=1,\ldots,n} \in \mathbb{K}[x]^{n\times 1}$, such that $\mathsf{rdeg}(\mathsf{f}) < [d_1, \ldots, d_n]$. Let ν be a positive integer. We define $\mathsf{q}_\nu = x^\nu \cdot \mathsf{f} \operatorname{quo} \mathsf{M}$. We have $\overline{\mathsf{q}_\nu} = \sum_{k=0}^{\nu-1} x^{k+1} \cdot \mathsf{a}_k$, where a_k is the vector such that at the i-th row we have the $d_i - 1$-th coefficient of the polynomial $x^k \cdot \mathsf{f} \bmod \mathsf{M}$, for $i = 1, \ldots, n$, $k = 0, \ldots, \nu - 1$.*

Proof. Since M is a modulus polynomial matrix and thus a row reduced matrix, by Theorem 17, there exists unique matrices $\mathsf{q}_k \in \mathbb{K}[x]^{n\times 1}$ and $\mathsf{r}_k \in \mathbb{K}[x]^{n\times 1}$ such that $x^k \cdot \mathsf{f} = \mathsf{M} \cdot \mathsf{q}_k + \mathsf{r}_k$, where $\mathsf{rdeg}(\mathsf{r}_k) < [d_1, \ldots, d_n]$ for $k = 0, \ldots, \nu - 1$.
We multiply both sides on the right by x and we get $x^{k+1} \cdot \mathsf{f} = x \cdot \mathsf{M} \cdot \mathsf{q}_k + x \cdot \mathsf{r}_k$. We can write r_k as $\mathsf{a}_k \cdot \mathbb{X}^{d-1} + \mathsf{b}_k$, where $\mathsf{rdeg}(\mathsf{b}_k) < [d_1, \ldots, d_n]$, as a_k is the vector such that at the i-th row we have the $d_i - 1$-th coefficient of the polynomial r_k.
Moreover, we can write $\mathsf{M} = \mathbb{X}^d - \mathsf{M}_0$ where $\mathsf{rdeg}(\mathsf{M}_0) < [d_1, \ldots, d_n]$ and $\mathbb{X}^d = \mathsf{diag}(x^{d_1}, x^{d_2}, \ldots, x^{d_n})$.
Thus, we can write the previous identity as $x^k \cdot \mathsf{f} = x \cdot \mathsf{M} \cdot \mathsf{q}_k + x \cdot (\mathsf{a}_k \cdot \mathbb{X}^{d-1} + \mathsf{b}_k)$ which can be rewritten as $x^{k+1} \cdot \mathsf{f} = x \cdot \mathsf{M} \cdot \mathsf{q}_k + \mathsf{a}_k \cdot \mathbb{X}^d + x \cdot \mathsf{b}_k$. We replace $\mathbb{X}^d$ by $\mathsf{M} + \mathsf{M}_0$ in the previous identity, which gives us $x^{k+1} \cdot \mathsf{f} = x \cdot \mathsf{M} \cdot \mathsf{q}_k + \mathsf{a}_k \cdot (\mathsf{M} + \mathsf{M}_0) + x \cdot \mathsf{b}_k$.
We obtain $x^{k+1} \cdot \mathsf{f} = \mathsf{M} \cdot (x \cdot \mathsf{q}_k + \mathsf{a}_k) + \mathsf{a}_k \cdot \mathsf{M}_0 + x \cdot \mathsf{b}_k$. We note that $(x \cdot \mathsf{q}_k + \mathsf{a}_k)$ is of degree strictly less than $k+1$ and $\mathsf{rdeg}(\mathsf{a}_k \cdot \mathsf{M}_0 + x \cdot \mathsf{b}_k) < [d_1, \ldots, d_n]$. Moreover, there exists unique $\mathsf{q}_{k+1} \in \mathbb{K}[x]^{n\times 1}$ and $\mathsf{r}_{k+1} \in \mathbb{K}[x]^{n\times 1}$ such that $x^{k+1} \cdot \mathsf{f} = \mathsf{M} \cdot \mathsf{q}_{k+1} + \mathsf{r}_{k+1}$ where $\mathsf{rdeg}(\mathsf{r}_{k+1}) < [d_1, \ldots, d_n]$ and $\mathsf{rdeg}(\mathsf{q}_{k+1}) < k + 1$.
Since, they have same degrees and by the uniqueness of the quotient and the remainder in the polynomial matrix division of same degree (see Theorem 17), we have that:

$$\begin{aligned}\mathsf{q}_{k+1} &= x \cdot \mathsf{q}_k + \mathsf{a}_k, \\ \mathsf{r}_{k+1} &= \mathsf{a}_k \cdot \mathsf{M}_0 + x \cdot \mathsf{b}_k.\end{aligned}$$

We focus on $\mathsf{q}_{k+1} = x \cdot \mathsf{q}_k + \mathsf{a}_k$ and by multiplying the identity by x^k and using the definition of the reversal of the quotion Q_k in (8), we get the following induction

$$\overline{\mathsf{q}_{k+1}} = x^{k+1} \cdot \mathsf{a}_k + \overline{\mathsf{q}_k}. \tag{13}$$

Applying the recurrence relation iteratively yields

$$\begin{aligned}&\overline{\mathsf{q}_0} = 0, \\ &\overline{\mathsf{q}_1} = x^1 \cdot \mathsf{a}_0 + \overline{\mathsf{q}_0} = \mathsf{a}_0 \cdot x, \\ &\quad\vdots \\ &\overline{\mathsf{q}_\nu} = x^\nu \cdot \mathsf{a}_{\nu-1} + \overline{\mathsf{q}_{\nu-1}} = \mathsf{a}_0 \cdot x + \mathsf{a}_1 \cdot x^2 + \cdots + \mathsf{a}_{\nu-1} \cdot x^\nu.\end{aligned}$$

We note that $\overline{\mathsf{q}_0} = 0$, since for the division $\mathsf{F} = \mathsf{M} \cdot \mathsf{q}_0 + \mathsf{R}_0$, Lemma 19 gives $\mathsf{deg}(\mathsf{q}_0) < 0$, and therefore $\mathsf{q}_0 = 0$. Therefore, for $k = \nu - 1$, we use (13) to get $\overline{\mathsf{q}_\nu} = \sum_{k=0}^{\nu-1} x^{k+1} \cdot \mathsf{a}_k$, which concludes the proof. ◀

4.2 Main result: matrix modular approximation

Due to space constraints, we provide the full proof of the following theorem in Appendix C.

▶ **Theorem 38.** *Let* $\mathsf{M} = [M_{ij}]_{i,j=1,\dots,n} \in \mathbb{K}[x]^{n\times n}$*, where* M *is a modulus polynomial matrix and* $\mathsf{rdeg}(M) = [d_1, \dots, d_n]$*,* $\mathsf{F} = [f_{ij}]_{j=1,\dots,m}^{i=1,\dots,n} \in \mathbb{K}[x]^{n\times m}$ *such that* $\mathsf{rdeg}(\mathsf{F}) < [d_1, \dots, d_n]$ *and* $\nu = \{\nu_1, \dots, \nu_m\}$ *a set of positive integers. Let* $C = \sum_{j=1}^{m} \nu_j$ *and* $D = \sum_{i=1}^{n} d_i$*. We suppose that* $n \in O(m)$ *and* $C \in O(D)$*.*
The problem $\mathsf{ApproxMatrixMod}(\mathsf{M}, \mathsf{F}, \nu)$ *can be solved in* $\tilde{O}(\mathsf{max}(m,n)^{\omega-1} \cdot D)$ *by Algorithm 4.*

Algorithm 4 Matrix modular approximation

Input: $\mathsf{M} \in \mathbb{K}[x]^{n\times n}$, $\mathsf{F} \in \mathbb{K}[x]^{n\times m}$, $\nu = \{\nu_1, \dots, \nu_m\}$
Output: $\mathsf{ApproxMatrixMod}(\mathsf{M}, \mathsf{F}, \nu)$

$\mathsf{A} \in \mathbb{K}^{D\times C}$;
for $j = 1$ **to** m **do**
  $\overline{\mathsf{F}}_{*j}$ is the vector of coefficients of each polynomial in F_{*j};
  $\mathsf{A}_j = \begin{bmatrix}\overline{\mathsf{F}}_{*j} & \mathbb{X}(\mathsf{M})\overline{\mathsf{F}}_{*j} & \cdots & \mathbb{X}(\mathsf{M})^{\nu_j-1}\overline{\mathsf{F}}_{*j}\end{bmatrix}$;
Get $\mathsf{G}, \mathsf{H} \in \mathbb{K}^{*\times \mathsf{max}(m,n)}$ such that $\phi_+(\mathsf{A}) = \mathsf{G} \cdot \mathsf{H}^{\mathsf{T}}$ using Lemma 54;
return $\mathsf{LinearSystem}[\mathbb{Z}_{D,0}, \mathbb{Z}_{C,0}^{\mathsf{T}}, O(\mathsf{max}(m,n))](\mathsf{G}, \mathsf{H})$;

Proof sketch. We use the same reduction strategy outlined in Theorem 27, i.e., we reduce the problem $\mathsf{ApproxMatrixMod}(\mathsf{M}, \mathsf{F}, \nu)$ to a problem of type $\mathsf{LinearSystem}[\mathbb{Z}_{D,0}, \mathbb{Z}_{C,0}^{\mathsf{T}}, O(\mathsf{max}(m,n))]$, as presented in Algorithm 4. The key difference lies in the definition of the matrix A, where we exploit the generalized multiplication matrix in Definition 30.
We define the matrix $\mathsf{A} \in \mathbb{K}^{D\times C}$ as the matrix $\mathsf{A} = \begin{bmatrix}\mathsf{A}_1 & \mathsf{A}_2 & \cdots & \mathsf{A}_m\end{bmatrix}$, where $\mathsf{A}_j \in \mathbb{K}^{D\times\nu_j}$, for $j = 1, \dots, m$, corresponds to the matrix

$$\mathsf{A}_j = \begin{bmatrix}\overline{\mathsf{F}}_{*j} & \mathbb{X}(\mathsf{M}) \cdot \overline{\mathsf{F}}_{*j} & \cdots & \mathbb{X}(\mathsf{M})^{\nu_j-1} \cdot \overline{\mathsf{F}}_{*j}\end{bmatrix}, \tag{14}$$

such that $\overline{\mathsf{F}} \in \mathbb{K}^{D\times m}$ is the matrix where we replace each polynomial of F by the vector of its coefficients.
We refer to Appendix C for the detailed construction of the ϕ_+-generators of length $O(\mathsf{max}(m,n))$ for the matrix A. Note that Theorem 36 and Lemma 31 are the main tools used in this construction. ◁

5 Solving quasi-Toeplitz like linear systems deterministically

In this section, we present a deterministic method to solve quasi-Toeplitz structured linear systems given by their ϕ_+-generators. Our approach achieves the same complexity as the best known algorithms [3, 1], thereby eliminating the randomization used in those works.
The Σ-LU decomposition of a quasi-Toeplitz matrix is a well-known result in linear algebra, that originated from [28].

▶ **Lemma 39.** *Let* $\mathsf{A} \in \mathbb{K}^{n\times m}$ *with* $\phi_+(\mathsf{A}) = \mathsf{G} \cdot \mathsf{H}^{\mathsf{T}}$ *such that* $\mathsf{G} \in \mathbb{K}^{n\times\alpha}$ *and* $\mathsf{H} \in \mathbb{K}^{m\times\alpha}$*. We can write* $\mathsf{A} = \sum_{i=1}^{\alpha} \mathbb{L}(f_i) \cdot \mathbb{U}(g_i)$*, where* f_i *is the* i*-th column of the matrix* G *and* g_i *is the* i*-th column of the matrix* H*.*

The following two lemmas are results commonly used in the literature (for instance [3]). We provide proofs of these results for the sake of completeness in Appendix D. Both results are the starting point for finding a polynomial representation of the structured linear systems we are interested in.

