Computing canonical bases of modules of univariate relations

Vincent Neiger, Thi Xuan Vu

Computing canonical bases of modules of univariate relations

Vincent Neiger
Technical University of Denmark
Kgs. Lyngby, Denmark
vinn@dtu.dk

Vu Thi Xuan
ENS de Lyon, LIP (CNRS, Inria, ENSL, UCBL)
Lyon, France
thi.vu@ens-lyon.fr

ABSTRACT

We study the computation of relations between elements of a finite-dimensional $\mathbb{K}[x]$-module. The latter is a quotient $\mathbb{K}[x]^n/\mathcal{M}$ specified by a basis $\mathbf{M} \in \mathbb{K}[x]^{n\times n}$ of $\mathcal{M}$. Then, on input $\mathbf{F} \in \mathbb{K}[x]^{m\times n}$, we seek canonical bases of the set of relations $\mathbf{p} \in \mathbb{K}[x]^{1\times m}$ such that $\mathbf{pF} = 0 \bmod \mathbf{M}$. This generalizes the computation of approximant bases, where the basis $\mathbf{M}$ is a diagonal of powers of x.

Focusing on a Hermite basis $\mathbf{M}$, our algorithm exploits the triangular shape to follow the divide-and-conquer approach used in fast approximant basis computation. Besides recent techniques for this approach, we rely on high-order lifting to perform fast modular products of the form $\mathbf{PF} \bmod \mathbf{M}$.

Our algorithm has a cost bound of $O^{\sim}(m^{\omega-1}D + n^{\omega}D/m)$ operations in $\mathbb{K}$, where $D = \deg(\det(\mathbf{M}))$ is the dimension of $\mathbb{K}[x]^n/\mathcal{M}$, $O^{\sim}(\cdot)$ indicates that logarithmic factors are omitted, and ω is the exponent of matrix multiplication. To the best of our knowledge, this had previously only been achieved for a diagonal matrix $\mathbf{M}$.

As a particular case, our algorithm computes the shifted Popov form of $\mathbf{M}$ within the same cost bound, up to logarithmic factors, as the previously fastest known algorithm, which is randomized.

KEYWORDS

Polynomial matrix; shifted Popov form; division with remainder; univariate equations; syzygy module.

1 INTRODUCTION

In what follows, $\mathbb{K}$ is a field, $\mathbb{K}[x]$ denotes the set of univariate polynomials in x over $\mathbb{K}$, and $\mathbb{K}[x]^{m\times n}$ denotes the set of $m \times n$ (univariate) polynomial matrices.

Relations. In this paper, we place ourselves in a univariate setting and we study the following type of *relations*, also known as *syzygies*.

We are given a (free) $\mathbb{K}[x]$-submodule $\mathcal{M} \subseteq \mathbb{K}[x]^n$ of rank n, specified by one of its bases, represented as the rows of a nonsingular matrix $\mathbf{M} \in \mathbb{K}[x]^{n\times n}$. In addition, we are given m elements $f_1, \ldots, f_m \in \mathbb{K}[x]^n/\mathcal{M}$, represented as a matrix $\mathbf{F} \in \mathbb{K}[x]^{m\times n}$. The kernel of the module morphism

$$\varphi_{\mathcal{M},f} : \begin{array}{ccc} \mathbb{K}[x]^m & \to & \mathbb{K}[x]^n/\mathcal{M} \\ (p_1, \ldots, p_m) & \mapsto & p_1 f_1 + \cdots + p_m f_m \end{array},$$

formed by relations between the elements, is known as a *syzygy module* [10]. Following the matrix viewpoint above, we write it as

$$\mathrm{Rel}(\mathbf{M}, \mathbf{F}) = \{\mathbf{p} \in \mathbb{K}[x]^{1\times m} \mid \mathbf{pF} = \mathbf{0} \bmod \mathbf{M}\},$$

where the notation $\mathbf{A} = \mathbf{0} \bmod \mathbf{M}$ stands for "$\mathbf{A} = \mathbf{QM}$ for some $\mathbf{Q}$", or in other words, the rows of $\mathbf{A}$ are in $\mathcal{M}$. In what follows, the elements of $\mathrm{Rel}(\mathbf{M}, \mathbf{F})$ are called *relations of* $\mathrm{Rel}(\mathbf{M}, \mathbf{F})$.

Examples of such relations are the following.

- *Hermite-Padé approximants:* these are relations for $n = 1$ and $\mathcal{M} = x^D\mathbb{K}[x]$; that is, given polynomials $f_1, \ldots, f_m$ modulo x^D, the corresponding Hermite-Padé approximants are the vectors $(p_1, \ldots, p_m) \in \mathbb{K}[x]^m$ such that
$$p_1 f_1 + \cdots + p_m f_m = 0 \bmod x^D.$$
Fast algorithms for finding this type of approximants include [3, 15, 19, 31, 37].
- *Multipoint Padé approximants:* the fast computation of relations when $\mathcal{M}$ is a product of ideals, corresponding to a diagonal basis $\mathbf{M} = \mathrm{diag}(M_1, \ldots, M_n)$, was studied in [2, 4, 19, 20, 26, 32]. Many of these references focus $M_1, \ldots, M_n$ which split over $\mathbb{K}$ with known roots and multiplicities; then, relations are known as multipoint Padé approximants [1], or also *interpolants* in [4, 20]. Here, a relation can be thought of as a solution to a linear system over $\mathbb{K}[x]$ in which the jth equation is modulo M_j.

Canonical relation bases. The module $\mathrm{Rel}(\mathbf{M}, \mathbf{F})$ is free of rank m since we have the inclusions $\mathbb{K}[x]^m \subseteq \mathrm{Rel}(\mathbf{M}, \mathbf{F}) \subseteq \det(\mathbf{M})\mathbb{K}[x]^m$ [8, Sec. 12.1, Thm. 4]. Hence, any of its bases can be represented as (the rows of) a nonsingular matrix in $\mathbb{K}[x]^{m\times m}$, hereafter called a *relation basis for* $\mathrm{Rel}(\mathbf{M}, \mathbf{F})$.

We are specifically interested in computing bases that have good properties with regards to the computations that we will perform to obtain them, and also that we may perform once we have them. For this, we will use the notion of *shifted Popov bases* [5, 27]. Such a basis is canonical in terms of the module it generates and of a *shift*, which is a tuple $\mathbf{s} \in \mathbb{Z}^n$ that specifies a monomial order on $\mathbb{K}[x]^n$. Furthermore, the degrees in shifted Popov bases are well controlled, which helps both to compute them faster than other families of bases (see [19] and [25, Sec. 1.2.2]), and to use them for further fast computations (see for example [28, Thm. 12]). Having a shifted Popov basis of a submodule of $\mathcal{M} \subseteq \mathbb{K}[x]^n$ is particularly useful for efficiently performing computations in the quotient $\mathbb{K}[x]^n/\mathcal{M}$, as can be observed in Section 3 of this paper.

Before formally defining shifted Popov bases, we note that they coincide with Gröbner bases for $\mathbb{K}[x]$-submodules of $\mathbb{K}[x]^n$, for a term-over-position monomial order on $\mathbb{K}[x]^n$ weighted by the entries of the shift [9, Chap. 15]. For more details about this link, we refer to [24, Chap. 6] or [25, Chap. 1] (the former is in a different context but directly carries over to our situation here).

For a shift $\mathbf{s} = (s_1, \ldots, s_n) \in \mathbb{Z}^n$, the $\mathbf{s}$-*degree* of a row vector $\mathbf{p} = [p_1, \ldots, p_n] \in \mathbb{K}[x]^{1\times n}$ is $\max_{1\leqslant j\leqslant n}(\deg(p_j) + s_j)$; the $\mathbf{s}$-*row degree* of a matrix $\mathbf{P} \in \mathbb{K}[x]^{m\times n}$ is $\mathrm{rdeg}_{\mathbf{s}}(\mathbf{P}) = (d_1, \ldots, d_m)$ with d_i the $\mathbf{s}$-degree of the ith row of $\mathbf{P}$. Then, the $\mathbf{s}$-*leading matrix* of $\mathbf{P} = [p_{i,j}]_{ij}$ is the matrix $\mathrm{lm}_{\mathbf{s}}(\mathbf{P}) \in \mathbb{K}^{m\times n}$ whose entry (i, j) is the coefficient of degree $d_i - s_j$ of $p_{i,j}$. Similarly, the list of column degrees of a matrix $\mathbf{P}$ is denoted by $\mathrm{cdeg}(\mathbf{P})$.

We will use the following definitions from [5, 21].

Definition 1.1. Let $\mathbf{P} \in \mathbb{K}[x]^{m\times m}$ be nonsingular, and let $\mathbf{s} \in \mathbb{Z}^m$. Then, $\mathbf{P}$ is said to be in

- $\mathbf{s}$-*reduced form* if $\mathrm{lm}_{\mathbf{s}}(\mathbf{P})$ is invertible;
- $\mathbf{s}$-*Popov form* if $\mathrm{lm}_{\mathbf{s}}(\mathbf{P})$ is unit lower triangular and $\mathrm{lm}_{\mathbf{0}}(\mathbf{P}^{\mathsf{T}})$ is the identity matrix.

Hereafter, when we introduce a matrix by saying that it is reduced, it is understood that it is nonsingular. Similar forms can be defined for when one considers modules generated by the *columns* of a matrix rather than by its rows; in the context of polynomial matrix division with remainder, we will use the notion of $\mathbf{P}$ in *column reduced* form, meaning that $\mathrm{lm}_{\mathbf{0}}(\mathbf{P}^{\mathsf{T}})$ is invertible. In particular, any matrix in shifted Popov form is column reduced.

Considering relation bases $\mathbf{P}$ for $\mathrm{Rel}(\mathbf{M}, \mathbf{F})$ in shifted Popov form offers a strong control over the degrees of their entries. As shifted (row) reduced bases, they satisfy the *predictable degree property* (see [12]), which is at the core of the correctness of a divide-and-conquer approach behind most algorithms for the two specific situations described above (see for example [3, 15, 16, 20]). In addition, being column reduced matrices they have small average column degree, which is central in the efficiency of fast algorithms for non-uniform shifts [19, 26]. Indeed, we will see in Lemma 2.3 that

$$|\mathrm{cdeg}(\mathbf{P})| = \deg(\det(\mathbf{P})) \leqslant \deg(\det(\mathbf{M})),$$

where $|\cdot|$ denotes the sum of the entries of a tuple, and the comparison of tuples is componentwise.

Another canonical form will have a central place in our work: a matrix $\mathbf{M} \in \mathbb{K}[x]^{n\times n}$ is in *Hermite form* if $\mathbf{M}$ is upper triangular and $\mathrm{lm}_{\mathbf{0}}(\mathbf{P}^{\mathsf{T}})$ is the identity matrix. We remark that, if $d \geqslant \deg(\det(\mathbf{M}))$ then $\mathbf{M}$ is equivalently in $(dn, d(n-1), \ldots, d)$-Popov form. Thus, one motivation for considering non-uniform shifts is to handle the computation of relation bases in Hermite form. To summarize, we are interested in the design of fast algorithms for Problem 1.

> *Problem 1.* Relation basis
> *Input:*
> - nonsingular matrix $\mathbf{M} \in \mathbb{K}[x]^{n\times n}$,
> - matrix $\mathbf{F} \in \mathbb{K}[x]^{m\times n}$,
> - shift $\mathbf{s} \in \mathbb{Z}^m$.
>
> *Output:*
> - the $\mathbf{s}$-Popov relation basis $\mathbf{P} \in \mathbb{K}[x]^{m\times m}$ for $\mathrm{Rel}(\mathbf{M}, \mathbf{F})$.

Relations modulo Hermite forms. Our main focus is on the case where $\mathbf{M}$ is in Hermite form and $\mathbf{F}$ is already reduced modulo $\mathbf{M}$.

