# Computing Popov and Hermite forms of rectangular polynomial matrices

Vincent Neiger, Johan Rosenkilde, Grigory Solomatov

# Computing Popov and Hermite forms of rectangular polynomial matrices

Vincent Neiger*
Univ. Limoges, CNRS, XLIM, UMR 7252
F-87000 Limoges, France
vincent.neiger@unilim.fr

Johan Rosenkilde
Technical University of Denmark
Kgs. Lyngby, Denmark
jsrn@jsrn.dk

Grigory Solomatov
Technical University of Denmark
Kgs. Lyngby, Denmark
grigorys93@gmail.com

## ABSTRACT

We consider the computation of two normal forms for matrices over the univariate polynomials: the Popov form and the Hermite form. For matrices which are square and nonsingular, deterministic algorithms with satisfactory cost bounds are known. Here, we present deterministic, fast algorithms for rectangular input matrices. The obtained cost bound for the Popov form matches the previous best known randomized algorithm, while the cost bound for the Hermite form improves on the previous best known ones by a factor which is at least the largest dimension of the input matrix.

## KEYWORDS

Polynomial matrix; Reduced form; Popov form; Hermite form.

## 1 INTRODUCTION

In this paper we deal with (univariate) polynomial matrices, i.e. matrices in $\mathbb{K}[x]^{m\times n}$ where $\mathbb{K}$ is a field admitting exact computation, typically a finite field. Given such an input matrix whose row space is the real object of interest, one may ask for a "better" basis for the row space, that is, another matrix which has the same row space but also has additional useful properties. Two important normal forms for such bases are the Popov form [20] and the Hermite form [10], whose definitions are recalled in this paper. The Popov form has rows which have the minimal possible degrees, while the Hermite form is in echelon form. A classical generalisation is the *shifted* Popov form of a matrix [1], where one incorporates degree weights on the columns: with zero shift this is the Popov form, while under some extremal shift this becomes the Hermite form [2]. We are interested in the efficient computation of these forms, which has been studied extensively along with the computation of the related but non-unique reduced forms [5, 12] and weak Popov forms [15].

Hereafter, complexity estimates count basic arithmetic operations in $\mathbb{K}$ on an algebraic RAM, and asymptotic cost bounds omit factors that are logarithmic in the input parameters, denoted by $O^{\sim}(\cdot)$. The exponent for matrix multiplication is denoted by $\omega$: two matrices in $\mathbb{K}^{m\times m}$ can be multiplied in $O(n^\omega)$ field operations. As shown in [4], the multiplication of two polynomials in $\mathbb{K}[x]$ of degree at most $d$ is done in $O^{\sim}(d)$ operations, and more generally the multiplication of two polynomial matrices in $\mathbb{K}[x]^{m\times m}$ of degree at most $d$ uses $O^{\sim}(m^\omega d)$ operations.

Consider a square, nonsingular $\mathbf{M} \in \mathbb{K}[x]^{m\times m}$ of degree $d$. For the computation of a reduced form of $\mathbf{M}$, the complexity $O^{\sim}(m^\omega d)$ was first achieved by a Las Vegas algorithm of Giorgi et al. [6]. All the subsequent work mentioned in the next paragraph achieved the same cost bound, which was taken as a target: up to logarithmic factors, it is the same as the cost for multiplying two matrices with dimensions and degree similar to those of $\mathbf{M}$.

The approach of [6] was de-randomized by Gupta et al. [8], while Sarkar and Storjohann [22] showed how to compute the Popov form from a reduced form; combining these results gives a deterministic algorithm for the Popov form. Gupta and Storjohann [7, 9] gave a Las Vegas algorithm for the Hermite form; a Las Vegas method for computing the shifted Popov form for any shift was described in [17]. Then, a deterministic Hermite form algorithm was given by Labahn et al. [13], which was one ingredient in a deterministic algorithm due to Neiger and Vu [18] for the arbitrary shift case.

The Popov form algorithms usually exploit the fact that, by definition, this form has degree at most $d = \deg(\mathbf{M})$. While no similarly strong degree bound holds for shifted Popov forms in general (including the Hermite form), these forms still share a remarkable property in the square, nonsingular case: each entry outside the diagonal has degree less than the entry on the diagonal in the same column. These diagonal entries are called *pivots* [12]. Furthermore, their degrees sum to $\deg(\det(\mathbf{M})) \leq md$, so that these forms can be represented with $O(m^2 d)$ field elements. This is especially helpful in the design of fast algorithms since this provides ways to control the degrees of the manipulated matrices.