▶ **Lemma 40.** *Let* $\mathsf{u}, \mathsf{v} \in \mathbb{K}^{n\times 1}$ *be two vectors and* $\mathsf{p} = \mathbb{L}(\mathsf{u}) \cdot \mathsf{v}$*. We have that* $p(x) = u(x) \cdot v(x) \bmod x^n$*, where* $u(x) = \sum_{i=0}^{n-1} u_i x^i$*,* $v(x) = \sum_{i=0}^{n-1} v_i x^i$ *and* $p(x) = \sum_{i=0}^{n-1} p_i x^i$ *are the polynomial representations of the vectors* u*,* v *and* p*, respectively.*

▶ **Lemma 41.** *Let* $\mathsf{u}, \mathsf{v} \in \mathbb{K}^{m\times 1}$ *be two vectors and* $\mathsf{p} = \mathbb{U}(\mathsf{u}) \cdot \mathsf{v}$. *We have that* $p(x) = \mathsf{rev}_{m-1}(u(x)) \cdot v(x) \ \mathsf{quo}\ x^{m-1}$, *where* $u(x) = \sum_{i=0}^{m-1} u_i x^i$, $v(x) = \sum_{i=0}^{m-1} v_i x^i$ *and* $p(x) = \sum_{i=0}^{m-1} p_i x^i$ *are the polynomial representations of the vectors* u, v *and* p, *respectively.*

In what follows, we introduce two approximation problems. The first is that of computing a Popov basis of solutions to a $\mathbb{K}[x]$-linear relation modulo a polynomial. The second is the problem where we seek rational functions all sharing the same denominator and satisfying a linear relation modulo a family of polynomials.

▶ **Definition 42.** *Given an approximation order* $n \in \mathbb{N}$, *a* $\mathsf{F} \in \mathbb{K}^{1\times\alpha}$ *with* $\mathsf{rdeg}(\mathsf{F}) < n$, *and a shift* $s \in \mathbb{Z}^\alpha$, *we define the problem* $\mathsf{ApproximantBasis}(\mathsf{F}, n)$ *as follows:*

ApproximantBasis	
Input:	$\mathsf{F} \in \mathbb{K}[x]^{1\times\alpha}$, n, $s = [s_1, \ldots, s_\alpha] \in \mathbb{Z}^\alpha$.
Output:	$\mathsf{P} \in \mathbb{K}[x]^{\alpha\times\alpha}$ *such that* P *is the* s*-popov basis of solutions* $\{\mathsf{p} \in \mathbb{K}[x]^{\alpha\times 1} \mid \mathsf{F}\cdot\mathsf{p} = 0 \bmod x^n\}$.

▶ **Theorem 43** (Theorem 1.1 in [14])**.** *Given an approximation order* $n \in \mathbb{N}$, *a matrix* $\mathsf{F} \in \mathbb{K}^{1\times\alpha}$ *with* $\mathsf{rdeg}(\mathsf{F}) < n$, *and a shift* $s \in \mathbb{Z}^\alpha$, *the problem* $\mathsf{ApproximantBasis}(\mathsf{F}, n)$ *can be solved in* $\tilde{O}(\alpha^{\omega-1} \cdot n)$.

▶ **Definition 44.** *Let* $\mathsf{F} \in \mathbb{K}^{t\times\alpha}$ *with* $t \leq \alpha$ *and* $\mathsf{rdeg}(\mathsf{F}) < n$. *Let* $T = (T_1, \ldots, T_t) \in \mathbb{N}^t$ *and* $N = (N_1, \ldots, N_\alpha) \in \mathbb{N}^\alpha$ *be degree bounds such that* $T_i < n$ *for* $i = 1, \ldots, t$ *and* $N_j < n$ *for* $j = 1, \ldots, \alpha$.
We define the problem $\mathsf{ApproxExtended}(\mathsf{F}, T, N)$ *as follows:*

ApproxExtended	
Input:	$\mathsf{F} \in \mathbb{K}[x]^{t\times\alpha}$, $T = (T_1, \ldots, T_t) \in \mathbb{N}^t$, $N = (N_1, \ldots, N_\alpha) \in \mathbb{N}^\alpha$.
Output:	*nonzero* $\mathsf{p} \in \mathbb{K}[x]^{t\times 1}$ *and* $\phi = (\phi_1, \ldots, \phi_\alpha) \in \mathbb{K}[x]^\alpha$ *such that* $\mathsf{F}\cdot\mathsf{p} = \phi \bmod x^n$, *with* $\mathsf{rdeg}(\mathsf{p}) < T$ *and* $\deg(\phi_j) < N_j$ *for* $j = 1, \ldots, \alpha$.

Note that we consider, in the above, a particular case of the result of Rosenkilde and Storjohann [25] in Theorem 45, where we assume that all the α moduli are x^n and that the number of rows of the matrix F is less than or equal to the number of its columns. For ease of notation, we denote the family of α moduli $(x^n, \ldots, x^n)$ as x^n.

▶ **Theorem 45** (Subcase of Theorem 1.7 in [25])**.** *Given a matrix* $\mathsf{F} \in \mathbb{K}^{t\times\alpha}$ *with* $\mathsf{rdeg}(\mathsf{F}) < n$ *and* $T = (T_1, \ldots, T_t) \in \mathbb{N}^t$ *and* $N = (N_1, \ldots, N_\alpha) \in \mathbb{N}^\alpha$ *be degree bounds such that* $T_i < n$ *for* $i = 1, \ldots, t$ *and* $N_j < n$ *for* $j = 1, \ldots, \alpha$, *the problem* $\mathsf{ApproxExtended}(\mathsf{F}, T, N)$ *can be solved in* $\tilde{O}(\alpha^{\omega-1} \cdot t \cdot n)$.

▶ **Theorem 46.** *Given* $\mathsf{G} \in \mathbb{K}^{n\times\alpha}$ *and* $\mathsf{H} \in \mathbb{K}^{m\times\alpha}$, *such that* $\phi_+(\mathsf{A}) = \mathsf{G}\cdot\mathsf{H}^\mathsf{T}$ *for a matrix* $\mathsf{A} \in \mathbb{K}^{n\times m}$, *and a vector* $\mathsf{v} \in \mathbb{K}^{n\times 1}$, *the problem* $\mathsf{LinearSystem}(\mathsf{G}, \mathsf{H}, \mathsf{v})$ *can be solved in* $\tilde{O}(\alpha^{\omega-1} \cdot n)$, *using Algorithm 5.*

Proof. We define

- $f_j(x) = \sum_{i=0}^{n-1} G_{ij} x^i$ for $j = 0, \ldots, \alpha - 1$,
- $g_j(x) = \sum_{i=0}^{m-1} H_{ij} x^i$ for $j = 0, \ldots, \alpha - 1$,
- $v(x) = \sum_{i=0}^{n-1} \mathsf{v}_i x^i$, where v_i is the i-th element of the vector v,
- $u(x) = \sum_{i=0}^{m-1} \mathsf{u}_i x^i$, where u_i is the i-th element of the unknown vector u.

We can then express the problem $\mathsf{LinearSystem}(\mathsf{G}, \mathsf{H}, \mathsf{v})$ using polynomials as follows, for any solution u,

$$\begin{aligned}\mathsf{A}\cdot\mathsf{u} = \mathsf{v} &\Leftrightarrow \sum_{i=1}^{\alpha} \mathbb{L}(\mathsf{G}_{*j}) \cdot \mathbb{U}(\mathsf{H}_{*j}) \cdot \mathsf{u} = \mathsf{v} \quad \text{(Since } \phi_+(\mathsf{A}) = \mathsf{G}\cdot\mathsf{H}^\mathsf{T} \text{ and by Lemma 39)} \\ &\Leftrightarrow \sum_{i=1}^{\alpha} f_i \cdot (\mathsf{rev}_{m-1}(g_i) \cdot u \ \mathsf{quo}\ x^{m-1}) \bmod x^n = v \quad \text{(by Lemmas 40 and 41).}\end{aligned}$$

We denote by $p_i(x)$ the polynomial $\mathsf{rev}_{m-1}(g_i) \cdot u(x) \ \mathsf{quo}\ x^{m-1}$ for $i = 0, \ldots, \alpha - 1$, and p the vector formed by the p_i's. We define $\mathsf{F} = [f_0, f_1, \ldots, f_{\alpha-1}]$. Computing all possible p_i's such that $\sum_{i=1}^{\alpha} f_i \cdot p_i = v \bmod x^n$, is equivalent to computing all possible vectors p such that $\mathsf{F}\cdot\mathsf{p} = v \bmod x^n$.
We can suppose without loss of generality that $m < n$ and we proceed as follows in this proof. To handle the case where $m \geq n$, we simply replace n by m in the rest of the proof.

Algorithm 5 Solving a structured linear system

Input: $\mathsf{G} = [G_{ij}]_{j=0,\dots,\alpha-1}^{i=0,\dots,n-1} \in \mathbb{K}^{n\times\alpha}$, $\mathsf{H} = [H_{ij}]_{j=0,\dots,\alpha-1}^{i=0,\dots,m-1} \in \mathbb{K}^{m\times\alpha}$, $\mathsf{v} = [\mathsf{v}_i]_{i=0,\dots,n-1} \in \mathbb{K}^{n\times 1}$.
Output: $\mathsf{LinearSystem}(\mathsf{G},\mathsf{H},\mathsf{v})$

// Transform to polynomial matrices
for $j = 0$ **to** $\alpha - 1$ **do**
  $f_j(x) = \sum_{i=0}^{n-1} G_{ij}x^i$;
$\mathsf{F} = [f_0(x), f_1(x), \dots, f_{\alpha-1}(x)]$;
for $j = 0$ **to** $\alpha - 1$ **do**
  $g_j(x) = \sum_{i=0}^{m-1} H_{ij}x^i$;
$\mathsf{g} = [g_0, g_1, \dots, g_{\alpha-1}]^\mathsf{T}$;
$v(x) = \sum_{i=0}^{n-1} \mathsf{v}_i x^i$;
// Approximant basis computation
$\begin{bmatrix}\mathsf{P} & \mathsf{s}\\ 0 & \mu\end{bmatrix} = \mathsf{ApproximantBasis}(\begin{bmatrix}\mathsf{F} & -v\end{bmatrix}, n)$;
if $\deg(\mu) > 0$ **then**
  return $\emptyset$;
else if $\deg(\mu) = 0$ **then**
  $\mu(x) = 1$;
  // Reverse
  $(\delta_1, \dots, \delta_\alpha) = \mathsf{rdeg}(\mathsf{P})$;
  $\bar{\mathsf{P}}(x) = \mathsf{P}(x^{-1})\mathsf{diag}(x^{\delta_1}, \dots, x^{\delta_\alpha})$;
  $\bar{\lambda}(x) = \mathsf{diag}(x^{n-1-\delta_1}, \dots, x^{n-1-\delta_\alpha})\lambda(x^{-1})$;
  $\mathsf{rev}_{n-1}(\mathsf{s}) = [\mathsf{rev}_{n-1}(\mathsf{s}_0), \dots, \mathsf{rev}_{n-1}(\mathsf{s}_{\alpha-1})]^\mathsf{T}$;
  // Retrieve the solution
  $\mathsf{B} = \begin{bmatrix}\bar{\mathsf{P}}^{-1}\cdot\mathsf{g} & -\bar{\mathsf{P}}^{-1}\cdot\mathsf{rev}_{n-1}(\mathsf{s})\end{bmatrix}$;
  $T = (n, 1), N = (n-1-\delta_1, \dots, n-1-\delta_\alpha)$;
  $\begin{bmatrix}u\\1\end{bmatrix}, \lambda = \mathsf{ApproxExtended}(\mathsf{B}, T, N)$;
  return $[\mathsf{coeff}(u, \alpha-1), \dots, \mathsf{coeff}(u, 0),]^\mathsf{T}$;