Theorem 1.2. *If $\mathbf{M}$ is in Hermite form with $\mathrm{cdeg}(\mathbf{F}) < \mathrm{cdeg}(\mathbf{M})$, there is a deterministic algorithm which solves Problem 1 using*

$$\widetilde{O}(m^{\omega-1}D + n^{\omega}D/m)$$

operations in $\mathbb{K}$, where $D = \deg(\det(\mathbf{M})) = |\mathrm{cdeg}(\mathbf{M})|$.

The algorithm and the proof corresponding to this result can be found in Section 5. Here, ω is so that we can multiply $m\times m$ matrices over $\mathbb{K}$ in $O(m^{\omega})$ operations in $\mathbb{K}$, the best known bound being $\omega < 2.38$ [7, 23]; in this paper, we only consider the case $\omega > 2$. The notation $\widetilde{O}(\cdot)$ means that we have omitted the logarithmic factors in the asymptotic bound.

To put this cost bound in perspective, the number of field elements used to represent the input $\mathbf{F}$ and $\mathbf{M}$ is at most $(m+n)D$. The output basis is represented using at most mD elements. Most often we have $n \in O(m)$, and then the cost $\widetilde{O}(m^{\omega-1}D)$ is satisfactory.

To the best of our knowledge, previous algorithms with a comparable cost bound focus on the case of a diagonal matrix $\mathbf{M}$.

The case of minimal approximant bases $\mathbf{M} = x^d\mathbf{I}_n$ has concentrated a lot of attention. After a first algorithm with cost quasi-linear in d was given [3], several improvements have achieved the cost bound $\widetilde{O}(m^{\omega-1}nd) = \widetilde{O}(m^{\omega-1}D)$ [15, 30, 37] under some assumption on the dimensions or on the shift.

In [20], the divide-and-conquer approach of [3] was carried over and made efficient in the more general case of interpolant bases $\mathbf{M} = \mathrm{diag}(M_1, \ldots, M_n)$ where the polynomials M_i split over $\mathbb{K}$ with known linear factors. Augmenting this approach with a new strategy focusing on degree information to efficiently compute the shifted Popov bases for arbitrary shifts, the cost bound $\widetilde{O}(m^{\omega-1}D)$ was obtained in [19] for such $\mathbf{M}$.

Then, the case of a diagonal matrix $\mathbf{M} = \mathrm{diag}(M_1, \ldots, M_n)$, with no assumption on the M_i's, was dealt with in [26] with the cost bound $\widetilde{O}(m^{\omega-1}D + n^{\omega}D/m)$. The main ingredient used therein, in addition to adapting tools developed for interpolant bases, was an efficient algorithm for the case $n = 1$, that is, when solving a single linear equation modulo a polynomial in $\mathbb{K}[x]$.

Here, we obtain the same cost bound as [26] for any matrix $\mathbf{M}$ in Hermite form. We remark that the cost $\widetilde{O}(m^{\omega-1}D)$, achieved for interpolant bases, is out of aim: when $n \gg m$, the mere number of field elements used to represent the input matrix $\mathbf{M}$ may be nD.

For more details and comparison with earlier algorithms focusing on specific cases of diagonal matrices $\mathbf{M}$, we refer the reader to [26, Sec. 1.2] and in particular Table 2 therein.

Our algorithm essentially follows the approach of [26]. In particular, it uses the algorithm developed there for $n = 1$. However, working modulo Hermite forms instead of diagonal matrices makes the computation of *residuals* much more involved. The residual is a modular product $\mathbf{PF}$ mod $\mathbf{M}$ which is computed after the first recursive call and is to be used as an input replacing $\mathbf{F}$ for the second recursive call. When $\mathbf{M}$ is diagonal, its computation boils down to the multiplication of $\mathbf{P}$ and $\mathbf{F}$, although some care has to be taken to account for their possibly unbalanced column degrees. However, when $\mathbf{M}$ is a Hermite form, computing $\mathbf{PF}$ mod $\mathbf{M}$ becomes a much greater challenge: we actually have to efficiently compute a matrix remainder instead of simply taking polynomial remainders for each column separately, while still taking the unbalanced degrees into account. Designing a fast algorithm for computing these residuals is the object of Section 3.

Shifted Popov forms of matrices. An important particular case of Problem 1 is the following question: given a shift $\mathbf{s} \in \mathbb{Z}^n$ and a nonsingular matrix $\mathbf{M} \in \mathbb{K}[x]^{n\times n}$, compute its $\mathbf{s}$-Popov form. Indeed, $\mathbf{M}$ is a relation basis for $\mathrm{Rel}(\mathbf{M}, \mathbf{I}_n)$ (see Lemma 2.6), and therefore the $\mathbf{s}$-Popov relation basis for $\mathrm{Rel}(\mathbf{M}, \mathbf{I}_n)$ is the $\mathbf{s}$-Popov form of $\mathbf{M}$.

To compute this relation basis efficiently, we start by computing the Hermite form $\mathbf{H}$ of $\mathbf{M}$, which can be done deterministically in

$\tilde{O}(n^\omega \lceil D_{\mathbf{M}}/n \rceil)$ operations [22]. Here, $D_{\mathbf{M}}$ is the *generic determinant bound* [17]; writing $\mathbf{M} = [a_{ij}]$, this is defined as

$$D_{\mathbf{M}} = \max_{\pi \in S_n} \sum_{1 \leqslant i \leqslant n} \overline{\deg}(a_{i,\pi_i})$$

where S_n is the set of permutations of $\{1, \ldots, n\}$, and where

$$\overline{\deg}(p) = \begin{cases} 0 & \text{if} \quad p = 0 \\ \deg(p) & \text{if} \quad p \neq 0 \end{cases}.$$

In particular, $D_{\mathbf{M}}/n$ is bounded from above by both the average of the degrees of the columns of $\mathbf{M}$ and that of its rows. For more details about this quantity, we refer to [17, Sec. 6] and [22, Sec. 2.3].

Since the rows of $\mathbf{H}$ generate the same module as $\mathbf{M}$, we have $\mathrm{Rel}(\mathbf{M}, \mathbf{I}_n) = \mathrm{Rel}(\mathbf{H}, \mathbf{I}_n)$ (see Lemma 2.4). Then, applying our algorithm for relations modulo $\mathbf{H}$ has a cost of $\tilde{O}(n^{\omega-1} \deg(\det(\mathbf{H})))$ operations, according to Theorem 1.2. This yields the next result.

Theorem 1.3. *Given a nonsingular matrix $\mathbf{M} \in \mathbb{K}[x]^{n \times n}$ and a shift $\mathbf{s} \in \mathbb{Z}^n$, there is a deterministic algorithm which computes the $\mathbf{s}$-Popov form of $\mathbf{M}$ using*

$$\widetilde{O}(n^\omega \lceil D_{\mathbf{M}}/n \rceil) \subseteq \widetilde{O}(n^\omega \deg(\mathbf{M}))$$

operations in $\mathbb{K}$.

A similar cost bound was obtained in [26], yet with a randomized algorithm. In [26], the approach to obtain the shifted Popov form is that of the randomized algorithm of [18] to compute Hermite forms. Namely, the first step is to compute the Smith form $\mathbf{S}$ of $\mathbf{M}$ along with a matrix $\mathbf{F}$ such that $\mathbf{M}$ is a relation basis for $\mathrm{Rel}(\mathbf{S}, \mathbf{F})$. One is left with a relation basis problem modulo a diagonal, solved efficiently and deterministically first in [18] when some output degree information is known, and then in [26] in general. For a more detailed comparison with earlier row reduction and Popov forms algorithms, we refer to [26, Sec. 1.1] and Table 1 therein.

Here, relying on the deterministic computation of the Hermite form of $\mathbf{M}$, our algorithm for relation bases modulo Hermite bases allows us to circumvent the computation of $\mathbf{S}$, for which the currently fastest known algorithm is Las Vegas randomized [29].

General relation bases. To solve the general case of Problem 1, one can proceed as follows:

- find the Hermite form $\mathbf{H}$ of $\mathbf{M}$, which costs $\tilde{O}(n^\omega \lceil D_{\mathbf{M}}/n \rceil)$;
- reduce $\mathbf{F}$ modulo $\mathbf{H}$, for example using Algorithm 1 which costs $\tilde{O}((m^\omega + n^\omega)(\lceil D/n \rceil + \deg(\mathbf{F})))$ in this case;
- apply Algorithm 5 for relations modulo a Hermite form.

Outline. We first present a few basic results about relation bases in Section 2. Then, we focus on the fast computation of residuals $\mathbf{PF} \bmod \mathbf{M}$ in Section 3, using in particular high-order lifting to handle the unbalanced column degrees in $\mathbf{P}$. After that, in Section 4 we explain how three situations can be directly dealt with using results from the literature: when $n = 1$, when the column degree of the output basis is known a priori, and when $D \leqslant m$. Finally, we give our algorithm for relations modulo Hermite forms in Section 5.

2 PRELIMINARIES

Division with remainder for polynomial matrices will be central in this paper, since Problem 1 involves working modulo a matrix $\mathbf{M}$.

Theorem 2.1. *Let $\mathbf{M} \in \mathbb{K}[x]^{n \times n}$ be column reduced. Then, for any $\mathbf{F} \in \mathbb{K}[x]^{m \times n}$, there exist unique polynomial matrices $(\mathbf{Q}, \mathbf{R})$ such that $\mathbf{F} = \mathbf{QM} + \mathbf{R}$ and $\mathrm{cdeg}(\mathbf{R}) < \mathrm{cdeg}(\mathbf{M})$.*

For more details, we refer to [13, Sec. IV.§2] and [21, Thm. 6.3-15]. As a consequence, we have the following description $\mathbb{K}[x]^n/\mathcal{M}$ according to the degrees in a column reduced basis of $\mathcal{M}$.

Lemma 2.2. *Let $\mathcal{M}$ be a $\mathbb{K}[x]$-submodule of $\mathbb{K}[x]^n$ of rank n. Let $\mathbf{M} \in \mathbb{K}[x]^{n \times n}$ whose rows form a basis of $\mathcal{M}$ and which is in column reduced form. Let $(d_1, \ldots, d_n) = \mathrm{cdeg}(\mathbf{M})$. Then,*

$$\mathbb{K}[x]^n/\mathcal{M} \;\cong\; \mathbb{K}[x]/(x^{d_1}) \times \cdots \times \mathbb{K}[x]/(x^{d_n}),$$

and in particular, the dimension of $\mathbb{K}[x]^n/\mathcal{M}$ as a $\mathbb{K}$-vector space is $d_1 + \cdots + d_n = \deg(\det(\mathbf{M}))$.

This allows us to bound the degree of the determinant of a relation basis, and therefore the sum of column degrees of any column reduced relation basis; for example, a shifted Popov relation basis. This control of the average column degree of the basis $\mathbf{P}$ will be central in the efficiency of our algorithms.

Lemma 2.3. *Let $\mathbf{F} \in \mathbb{K}[x]^{m \times n}$, and let $\mathbf{M} \in \mathbb{K}[x]^{n \times n}$ be nonsingular. Then, any relation basis $\mathbf{P} \in \mathbb{K}[x]^{m \times m}$ for $\mathrm{Rel}(\mathbf{M}, \mathbf{F})$ is such that $\deg(\det(\mathbf{P})) \leqslant \deg(\det(\mathbf{M}))$. In particular, if $\mathbf{P}$ is $\mathbf{0}$-column reduced, then $|\mathrm{cdeg}(\mathbf{P})| \leqslant \deg(\det(\mathbf{M}))$.*

Proof. Since all relation bases for $\mathrm{Rel}(\mathbf{M}, \mathbf{F})$ have the same determinant up to a multiplicative constant from $\mathbb{K}$, we may assume without loss of generality that $\mathbf{P}$ is $\mathbf{0}$-column reduced. Then, we have $|\mathrm{cdeg}(\mathbf{P})| = \deg(\det(\mathbf{P}))$, and Lemma 2.2 ensures that $\deg(\det(\mathbf{P}))$ is the vector space dimension of $\mathbb{K}[x]^m/\mathrm{Rel}(\mathbf{M}, \mathbf{F})$.