We denote by $\mathsf{E} \in \mathbb{K}[x]^{1\times(\alpha+1)}$ the matrix formed by the polynomials $f_i(x)$ and the polynomial $-v(x)$, i.e., $\mathsf{E} = \begin{bmatrix}\mathsf{F} & -v(x)\end{bmatrix}$. From Theorem 43, we can compute $\mathsf{Q} = \mathsf{ApproximantBasis}(\mathsf{E}, n, [0,\dots,0,n])$ in $O(\alpha^{\omega-1}\cdot\mathcal{M}(n)+\alpha^{\omega-1}\cdot n\cdot\log(\alpha)) = \tilde{O}(\alpha^{\omega-1}\cdot n)$ operations in $\mathbb{K}$, where $[0,\dots,0,n] \in \mathbb{Z}^{\alpha+1}$. By Lemma 47, we get that $\mathsf{Q} = \begin{bmatrix}\mathsf{P} & \mathsf{s}\\ 0 & \mu(x)\end{bmatrix}$, where $\mathsf{P} \in \mathbb{K}[x]^{\alpha\times\alpha}$ is the $[0,\dots,0]$-Popov basis of solutions to the equation $\mathsf{F}\cdot\mathsf{p} = 0 \bmod x^n$, a monic polynomial $\mu(x)$ of minimal degree and $\mathsf{s} \in \mathbb{K}[x]^{\alpha\times 1}$ are such that $\mathsf{F}\cdot\mathsf{s} = \mu\cdot v \bmod x^n$.
We now distinguish two cases:

- If μ has degree ≥ 1, then there is not a solution to the equation $\mathsf{F}\cdot\mathsf{p} = v \bmod x^n$, as μ is of minimal degree. Therefore, there is no polynomial solution $(p_1,\dots,p_\alpha)$ to the equation $\sum_i f_i p_i = v \bmod x^n$; in this case, the last equation cannot hold and therefore $\mathsf{A}\cdot\mathsf{u} = \mathsf{v}$ has no solution u by the above equivalence.
- Otherwise, if μ has degree 0, then we can assume without loss of generality that $\mu = 1$. In this case, we can write any solution p to the equation $\mathsf{F}\cdot\mathsf{p} = v \bmod x^n$ as $\mathsf{p} = \mathsf{P}\cdot\lambda + \mathsf{s}$, where $\lambda \in \mathbb{K}[x]^{\alpha\times 1}$ is a vector of polynomials. This means that any solution to the equation can be expressed as a linear combination of the columns of the matrix P plus a vector s.

Going back to the equivalence and assuming $\mu = 1$, for any solution u,

$$\begin{aligned}\mathsf{A}\cdot\mathsf{u} = \mathsf{v} &\Leftrightarrow \sum_{i=1}^{\alpha} f_i \cdot (\mathsf{rev}_{n-1}(g_i)\cdot u \ \mathsf{quo}\ x^{n-1}) \bmod x^n = v\\ &\Leftrightarrow \exists\lambda \in \mathbb{K}[x]^{\alpha\times 1}, \mathsf{p} = \mathsf{P}\lambda + \mathsf{s}.\end{aligned}$$

We fix a $\lambda \in \mathbb{K}[x]^{\alpha\times 1}$.
Since Q is the $[0,\dots,0,n]$-Popov basis of $\{\mathsf{q} \in \mathbb{K}^{\alpha\times 1} \mid \mathsf{E}\cdot\mathsf{q} = 0 \bmod x^n\}$, we have that $\det(\mathsf{Q}) = x^n$. Therefore, we get that $\sum(\mathsf{rdeg}(\mathsf{Q})) \leq \deg(det(\mathsf{Q})) = n$. We can't reduce the degree of $det(\mathsf{Q})$ further, as it would mean there exists another linear combination of the columns of Q. Thus, we have $\sum(\mathsf{rdeg}(\mathsf{P})) \leq n$.
Note that $\mathsf{rdeg}(\mathsf{p}) \leq n-1$ since $\deg(\mathsf{rev}_{n-1}(g_i)) \leq n-1$ and $\deg(u) \leq n-1$. On the other hand, $\mathsf{rdeg}(\mathsf{s}) < \mathsf{rdeg}(\mathsf{P}) \leq n$, so we can deduce that $\mathsf{rdeg}(\mathsf{P}\lambda) < n-1$. Thus we can reverse the identity $\mathsf{p} = \mathsf{P}\lambda + \mathsf{s}$ with respect to degree $n-1$: $\mathsf{rev}_{n-1}(\mathsf{p}) = \mathsf{rev}_{n-1}(\mathsf{P}\lambda) + \mathsf{rev}_{n-1}(\mathsf{s})$. We recall that $p_i = \mathsf{rev}_{n-1}(g_i)\cdot u \ \mathsf{quo}\ x^{n-1}$, so we can write $\mathsf{rev}_{n-1}(p_i) = g_i\cdot\mathsf{rev}_{n-1}(u) \bmod x^n$ for $i = 1,\dots,\alpha$.
We focus on $\mathsf{rev}_{n-1}(\mathsf{P}\lambda)$. We have $\mathsf{rdeg}(\mathsf{P}) = (\delta_1,\dots,\delta_\alpha)$ with all $\delta_i \geq 0$ and $\delta_1+\cdots+\delta_\alpha \leq n$, and since P is in Popov form (zero shift), we also have $\mathsf{cdeg}(\mathsf{P}) = (\delta_1,\dots,\delta_\alpha)$. Moreover, since P is column reduced (as it is in an Popov form), we can apply the predictable degree property (Proposition 6.3.13 in [15]), to get $\deg(\mathsf{P}\lambda) = \mathsf{rdeg}_{\mathsf{cdeg}(\mathsf{P})}(\lambda) = \mathsf{max}_{i=1,\dots,\alpha}(\deg(\lambda_i)+\delta_i)$. Therefore, for $i = 1,\dots,\alpha$, we have $\deg(\lambda_i) \leq n-1-\delta_i$.

Consequently, we can write

$$\begin{aligned}\mathsf{rev}_{n-1}(\mathsf{P}\lambda) &= x^{n-1}\mathsf{P}(x^{-1})\lambda(x^{-1})\\ &= \mathsf{P}(x^{-1})\mathsf{diag}(x^{n-1},\dots,x^{n-1})\lambda(x^{-1})\\ &= \mathsf{P}(x^{-1})\mathsf{diag}(x^{\delta_1},\dots,x^{\delta_\alpha})\mathsf{diag}(x^{n-1-\delta_1},\dots,x^{n-1-\delta_\alpha})\lambda(x^{-1}) = \bar{\mathsf{P}}(x)\bar{\lambda}(x).\end{aligned}$$

where we denote by $\bar{\mathsf{P}}(x)$ the column-wise reverse of P (with respect to degrees $\delta_1,\dots,\delta_\alpha$), and by $\bar{\lambda}$ the reverse of λ (with respect to degrees $n-1-\delta_1,\dots,n-1-\delta_\alpha$).
We denote by g the vector of polynomials $[g_1, g_2, \dots, g_\alpha]^\mathsf{T}$.
Therefore, we obtain an approximation problem where we have to find u and λ such that $\mathsf{deg}(\lambda_i) \leq n-1-\delta_i$ and $\mathsf{deg}(u) \leq n-1$, as follows:

$$\begin{aligned}\mathsf{A}\cdot\mathsf{u} = \mathsf{v} &\Leftrightarrow \mathsf{g}\cdot\mathsf{rev}_{n-1}(u) = (\bar{\mathsf{P}}\bar{\lambda} + \mathsf{rev}_{n-1}(\mathsf{s}))\bmod x^n\\ &\Leftrightarrow \mathsf{g}\cdot\mathsf{rev}_{n-1}(u) - \mathsf{rev}_{n-1}(\mathsf{s}) = \bar{\mathsf{P}}\bar{\lambda} \bmod x^n\\ &\Leftrightarrow \bar{\mathsf{P}}^{-1}\cdot\mathsf{g}\cdot\mathsf{rev}_{n-1}(u) - \bar{\mathsf{P}}^{-1}\cdot\mathsf{rev}_{n-1}(\mathsf{s}) = \bar{\lambda} \bmod x^n\\ &\Leftrightarrow \begin{bmatrix}\bar{\mathsf{P}}^{-1}\cdot\mathsf{g} & -\bar{\mathsf{P}}^{-1}\cdot\mathsf{rev}_{n-1}(\mathsf{s})\end{bmatrix}\cdot\begin{bmatrix}\mathsf{rev}_{n-1}(u)\\ 1\end{bmatrix} = \bar{\lambda} \bmod x^n.\end{aligned}$$

Note that $\bar{\mathsf{P}}$ is invertible modulo x^n. This is built on the fact that P is in Popov form, which implies that $\mathsf{clm}_0(\mathsf{P})$ is invertible. Therefore, the matrix of constant terms of $\bar{\mathsf{P}}$ is also invertible, and thus the determinant of $\bar{\mathsf{P}}$ is nonzero (its evaluation at 0 is nonzero).
We consider the problem $\mathsf{ApproxExtended}$ with the inputs $[\bar{\mathsf{P}}^{-1}\cdot\mathsf{g}, -\bar{\mathsf{P}}^{-1}\cdot\mathsf{rev}_{n-1}(\mathsf{s})] \in \mathbb{K}[x]^{\alpha\times 2}$, $T=(n,1)\in\mathbb{N}^2$ and $N=(n-1-\delta_1,\dots,n-1-\delta_\alpha)\in\mathbb{N}^\alpha$. We can assume that $n\geq 2$, as the case $n=1$ is trivial and we indeed have $\mathsf{rdeg}([\bar{\mathsf{P}}^{-1}\cdot\mathsf{g}, -\bar{\mathsf{P}}^{-1}\cdot\mathsf{rev}_{n-1}(\mathsf{s})]) < n$.
We get a solution $\begin{bmatrix}\mathsf{rev}_{n-1}(u)\\ \beta\end{bmatrix}$ and $\bar{\lambda} = (\bar{\lambda}_1,\dots,\bar{\lambda}_\alpha)$ such that $\mathsf{rdeg}(\begin{bmatrix}\mathsf{rev}_{n-1}(u)\\ \beta\end{bmatrix}) < T$ and $\mathsf{deg}(\bar{\lambda}_j) < n-1-\delta_j$.
Since $\mathsf{deg}(\beta)\leq 0$, we may assume $\beta = 1$ as the algorithm in [25] returns such a solution.
The computation is done in $\tilde{O}(n^{\omega-1}\cdot n)$ by Theorem 45. Finally, we can retrieve the solution of the problem $\mathsf{LinearSystem}(\mathsf{G},\mathsf{H},\mathsf{v})$ by taking the vector of coefficients of u. The total cost is $\tilde{O}(\alpha^{\omega-1}\cdot n)$. ◀

▶ **Lemma 47.** *Let $\mathsf{F}\in\mathbb{K}[x]^{1\times m}$ be a polynomial matrix such that $\mathsf{rdeg}(\mathsf{F}) < n$, $v\in\mathbb{K}[x]$ be a polynomial such that $\mathsf{deg}(v) < n$ and $s=[0,\dots,0,d]\in\mathbb{Z}^{m+1}$. We define $\mathsf{G} = \begin{bmatrix}\mathsf{F} & -v\end{bmatrix}\in\mathbb{K}[x]^{1\times(m+1)}$ and $\mathsf{Q}\in\mathbb{K}[x]^{(m+1)\times(m+1)}$ such that $\mathsf{Q} = \mathsf{ApproximantBasis}(\mathsf{G},n,s)$. Then, Q corresponds to the matrix $\begin{bmatrix}\mathsf{P} & \mathsf{s}\\ 0 & \mu\end{bmatrix}$, where*