On the other hand, $\mathrm{Rel}(\mathbf{M}, \mathbf{F})$ being the kernel of the morphism $\varphi_{\mathcal{M},f}$ (see Section 1), we have that $\mathbb{K}[x]^m/\mathrm{Rel}(\mathbf{M}, \mathbf{F})$ is isomorphic to a submodule of $\mathbb{K}[x]^n/\mathcal{M}$, where $\mathcal{M}$ is the row space $\mathbf{M}$. Thus, $\deg(\det(\mathbf{P}))$ is at most the vector space dimension of $\mathbb{K}[x]^m/\mathcal{M}$, which is $\deg(\det(\mathbf{M}))$ according to Lemma 2.2. □

The next lemma formalizes the facts that, when computing relations of $\mathrm{Rel}(\mathbf{M}, \mathbf{F})$,

- any other basis of the module generated by $\mathbf{M}$ may be used ($\mathbf{UM}$ for $\mathbf{U}$ unimodular);
- the set of relations is not changed if $\mathbf{F}$ and $\mathbf{M}$ are both right-multiplied by the same nonsingular matrix;
- the input $\mathbf{F}$ may be considered modulo $\mathbf{M}$.

Lemma 2.4. *Let $\mathbf{F} \in \mathbb{K}[x]^{m \times n}$, and let $\mathbf{M} \in \mathbb{K}[x]^{n \times n}$ be nonsingular. Then, for any nonsingular $\mathbf{A} \in \mathbb{K}[x]^{n \times n}$, any matrix $\mathbf{B} \in \mathbb{K}[x]^{m \times n}$, and any unimodular $\mathbf{U} \in \mathbb{K}[x]^{m \times m}$, we have*

$$\mathrm{Rel}(\mathbf{M}, \mathbf{F}) = \mathrm{Rel}(\mathbf{UM}, \mathbf{F}) = \mathrm{Rel}(\mathbf{MA}, \mathbf{FA}) = \mathrm{Rel}(\mathbf{M}, \mathbf{F} + \mathbf{BM}).$$

As we will see now, a consequence is that we may easily discard the identity columns in $\mathbf{M}$; in our algorithms, where we work with $\mathbf{M}$ in shifted Popov form, this means that we can assume that all entries of $\mathrm{cdeg}(\mathbf{M})$ are positive.

Corollary 2.5. *Let $\mathbf{F} \in \mathbb{K}[x]^{m \times n}$, and let $\mathbf{M} \in \mathbb{K}[x]^{n \times n}$ be nonsingular. Suppose that $\mathbf{M}$ has at least $k \in \mathbb{Z}_{>0}$ identity columns, and that the corresponding columns of $\mathbf{F}$ are zero. Then, let $\pi \in \mathbb{K}^{n \times n}$ denote a permutation matrix such that*

$$\pi^{-1}\mathbf{M}\pi = \begin{bmatrix} \mathbf{I}_k & \mathbf{B} \\ \mathbf{0} & \mathbf{N} \end{bmatrix} \quad \text{and} \quad \mathbf{F}\pi = \begin{bmatrix} \mathbf{0} & \mathbf{G} \end{bmatrix},$$

where $\mathbf{G} \in \mathbb{K}[x]^{m\times(n-k)}$. *Then,* $\mathrm{Rel}(\mathbf{M},\mathbf{F}) = \mathrm{Rel}(\mathbf{N},\mathbf{G})$.

Proof. According to Lemma 2.4, we have

$$\pi^{-1}\mathbf{M}\pi\mathbf{A} = \begin{bmatrix}\mathbf{I}_k & \mathbf{0}\\ \mathbf{0} & \mathbf{N}\end{bmatrix} \text{ with } \mathbf{A} = \begin{bmatrix}\mathbf{I}_k & -\mathbf{B}\\ \mathbf{0} & \mathbf{I}_{n-k}\end{bmatrix},$$

hence $\mathrm{Rel}(\mathbf{M},\mathbf{F}) = \mathrm{Rel}(\mathrm{diag}(\mathbf{I}_k,\mathbf{N}), \mathbf{F}\pi\mathbf{A})$, where $\mathbf{F}\pi\mathbf{A} = [\mathbf{0}\ \ \mathbf{G}]$. □

An important consequence of Lemma 2.4 concerns the use of relation basis algorithms to compute shifted Popov forms of matrices. Lemma 2.4 together with the next lemma show that to obtain the $\mathbf{s}$-Popov form of $\mathbf{M}$ it is correct to first compute the Hermite form $\mathbf{H}$ of $\mathbf{M}$ and then return the $\mathbf{s}$-Popov relation basis for $\mathrm{Rel}(\mathbf{H},\mathbf{I}_m)$.

Lemma 2.6. *Let* $\mathbf{M} \in \mathbb{K}[x]^{m\times m}$ *be nonsingular. Then,* $\mathbf{M}$ *is a relation basis for* $\mathrm{Rel}(\mathbf{M},\mathbf{I}_m)$. *As a consequence, for* $\mathbf{s} \in \mathbb{Z}^m$, *the* $\mathbf{s}$*-Popov form of* $\mathbf{M}$ *is the* $\mathbf{s}$*-Popov relation basis for* $\mathrm{Rel}(\mathbf{UM},\mathbf{I}_m)$, *for any unimodular matrix* $\mathbf{U} \in \mathbb{K}[x]^{n\times n}$.

Proof. Let $\mathbf{P} \in \mathbb{K}[x]^{m\times m}$ be a relation basis for $\mathrm{Rel}(\mathbf{M},\mathbf{I}_m)$. Then, $\mathbf{PI}_m = \mathbf{QM}$ for some $\mathbf{Q} \in \mathbb{K}[x]^{m\times m}$. On the other hand, since the rows of $\mathbf{M}$ belong to $\mathrm{Rel}(\mathbf{M},\mathbf{I}_m)$, we have $\mathbf{M} = \mathbf{RP}$ for some $\mathbf{R} \in \mathbb{K}[x]^{m\times m}$. Since $\mathbf{P}$ is nonsingular, $\mathbf{P} = \mathbf{QRP}$ implies that $\mathbf{QR} = \mathbf{I}_m$, and therefore $\mathbf{R}$ is unimodular. Hence, $\mathbf{M} = \mathbf{RP}$ is a relation basis for $\mathrm{Rel}(\mathbf{M},\mathbf{I}_m)$. □

3 COMPUTING MODULAR PRODUCTS

In this section, we focus on algorithms for division with remainder for matrices, aiming at achieving fast computation of modular products such as those that arise in our relation basis algorithm.

3.1 Fast division with remainder

In what follows, we will write $\mathbf{F}$ quo $\mathbf{M}$ and $\mathbf{F}$ rem $\mathbf{M}$ for the quotient and the remainder in the division of $\mathbf{F}$ by $\mathbf{M}$.

The fast division algorithm for univariate polynomials starts by computing the reversed quotient via Newton iteration, and then deduces the remainder [14, Chap. 9]. This directly translates into the context of polynomial matrices, as was noted for example in the proof of [15, Lem. 3.4] or in [36, Chap. 10].

In the latter reference, it is showed how to efficiently compute the remainder $\mathcal{E}$ rem $\mathbf{M}$ where $\mathcal{E}$ is a matrix as in Eq. (1); this is not general enough for our purpose. Other algorithms for matrix division have been discussed, such as in [6, 11, 33–35], yet none seems to achieve the speed we desire. Thus, the aim of this section is to detail this extension of fast polynomial division to fast matrix division, so as to prepare the ground for tackling the computation of residuals in Section 3.2.

As said above, our division algorithm will first compute the quotient. In the next lemma, we show that the degrees of its entries are well controlled. This follows from the reducedness of the divisor, which ensures that when it is multiplied by the quotient no high-degree cancellation can occur.

Lemma 3.1. *Let* $\mathbf{M} \in \mathbb{K}[x]^{n\times n}$, $\mathbf{F} \in \mathbb{K}[x]^{m\times n}$, *and* $\delta \in \mathbb{Z}_{>0}$ *be such that* $\mathbf{M}$ *is column reduced and* $\mathrm{cdeg}(\mathbf{F}) < \mathrm{cdeg}(\mathbf{M}) + (\delta,\dots,\delta)$. *Then,* $\deg(\mathbf{F} \operatorname{quo} \mathbf{M}) < \delta$.

Proof. Writing $\mathbf{d} = \mathrm{cdeg}(\mathbf{M}) \in \mathbb{Z}^n_{\geqslant 0}$, we remark that the $\mathbf{0}$-column leading matrix of $\mathbf{M}$ is equal to its $-\mathbf{d}$-row leading matrix; that is, $\mathrm{lm}_{\mathbf{0}}(\mathbf{M}^{\mathsf{T}})^{\mathsf{T}} = \mathrm{lm}_{-\mathbf{d}}(\mathbf{M})$. Thus, $\mathbf{M}$ being $\mathbf{0}$-column reduced, it is also $-\mathbf{d}$-row reduced.

This reducedness implies that the predictable degree property holds [21, Thm. 6.3-13] when multiplying $\mathbf{M}$. Let $\mathbf{Q},\mathbf{R} \in \mathbb{K}[x]^{m\times n}$ denote the quotient and remainder in the division of $\mathbf{F}$ by $\mathbf{M}$: since $\mathrm{rdeg}_{-\mathbf{d}}(\mathbf{M}) = \mathbf{0}$, this property states that $\mathrm{rdeg}_{-\mathbf{d}}(\mathbf{QM}) = \mathrm{rdeg}_{\mathbf{0}}(\mathbf{Q})$.

On the other hand, our assumption $\mathrm{cdeg}(\mathbf{F}) < \mathbf{d} + (\delta,\dots,\delta)$ and the fact that $\mathrm{cdeg}(\mathbf{R}) < \mathbf{d}$ imply that $\mathrm{cdeg}(\mathbf{F}-\mathbf{R}) < \mathbf{d} + (d,\dots,d)$, hence $\mathrm{rdeg}_{-\mathbf{d}}(\mathbf{F}-\mathbf{R}) < (\delta,\dots,\delta)$. Since $\mathbf{F}-\mathbf{R} = \mathbf{QM}$, from the identity in the previous paragraph we obtain $\mathrm{rdeg}_{\mathbf{0}}(\mathbf{Q}) < (\delta,\dots,\delta)$, and therefore $\deg(\mathbf{Q}) < \delta$. □

This implies the following result concerning modular products.

Corollary 3.2. *Let* $\mathbf{M} \in \mathbb{K}[x]^{n\times n}$ *and* $\mathbf{F} \in \mathbb{K}[x]^{m\times n}$ *be such that* $\mathbf{M}$ *is column reduced and* $\mathrm{cdeg}(\mathbf{F}) < \mathrm{cdeg}(\mathbf{M})$, *and let* $\mathbf{P} \in \mathbb{K}[x]^{k\times m}$. *Then,* $\mathrm{rdeg}((\mathbf{PF}) \operatorname{quo} \mathbf{M}) < \mathrm{rdeg}(\mathbf{P})$.

Proof. For the case $k = 1$, the inequality follows from Lemma 3.1 since $\mathrm{cdeg}(\mathbf{PF}) \leqslant (\delta,\dots,\delta) + \mathrm{cdeg}(\mathbf{F}) < (\delta,\dots,\delta) + \mathrm{cdeg}(\mathbf{M})$, where $\delta = \deg(\mathbf{P})$. Then, the general case $k \in \mathbb{Z}_{>0}$ follows by considering separately each row of $\mathbf{P}$. □

Going back to the division $\mathbf{F} = \mathbf{QM} + \mathbf{R}$, to obtain the reversed quotient we will right-multiply the reversed $\mathbf{F}$ by an expansion of the inverse of the reversed $\mathbf{M}$. This operation is efficiently performed by means of high-order lifting; we will use the next result.

Lemma 3.3. *Let* $\mathbf{M} \in \mathbb{K}[x]^{n\times n}$ *with* $\mathbf{M}(0)$ *nonsingular, and let* $\mathbf{F} \in \mathbb{K}[x]^{m\times n}$. *Then, defining* $d = \lceil |\mathrm{cdeg}(\mathbf{M})|/n \rceil$, *the truncated* x*-adic expansion* $\mathbf{FM}^{-1} \bmod x^{kd}$ *can be computed deterministically using* $O^{\sim}(\lceil mk/n \rceil n^{\omega} d)$ *operations in* $\mathbb{K}$.

This is a minor extension of [29, Prop. 15], incorporating the average column degree of the matrix $\mathbf{M}$ instead of the largest degree of its entries. This can be done by means of partial column linearization [17, Sec. 6], as follows. One first expands the high-degree columns of $\mathbf{M}$ and inserts elementary rows to obtain a matrix $\overline{\mathbf{M}} \in \mathbb{K}[x]^{\overline{n}\times\overline{n}}$ such that $n \leqslant \overline{n} < 2n$, $\deg(\overline{\mathbf{M}}) \leqslant \delta$, and $\mathbf{M}^{-1}$ is the $n\times n$ principal leading submatrix of $\overline{\mathbf{M}}^{-1}$ [17, Thm. 10 and Cor. 2]. Then, defining $\overline{\mathbf{F}} = [\mathbf{F}\ \ \mathbf{0}] \in \mathbb{K}[x]^{m\times\overline{n}}$, we have that $\mathbf{FM}^{-1}$ is the submatrix of $\overline{\mathbf{F}}\,\overline{\mathbf{M}}^{-1}$ formed by its first n columns. Thus, the sought truncated expansion is obtained by computing $\overline{\mathbf{F}}\,\overline{\mathbf{M}}^{-1} \bmod x^{k\delta}$, which is done efficiently by [29, Alg. 4] with the choice $X = x^{\delta}$; this is valid since this polynomial is coprime to $\det(\overline{\mathbf{M}}) = \det(\mathbf{M})$ and its degree is at least the degree of $\overline{\mathbf{M}}$.

Proposition 3.4. *Algorithm 1 is correct. Assuming that both* $m\delta$ *and* n *are in* $O(D)$, *where* $D = |\mathrm{cdeg}(\mathbf{M})|$, *this algorithm uses* $O^{\sim}(\lceil m/n \rceil n^{\omega-1} D)$ *operations in* $\mathbb{K}$.

Proof. Let $\mathbf{Q}$ and $\mathbf{R}$ denote the quotient and remainder in the division of $\mathbf{F}$ by $\mathbf{M}$. Writing $\mathrm{cdeg}(\mathbf{M}) = (d_1,\dots,d_n)$, we have the bounds $\mathrm{cdeg}(\mathbf{F}) < (\delta + d_1,\dots,\delta + d_n)$, $\mathrm{cdeg}(\mathbf{R}) < (d_1,\dots,d_n)$, and $\deg(\mathbf{Q}) < \delta$ according to Lemma 3.1. Thus, we can define the

Algorithm 1. PM-QuoRem
Input:

- $\mathbf{M} \in \mathbb{K}[x]^{n\times n}$ column reduced,
- $\mathbf{F} \in \mathbb{K}[x]^{m\times n}$,
- $\delta \in \mathbb{Z}_{>0}$ such that $\mathrm{cdeg}(\mathbf{F}) < \mathrm{cdeg}(\mathbf{M}) + (\delta, \dots, \delta)$.

Output: the quotient $\mathbf{F}$ quo $\mathbf{M}$ and the remainder $\mathbf{F}$ rem $\mathbf{M}$.

1. `/* reverse order of coefficients */`
   $(d_1, \dots, d_n) \leftarrow \mathrm{cdeg}(\mathbf{M})$
   $\mathbf{M}_{\mathrm{rev}} = \mathbf{M}(x^{-1})\,\mathrm{diag}(x^{d_1}, \dots, x^{d_n})$
   $\mathbf{F}_{\mathrm{rev}} = \mathbf{F}(x^{-1})\,\mathrm{diag}(x^{\delta+d_1-1}, \dots, x^{\delta+d_n-1})$
2. `/* compute quotient via expansion */`
   $\mathbf{Q}_{\mathrm{rev}} \leftarrow \mathbf{F}_{\mathrm{rev}}\mathbf{M}_{\mathrm{rev}}^{-1} \bmod x^{\delta}$
   $\mathbf{Q} \leftarrow x^{\delta-1}\mathbf{Q}_{\mathrm{rev}}(x^{-1})$
3. *Return* $(\mathbf{Q}, \mathbf{F} - \mathbf{Q}\mathbf{M})$

reversals of these polynomial matrices as

$$\begin{aligned}
\mathbf{M}_{\mathrm{rev}} &= \mathbf{M}(x^{-1})\,\mathrm{diag}(x^{d_1}, \dots, x^{d_n}),\\
\mathbf{F}_{\mathrm{rev}} &= \mathbf{F}(x^{-1})\,\mathrm{diag}(x^{\delta+d_1-1}, \dots, x^{\delta+d_n-1}),\\
\mathbf{Q}_{\mathrm{rev}} &= x^{\delta-1}\mathbf{Q}(x^{-1}),\\
\mathbf{R}_{\mathrm{rev}} &= \mathbf{R}(x^{-1})\,\mathrm{diag}(x^{d_1-1}, \dots, x^{d_n-1}),
\end{aligned}$$

for which the same degree bounds hold. Then, right-multiplying both sides of the identity $\mathbf{F}(x^{-1}) = \mathbf{Q}(x^{-1})\mathbf{M}(x^{-1}) + \mathbf{R}(x^{-1})$ by the matrix $\mathrm{diag}(x^{\delta+d_1-1}, \dots, x^{\delta+d_n-1})$ (which is nonsingular), we obtain $\mathbf{F}_{\mathrm{rev}} = \mathbf{Q}_{\mathrm{rev}}\mathbf{M}_{\mathrm{rev}} + x^{\delta}\mathbf{R}_{\mathrm{rev}}$.

Now, note that the constant term $\mathbf{M}_{\mathrm{rev}}(0) \in \mathbb{K}^{n\times n}$ is equal to the column leading matrix of $\mathbf{M}$, which is invertible since $\mathbf{M}$ is column reduced, hence $\mathbf{M}_{\mathrm{rev}}$ is invertible (over the fractions). Thus, since $\deg(\mathbf{Q}_{\mathrm{rev}}) < \delta$, this reversed quotient matrix is completely determined from the truncated expansion $\mathbf{Q}_{\mathrm{rev}} = \mathbf{F}_{\mathrm{rev}}\mathbf{M}_{\mathrm{rev}}^{-1} \bmod x^{\delta}$; this proves the correctness of the algorithm.

Concerning the cost bound, Step **2** uses $O^{\sim}(\lceil(m\delta)/(nd)\rceil n^{\omega} d)$ operations according to Lemma 3.3, where $d = \lceil D/n\rceil$. Our assumptions imply $d \in \Theta(D/n)$ as well as $m\delta/(nd) \in O(1)$, so that this cost bound is in $O^{\sim}(n^{\omega-1}D)$.

In Step **3**, we multiply the $m \times n$ matrix $\mathbf{Q}$ of degree less than δ with the $n \times n$ matrix $\mathbf{M}$ such that $|\mathrm{cdeg}(\mathbf{M})| = D$. First consider the case $m \leqslant n$. To perform this product efficiently, we expand the rows of $\mathbf{Q}$ so as to obtain a $O(n) \times n$ matrix $\overline{\mathbf{Q}}$ of degree in $O(\lceil m\delta/n\rceil)$ and such that $\mathbf{Q}\mathbf{F}$ is easily retrieved from $\overline{\mathbf{Q}}\mathbf{F}$ (see Section 3.2 for more details about how such column expansions are carried out). Thus, this product is done in $O^{\sim}(n^{\omega-1}D)$, since $\lceil m\delta/n\rceil \in O(D/n)$. On the other hand, if $m > n$, we have $\delta \in O(D/m) \subseteq O(D/n)$. Thus, computing the product $\mathbf{Q}\mathbf{F}$ uses $\lceil m/n\rceil$ products of $n \times n$ matrices of degree $O(D/n)$, which cost each $O^{\sim}(n^{\omega-1}D)$ operations; hence the total cost $O^{\sim}(mn^{\omega-2}D)$ when $m > n$. □

3.2 Fast residual computation

Computing residuals is performing modular products $(\mathbf{P}\mathbf{F}) \operatorname{rem} \mathbf{M}$ where $\mathbf{M} \in \mathbb{K}[x]^{n\times n}$ is column reduced, and $\mathbf{P} \in \mathbb{K}[x]^{m\times m}$ and $\mathbf{F} \in \mathbb{K}[x]^{m\times n}$ are such that $|\mathrm{cdeg}(\mathbf{P})| \leqslant |\mathrm{cdeg}(\mathbf{M})|$ and $\mathrm{cdeg}(\mathbf{F}) < \mathrm{cdeg}(\mathbf{M})$. The difficulty in designing a fast algorithm comes from the non-uniformity of the column degrees of $\mathbf{P}$, so that we cannot compute the product $\mathbf{P}\mathbf{F}$ within the target cost bound.

To start with, we use the same strategy as in [19, 26]: we make the column degrees of $\mathbf{P}$ uniform, at the price of introducing another, simpler matrix $\mathcal{E}$ for which we will need to compute $(\mathcal{E}\mathbf{F}) \bmod \mathbf{M}$.

Let $(\delta_1, \dots, \delta_m) = \mathrm{cdeg}(\mathbf{P})$, $\delta = \lceil(\delta_1 + \cdots + \delta_m)/m\rceil \geqslant 1$, and for $i \in \{1, \dots, m\}$ write $\delta_i = (\alpha_i - 1)\delta + \beta_i$ with $\alpha_i = \lceil\delta_i/\delta\rceil$ and $1 \leqslant \beta_i \leqslant \delta$ if $\delta_i > 0$, and with $\alpha_i = 1$ and $\beta_i = 0$ if $\delta_i = 0$. Then, let $\overline{m} = \alpha_1 + \cdots + \alpha_m$, and define the matrix $\mathcal{E} \in \mathbb{K}[x]^{\overline{m}\times m}$ as

$$\mathcal{E} = \begin{bmatrix} 1 & & & \\ x^{\delta} & & & \\ \vdots & & & \\ x^{(\alpha_1-1)\delta} & & & \\ & \ddots & & \\ & & 1 & \\ & & x^{\delta} & \\ & & \vdots & \\ & & x^{(\alpha_m-1)\delta} & \end{bmatrix}; \tag{1}$$

define also the expanded column degrees $\overline{\boldsymbol{\delta}} \in \mathbb{Z}_{\geqslant 0}^{\overline{m}}$ as

$$\overline{\boldsymbol{\delta}} = (\underbrace{\delta, \dots, \delta, \beta_1}_{\alpha_1}, \dots, \underbrace{\delta, \dots, \delta, \beta_m}_{\alpha_m}). \tag{2}$$

Then, expanding the columns of $\mathbf{P}$ yields a matrix $\overline{\mathbf{P}} \in \mathbb{K}[x]^{m\times\overline{m}}$ such that $\mathbf{P} = \overline{\mathbf{P}}\mathcal{E}$ and $\deg(\overline{\mathbf{P}}) \leqslant \delta$. (Note that $\overline{\mathbf{P}}$ can be made unique by specifying more constraints on $\mathrm{cdeg}(\overline{\mathbf{P}})$.) The aim of this construction is that the dimension is at most doubled while the degree of the expanded matrix becomes the average column degree of $\mathbf{P}$. Precisely, $m \leqslant \overline{m} < 2m$ and $\max(\overline{\boldsymbol{\delta}}) = \delta = \lceil|\mathrm{cdeg}(\mathbf{P})|/m\rceil$.