- *$\mathsf{P}\in\mathbb{K}[x]^{m\times m}$ is the $[0,\dots,0]$-Popov basis of $\{\mathsf{p}\in\mathbb{K}^{m\times 1} \mid \mathsf{F}\cdot\mathsf{p} = 0 \bmod x^n\}$,*
- *$\mathsf{s}\in\mathbb{K}[x]^{m\times 1}$ and $\mu\in\mathbb{K}[x]$ of minimal degree such that $\mathsf{F}\cdot\mathsf{s} = v(x)\cdot\mu(x) \bmod x^n$.*

Proof. We write Q as $\begin{bmatrix}\mathsf{P} & \mathsf{s}\\ \mathsf{u} & \mu\end{bmatrix}$, where $\mathsf{P}\in\mathbb{K}[x]^{m\times m}$, $\mathsf{s}\in\mathbb{K}[x]^{m\times 1}$, $\mathsf{u} = [\mathsf{u}_1,\dots,\mathsf{u}_m]\in\mathbb{K}[x]^{1\times m}$ and $\mu\in\mathbb{K}[x]$.
Since Q is $\mathsf{ApproximantBasis}(\mathsf{G},n,s)$, it is a $[0,\dots,0,d]$-Popov basis of $\{\mathsf{p}\in\mathbb{K}^{m+1\times 1} \mid \mathsf{G}\cdot\mathsf{p} = 0 \bmod x^n\}$. By Definition 21, we have $[\mathsf{deg}(\mathsf{u}_1)+d,\dots,\mathsf{deg}(\mathsf{u}_m)+d] < \mathsf{cdeg}_0(\mathsf{P})$. Since $\mathsf{cdeg}_s(\mathsf{Q}) < [d,\dots,d]$, we can deduce that $\mathsf{cdeg}_0(\mathsf{P}) < [d,\dots,d]$ and thus $[\mathsf{deg}(\mathsf{u}_1),\dots,\mathsf{deg}(\mathsf{u}_m)] < [0,\dots,0]$. Therefore, we get that u_i is the zero polynomial for $i=1,\dots,m$.

Moreover, we have $\mathsf{G}\cdot\mathsf{Q} = 0 \bmod x^n$, which means that $\begin{bmatrix}\mathsf{F} & -v(x)\end{bmatrix}\cdot\begin{bmatrix}\mathsf{P} & \mathsf{s}\\ 0 & \mu\end{bmatrix} = 0 \bmod x^n$. This means that $\begin{bmatrix}\mathsf{F}\cdot\mathsf{P} & \mathsf{F}\cdot\mathsf{s} - v\cdot\mu\end{bmatrix} = 0 \bmod x^n$. Therefore, we get that $\mathsf{F}\cdot\mathsf{P} = 0 \bmod x^n$ and $\mathsf{F}\cdot\mathsf{s} = v\cdot\mu \bmod x^n$.
Finally, we add that P is the $[0,\dots,0]$-Popov basis of $\{\mathsf{p}\in\mathbb{K}^{m\times 1} \mid \mathsf{F}\cdot\mathsf{p} = 0 \bmod x^n\}$ since it is the submatrix of a matrix in $[0,\dots,0,d]$-Popov form. ◀

6 Conclusion

We refer to Section 1.5 for a summary of the results presented in this report and perspectives for future work.

References

[1] A. Bostan, C.-P. Jeannerod, C. Mouilleron and É. Schost. 'On Matrices With Displacement Structure: Generalized Operators and Faster Algorithms'. In: *SIAM Journal on Matrix Analysis and Applications* 38.3 (2017), pp. 733–775. DOI: 10.1137/16M1062855. eprint: https://doi.org/10.1137/16M1062855. URL: https://doi.org/10.1137/16M1062855.

[2] Alin Bostan, Frédéric Chyzak, Marc Giusti, Romain Lebreton, Grégoire Lecerf, Bruno Salvy and Éric Schost. *Algorithmes Efficaces en Calcul Formel*. French. 686 pages. Imprimé par CreateSpace. Aussi disponible en version électronique. Palaiseau: Frédéric Chyzak (auto-édit.), Sept. 2017. ISBN: 979-10-699-0947-2. URL: https://hal.archives-ouvertes.fr/AECF/.

[3] Alin Bostan, Claude-Pierre Jeannerod and Éric Schost. 'Solving structured linear systems with large displacement rank'. In: *Theoretical Computer Science* 407.1 (2008), pp. 155–181. ISSN: 0304-3975. DOI: https://doi.org/10.1016/j.tcs.2008.05.014. URL: https://www.sciencedirect.com/science/article/pii/S0304397508003940.

[4] Alin Bostan and Ryuhei Mori. 'A Simple and Fast Algorithm for Computing the N-th Term of a Linearly Recurrent Sequence'. In: *2021 Symposium on Simplicity in Algorithms (SOSA)*, pp. 118–132. DOI: 10.1137/1.9781611976496.14. eprint: https://epubs.siam.org/doi/pdf/10.1137/1.9781611976496.14. URL: https://epubs.siam.org/doi/abs/10.1137/1.9781611976496.14.

[5] Richard P Brent, Fred G Gustavson and David YY Yun. 'Fast solution of Toeplitz systems of equations and computation of Padé approximants'. In: *Journal of Algorithms* 1.3 (1980), pp. 259–295.

[6] Peter Bürgisser, Michael Clausen and Mohammad Amin Shokrollahi. *Algebraic complexity theory*. Vol. 315. Grundlehren der mathematischen Wissenschaften. Springer, 1997. DOI: 10.1007/978-3-662-03338-8.

[7] Muhammad F. I. Chowdhury, Claude-Pierre Jeannerod, Vincent Neiger, Éric Schost and Gilles Villard. 'Faster Algorithms for Multivariate Interpolation With Multiplicities and Simultaneous Polynomial Approximations'. In: *IEEE Transactions on Information Theory* 61.5 (2015), pp. 2370–2387. DOI: 10.1109/TIT.2015.2416068.

[8] F.R. Gantmacher. *The Theory of Matrices*. Chelsea Publishing Company, 1980. URL: https://books.google.fr/books?id=ebMuywEACAAJ.

[9] FR Gantmacher. *The Theory of Matrices (Chelsea, New York)*. 1959, p. 16.

[10] Joachim von zur Gathen and Jürgen Gerhard. *Modern Computer Algebra*. 3rd ed. Cambridge University Press, 2013.

[11] Ch. Hermite. 'Sur la généralisation des fractions continues algébriques'. In: *Annali di Matematica Pura ed Applicata (1867-1897)* 21.1 (Jan. 1893), pp. 289–308. ISSN: 0373-3114. DOI: 10.1007/BF02420446. URL: https://doi.org/10.1007/BF02420446.

[12] Michael Heymann. *Structure and realization problems in the theory of dynamical systems*. C.LS.M. Courses and Lectures 204. Vienna: Springer-Verlag, 1975.

[13] Jin-Jen Hsue and Andrew E. Yagle. 'Fast algorithms for solving Toeplitz systems of equations using number-theoretic transforms'. In: *Signal Processing* 44.1 (1995), pp. 89–101. ISSN: 0165-1684. DOI: https://doi.org/10.1016/0165-1684(95)00017-8. URL: https://www.sciencedirect.com/science/article/pii/0165168495000178.

[14] Claude-Pierre Jeannerod, Vincent Neiger and Gilles Villard. 'Fast computation of approximant bases in canonical form'. In: *Journal of Symbolic Computation* 98 (2020). Special Issue on Symbolic and Algebraic Computation: ISSAC 2017, pp. 192–224. ISSN: 0747-7171. DOI: https://doi.org/10.1016/j.jsc.2019.07.011. URL: https://www.sciencedirect.com/science/article/pii/S0747717119300768.

[15] T. Kailath. *Linear Systems*. Information and System Sciences Series. Prentice-Hall, 1980. URL: https://books.google.fr/books?id=ggYqAQAAMAAJ.

[16] Thomas Kailath, Sun-Yuan Kung and Martin Morf. 'Displacement ranks of matrices and linear equations'. In: *Journal of Mathematical Analysis and Applications* 68.2 (1979), pp. 395–407. ISSN: 0022-247X. DOI: https://doi.org/10.1016/0022-247X(79)90124-0. URL: https://www.sciencedirect.com/science/article/pii/0022247X79901240.

[17] Erich Kaltofen. 'Asymptotically fast solution of Toeplitz-like singular linear systems'. In: *Proceedings of the International Symposium on Symbolic and Algebraic Computation*. ISSAC '94. Oxford, United Kingdom: Association for Computing Machinery, 1994, pp. 297–304. ISBN: 0897916387. DOI: 10.1145/190347.190431. URL: https://doi.org/10.1145/190347.190431.

[18] C. C. MacDutfee. *The Theory of Matrices*. Berlin: Springer, 1933.

[19] M. Morf. 'Doubling algorithms for Toeplitz and related equations'. In: *ICASSP '80. IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 5. 1980, pp. 954–959. DOI: 10.1109/ICASSP.1980.1171074.

[20] Vincent Neiger. 'Fast Computation of Shifted Popov Forms of Polynomial Matrices via Systems of Modular Polynomial Equations'. In: *Proceedings of the 2016 ACM International Symposium on Symbolic and Algebraic Computation*. ISSAC '16. Waterloo, ON, Canada: Association for Computing Machinery, 2016, pp. 365–372. ISBN: 9781450343800. DOI: 10.1145/2930889.2930936. URL: https://doi.org/10.1145/2930889.2930936.

[21] Vincent Neiger, Clément Pernet and Gilles Villard. 'Computing Krylov iterates in the time of matrix multiplication'. In: *Proceedings of the 2024 International Symposium on Symbolic and Algebraic Computation*. ISSAC '24. Raleigh, NC, USA: Association for Computing Machinery, 2024, pp. 419–428. ISBN: 9798400706967. DOI: 10.1145/3666000.3669715. URL: https://doi.org/10.1145/3666000.3669715.

[22] Vincent Neiger and Thi Xuan Vu. 'Computing Canonical Bases of Modules of Univariate Relations'. In: *CoRR* abs/1705.10649 (2017). arXiv: 1705.10649. URL: http://arxiv.org/abs/1705.10649.

[23] H. Padé. 'Sur la généralisation des fractions continues algébriques'. fr. In: *Journal de Mathématiques Pures et Appliquées* 4e série, 10 (1894), pp. 291–329. URL: https://www.numdam.org/item/JMPA_1894_4_10__291_0/.

[24] Victor Y. Pan. *Structured matrices and polynomials: unified superfast algorithms*. Berlin, Heidelberg: Springer-Verlag, 2001. ISBN: 0817642404.

[25] Johan Rosenkilde and Arne Storjohann. 'Algorithms for simultaneous Hermite–Padé approximations'. In: *Journal of Symbolic Computation* 102 (2021), pp. 279–303. ISSN: 0747-7171. DOI: https://doi.org/10.1016/j.jsc.2019.07.026. URL: https://www.sciencedirect.com/science/article/pii/S0747717119301300.

[26] Arne Storjohann. 'Algorithms for matrix canonical forms'. In: 2000. URL: https://api.semanticscholar.org/CorpusID:118041830.

[27] Volker Strassen. 'Vermeidung von Divisionen.' ger. In: *Journal für die reine und angewandte Mathematik* 264 (1973), pp. 184–202. URL: http://eudml.org/doc/151394.

[28] William F. Trench. 'An Algorithm for the Inversion of Finite Toeplitz Matrices'. In: *Journal of the Society for Industrial and Applied Mathematics* 12.3 (1964), pp. 515–522. DOI: 10.1137/0112045. eprint: https://doi.org/10.1137/0112045. URL: https://doi.org/10.1137/0112045.

[29] Virginia Vassilevska Williams, Yinzhan Xu, Zixuan Xu and Renfei Zhou. 'New Bounds for Matrix Multiplication: from Alpha to Omega'. In: *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 3792–3835. DOI: 10.1137/1.9781611977912.134. eprint: https://epubs.siam.org/doi/pdf/10.1137/1.9781611977912.134. URL: https://epubs.siam.org/doi/abs/10.1137/1.9781611977912.134.