Thus, the main tasks are to compute $\overline{\mathbf{F}} = (\mathcal{E}\mathbf{F}) \operatorname{rem} \mathbf{M}$, and after that, $(\overline{\mathbf{P}}\overline{\mathbf{F}}) \operatorname{rem} \mathbf{M}$. Since $\overline{\mathbf{P}}$ has small degree, the latter is done by computing the product and performing the division with Algorithm 1. We now study the computation of $\overline{\mathbf{F}} = (\mathcal{E}\mathbf{F}) \operatorname{rem} \mathbf{M}$.

Algorithm 2. RemOfShifts
Input:

- $\mathbf{M} \in \mathbb{K}[x]^{n\times n}$ column reduced,
- $\mathbf{F} \in \mathbb{K}[x]^{m\times n}$ such that $\mathrm{cdeg}(\mathbf{F}) < \mathrm{cdeg}(\mathbf{M})$,
- $\delta \in \mathbb{Z}_{>0}$,
- $k \in \mathbb{Z}_{\geqslant 0}$.

Output: the list of remainders $\left((x^{r\delta}\mathbf{F}) \operatorname{rem} \mathbf{M}\right)_{0\leqslant r<2^k}$.

1. *If* $k = 0$ *then Return* $\mathbf{F}$
2. *Else*
   a. $(*, \mathbf{G}) \leftarrow$ PM-QuoRem$(\mathbf{M}, x^{2^{k-1}\delta}\mathbf{F}, 2^{k-1}\delta)$
   b. $\left(\begin{bmatrix}\mathbf{R}_{r0}\\ \mathbf{R}_{r1}\end{bmatrix}\right)_{0\leqslant r<2^{k-1}} \leftarrow$ RemOfShifts$\left(\mathbf{M}, \begin{bmatrix}\mathbf{F}\\ \mathbf{G}\end{bmatrix}, \delta, k-1\right)$
   c. *Return* $(\mathbf{R}_{r0})_{0\leqslant r<2^{k-1}} \cup (\mathbf{R}_{r1})_{0\leqslant r<2^{k-1}}$

Proposition 3.5. *Algorithm 2 is correct. Assuming that both* $2^k m\delta$ *and* n *are in* $O(D)$*, where* $D = |\mathrm{cdeg}(\mathbf{M})|$*, this algorithm uses* $O^{\sim}((2^k mn^{\omega-2} + kn^{\omega-1})D)$ *operations in* $\mathbb{K}$.

Proof. The correctness follows from the division properties

$$(x^{(2^{k-1}+r)\delta}\mathbf{F}) \operatorname{rem} \mathbf{M} = (x^{r\delta}((x^{2^{k-1}\delta}\mathbf{F}) \operatorname{rem} \mathbf{M})) \operatorname{rem} \mathbf{M}$$

and

$$\begin{bmatrix}\mathbf{A}\\ \mathbf{B}\end{bmatrix} \operatorname{rem} \mathbf{M} = \begin{bmatrix}\mathbf{A} \operatorname{rem} \mathbf{M}\\ \mathbf{B} \operatorname{rem} \mathbf{M}\end{bmatrix}.$$

Now let us assume that $2^k m\delta \in O(D)$. Then, we remark that the assumptions in Proposition 3.4 about the input parameters for PM-QuoRem are always satisfied in recursive calls, since the row dimension m is doubled while the exponent $2^k\delta$ is halved. Then, from Proposition 3.4 we obtain that the algorithm uses

$$\widetilde{O}\left(\left(\sum_{0\leqslant r\leqslant k-1}\left\lceil\frac{2^r m}{n}\right\rceil\right)n^{\omega-1}D\right);$$

the announced cost bound follows. □

Algorithm 3. Residual
Input:

- $\mathbf{M} \in \mathbb{K}[x]^{n\times n}$ column reduced,
- $\mathbf{F} \in \mathbb{K}[x]^{m\times n}$ such that $\operatorname{cdeg}(\mathbf{F}) < \operatorname{cdeg}(\mathbf{M})$,
- $\mathbf{P} \in \mathbb{K}[x]^{m\times m}$.

Output: the remainder $(\mathbf{PF}) \operatorname{rem} \mathbf{M}$.

1. `/* expand high-degree columns of P */`
$(\delta_i)_{1\leqslant i\leqslant m} \leftarrow \operatorname{cdeg}(\mathbf{P})$
$\delta \leftarrow \lceil(\delta_1 + \cdots + \delta_m)/m\rceil$
$\alpha_i \leftarrow \max(1, \lceil\delta_i/\delta\rceil)$ for $1 \leqslant i \leqslant m$
$\overline{m} \leftarrow \alpha_1 + \cdots + \alpha_m$
$\overline{\mathbf{P}} \in \mathbb{K}[x]^{\overline{m}\times m} \leftarrow$ matrix such that $\mathbf{P} = \overline{\mathbf{P}}\mathcal{E}$ and $\deg(\overline{\mathbf{P}}) \leqslant \delta$ for $\mathcal{E}$ as in Eq. (1)
2. `/* for E as in Eq. (1), compute F = EF rem M */`
For $1 \leqslant i \leqslant m$ such that $\alpha_i = 1$ *do*
$\overline{\mathbf{F}}_i \in \mathbb{K}[x]^{\alpha_i\times n} \leftarrow$ row i of $\mathbf{F}$
For $1 \leqslant k \leqslant \lceil\log_2(\max_i(\alpha_i))\rceil$ *do*
$(i_1, \ldots, i_\ell) \leftarrow \{i \in \{1, \ldots, m\} \mid 2^{k-1} < \alpha_i \leqslant 2^k\}$
$\mathbf{G} \leftarrow$ submatrix of $\mathbf{F}$ formed by its rows $i_1, \ldots, i_\ell$
$(\mathbf{R}_r)_{0\leqslant r<2^k} \leftarrow$ RemOfShifts$(\mathbf{M}, \mathbf{G}, \delta, k)$
For $1 \leqslant j \leqslant \ell$ *do*
$\overline{\mathbf{F}}_{i_j} \in \mathbb{K}[x]^{\alpha_{i_j}\times n} \leftarrow$ stack the rows j of $(\mathbf{R}_r)_{0\leqslant r<\alpha_{i_j}}$
$\overline{\mathbf{F}} \leftarrow \begin{bmatrix}\overline{\mathbf{F}}_1^{\mathsf{T}} & \cdots & \overline{\mathbf{F}}_m^{\mathsf{T}}\end{bmatrix}^{\mathsf{T}} \in \mathbb{K}[x]^{\overline{m}\times n}$
3. `/* left-multiply by the expanded P */`
$\mathbf{G} \leftarrow \overline{\mathbf{P}}\,\overline{\mathbf{F}}$
4. `/* complete the remainder computation */`
$(*, \mathbf{R}) \leftarrow$ PM-QuoRem$(\mathbf{M}, \mathbf{G}, \delta)$
Return $\mathbf{R}$

Proposition 3.6. *Algorithm 3 is correct. Assuming that all of* $|\operatorname{cdeg}(\mathbf{P})|$, m, *and* n *are in* $O(D)$, *where* $D = |\operatorname{cdeg}(\mathbf{M})|$, *this algorithm uses* $O^{\sim}((m^{\omega-1} + n^{\omega-1})D)$ *operations in* $\mathbb{K}$.

Proof. Let us consider $\mathcal{E} \in \mathbb{K}[x]^{\overline{m}\times m}$ defined as in Eq. (1) from the parameters δ and $\alpha_1, \ldots, \alpha_m$ in Step **1**. We claim that the matrix $\overline{\mathbf{F}}$ computed at Step **2** is equal to $(\mathcal{E}\mathbf{F}) \operatorname{rem} \mathbf{M}$. Then, having $\operatorname{cdeg}(\overline{\mathbf{P}}\,\overline{\mathbf{F}}) < \operatorname{cdeg}(\mathbf{M}) + (\delta, \ldots, \delta)$, the correctness of PM-QuoRem ensures that $\mathbf{R} = (\overline{\mathbf{P}}\,\overline{\mathbf{F}}) \operatorname{rem} \mathbf{M}$. The correctness follows by construction of $\overline{\mathbf{P}}$ since $(\overline{\mathbf{P}}\,\overline{\mathbf{F}}) \operatorname{rem} \mathbf{M} = (\overline{\mathbf{P}}\mathcal{E}\mathbf{F}) \operatorname{rem} \mathbf{M} = (\mathbf{PF}) \operatorname{rem} \mathbf{M}$.

To prove our claim, it is enough to show that, for $1 \leqslant i \leqslant m$, the ith block $\overline{\mathbf{F}}_i$ of $\overline{\mathbf{F}}$ is the matrix formed by stacking the remainders involving the row i of $\mathbf{F}$, that is, $((x^{r\delta}\mathbf{F}_{i,*}) \operatorname{rem} \mathbf{M})_{0\leqslant r<\alpha_i}$. If $\alpha_i = 1$ then this is clear from the first *For* loop. Otherwise, let $k \in \mathbb{Z}_{>0}$ be such that $2^{k-1} < \alpha_i \leqslant 2^k$. Then, at the kth iteration of the second loop, we have $i_j = i$ for some $1 \leqslant j \leqslant \ell$. Thus, the correctness of RemOfShifts implies that, for $0 \leqslant r < 2^k$, the row j of $\mathbf{R}_r$ is $(x^{r\delta}\mathbf{G}_{j,*}) \operatorname{rem} \mathbf{M} = (x^{r\delta}\mathbf{F}_{i,*}) \operatorname{rem} \mathbf{M}$. Since $2^k \geqslant \alpha_i$, this contains the wanted remainders and the claim follows.

Let us show the cost bound, assuming that $|\operatorname{cdeg}(\mathbf{P})|$, m, and n are in $O(D)$. Note that this implies $m\delta \in O(D)$.

We first study the cost of the iteration k of the second loop of Step **2**. We have that $2^{k-1}\ell \leqslant \alpha_1 + \cdots + \alpha_m = \overline{m} \leqslant 2m$, the row dimension of $\mathbf{G}$ is ℓ, and $k \leqslant \lceil\log(\max_i(\alpha_i))\rceil \in O(\log(m))$. Thus, the call to RemOfShifts costs $O^{\sim}((mn^{\omega-2} + n^{\omega-1})D)$ operations according to Proposition 3.5, and the same cost bound holds for the whole Step **2**. Concerning Step **4**, the cost bound $O^{\sim}(\lceil m/n\rceil n^{\omega-1}D)$ follows directly from Proposition 3.4.