[30] W.A. Wolovich. *Linear Multivariable Systems*. Applied Mathematical Sciences. Springer New York, 1974. ISBN: 9780387901015. URL: https://books.google.fr/books?id=mhJuzwEACAAJ.

A Column reduced matrices and division

Our previous discussions in Section 2.2 about row reduced polynomial matrices have counterparts, which are column reduced polynomial matrices. We define the column reduced polynomial matrices and the right division of polynomial matrices by column reduced polynomial matrices in this appendix.

▶ **Definition 48.** *Let* $\mathsf{M} = [M_{ij}]_{i,j=1,\ldots,n} \in \mathbb{K}[x]^{n\times n}$ *be a polynomial matrix,* $s = [s_1, \ldots, s_m]$ *be a list of non-negative integers and* $\mathsf{rdeg}_s(M) = [d_1, \ldots, d_n]$*. We define* the row leading coefficients matrix of M with respect to s *as the matrix* $[\mathsf{coeff}(M_{ij}, d_i - s_j)]_{i,j=1,\ldots,n} \in \mathbb{K}^{n\times m}$*. We denote this matrix by* $\mathsf{clm}_s(\mathsf{M})$*.*

▶ **Definition 49.** *Let* $s = [s_1, \ldots, s_n]$ *be a list of non-negative integers. A nonsingular matrix* $\mathsf{M} = [M_{ij}]_{i,j=1,\ldots,n} \in \mathbb{K}[x]^{n\times n}$*, such that* $\mathsf{cdeg}_s(M) = [d_1, \ldots, d_n]$ *is called* s-column reduced *if the matrix* clm_s *is invertible.*
We say that M *is* column reduced *if* $s = [0, \ldots, 0]$*.*

For instance, the matrix $\mathsf{M} = \begin{bmatrix} x & x^2 \\ x & x^3 \end{bmatrix}$ is column reduced, while the matrix $\mathsf{M} = \begin{bmatrix} x & x \\ x & x^3 \end{bmatrix}$ is not column reduced.

▶ **Theorem 50** (Theorem 6.3.15 in [15])**.** *For any column reduced* $\mathsf{M} = [M_{ij}]_{i,j=1,\ldots,m} \in \mathbb{K}[x]^{m\times m}$*, and any* $\mathsf{F} = [f_{ij}]_{j=1,\ldots,m}^{i=1,\ldots,n} \in \mathbb{K}[x]^{n\times m}$*, there exist unique matrices* $\mathsf{Q} = [q_{ij}]_{j=1,\ldots,m}^{i=1,\ldots,n} \in \mathbb{K}[x]^{n\times m}$ *and* $\mathsf{R} = [r_{ij}]_{j=1,\ldots,m}^{i=1,\ldots,n} \in \mathbb{K}[x]^{n\times m}$ *such that* $\mathsf{cdeg}(\mathsf{R}) < \mathsf{cdeg}(\mathsf{M})$ *and*

$$\mathsf{F} = \mathsf{Q} \cdot \mathsf{M} + \mathsf{R},$$

where R *is the right remainder matrix and* Q *is the right quotient matrix.*
We denote by F rquo M *the matrix quotient* Q *and by* F rrem M *the matrix remainder* R *for the right division of the polynomial matrix* F *by the polynomial matrix* M*.*

▶ **Lemma 51.** *Let* $\mathsf{M} = [M_{ij}] \in \mathbb{K}[x]^{n\times n}$ *a column reduced polynomial matrix such that* $\mathsf{rdeg}(\mathsf{M}) = [d_1, \ldots, d_n]$ *where* d_i *is a positive integer for* $i = 1, \ldots, n$*. Let* $\mathsf{F} = [f_{ij}] \in \mathbb{K}[x]^{n\times m}$*, such that* $\mathsf{rdeg}(\mathsf{F}) < [d_1 + k, \ldots, d_n + k]$ *where* k *is a positive integer. Let* $\mathsf{Q} = [q_{ij}]_{j=1,\ldots,m}^{i=1,\ldots,n} \in \mathbb{K}[x]^{n\times m}$ *be* F rquo M*. Then, we have* $\mathsf{deg}(\mathsf{Q}) < k$*.*

Proof. We start this proof by proving the following property: $\mathsf{clm}_0(\mathsf{M}) = \mathsf{rlm}_{-\mathsf{cdeg}(\mathsf{M})}(\mathsf{M})$.

- We have on one side, $\forall i, j \in \{1, \ldots, n\}$:

$$\begin{aligned}
\mathsf{elem}(\mathsf{clm}_0(\mathsf{M}), i, j) &= \mathsf{elem}(\mathsf{rlm}_0(\mathsf{M}^\mathsf{T})^\mathsf{T}, i, j) \\
&= \mathsf{elem}(\mathsf{rlm}_0(\mathsf{M}^\mathsf{T}), j, i) \\
&= \mathsf{coeff}(M_{ij}, \mathsf{elem}(\mathsf{rdeg}(\mathsf{M}^\mathsf{T}), j)) \text{ (by Definition 15)} \\
&= \mathsf{coeff}(M_{ij}, \mathsf{max}_{k=1,\ldots,n}(\mathsf{deg}(M_{jk}^\mathsf{T}) + 0)) \\
&= \mathsf{coeff}(M_{ij}, \mathsf{max}_{k=1,\ldots,n}(\mathsf{deg}(M_{kj}) + 0)) \\
&= \mathsf{coeff}(M_{ij}, \mathsf{elem}(\mathsf{cdeg}(\mathsf{M}), j)) \text{ (by Definition 48)}
\end{aligned}$$

- On the other side, we have $\forall i, j \in \{1, \ldots, n\}$:

$$\begin{aligned}
\mathsf{elem}(\mathsf{rlm}_{-\mathsf{cdeg}(\mathsf{M})}(\mathsf{M}), i, j) &= \mathsf{coeff}(\mathsf{elem}(\mathsf{rdeg}_{-\mathsf{cdeg}(\mathsf{M})}(\mathsf{M}), i) + \mathsf{elem}(\mathsf{cdeg}(\mathsf{M}), j), M_{ij}) \text{ (by Definition 48)} \\
&= \mathsf{coeff}(\mathsf{max}_{k=1,\ldots,n}(\mathsf{deg}(M_{ik}) - \mathsf{elem}(\mathsf{cdeg}(\mathsf{M}), k)) + \mathsf{elem}(\mathsf{cdeg}(\mathsf{M}), j), M_{ij}) \\
&= \mathsf{coeff}(\mathsf{max}_{k=1,\ldots,n}(\mathsf{deg}(M_{ik}) - \mathsf{max}_{\ell=1,\ldots,n}(\mathsf{deg}(M_{\ell k}) + 0)) + \mathsf{elem}(\mathsf{cdeg}(\mathsf{M}), j), M_{ij}) \\
&= \mathsf{coeff}(\mathsf{elem}(\mathsf{cdeg}(\mathsf{M}), j), M_{ij}) \text{ (since } \mathsf{deg}(M_{ik}) \leq \mathsf{max}_{\ell=1,\ldots,n}(\mathsf{deg}(M_{\ell k})) \text{ for all } i, k)
\end{aligned}$$

Thus, we conclude that

$$\mathsf{clm}_0(\mathsf{M}) = \mathsf{rlm}_{-\mathsf{cdeg}(\mathsf{M})}(\mathsf{M}). \tag{15}$$

Now, we show that $\mathsf{rdeg}_{-\mathsf{cdeg}(\mathsf{M})}(\mathsf{M}) = [0, \ldots, 0]$. Since we shift the degrees of each row of M by its leading coefficient degree, we have that all degrees after the shifts are either 0 or negative. We consider that an entry with negative degree is equivalent to a zero entry and thus has a degree of $-\infty$. Therefore, the only possible values for the degrees of the entries of $\mathsf{rdeg}_{-\mathsf{cdeg}(\mathsf{M})}(\mathsf{M})$ are 0 or $-\infty$. Moreover, if we have $-\infty$ in the degree of an entry, then all degrees of that row in the shifted matrix are $-\infty$, which means that the row is entirely zero. If we assume this, then $\mathsf{rlm}_{-\mathsf{cdeg}(\mathsf{M})}(\mathsf{M})$ is a matrix with a row of zeros, which means that the matrix is not invertible. This a contradiction since we know that $\mathsf{clm}_0(\mathsf{M}) = \mathsf{rlm}_{-\mathsf{cdeg}(\mathsf{M})}(\mathsf{M})$ by (15) and $\mathsf{clm}_0(\mathsf{M})$ is invertible by Definition 48. Thus, we conclude that

$$\mathsf{rdeg}_{-\mathsf{cdeg}(\mathsf{M})}(\mathsf{M}) = [0, \ldots, 0].$$

Let $\mathsf{R} = \mathsf{rem}\ (\mathsf{F}, \mathsf{M})$. By Theorem 50, we have $\mathsf{F} = \mathsf{Q} \cdot \mathsf{M} + \mathsf{R}$ and $\mathsf{cdeg}(\mathsf{R}) < [d_1, \ldots, d_n]$. Since $\mathsf{F} - \mathsf{R} = \mathsf{Q} \cdot \mathsf{M}$, we have $\mathsf{cdeg}(\mathsf{F} - \mathsf{R}) = \mathsf{cdeg}(\mathsf{Q} \cdot \mathsf{M})$. We use the fact that $\mathsf{cdeg}(\mathsf{F}) < [d_1 + k, \ldots, d_n + k]$ and $\mathsf{cdeg}(\mathsf{R}) < [d_1, \ldots, d_n]$ to deduce $\mathsf{cdeg}(\mathsf{F} - \mathsf{R}) < [d_1 + k, \ldots, d_n + k]$ and thus $\mathsf{cdeg}(\mathsf{Q} \cdot \mathsf{M}) < [d_1 + k, \ldots, d_n + k]$. By the same argument as above, we have

$$\mathsf{rdeg}_{-\mathsf{cdeg}(\mathsf{M})}(\mathsf{Q} \cdot \mathsf{M}) < [k, \ldots, k].$$

Since M is $-\mathsf{cdeg}(\mathsf{M})$-row reduced (as it is column reduced) and $\mathsf{rdeg}_{-\mathsf{cdeg}(\mathsf{M})}(\mathsf{M}) = [0, \ldots, 0]$, we can apply the counterpart of the predictable degree property (see Theorem 6.3.13 in [15]) on each row Q_{i*} of Q for $i = 1, \ldots, n$ to get:

$$\begin{aligned}
\mathsf{rdeg}_{-\mathsf{cdeg}(\mathsf{M})}(\mathsf{M} \cdot Q_{i*}) &= \mathsf{max}_{j=1,\ldots,n}(\mathsf{deg}(Q_{ij}) + \mathsf{rdeg}_{-\mathsf{cdeg}(\mathsf{M})}(M_{i*})) \\
&= \mathsf{max}_{j=1,\ldots,n}(\mathsf{deg}(Q_{ij}) + 0) \\
&= \mathsf{rdeg}(Q_{i*}).
\end{aligned}$$

Thus, we have $\mathsf{rdeg}(Q_{i*}) < k$ for $i = 1, \ldots, n$, which means that $\mathsf{rdeg}(\mathsf{Q}) < [k, \ldots, k]$.
Finally, we conclude with $\mathsf{deg}(\mathsf{Q}) = \mathsf{max}(\mathsf{rdeg}(\mathsf{Q})) < k$. ◀

B No loss of generality

As stated previously, in Section 4, we can assume without loss of generality that the matrix M is a modulus polynomial matrix (see Definition 22). We show in what follows how to reduce the problem ApproxMatrixMod given a modulus that is a regular nonsingular matrix M to the problem ApproxMatrixMod given a modulus that is a modulus polynomial matrix, while ensuring the degree bounds via matrix division with remainder (see Theorem 17).
We have previously defined 0-popov matrices in Definition 21. For this section, we will use the counterpart definition, such that for $s = [s_1, \ldots, s_n]$ a list of non-negative integers, a polynomial matrix $\mathsf{M} \in \mathbb{K}[x]^{n\times n}$ is in *s-Popov form* if $\mathsf{rlm}_0(\mathsf{M}) = \mathbb{I}_n$ and $\mathsf{clm}_s(\mathsf{M})$ is a unit lower triangular matrix (lower triangular matrix with all diagonal entries equal to 1). We say that M is in *Popov form* if $s = [0, \ldots, 0]$.