The product at Step **3** involves the $m \times \overline{m}$ matrix $\overline{\mathbf{P}}$ whose degree is at most δ and the $\overline{m}\times n$ matrix $\overline{\mathbf{F}}$ such that $\operatorname{cdeg}(\overline{\mathbf{F}}) < \operatorname{cdeg}(\mathbf{M})$; we recall that $\overline{m} \leqslant 2m$. If $n \geqslant m$, we expand the columns of $\overline{\mathbf{F}}$ similarly to how $\overline{\mathbf{P}}$ was obtained from $\mathbf{P}$: this yields a $\overline{m} \times (\leqslant 2n)$ matrix of degree at most $\lceil D/n\rceil$, whose left-multiplication by $\overline{\mathbf{P}}$ directly yields $\overline{\mathbf{P}}\,\overline{\mathbf{F}}$ by compressing back the columns. Thus, this product is done in $O^{\sim}(m^{\omega-2}nD)$ operations since both δ and D/n are in $O(D/m)$ when $n \geqslant m$. If $m \geqslant n$, we do a similar column expansion of $\overline{\mathbf{F}}$, yet into a matrix with $O(m)$ columns and degree $O(D/m)$; thus, the product can be performed in $O^{\sim}(m^{\omega-1}D)$ operations in this case. □

4 FAST ALGORITHMS IN SPECIFIC CASES

4.1 When the input module is an ideal

Here, we focus on Problem 1 when $n = 1$, which is a base case of our main algorithm. This is the situation where $\mathbf{M}$ is a nonzero polynomial in $\mathbb{K}[x]$, or in other words, the input module $\mathcal{M}$ is the ideal of $\mathbb{K}[x]$ generated by $\mathbf{M}$. Thus, we are looking for the $\mathbf{s}$-Popov basis for the set of relations between m elements of $\mathbb{K}[x]/(\mathbf{M})$. A fast algorithm for this specific case has been given in [26, Sec. 2.2]; precisely, the following result is achieved by running [26, Alg. 2] on input $\mathbf{M}, \mathbf{F}, \mathbf{s}, 2D$.

Proposition 4.1. *Assuming* $n = 1$ *and* $\deg(\mathbf{F}) < D = \deg(\mathbf{M})$, *there is an algorithm which solves Problem 1 using* $O^{\sim}(m^{\omega-1}D)$ *operations in* $\mathbb{K}$.

4.2 When the output minimal degree is known

Now, we focus on the situation where we know in advance the $\mathbf{s}$-minimal degree of the relation module, that is, the column degree of its $\mathbf{s}$-Popov basis. Our motivation is that, in our main algorithm, we will use the technique introduced in [19] to control the degrees in the computation in the case of arbitrary shifts. Namely, we find recursively this $\mathbf{s}$-minimal degree, and then the remaining task is to compute the $\mathbf{s}$-Popov relation basis using this knowledge.

The same question was tackled in [26, Sec. 2.1] in the specific case of a diagonal matrix $\mathbf{M}$. Here, we adapt this approach to our context, relying in particular on the fast computation of $(\mathcal{E}\mathbf{F}) \operatorname{rem} \mathbf{M}$ designed in Section 3.2. A first task is to see that [26, Lem. 2.1] is still valid in our more general setting: in Lemma 4.2, we show that it extends as soon as $\mathbf{M}$ is column reduced, mainly thanks to the degree property in Corollary 3.2. Then, for the sake of presentation, we give the slightly modified [26, Alg. 1].

Lemma 4.2. *Let $\mathbf{M} \in \mathbb{K}[x]^{n\times n}$ be column reduced, let $\mathbf{F} \in \mathbb{K}[x]^{m\times n}$ be such that $\operatorname{cdeg}(\mathbf{F}) < \operatorname{cdeg}(\mathbf{M})$, let $\mathbf{s} \in \mathbb{Z}^m$. Furthermore, let $\mathbf{P} \in \mathbb{K}[x]^{m\times m}$, and let $\mathbf{w} \in \mathbb{Z}^n$ be such that $\max(\mathbf{w}) \leqslant \min(\mathbf{s})$. Then, $\mathbf{P}$ is the $\mathbf{s}$-Popov relation basis for $\operatorname{Rel}(\mathbf{H}, \mathbf{F})$ if and only if $\begin{bmatrix}\mathbf{P} & \mathbf{Q}\end{bmatrix}$ is the $\mathbf{u}$-Popov kernel basis of $\begin{bmatrix}\mathbf{F}^\mathsf{T} & \mathbf{M}\end{bmatrix}^\mathsf{T}$ for some $\mathbf{Q} \in \mathbb{K}[x]^{m\times n}$ and $\mathbf{u} = (\mathbf{s}, \mathbf{w}) \in \mathbb{Z}^{m+n}$. In this case, $\deg(\det(\mathbf{Q})) < \deg(\det(\mathbf{P}))$ and $\begin{bmatrix}\mathbf{P} & \mathbf{Q}\end{bmatrix}$ has $\mathbf{s}$-pivot index $(1, 2, \ldots, m)$.*

Proof. Let $\mathbf{N} = \begin{bmatrix}\mathbf{F}^\mathsf{T} & \mathbf{M}\end{bmatrix}^\mathsf{T}$. It is easily verified that $\mathbf{P}$ is a relation basis for $\operatorname{Rel}(\mathbf{H}, \mathbf{F})$ if and only if there is some $\mathbf{Q} \in \mathbb{K}[x]^{m\times n}$ such that $\begin{bmatrix}\mathbf{P} & \mathbf{Q}\end{bmatrix}$ is a kernel basis of $\mathbf{N}$.

Then, for any matrix $\begin{bmatrix}\mathbf{P} & \mathbf{Q}\end{bmatrix} \in \mathbb{K}[x]^{m\times(m+n)}$ in the kernel of $\mathbf{N}$, we have $\mathbf{PF} = -\mathbf{QM}$ and therefore Corollary 3.2 shows that $\operatorname{rdeg}(\mathbf{Q}) < \operatorname{rdeg}(\mathbf{P})$; since $\max(\mathbf{w}) \leqslant \min(\mathbf{s})$, this implies $\operatorname{rdeg}_{\mathbf{w}}(\mathbf{Q}) < \operatorname{rdeg}_{\mathbf{s}}(\mathbf{P})$. Thus, we have $\operatorname{lm}_{\mathbf{u}}(\begin{bmatrix}\mathbf{P} & \mathbf{Q}\end{bmatrix}) = \begin{bmatrix}\operatorname{lm}_{\mathbf{s}}(\mathbf{P}) & \mathbf{0}\end{bmatrix}$, and therefore $\mathbf{P}$ is in $\mathbf{s}$-Popov form if and only if $\begin{bmatrix}\mathbf{P} & \mathbf{Q}\end{bmatrix}$ is in $\mathbf{u}$-Popov form with $\mathbf{s}$-pivot index $(1, \ldots, m)$. □

Algorithm 4. KnownDegreeRelations
Input:

- column reduced matrix $\mathbf{M} \in \mathbb{K}[x]^{n\times n}$,
- matrix $\mathbf{F} \in \mathbb{K}[x]^{m\times n}$ such that $\operatorname{cdeg}(\mathbf{F}) < \operatorname{cdeg}(\mathbf{M})$,
- shift $\mathbf{s} \in \mathbb{Z}^m$,
- $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_m)$ the $\mathbf{s}$-minimal degree of $\operatorname{Rel}(\mathbf{M}, \mathbf{F})$.

Output: the $\mathbf{s}$-Popov relation basis for $\operatorname{Rel}(\mathbf{M}, \mathbf{F})$.

1. `/* define partial linearization parameters */`
   $\delta \leftarrow \lceil(\delta_1 + \cdots + \delta_m)/m\rceil$,
   $\alpha_i \leftarrow \max(1, \lceil\delta_i/\delta\rceil)$ for $1 \leqslant i \leqslant m$,
   $\overline{m} \leftarrow \alpha_1 + \cdots + \alpha_m$,
   $\overline{\boldsymbol{\delta}} \leftarrow$ tuple as in Eq. (2)
2. `/* for` $\mathcal{E}$ `as in Eq. (1), compute` $\overline{\mathbf{F}} = \mathcal{E}\mathbf{F} \operatorname{rem} \mathbf{M}$ `*/`
   $\overline{\mathbf{F}} \leftarrow$ follow Step **2** of Algorithm 3 (Residual)
3. `/* compute the kernel basis */`
   $\mathbf{u} \leftarrow (-\overline{\boldsymbol{\delta}}, -\delta, \ldots, -\delta) \in \mathbb{Z}^{\overline{m}+n}$
   $\boldsymbol{\tau} \leftarrow (\operatorname{cdeg}(\mathbf{M}_{*,j}) + \delta + 1)_{1\leqslant j\leqslant n}$
   $\overline{\mathbf{P}} \leftarrow \mathbf{u}$-Popov approximant basis for $\begin{bmatrix}\overline{\mathbf{F}} \\ \mathbf{M}\end{bmatrix}$ and orders $\boldsymbol{\tau}$
4. `/* retrieve the relation basis */`
   $\mathbf{P} \leftarrow$ the principal $\overline{m} \times \overline{m}$ submatrix of $\overline{\mathbf{P}}$
   Return the submatrix of $\mathbf{P}\mathcal{E}$ formed by the rows at indices $\alpha_1 + \cdots + \alpha_i$ for $1 \leqslant i \leqslant m$

Proposition 4.3. *Algorithm 4 is correct, and assuming that m and n are in $O(D)$, where $D = |\operatorname{cdeg}(\mathbf{M})|$, it uses $O^{\tilde{}}(m^{\omega-1}D + n^\omega D/m)$ operations in $\mathbb{K}$.*

Proof. The correctness follows from the material in [26, Sec. 2.1] and [19, Sec. 4]. Concerning the cost bound, we first note that we have $\delta_1 + \cdots + \delta_m \leqslant D$ according to Lemma 2.3. Thus, the cost analysis in Proposition 3.6 shows that Step **2** uses $O^{\tilde{}}((mn^{\omega-2} + n^{\omega-1})D)$ operations. [19, Thm. 1.4] states that the approximant basis computation at Step **3** uses $O^{\tilde{}}((m+n)^{\omega-1}(1+n/m)D)$ operations, since the row dimension of the input matrix is $\overline{m} + n \leqslant 2m + n$ and the sum of the orders is $|\boldsymbol{\tau}| = |\operatorname{cdeg}(\mathbf{M})| + n(\delta+1) \leqslant D(1+n/m)$. □

4.3 When fast linear algebra is fast

Our goal here is to detail how previous work can be used to handle a specific case: when the vector space dimension $\deg(\det(\mathbf{M}))$ of the input module is small compared to the number m of input elements. Precisely, we rely on an interpretation of Problem 1 as a question of dense linear algebra over $\mathbb{K}$, which is solved efficiently by [20, Algorithm 9]. This yields the following result.

Proposition 4.4. *Assuming that $\mathbf{M}$ is in shifted Popov form, and that $\operatorname{cdeg}(\mathbf{F}) < \operatorname{cdeg}(\mathbf{M})$, there is a deterministic algorithm which solves Problem 1 using $O^{\tilde{}}(D^\omega\lceil m/D\rceil)$ operations in $\mathbb{K}$, where $D = \deg(\det(\mathbf{M})) = |\operatorname{cdeg}(\mathbf{M})|$.*

This cost bound is $O^{\tilde{}}(D^{\omega-1}m) \subseteq O^{\tilde{}}(m^{\omega-1}D)$ when $D \in O(m)$. To see why relying on fast linear algebra is sufficient to obtain a fast algorithm when $D \in O(m)$, we note that this implies that the average column degree of the $\mathbf{s}$-Popov relation basis is at most $D/m \in O(1)$. For example, if $D \leqslant m$, most entries in this basis are constants from $\mathbb{K}$: we are essentially dealing with matrices over $\mathbb{K}$. However, when $m \in O(D)$, this linear algebra solution uses $O^{\tilde{}}(D^\omega)$ operations, which largely exceeds our target cost.