▶ **Lemma 52.** *Given a nonsingular $\mathsf{M} \in \mathbb{K}[x]^{n\times n}$, there exists a unimodular matrix $\mathsf{U} \in \mathbb{K}[x]^{n\times n}$ such that*
- *$\mathsf{M}\cdot\mathsf{U}$ is a modulus polynomial matrix,*
- *Given $\mathsf{F} \in \mathbb{K}[x]^{n\times m}$ with $\mathsf{rdeg}(\mathsf{F}) < \mathsf{rdeg}(\mathsf{M})$, there exists $\mathsf{G} \in \mathbb{K}[x]^{n\times m}$ with $\mathsf{rdeg}(\mathsf{G}) < \mathsf{rdeg}(\mathsf{M})$ where $\mathsf{G} = \mathsf{F} \mathsf{\ rem\ } \mathsf{M}$ and the set of solutions of $\mathsf{ApproxMatrixMod}(\mathsf{M}, \mathsf{F}, \nu)$ is the same as the set of solutions $\mathsf{ApproxMatrixMod}(\mathsf{M}\cdot\mathsf{U}, \mathsf{G}, \nu)$.*

Proof. We prove each point of the lemma separately.
- Since M is a nonsingular matrix, we can transform this matrix to a Popov form by multiplying it on the right by a unimodular matrix U. The construction of this unimodular matrix is detailed in pages 484-486 of [15]. Therefore, we have that

  $$\forall \mathsf{M} \in \mathbb{K}[x]^{n\times n} \text{ nonsingular, } \exists \mathsf{U} \text{ unimodular s.t. } \mathsf{M}\cdot\mathsf{U} \text{ is 0-Popov.}$$

  Moreover, we show that a 0-Popov matrix is a modulus polynomial matrix.
  Indeed, we have that $\mathsf{rlm}_0(\mathsf{M}) = \mathbb{I}_n$.
  The diagonal of 1's in the matrix $\mathsf{rlm}_0(\mathsf{M})$ means that the polynomials with the leading degrees of M are the ones on the diagonal and that these polynomials are monic.
  Since the rest of the entries in the matrix $\mathsf{rlm}_0(\mathsf{M})$ are 0, we have that all the other polynomials in the matrix M are of degree strictly less than the leading degrees of the polynomial on the diagonal of its corresponding row.
  Therefore, we have that M is a modulus polynomial matrix.

- Let U be the unimodular matrix such that $\mathsf{M}\cdot\mathsf{U}$ is a modulus polynomial matrix. Consider the sets $\{\mathsf{g} \in \mathbb{K}^{m\times 1} \mid \mathsf{F}\cdot\mathsf{g} = 0 \bmod \mathsf{M}\}$ and $\{\mathsf{h} \in \mathbb{K}^{m\times 1} \mid \mathsf{G}\cdot\mathsf{h} = 0 \bmod \mathsf{M}\cdot\mathsf{U}\}$.
  The latter corresponds to the set of solutions to the equation $\mathsf{F} \mathsf{\ rem\ } (\mathsf{M}\cdot\mathsf{U})\cdot\mathsf{h} = 0 \bmod \mathsf{M}$, which is equivalent to the set of solutions to the equation $\mathsf{F}\cdot\mathsf{h} = 0 \bmod (\mathsf{M}\cdot\mathsf{U})$.

  The multiplication by the unimodular matrix U does not change the basis of solutions $\{\mathsf{g} \in \mathbb{K}^{m\times 1} \mid \mathsf{F}\cdot\mathsf{g} = 0 \bmod \mathsf{M}\}$, i.e., $0 \bmod \mathsf{M}$ and $0 \bmod (\mathsf{M}\cdot\mathsf{U})$ generate the same vector space.
  The columns of $\mathsf{M}\cdot\mathsf{U}$ are linear combinations of the columns of M, and thus the solutions to the equation $\mathsf{F}\cdot\mathsf{g} = 0 \bmod \mathsf{M}$ are also solutions to the equation $\mathsf{F}\cdot\mathsf{h} = 0 \bmod (\mathsf{M}\cdot\mathsf{U})$.
  Moreover, since U is unimodular, it is invertible and thus we can write $\mathsf{M} = \mathsf{M}\cdot\mathsf{U}\cdot\mathsf{U}^{-1}$. Similarly, the columns of M are linear combinations of the columns of $\mathsf{M}\cdot\mathsf{U}$, and thus the solutions to the equation $\mathsf{F}\cdot\mathsf{h} = 0 \bmod (\mathsf{M}\cdot\mathsf{U})$ are also solutions to the equation $\mathsf{F}\cdot\mathsf{g} = 0 \bmod \mathsf{M}$.

  Therefore, we have that $\{\mathsf{g} \in \mathbb{K}^{m\times 1} \mid \mathsf{F}\cdot\mathsf{g} = 0 \bmod \mathsf{M}\} = \{\mathsf{h} \in \mathbb{K}^{m\times 1} \mid \mathsf{G}\cdot\mathsf{h} = 0 \bmod (\mathsf{M}\cdot\mathsf{U})\}$.

◀

C Matrix modular approximation: proof of Theorem 38

We restate the theorem and provide its complete proof.

▶ **Theorem 38.** *Let* $\mathsf{M} = [M_{ij}]_{i,j=1,\dots,n} \in \mathbb{K}[x]^{n\times n}$*, where* M *is a modulus polynomial matrix and* $\mathsf{rdeg}(M) = [d_1,\dots,d_n]$*,* $\mathsf{F} = [f_{ij}]_{j=1,\dots,m}^{i=1,\dots,n} \in \mathbb{K}[x]^{n\times m}$ *such that* $\mathsf{rdeg}(\mathsf{F}) < [d_1,\dots,d_n]$ *and* $\nu = \{\nu_1,\dots,\nu_m\}$ *a set of positive integers. Let* $C = \sum_{j=1}^{m}\nu_j$ *and* $D = \sum_{i=1}^{n} d_i$*. We suppose that* $n \in O(m)$ *and* $C \in O(D)$*.*
The problem $\mathsf{ApproxMatrixMod}(\mathsf{M},\mathsf{F},\nu)$ *can be solved in* $\tilde{O}(\mathsf{max}(m,n)^{\omega-1}\cdot D)$ *by Algorithm 4.*

Proof. The approach is to reduce the problem $\mathsf{ApproxMatrixMod}(\mathsf{M},\mathsf{F},\nu)$ to the problem of solving a linear system with a quasi-Toeplitz structure. We can use known Algorithm 5 to solve such a linear system in the desired complexity.
Formally, we reduce $\mathsf{ApproxMatrixMod}(\mathsf{M},\mathsf{F},\nu)$ to a problem of type $\mathsf{LinearSystem}[\mathbb{Z}_{D,0},\mathbb{Z}_{C,0}^{\mathsf{T}},O(\mathsf{max}(m,n))]$, as presented in Algorithm 4.
We define the matrix $\mathsf{A}\in\mathbb{K}^{D\times C}$ as the matrix $\mathsf{A} = \begin{bmatrix}\mathsf{A}_1 & \mathsf{A}_2 & \cdots & \mathsf{A}_m\end{bmatrix}$, where $\mathsf{A}_j \in \mathbb{K}^{D\times\nu_j}$, for $j = 1,\dots,m$, corresponds to the matrix

$$\mathsf{A}_j = \begin{bmatrix}\overline{\mathsf{F}}_{*j} & \mathbb{X}(\mathsf{M})\cdot\overline{\mathsf{F}}_{*j} & \cdots & \mathbb{X}(\mathsf{M})^{\nu_j-1}\cdot\overline{\mathsf{F}}_{*j}\end{bmatrix}, \tag{16}$$

such that $\overline{\mathsf{F}}\in\mathbb{K}^{D\times m}$ is the matrix where we replace each polynomial of F by the vector of its coefficients.

▶ Notation 53. For ease of notation, we refer to the matrix $\mathbb{X}(\mathsf{M})$ as $\mathbb{X}$ for the rest of this proof.

Lemma 54 establishes that we can construct a ϕ_+-generator of length $O(\mathsf{max}(m,n))$ for the matrix A.

▶ **Lemma 54.** *Let* A *be the matrix defined in* (16)*. We construct* ϕ_+*-generators of length* $O(\mathsf{max}(m,n))$ *for* A *in* $\tilde{O}(n^{\omega-1}\cdot D + m\cdot n^{\omega-2}\cdot D + n^{\omega-1}\cdot C)$ *operations in* $\mathbb{K}$*.*

Proof. We know by Definition 8 that, if A is a quasi-Toeplitz matrix, then $\phi_+(\mathsf{A}) = A - \mathbb{Z}_{D,0}\cdot A\cdot\mathbb{Z}_{C,0}^{\mathsf{T}}$. We use the fact that $\mathbb{Z}_{D,0} = \mathbb{X} - \mathbb{Y} + \mathsf{N}\cdot\mathsf{E} + \delta$ where $\mathbb{X}$ is the matrix defined in (6), $\mathbb{Y}$ is the matrix defined as

$$\mathbb{Y} = \begin{bmatrix} 0 & \mathsf{C}_{12} & \cdots & \mathsf{C}_{1m}\\ \mathsf{C}_{21} & 0 & \cdots & \mathsf{C}_{2m}\\ \vdots & \vdots & \ddots & \vdots\\ \mathsf{C}_{n1} & \mathsf{C}_{n2} & \cdots & 0\end{bmatrix},$$

where $\delta\in\mathbb{K}^{D\times D}$ as the matrix $[\delta_{ab}]_{a,b=1,\dots,D}$ where $\delta_{ab} = 1$ if $a = \sum_{k=1}^{i} d_k + 1$ and $b = \sum_{k=1}^{i} d_k$ for $i = 1,\dots,n$, and $\delta_{ab} = 0$ otherwise, for $a,b = 1,\dots,D$, $\mathsf{N}\in\mathbb{K}^{D\times n}$ and $\mathsf{E}\in\mathbb{K}^{n\times D}$ are block diagonal matrices such that $\mathsf{M} = \mathsf{diag}(\mathsf{m}_1,\dots,\mathsf{m}_n)$ with $\mathsf{m}_i\in\mathbb{K}^{d_i\times 1}$; the vector of coefficients of the polynomial M_{ii}, and $\mathsf{E} = \mathsf{diag}(\mathsf{e}_1,\dots,\mathsf{e}_n)$ with $\mathsf{e}_i = [0,\dots,0,1]\in\mathbb{K}^{1\times d_i}$ for $i = 1,\dots,n$.
Consequently, we can rewrite $\phi_+(\mathsf{A})$ as follows:

$$\phi_+(\mathsf{A}) = \mathsf{A} - \mathbb{X}\cdot\mathsf{A}\cdot\mathbb{Z}_{C,0}^{\mathsf{T}} + \mathbb{Y}\cdot\mathsf{A}\cdot\mathbb{Z}_{C,0}^{\mathsf{T}} - \delta\cdot\mathsf{A}\cdot\mathbb{Z}_{C,0}^{\mathsf{T}} - \mathsf{N}\cdot\mathsf{E}\cdot\mathsf{A}\cdot\mathbb{Z}_{C,0}^{\mathsf{T}}. \tag{17}$$