We now describe how to translate our problem into this $\mathbb{K}$-linear algebra framework. Let $\mathcal{M}$ denote the row space of $\mathbf{M}$; we assume that $\mathbf{M}$ has no identity column. In order to compute in the quotient $\mathbb{K}[x]^n/\mathcal{M}$, which has finite dimension D, it is customary to make use of the *multiplication matrix* of x with respect to a given monomial basis. Here, since the basis $\mathbf{M}$ of $\mathcal{M}$ is in shifted Popov form with column degree $(d_1, \ldots, d_n) \in \mathbb{Z}^n_{>0}$, Lemma 2.2 suggests to use the monomial basis

$$\{(x^i, 0, \ldots, 0), 0 \leqslant i < d_1\} \cup \cdots \cup \{(0, \ldots, 0, x^i), 0 \leqslant i < d_n\}.$$

Above, we have represented an element in $\mathbb{K}[x]^n/\mathcal{M}$ by a polynomial vector $\mathbf{f} \in \mathbb{K}[x]^{1\times n}$ such that $\operatorname{cdeg}(\mathbf{f}) < (d_1, \ldots, d_n)$. In the linear algebra viewpoint, we rather represent it by a constant vector $\mathbf{e} \in \mathbb{K}^{1\times D}$, which is formed by the concatenations of the coefficient vectors of the entries of $\mathbf{f}$. Applying this to each row of the input matrix $\mathbf{F}$ yields a constant matrix $\mathbf{E} \in \mathbb{K}^{m\times D}$, which is another representation of the same m elements in the quotient.

Besides, the multiplication matrix $\mathbf{X} \in \mathbb{K}^{D\times D}$ is the matrix such that $\mathbf{eX} \in \mathbb{K}^{1\times D}$ corresponds to the remainder in the division of $x\mathbf{f}$ by $\mathbf{M}$. Since the basis $\mathbf{M}$ is in shifted Popov form, the computation of $\mathbf{X}$ is straightforward. Indeed, writing $\mathbf{M} = \operatorname{diag}(x^{d_1}, \ldots, x^{d_n}) - \mathbf{A}$ where $\mathbf{A} \in \mathbb{K}[x]^{n\times n}$ is such that $\operatorname{cdeg}(\mathbf{A}) < (d_1, \ldots, d_n)$, then

- the row $d_1 + \cdots + d_{i-1} + j$ of $\mathbf{X}$ is the unit vector with 1 at index $d_1 + \cdots + d_{i-1} + j + 1$, for $1 \leqslant j < d_i$ and $1 \leqslant i \leqslant n$,
- the row $d_1 + \cdots + d_i$ of $\mathbf{X}$ is the concatenation of the coefficient vectors of the row i of $\mathbf{A}$, for $1 \leqslant i \leqslant n$.

That is, writing $\mathbf{A} = [a_{ij}]_{1 \leqslant i,j \leqslant n}$ and denoting by $\{a_{ij}^k, 0 \leqslant k < d_j\}$ the coefficients of a_{ij}, the multiplication matrix $\mathbf{X} \in \mathbb{K}^{D \times D}$ is

$$\begin{bmatrix}
 & 1 & & & & & & & \\
 & & \ddots & & & & & & \\
 & & & 1 & & & & & \\
a_{11}^{(0)} & a_{11}^{(1)} & \cdots & a_{11}^{(d_1-1)} & \cdots & a_{1n}^{(0)} & a_{1n}^{(1)} & \cdots & a_{1n}^{(d_n-1)} \\
 & & & & \ddots & & & & \\
 & & & & & & 1 & & \\
 & & & & & & & \ddots & \\
 & & & & & & & & 1 \\
a_{n1}^{(0)} & a_{n1}^{(1)} & \cdots & a_{n1}^{(d_1-1)} & \cdots & a_{nn}^{(0)} & a_{nn}^{(1)} & \cdots & a_{nn}^{(d_n-1)}
\end{bmatrix}.$$

5 RELATIONS MODULO HERMITE FORMS

In this section, we give our main algorithm, along with a proof of correctness. The cost bound is given under the assumption that the Hermite matrix $\mathbf{H}$ has no identity column; how to reduce to this case by discarding columns of $\mathbf{H}$ and $\mathbf{F}$ was discussed in Corollary 2.5.

Algorithm 5. RelationsModHermite
Input:

- matrix $\mathbf{H} \in \mathbb{K}[x]^{n \times n}$ in Hermite form,
- matrix $\mathbf{F} \in \mathbb{K}[x]^{m \times n}$ such that $\operatorname{cdeg}(\mathbf{F}) < \operatorname{cdeg}(\mathbf{H})$,
- shift $\mathbf{s} \in \mathbb{Z}^m$.

Output: the $\mathbf{s}$-Popov relation basis for $\operatorname{Rel}(\mathbf{H}, \mathbf{F})$.

1. *If* $D = |\operatorname{cdeg}(\mathbf{H})| \leqslant m$:
   a. build $\mathbf{X} \in \mathbb{K}^{D \times D}$ from $\mathbf{H}$ as in Section 4.3
   b. build $\mathbf{E} \in \mathbb{K}^{m \times D}$ from $\mathbf{F}$ as in Section 4.3
   c. $\mathbf{P} \leftarrow$ [20, Algorithm 9] on input $(\mathbf{E}, \mathbf{X}, \mathbf{s}, 2^{\lceil \log_2(D) \rceil})$
   d. *Return* $\mathbf{P}$
2. *Else if* $n = 1$ *then*
   a. $\mathbf{P} \leftarrow$ [26, Alg. 2] on input $(\mathbf{H}, \mathbf{F}, \mathbf{s}, 2D)$
   b. *Return* $\mathbf{P}$
3. *Else*:
   a. $n_1 \leftarrow \lfloor n/2 \rfloor$; $n_2 \leftarrow \lceil n/2 \rceil$
   b. $\mathbf{H}_1$ and $\mathbf{H}_2 \leftarrow$ the $n_1 \times n_1$ leading and $n_2 \times n_2$ trailing principal submatrices of $\mathbf{H}$
   c. $\mathbf{F}_1 \leftarrow$ first n_1 columns of $\mathbf{F}$
   d. $\mathbf{P}_1 \leftarrow$ RelationsModHermite$(\mathbf{H}_1, \mathbf{F}_1, \mathbf{s})$
   e. $\boldsymbol{\delta}_1 \leftarrow$ diagonal degrees of $\mathbf{P}_1$
   f. $\mathbf{G} \leftarrow$ last n_2 columns of Residual$(\mathbf{P}_1, \mathbf{F}, \mathbf{H})$
   g. $\mathbf{P}_2 \leftarrow$ RelationsModHermite$(\mathbf{H}_2, \mathbf{G}, \mathbf{s} + \boldsymbol{\delta}_1)$
   h. $\boldsymbol{\delta}_2 \leftarrow$ diagonal degrees of $\mathbf{P}_2$
   i. *Return* KnownDegreeRelations$(\mathbf{H}, \mathbf{F}, \mathbf{s}, \boldsymbol{\delta}_1 + \boldsymbol{\delta}_2)$

Proposition 5.1. *Algorithm 5 is correct and, assuming the entries* $\operatorname{cdeg}(\mathbf{H})$ *are positive, it uses* $O^{\sim}(m^{\omega-1}D + n^{\omega}D/m)$ *operations in* $\mathbb{K}$, *where* $D = |\operatorname{cdeg}(\mathbf{H})|$.

Proof. The correctness and the cost bound for Step **1** follows from the discussion in Section 4.3, as summarized in Proposition 4.4. According to [26, Prop. 2.10], Step **2** correctly computes the $\mathbf{s}$-Popov relation basis and uses $O^{\sim}(m^{\omega-1}D)$ operations in $\mathbb{K}$.

Now, we focus on the correctness of Step **3**, assuming that the two recursive calls at Steps **3.d** and **3.g** correctly compute the shifted Popov relation bases. Since KnownDegreeRelations is correct, it is enough to prove that the $\mathbf{s}$-minimal degree of $\operatorname{Rel}(\mathbf{H}, \mathbf{F})$ is $\boldsymbol{\delta}_1 + \boldsymbol{\delta}_2$; for this, we will show that $\mathbf{P}_2\mathbf{P}_1$ is a relation basis for $\operatorname{Rel}(\mathbf{H}, \mathbf{F})$ whose $\mathbf{s}$-Popov form has column degree $\boldsymbol{\delta}_1 + \boldsymbol{\delta}_2$.

The fact that the $\mathbf{s}$-Popov form of $\mathbf{P}_2\mathbf{P}_1$ has column degree $\boldsymbol{\delta}_1 + \boldsymbol{\delta}_2$ follows from [19, Sec. 3], since $\mathbf{P}_1$ is in $\mathbf{s}$-Popov form and $\mathbf{P}_2$ is in $\mathbf{t}$-Popov form, where $\mathbf{t} = \mathbf{s} + \boldsymbol{\delta}_1 = \operatorname{rdeg}_{\mathbf{s}}(\mathbf{P}_1)$. It remains to prove that $\mathbf{P}_2\mathbf{P}_1$ is a relation basis for $\operatorname{Rel}(\mathbf{H}, \mathbf{F})$.

Let $\mathbf{Q}, \mathbf{R} \in \mathbb{K}[x]^{m \times n}$ denote the quotient and remainder in the division of $\mathbf{P}_1\mathbf{F}$ by $\mathbf{H}$. We write $\mathbf{Q} = [\mathbf{Q}_\ell \;\; \mathbf{Q}_r]$ with $\mathbf{Q}_\ell \in \mathbb{K}[x]^{m \times n_1}$ and $\mathbf{Q}_r \in \mathbb{K}[x]^{m \times n_2}$, and similarly $\mathbf{R} = [\mathbf{R}_\ell \;\; \mathbf{G}]$.

First, we claim that $\mathbf{R}_\ell = \mathbf{0}$. Indeed, we have $\mathbf{P}_1\mathbf{F}_1 = \mathbf{Q}_\ell\mathbf{H}_1 + \mathbf{R}_\ell$, with $\operatorname{cdeg}(\mathbf{R}_\ell) < \operatorname{cdeg}(\mathbf{H}_1)$. Besides, $\mathbf{H}_1$ is in Hermite form and therefore column reduced. Thus, $\mathbf{R}_\ell = (\mathbf{P}_1\mathbf{F}) \operatorname{rem} \mathbf{H}_1$, and the assumption $\mathbf{P}_1\mathbf{F}_1 = 0 \bmod \mathbf{H}_1$ ensures that $\mathbf{R}_\ell = \mathbf{0}$.

Second, we verify that $\mathbf{P}_2\mathbf{P}_1\mathbf{F} = 0 \bmod \mathbf{H}$. Let $\mathbf{Q}_2 \in \mathbb{K}[x]^{m \times n_2}$ denote the quotient $\mathbf{P}_2\mathbf{G} = \mathbf{Q}_2\mathbf{H}_2$. Then,

$$\mathbf{P}_2\mathbf{P}_1\mathbf{F} = \mathbf{P}_2\mathbf{Q}\mathbf{H} + [\mathbf{0} \;\; \mathbf{P}_2\mathbf{G}] = \left(\mathbf{P}_2\mathbf{Q} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_2 \end{bmatrix}\right)\mathbf{H}.$$

Now, to prove that $\mathbf{P}_2\mathbf{P}_1$ generates $\operatorname{Rel}(\mathbf{H}, \mathbf{F})$, we let $\mathbf{p} \in \operatorname{Rel}(\mathbf{H}, \mathbf{F})$ and we prove that $\mathbf{p} = \boldsymbol{\lambda}\mathbf{P}_2\mathbf{P}_1$ for some $\boldsymbol{\lambda} \in \mathbb{K}[x]^{1 \times m}$. Let $\mathbf{q}$ denote the quotient $\mathbf{p}\mathbf{F} = \mathbf{q}\mathbf{H}$, and write $\mathbf{p} = [\mathbf{p}_\ell \;\; \mathbf{p}_r]$ with $\mathbf{p}_\ell \in \mathbb{K}[x]^{1 \times n_1}$ and $\mathbf{p}_r \in \mathbb{K}[x]^{1 \times n_2}$, and similarly $\mathbf{q} = [\mathbf{q}_\ell \;\; \mathbf{q}_r]$. We have

$$\mathbf{p}\mathbf{F} = [\mathbf{q}_\ell\mathbf{H}_1 \quad \mathbf{q}_\ell\mathbf{A} + \mathbf{q}_r\mathbf{H}_2], \text{ where } \mathbf{H} = \begin{bmatrix} \mathbf{H}_1 & \mathbf{A} \\ \mathbf{0} & \mathbf{H}_2 \end{bmatrix}. \tag{3}$$