We then compute the generators as follows:

- For $\mathsf{A} - \mathbb{X}\cdot\mathsf{A}\cdot\mathbb{Z}_{C,0}^{\mathsf{T}}$:
  We can write $\mathsf{A} - \mathbb{X}\cdot\mathsf{A}\cdot\mathbb{Z}_{C,0}^{\mathsf{T}}$ as $\begin{bmatrix}\mathsf{B}_1 & \mathsf{B}_2 & \cdots & \mathsf{B}_m\end{bmatrix}$ where $\mathsf{B}_j\in\mathbb{K}^{D\times\nu_j}$, for $j = 1,\dots,m$ is

$$\mathsf{B}_j = \begin{cases}\begin{bmatrix}\overline{\mathsf{F}}_{*j} & 0 & \cdots & 0\end{bmatrix} & \text{if } j = 1\\ \begin{bmatrix}\overline{\mathsf{F}}_{*j} - \mathbb{X}^{\nu_{j-1}}\cdot\overline{\mathsf{F}}_{*(j-1)} & 0 & \cdots & 0\end{bmatrix} & \text{if } j > 1\end{cases}$$

  We can write $\mathsf{B} = \mathsf{Y}\cdot\mathsf{Z}^{\mathsf{T}}$ where $\mathsf{Y}\in\mathbb{K}^{D\times m}$ is the matrix $\begin{bmatrix}\mathsf{Y}_j\end{bmatrix}_{j=1,\dots,m}$ where $\mathsf{Y}_j\in\mathbb{K}^{D\times 1}$ is defined as

$$\mathsf{Y}_j = \begin{cases}\begin{bmatrix}\overline{\mathsf{F}}_{*j}\end{bmatrix} & \text{if } j = 1\\ \begin{bmatrix}\overline{\mathsf{F}}_{*j} - \mathbb{X}^{\nu_{j-1}}\cdot\overline{\mathsf{F}}_{*(j-1)}\end{bmatrix} & \text{if } j > 1\end{cases}$$

and $\mathsf{Z} \in \mathbb{K}^{C \times m}$ is the matrix $\left[\mathsf{Z}_j\right]$ for $j = 1, \ldots, m$ where $\mathsf{Z}_j \in \mathbb{K}^{C \times 1}$ is defined as the vector of all zeros except for the first position and the $\sum_{k=1}^{j-1} \nu_k + 1$-th position for $j = 2, \ldots, m$, which is 1.
The computation of Y is bounded by the computation of $\mathbb{X}^{\nu_{j-1}} \cdot \overline{\mathsf{F}}_{*(j-1)}$ for $j = 2, \ldots, m$, which corresponds to the computation of the coefficients of the polynomial $x^{\nu_j} \cdot \mathsf{F}_{*j} \mathsf{\ rem\ } M$, and can be done in $\tilde{O}(n^{\omega-1} \cdot D + m \cdot n^{\omega-2} \cdot D)$ operations in $\mathbb{K}$, as proven in Lemma 31.

- For $\delta \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}}$:
We can write $\delta \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} = \mathsf{T} \cdot \mathsf{U}^{\mathsf{T}}$ where $\mathsf{T} \in \mathbb{K}^{D \times n}$ and $\mathsf{U} \in \mathbb{K}^{C \times n}$ are the matrices such that $\mathsf{T} = [\mathsf{T}_{ij}]_{i,j=1,\ldots,n}$ and $\mathsf{U} = [\mathsf{U}_{ji}]_{j=1,\ldots,m}^{i=1,\ldots,n}$, where $\mathsf{T}_{ij} \in \mathbb{K}^{d_i \times 1}$ and $\mathsf{U}_{ji} \in \mathbb{K}^{\nu_j \times 1}$ are defined as follows:

$$\mathsf{T}_{ij} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \text{ for } \begin{cases} j = 1 \\ i = 1, \ldots, n \end{cases} \quad \mathsf{T}_{ij} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \text{ for } \begin{cases} j = 2, \ldots, n \\ i = 1, \ldots, n \end{cases}$$

$$\mathsf{U}_{ji} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \text{ for } \begin{cases} j = 1 \\ i = 1, \ldots, n \end{cases} \quad \mathsf{U}_{ji} = \begin{bmatrix} 0 \\ \mathsf{coeff}(\mathsf{F}_{*j}, i, d_i - 1) \\ \mathsf{coeff}(x \cdot \mathsf{F}_{*j} \mathsf{\ rem\ } \mathsf{M}, i, d_i - 1) \\ \vdots \\ \mathsf{coeff}(x^{\nu_j - 2} \cdot \mathsf{F}_{*j} \mathsf{\ rem\ } \mathsf{M}, i, d_i - 1) \end{bmatrix} \text{ for } \begin{cases} j = 2, \ldots, m \\ i = 1 \end{cases}$$

$$\mathsf{U}_{ji} = \begin{bmatrix} \mathsf{coeff}(x^{\nu_j - 1} \cdot \mathsf{F}_{*j} \mathsf{\ rem\ } \mathsf{M}, i - 1, d_{i-1} - 1) \\ \mathsf{coeff}(\mathsf{F}_{*j}, i, d_i - 1) \\ \mathsf{coeff}(x \cdot \mathsf{F}_{*j} \mathsf{\ rem\ } \mathsf{M}, i, d_i - 1) \\ \vdots \\ \mathsf{coeff}(x^{\nu_j - 2} \cdot \mathsf{F}_{*j} \mathsf{\ rem\ } \mathsf{M}, i, d_i - 1) \end{bmatrix} \text{ for } \begin{cases} j = 2, \ldots, m \\ i = 2, \ldots, n \end{cases}$$

Note that $\mathsf{coeff}(x^{\nu_j - 1} \cdot \mathsf{F}_{*j} \mathsf{\ rem\ } \mathsf{M}, i, d_i - 1)$ corresponds to $\mathsf{elem}(\mathsf{F}_{*j} \cdot \mathbb{X}^{\nu_j - 1}, \sum_{k=1}^{i}(d_k) - 1)$.
The computation of U corresponds to the computation of the $d_i - 1$-th coefficient of the polynomial at row i and column j of the matrix $x^{k_j} \cdot \mathsf{F} \mathsf{\ rem\ } M$, for $i = 1, \ldots, n$, $k_j = 0, \ldots, \nu_j - 1$ and $j = 1, \ldots, m$ (i.e. the problem $\mathsf{LastCoeffModMat}(\mathsf{M}, \mathsf{F}, \nu)$, where $\nu = \{\nu_1, \ldots, \nu_m\}$) and thus can be done in $\tilde{O}(n^{\omega-1} \cdot D + m \cdot n^{\omega-2} \cdot D + n^{\omega-1} \cdot C)$ operations in $\mathbb{K}$ as shown in Theorem 36.

- For $\mathbb{Y} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}}$:
We define $\mathsf{R} \in \mathbb{K}^{D \times n}$ as follows

$$\mathsf{R} = \begin{bmatrix} \mathbf{0} & \mathsf{m}_{12} & \cdots & \mathsf{m}_{1m} \\ \mathsf{m}_{21} & \mathbf{0} & \cdots & \mathsf{m}_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \mathsf{m}_{n1} & \mathsf{m}_{n2} & \cdots & \mathbf{0} \end{bmatrix},$$

where $\mathsf{m}_{ij} \in \mathbb{K}^{1 \times \nu_j}$ is the vector of coefficients of the polynomial M_{ij} for $i, j = 1, \ldots, n$.
We have that $\mathbb{Y} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} = \mathsf{R} \cdot \mathsf{U}^{\mathsf{T}}$. The computation here is bounded by the cost of computing U, as in the previous case. Therefore, the total cost of computing $\mathbb{Y} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}}$ is bounded by $\tilde{O}(n^{\omega-1} \cdot D + m \cdot n^{\omega-2} \cdot D + n^{\omega-1} \cdot C)$ operations in $\mathbb{K}$ (see Theorem 36).

- For $\mathsf{N} \cdot \mathsf{E} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}}$:
Let $\mathsf{P} \in \mathbb{K}^{C \times n}$ such that $\mathsf{P} = \mathsf{A}^{\mathsf{T}} \cdot \mathsf{E}^{\mathsf{T}}$ and $\mathsf{Q} \in \mathbb{K}^{C \times n}$ such that $\mathsf{Q} = \mathbb{Z}_{C,0} \cdot \mathsf{P}$. We can write $\mathsf{N} \cdot \mathsf{E} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} = \mathsf{N} \cdot \mathsf{Q}^{\mathsf{T}}$.

In order to compute Q, we need to compute P first. We have $\mathsf{P} = [\mathsf{P}_{ji}]_{i=1,\ldots,n}^{j=1,\ldots,m}$ and $\mathsf{P}_{ji} \in \mathbb{K}^{\nu_j \times 1}$, such that

$$\mathsf{P}_{ji} = \begin{bmatrix} \mathsf{coeff}(\mathsf{F}_{*j}, i, d_i - 1) \\ \mathsf{coeff}(x \cdot \mathsf{F}_{*j} \text{ rem } \mathsf{M}, i, d_i - 1) \\ \vdots \\ \mathsf{coeff}(x^{\nu_j - 1} \cdot \mathsf{F}_{*j} \text{ rem } \mathsf{M}, i, d_i - 1) \end{bmatrix},$$

for $i = 1, \ldots, n$ and $j = 1, \ldots, m$.

By Theorem 36, we can compute P_{ji} in $\tilde{O}(n^{\omega-1} \cdot D + m \cdot n^{\omega-2} \cdot D + n^{\omega-1} \cdot C)$ operations in $\mathbb{K}$. The computation of Q is the cost of the multiplication of the matrix the matrix $\mathbb{Z}_{C,0}$ by P, which corresponds to a vector matrix multiplication and thus can be done in $\tilde{O}(n^{\omega-1} \cdot C)$ operations in $\mathbb{K}$.

Therefore, the total cost of computing $\mathsf{N} \cdot \mathsf{E} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}}$ is bounded by $\tilde{O}(n^{\omega-1} \cdot D + m \cdot n^{\omega-2} \cdot D + n^{\omega-1} \cdot C)$.

To summarize, we recall that $\phi_+(\mathsf{A}) = \mathsf{A} - \mathbb{Z}_{D,0} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} + \mathbb{Y} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} - \delta \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} - \mathsf{N} \cdot \mathsf{E} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}}$ and we now have:

- $\mathsf{A} - \mathbb{X} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} = \mathsf{Y} \cdot \mathsf{Z}^{\mathsf{T}}$, where $\mathsf{Y} \in \mathbb{K}^{D \times m}$ and $\mathsf{Z} \in \mathbb{K}^{C \times m}$, can be computed in $\tilde{O}(n^{\omega-1} \cdot D + m \cdot n^{\omega-2} \cdot D)$.
- $\delta \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} = \mathsf{T} \cdot \mathsf{U}^{\mathsf{T}}$, where $\mathsf{T} \in \mathbb{K}^{D \times n}$ and $\mathsf{U} \in \mathbb{K}^{C \times n}$, can be computed in $\tilde{O}(n^{\omega-1} \cdot D + m \cdot n^{\omega-2} \cdot D + n^{\omega-1} \cdot C)$.
- $\mathbb{Y} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} = \mathsf{R} \cdot \mathsf{U}^{\mathsf{T}}$, where $\mathsf{R} \in \mathbb{K}^{D \times n}$ and $\mathsf{U} \in \mathbb{K}^{C \times n}$, can be computed in $\tilde{O}(n^{\omega-1} \cdot D + m \cdot n^{\omega-2} \cdot D + n^{\omega-1} \cdot C)$.
- $\mathsf{N} \cdot \mathsf{E} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} = \mathsf{N} \cdot \mathsf{Q}^{\mathsf{T}}$, where $\mathsf{N} \in \mathbb{K}^{D \times n}$ and $\mathsf{Q} \in \mathbb{K}^{C \times n}$, can be computed in $\tilde{O}(n^{\omega-1} \cdot D + m \cdot n^{\omega-2} \cdot D + n^{\omega-1} \cdot C)$.