In particular, $\mathbf{p} \in \operatorname{Rel}(\mathbf{H}_1, \mathbf{F}_1)$ and therefore $\mathbf{p} = \boldsymbol{\mu}\mathbf{P}_1$ for some $\boldsymbol{\mu} \in \mathbb{K}[x]^{1 \times m}$. This yields

$$\begin{aligned}\mathbf{p}\mathbf{F} &= \boldsymbol{\mu}\mathbf{P}_1\mathbf{F} = \boldsymbol{\mu}(\mathbf{Q}\mathbf{H} + \mathbf{R}) \\ &= [\boldsymbol{\mu}\mathbf{Q}_\ell\mathbf{H}_1 \quad \boldsymbol{\mu}\mathbf{Q}_\ell\mathbf{A} + \boldsymbol{\mu}\mathbf{Q}_r\mathbf{H}_2 + \boldsymbol{\mu}\mathbf{G}].\end{aligned}$$

Comparing the left block with that in Eq. (3) gives $\boldsymbol{\mu}\mathbf{Q}_\ell = \mathbf{q}_\ell$; then comparing the right blocks leads to $\mathbf{q}_r\mathbf{H}_2 = \boldsymbol{\mu}\mathbf{Q}_r\mathbf{H}_2 + \boldsymbol{\mu}\mathbf{G}$. The latter identity shows $\boldsymbol{\mu} \in \operatorname{Rel}(\mathbf{H}_2, \mathbf{G})$, and thus $\boldsymbol{\mu} = \boldsymbol{\lambda}\mathbf{P}_2$ for some $\boldsymbol{\lambda} \in \mathbb{K}[x]^{1 \times m}$, hence finally $\mathbf{p} = \boldsymbol{\lambda}\mathbf{P}_2\mathbf{P}_1$.

Concerning the cost of Step **3**, we remark that $m < D$, that $n \leqslant D$ is ensured by $\operatorname{cdeg}(\mathbf{H}) > 0$, and that $\boldsymbol{\delta}_1 + \boldsymbol{\delta}_2 \leqslant D$ according to Lemma 2.3. Furthermore, there are two recursive calls with module dimension about $n/2$, and with $\mathbf{H}_1$ and $\mathbf{H}_2$ that are in Hermite form and have determinant degrees $D_1 = \deg(\det(\mathbf{H}_1))$ and $D_2 = \deg(\det(\mathbf{H}_2))$ such that $D = D_1 + D_2$; and the entries of both $\operatorname{cdeg}(\mathbf{H}_1)$ and $\operatorname{cdeg}(\mathbf{H}_2)$ are all positive.

In particular, the assumptions on the parameters in Propositions 3.6 and 4.3, concerning the computation of the residual at Step **3.f** and of the relation basis when the degrees are known at Step **3.i**, are satisfied. Thus, these steps use $O^{\sim}((m^{\omega-1} + n^{\omega-1})D)$ and $O^{\sim}(m^{\omega-1}D + n^{\omega}D/m)$ operations, respectively. The announced cost bound follows. □

ACKNOWLEDGMENTS

The authors would like to thank Claude-Pierre Jeannerod for the interesting discussions, and Arne Storjohann for his helpful comments on high-order lifting. Vu Thi Xuan gratefully acknowledges financial support provided though the scholarship *Explora Doc* from *Région Rhône-Alpes, France.*

REFERENCES

[1] G. A. Baker and P. R. Graves-Morris. 1996. *Padé Approximants.* Cambridge University Press.

[2] B. Beckermann. 1992. A reliable method for computing M-Padé approximants on arbitrary staircases. *J. Comput. Appl. Math.* 40, 1 (1992), 19–42. DOI: http://dx.doi.org/10.1016/0377-0427(92)90039-Z

[3] B. Beckermann and G. Labahn. 1994. A Uniform Approach for the Fast Computation of Matrix-Type Padé Approximants. *SIAM J. Matrix Anal. Appl.* 15, 3 (July 1994), 804–823. DOI: http://dx.doi.org/10.1137/S0895479892230031

[4] B. Beckermann and G. Labahn. 1997. Recursiveness in matrix rational interpolation problems. *J. Comput. Appl. Math.* 77, 1–2 (1997), 5–34. DOI: http://dx.doi.org/10.1016/S0377-0427(96)00120-3

[5] B. Beckermann, G. Labahn, and G. Villard. 1999. Shifted Normal Forms of Polynomial Matrices. In *ISSAC'99.* ACM, 189–196. DOI: http://dx.doi.org/10.1145/309831.309929

[6] B. Codenotti and G. Lotti. 1989. A fast algorithm for the division of two polynomial matrices. *IEEE Trans. Automat. Control* 34, 4 (Apr 1989), 446–448. DOI: http://dx.doi.org/10.1109/9.28020

[7] D. Coppersmith and S. Winograd. 1990. Matrix multiplication via arithmetic progressions. *J. Symbolic Comput.* 9, 3 (1990), 251–280. DOI: http://dx.doi.org/10.1016/S0747-7171(08)80013-2

[8] D. S. Dummit and R. M. Foote. 2004. *Abstract Algebra.* John Wiley & Sons.

[9] D. Eisenbud. 1995. *Commutative Algebra: with a View Toward Algebraic Geometry.* Springer, New York, Berlin, Heildelberg. DOI: http://dx.doi.org/10.1007/978-1-4612-5350-1

[10] D. Eisenbud. 2005. *The Geometry of Syzygies.* Springer, New York, Berlin, Heildelberg. DOI: http://dx.doi.org/10.1007/b137572

[11] P. Favati and G. Lotti. 1991. Parallel algorithms for matrix polynomial division. *Computers and Mathematics with Applications* 22, 7 (1991), 37–42. DOI: http://dx.doi.org/10.1016/0898-1221(91)90179-8

[12] G. D. Forney, Jr. 1975. Minimal Bases of Rational Vector Spaces, with Applications to Multivariable Linear Systems. *SIAM Journal on Control* 13, 3 (1975), 493–520. DOI: http://dx.doi.org/10.1137/0313029

[13] F. R. Gantmacher. 1959. *The Theory of Matrices.* Chelsea.

[14] J. von zur Gathen and J. Gerhard. 2013. *Modern Computer Algebra (third edition).* Cambridge University Press. i–xiii, 1–795 pages. DOI: http://dx.doi.org/10.1017/CBO9781139856065

[15] P. Giorgi, C.-P. Jeannerod, and G. Villard. 2003. On the complexity of polynomial matrix computations. In *ISSAC'03.* ACM, 135–142. DOI: http://dx.doi.org/10.1145/860854.860889

[16] P. Giorgi and R. Lebreton. 2014. Online Order Basis Algorithm and Its Impact on the Block Wiedemann Algorithm. In *ISSAC'14.* ACM, New York, NY, USA, 202–209. DOI: http://dx.doi.org/10.1145/2608628.2608647

[17] S. Gupta, S. Sarkar, A. Storjohann, and J. Valeriote. 2012. Triangular x-basis decompositions and derandomization of linear algebra algorithms over $K[x]$. *J. Symbolic Comput.* 47, 4 (2012), 422–453. DOI: http://dx.doi.org/10.1016/j.jsc.2011.09.006

[18] S. Gupta and A. Storjohann. 2011. Computing Hermite Forms of Polynomial Matrices. In *ISSAC'11.* ACM, 155–162. DOI: http://dx.doi.org/10.1145/1993886.1993913

[19] C.-P. Jeannerod, V. Neiger, É. Schost, and G. Villard. 2016. Fast computation of minimal interpolation bases in Popov form for arbitrary shifts. In *ISSAC'16.* ACM, 295–302. DOI: http://dx.doi.org/10.1145/2930889.2930928

[20] C.-P. Jeannerod, V. Neiger, É. Schost, and G. Villard. 2017. Computing minimal interpolation bases. *J. Symbolic Comput.* (in press) (2017), ?–? DOI: http://dx.doi.org/10.1016/j.jsc.2016.11.015

[21] T. Kailath. 1980. *Linear Systems.* Prentice-Hall.

[22] G. Labahn, V. Neiger, and W. Zhou. 2016. Fast, deterministic computation of the Hermite normal form and determinant of a polynomial matrix. *In revision* (2016). http://arxiv.org/abs/1607.04176

[23] F. Le Gall. 2014. Powers of Tensors and Fast Matrix Multiplication. In *ISSAC'14.* ACM, 296–303. DOI: http://dx.doi.org/10.1145/2608628.2608664

[24] J. Middeke. 2011. *A computational view on normal forms of matrices of Ore polynomials.* Ph.D. Dissertation. Research Institute for Symbolic Computation (RISC). http://www.risc.jku.at/publications/download/risc_4377/diss.pdf

[25] V. Neiger. 2016. *Bases of relations in one or several variables: fast algorithms and applications.* Ph.D. Dissertation. École Normale Supérieure de Lyon. https://tel.archives-ouvertes.fr/tel-01431413

[26] V. Neiger. 2016. Fast computation of shifted Popov forms of polynomial matrices via systems of modular polynomial equations. In *ISSAC'16.* ACM, 365–372. DOI: http://dx.doi.org/10.1145/2930889.2930936

[27] V. M. Popov. 1972. Invariant Description of Linear, Time-Invariant Controllable Systems. *SIAM Journal on Control* 10, 2 (1972), 252–264. DOI: http://dx.doi.org/10.1137/0310020

[28] J. Rosenkilde and A. Storjohann. 2016. Algorithms for Simultaneous Padé Approximations. In *ISSAC'16.* ACM, New York, NY, USA, 405–412. DOI: http://dx.doi.org/10.1145/2930889.2930933

[29] A. Storjohann. 2003. High-order lifting and integrality certification. *J. Symbolic Comput.* 36, 3-4 (2003), 613–648. DOI: http://dx.doi.org/10.1016/S0747-7171(03)00097-X

[30] A. Storjohann. 2006. Notes on computing minimal approximant bases. In *Challenges in Symbolic Computation Software (Dagstuhl Seminar Proceedings).* http://drops.dagstuhl.de/opus/volltexte/2006/776

[31] M. Van Barel and A. Bultheel. 1991. The computation of non-perfect Padé-Hermite approximants. *Numer. Algorithms* 1, 3 (1991), 285–304. DOI: http://dx.doi.org/10.1007/BF02142327

[32] M. Van Barel and A. Bultheel. 1992. A general module theoretic framework for vector M-Padé and matrix rational interpolation. *Numer. Algorithms* 3 (1992), 451–462. DOI: http://dx.doi.org/10.1007/BF02141952

[33] Qing-Guo Wang and Chun-Hui Zhou. 1986. An efficient division algorithm for polynomial matrices. *IEEE Trans. Automat. Control* 31, 2 (Feb 1986), 165–166. DOI: http://dx.doi.org/10.1109/TAC.1986.1104212

[34] W. Wolovich. 1984. A division algorithm for polynomial matrices. *IEEE Trans. Automat. Control* 29, 7 (Jul 1984), 656–658. DOI: http://dx.doi.org/10.1109/TAC.1984.1103609

[35] Shou-Yuan Zhang and Chi-Tsong Chen. 1983. An algorithm for the division of two polynomial matrices. *IEEE Trans. Automat. Control* 28, 2 (Feb 1983), 238–240. DOI: http://dx.doi.org/10.1109/TAC.1983.1103203

[36] W. Zhou. 2012. *Fast Order Basis and Kernel Basis Computation and Related Problems.* Ph.D. Dissertation. University of Waterloo.

[37] W. Zhou and G. Labahn. 2012. Efficient Algorithms for Order Basis Computation. *J. Symbolic Comput.* 47, 7 (2012), 793–819. DOI: http://dx.doi.org/10.1016/j.jsc.2011.12.009