Therefore, we can compute $\phi_+(\mathsf{A}) = \mathsf{G} \cdot \mathsf{H}^{\mathsf{T}}$ where $\mathsf{G} = \begin{bmatrix} \mathsf{Y} & -\mathsf{T} & \mathsf{R} & -\mathsf{N} \end{bmatrix}$ and $\mathsf{H} = \begin{bmatrix} \mathsf{Z} & \mathsf{U} & \mathsf{U} & \mathsf{Q} \end{bmatrix}$ in $\tilde{O}(n^{\omega-1} \cdot D + m \cdot n^{\omega-2} \cdot D + n^{\omega-1} \cdot C)$ operations in $\mathbb{K}$. We have $\mathsf{G} \in \mathbb{K}^{D \times (m+3n)}$ and $\mathsf{H} \in \mathbb{K}^{C \times (m+3n)}$. Since the number of columns of G is $(m + 3n)$, we have a ϕ_+-generator of length $O((m + 3n) \in \mathsf{max}(m, n))$ for the matrix A, computed in $\tilde{O}(n^{\omega-1} \cdot D + m \cdot n^{\omega-2} \cdot D + n^{\omega-1} \cdot C) = \tilde{O}(\mathsf{max}(m, n)^{\omega-1} \cdot D)$ operations in $\mathbb{K}$, which concludes the proof of Lemma 54. ◀

After computing the ϕ_+-generator of length $O(\mathsf{max}(m, n))$ for the matrix A, we have formally reduced the problem $\mathsf{ApproxMatrixMod}(\mathsf{M}, \mathsf{F}, \nu)$ to the problem of solving a linear system of equations with a quasi-Toeplitz matrix A, i.e., $\mathsf{LinearSystem}[\mathbb{Z}_{D,0}, \mathbb{Z}_{C,0}^{\mathsf{T}}, \mathsf{max}(m, n)](\mathsf{G}, \mathsf{H})$, which can be solved in $\tilde{O}(\mathsf{max}(m, n)^{\omega-1} \cdot D)$ operations in $\mathbb{K}$ (see Theorem 12). Finally, we recover the solution g of the problem $\mathsf{ApproxMatrixMod}(\mathsf{M}, \mathsf{F}, \nu)$ from the solution u of the linear system $\mathsf{A} \cdot \mathsf{u} = \begin{bmatrix} 0 & \cdots & 0 \end{bmatrix}^{\mathsf{T}}$. We conclude that the problem $\mathsf{ApproxMatrixMod}(\mathsf{M}, \mathsf{F}, \nu)$ can be solved in $\tilde{O}(\mathsf{max}(m, n)^{\omega-1} \cdot D)$ operations in $\mathbb{K}$. ◀

D From quasi-Toeplitz structure to polynomial representation

In this section, we provide formal proofs for Lemmas 40 and 41, which show how to express the polynomial representation of a lower and upper triangular Toeplitz matrix, respectively. We restate the lemmas here for the sake of clarity.

▶ **Lemma 40.** *Let* $\mathsf{u}, \mathsf{v} \in \mathbb{K}^{n\times 1}$ *be two vectors and* $\mathsf{p} = \mathbb{L}(\mathsf{u}) \cdot \mathsf{v}$. *We have that* $p(x) = u(x) \cdot v(x) \bmod x^n$, *where* $u(x) = \sum_{i=0}^{n-1} u_i x^i$, $v(x) = \sum_{i=0}^{n-1} v_i x^i$ *and* $p(x) = \sum_{i=0}^{n-1} p_i x^i$ *are the polynomial representations of the vectors* u, v *and* p, *respectively.*

Proof. We have the vector $\mathsf{u} = [\mathsf{u}_0, \mathsf{u}_1, \ldots, \mathsf{u}_{n-1}]^\mathsf{T} \in \mathbb{K}^{n\times 1}$ and the vector $\mathsf{v} = [\mathsf{v}_0, \mathsf{v}_1, \ldots, \mathsf{v}_{n-1}]^\mathsf{T} \in \mathbb{K}^{n\times 1}$. We consider $\mathsf{p} = [\mathsf{p}_0, \mathsf{p}_1, \ldots, \mathsf{p}_{n-1}]^\mathsf{T} \in \mathbb{K}^{n\times 1}$ as the vector we want to compute, given by the following equation:

$$\mathsf{p} = \mathbb{L}(\mathsf{u}) \cdot \mathsf{v} \Leftrightarrow \begin{bmatrix} \mathsf{p}_0 \\ \mathsf{p}_1 \\ \vdots \\ \mathsf{p}_{n-1} \end{bmatrix} = \begin{bmatrix} \mathsf{u}_0 & 0 & \cdots & 0 \\ \mathsf{u}_1 & \mathsf{u}_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathsf{u}_{n-1} & \mathsf{u}_{n-2} & \cdots & \mathsf{u}_0 \end{bmatrix} \cdot \begin{bmatrix} \mathsf{v}_0 \\ \mathsf{v}_1 \\ \vdots \\ \mathsf{v}_{n-1} \end{bmatrix}$$

$$\Leftrightarrow \begin{bmatrix} \mathsf{p}_0 \\ \mathsf{p}_1 \\ \vdots \\ \mathsf{p}_{n-1} \end{bmatrix} = \begin{bmatrix} \mathsf{u}_0\mathsf{v}_0 \\ \mathsf{u}_1\mathsf{v}_0 + \mathsf{u}_0\mathsf{v}_1 \\ \vdots \\ \sum_{i=0}^{n-1} \mathsf{u}_i\mathsf{v}_{n-1-i} \end{bmatrix}$$

We define the following polynomials:

- $u(x) = \sum_{i=0}^{n-1} \mathsf{u}_i x^i$,
- $v(x) = \sum_{i=0}^{n-1} \mathsf{v}_i x^i$,
- $p(x) = \sum_{i=0}^{n-1} \mathsf{p}_i x^i$.

We can verify that $u(x) \cdot v(x) \bmod x^n = p(x)$, as follows:

$$\begin{aligned} u(x) \cdot v(x) &= \left(\sum_{i=0}^{n-1} \mathsf{u}_i x^i\right) \cdot \left(\sum_{j=0}^{n-1} \mathsf{v}_j x^j\right) \bmod x^n \\ &= \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} \mathsf{u}_i\mathsf{v}_j x^{i+j} \bmod x^n \\ &= \sum_{k=0}^{2(n-1)} \left(\sum_{i=0}^{k} \mathsf{u}_i\mathsf{v}_{k-i}\right) x^k \bmod x^n \\ &= \sum_{k=0}^{n-1} \left(\sum_{i=0}^{k} \mathsf{u}_i\mathsf{v}_{k-i}\right) x^k \\ &= \sum_{k=0}^{n-1} \mathsf{p}_k x^k \\ &= p(x) \end{aligned}$$

◀

▶ **Lemma 41.** *Let* $\mathsf{u}, \mathsf{v} \in \mathbb{K}^{m\times 1}$ *be two vectors and* $\mathsf{p} = \mathbb{U}(\mathsf{u}) \cdot \mathsf{v}$. *We have that* $p(x) = \mathsf{rev}_{m-1}(u(x)) \cdot v(x) \operatorname{quo} x^{m-1}$, *where* $u(x) = \sum_{i=0}^{m-1} u_i x^i$, $v(x) = \sum_{i=0}^{m-1} v_i x^i$ *and* $p(x) = \sum_{i=0}^{m-1} p_i x^i$ *are the polynomial representations of the vectors* u, v *and* p, *respectively.*

Proof. We have the vector $\mathsf{u} = [\mathsf{u}_0, \mathsf{u}_1, \ldots, \mathsf{u}_{n-1}]^\mathsf{T} \in \mathbb{K}^{n\times 1}$ and the vector $\mathsf{v} = [\mathsf{v}_0, \mathsf{v}_1, \ldots, \mathsf{v}_{n-1}]^\mathsf{T} \in \mathbb{K}^{n\times 1}$. We consider $\mathsf{p} = [\mathsf{p}_0, \mathsf{p}_1, \ldots, \mathsf{p}_{n-1}]^\mathsf{T} \in \mathbb{K}^{n\times 1}$ as the vector we want to compute, given by the following equation:

$$
\begin{aligned}
\mathsf{p} = \mathbb{U}(\mathsf{u}) \cdot \mathsf{v} \Leftrightarrow \begin{bmatrix} \mathsf{p}_0 \\ \mathsf{p}_1 \\ \vdots \\ \mathsf{p}_{n-1} \end{bmatrix} &= \begin{bmatrix} \mathsf{u}_0 & \mathsf{u}_1 & \cdots & \mathsf{u}_{n-1} \\ 0 & \mathsf{u}_0 & \cdots & \mathsf{u}_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathsf{u}_0 \end{bmatrix} \cdot \begin{bmatrix} \mathsf{v}_0 \\ \mathsf{v}_1 \\ \vdots \\ \mathsf{v}_{n-1} \end{bmatrix} \\
\Leftrightarrow \begin{bmatrix} \mathsf{p}_0 \\ \mathsf{p}_1 \\ \vdots \\ \mathsf{p}_{n-1} \end{bmatrix} &= \begin{bmatrix} \mathsf{u}_0\mathsf{v}_0 + \mathsf{u}_1\mathsf{v}_1 + \cdots + \mathsf{u}_{n-1}\mathsf{v}_{n-1} \\ \mathsf{u}_0\mathsf{v}_1 + \mathsf{u}_1\mathsf{v}_2 + \cdots + \mathsf{u}_{n-2}\mathsf{v}_{n-1} \\ \vdots \\ \mathsf{u}_0\mathsf{v}_{n-1} \end{bmatrix} \\
\Leftrightarrow \begin{bmatrix} \mathsf{p}_0 \\ \mathsf{p}_1 \\ \vdots \\ \mathsf{p}_{n-1} \end{bmatrix} &= \begin{bmatrix} \sum_{i=0}^{n-1} \mathsf{u}_i\mathsf{v}_i \\ \sum_{i=0}^{n-2} \mathsf{u}_i\mathsf{v}_{1+i} \\ \vdots \\ \sum_{i=0}^{0} \mathsf{u}_i\mathsf{v}_{n-1+i} \end{bmatrix}
\end{aligned}
$$

We define the following polynomials:

- $u(x) = \sum_{i=0}^{n-1} \mathsf{u}_i x^i$,
- $v(x) = \sum_{i=0}^{n-1} \mathsf{v}_i x^i$,
- $p(x) = \sum_{i=0}^{n-1} \mathsf{p}_i x^i$.

We can verify that $\mathsf{rev}_{n-1}(u(x)) \cdot v(x) \ \mathsf{quo}\ x^{n-1} = p(x)$, as follows:

$$
\begin{aligned}
\mathsf{rev}_{n-1}(u(x)) \cdot v(x) &= \left(\sum_{i=0}^{n-1} \mathsf{u}_i x^{n-1-i}\right) \cdot \left(\sum_{j=0}^{n-1} \mathsf{v}_j x^j\right) \ \mathsf{quo}\ x^{n-1} \\
&= \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} \mathsf{u}_i\mathsf{v}_j x^{(n-1-i)+j} \ \mathsf{quo}\ x^{n-1} \\
&= \sum_{k=0}^{2(n-1)} \left(\sum_{i=0}^{k} \mathsf{u}_i\mathsf{v}_{k+i}\right) x^{n-1-k} \ \mathsf{quo}\ x^{n-1} \\
&= \sum_{k=0}^{n-1} \left(\sum_{i=0}^{k} \mathsf{u}_i\mathsf{v}_{k-i}\right) x^k \\
&= \sum_{k=0}^{n-1} \mathsf{p}_k x^k \\
&= p(x)
\end{aligned}
$$

◀