These degree constraints still exist but become weaker in the case of rectangular shifted Popov forms, say $m \times n$ with $m < n$. Such a normal form does have $m$ columns containing pivots, whose average degree is at most the degree $d$ of the input matrix $\mathbf{M}$. Yet it also contains $n-m$ columns without pivots, which may all have large degree: up to $\Theta(md)$ in the case of the Hermite form. As a result, a dense representation of the latter form may require $\Omega(m^2 nd)$ field elements, a factor of $m$ larger than for the input matrix $\mathbf{M}$. Take for example any matrix $\mathbf{U} \in \mathbb{K}[x]^{m\times m}$ of degree $d$ which is unimodular, meaning that $\mathbf{U}^{-1}$ has polynomial entries. Then, the Hermite form of $[\mathbf{U} \;\; \mathbf{I}_m \;\; \cdots \;\; \mathbf{I}_m]$ is $[\mathbf{I}_m \;\; \mathbf{U}^{-1} \;\; \cdots \;\; \mathbf{U}^{-1}]$, and all the entries of $\mathbf{U}^{-1}$ may have degree in $\Omega(md)$. On the other hand, the Popov form, having minimal degree, has at most the size $O(mnd)$ of the input $\mathbf{M}$. Thus, unlike in the nonsingular case, one would here set different target costs for the computation of Popov and Hermite forms, such as $O^{\sim}(m^{\omega-1}nd)$ for the former and $O^{\sim}(m^\omega nd)$ for the latter (note that the exponent affects the small dimension).

For a rectangular matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$, Mulders and Storjohann [15] gave an iterative Popov form algorithm which costs $O(rmnd^2)$, where $r$ is the rank of $\mathbf{M}$. Beckermann et al. [3] obtain the shifted Popov form for any shift by computing a basis of the left kernel

*Part of the research leading to this work was conducted while Vincent Neiger was with Technical University of Denmark, Kgs. Lyngby, Denmark, with funding from the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme (FP7/2007-2013) under REA grant agreement number 609405 (COFUNDPostdocDTU).

of $[\mathbf{M}^\mathsf{T} \ \ \mathbf{I}_n]^\mathsf{T}$. This approach also produces a matrix which transforms $\mathbf{M}$ into its normal form and whose degree can be in $\Omega(md)$: efficient algorithms usually avoid computing this transformation. To compute the sought kernel basis, the fastest known method is to compute a shifted Popov approximant basis of the $(m+n) \times n$ matrix above, at an order which depends on the shift. [3] relies on a fraction-free algorithm for the latter computation, and hence lends itself well to cases where $\mathbb{K}$ is not finite. In our context, following this approach with the fastest known approximant basis algorithm [11] yields the cost bounds $O^\sim((m+n)^{\omega-1}nmd)$ for the Popov form and $O^\sim((m+n)^{\omega-1}n^2md)$ for the Hermite form. For the latter this is the fastest existing algorithm, to the best of our knowledge.

For $\mathbf{M}$ with full rank and $m \leq n$, Sarkar [21] showed a Las Vegas algorithm for the Popov form achieving the cost $O^\sim(m^{\omega-1}nd)$. This algorithm uses random column operations to compress $\mathbf{M}$ into an $m \times m$ matrix, which is then transformed into a reduced form. Applying the same transformation on $\mathbf{M}$ yields a reduced form with high probability, and from there the Popov form can be obtained. Lowering this cost further seems difficult, as indicated in the square case by the reduction from polynomial matrix multiplication to Popov form computation described in [22, Thm. 22].

For a matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ which is rank-deficient or has $m > n$, the computation of a basis of the row space of $\mathbf{M}$ was handled by Zhou and Labahn [28] with cost $O^\sim(m^{\omega-1}(m+n)d)$. Their algorithm is deterministic, and the output basis $\mathbf{B} \in \mathbb{K}[x]^{r\times n}$ has degree at most $d$. This may be used as a preliminary step: the normal form of $\mathbf{M}$ is also that of $\mathbf{B}$, and the latter has full rank with $r \leq n$.

We stress that, from a rectangular matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$, it seems difficult in general to predict which columns of its shifted Popov form will be pivot-free. For this reason, there seems to be no obvious deterministic reduction from the rectangular case to the square case, even when $n$ is only slightly larger than $m$. Sarkar's algorithm is a Las Vegas reduction, *compressing* the matrix to a nonsingular $m \times m$ matrix; another Las Vegas reduction consists in *completing* the matrix to a nonsingular $n \times n$ matrix (see Section 3).

In the nonsingular case, exploiting information on the pivots has led to algorithmic improvements for normal form algorithms [9, 11, 13, 22]. Following this, we put our effort into two computational tasks: finding the location of the pivots in the normal form (the *pivot support*), and using this knowledge to compute this form.

Our first contribution is to show how to efficiently determine the pivot support of the Popov form of $\mathbf{M}$. For this, we use a factorization from [27], writing $\mathbf{M}$ as a column basis multiplied by some kernel basis (Section 4.1). While this is mainly efficient for $n \in O(m)$, using this method repeatedly on well-chosen submatrices of $\mathbf{M}$ with about $2m$ columns allows us to find the pivot support using $O^\sim(m^{\omega-1}nd)$ operations for any dimensions $m \leq n$ (Section 4.2).

In our second main contribution, we consider the shifted Popov form of $\mathbf{M}$, for any shift. We show that once its pivot support is known, then this form can be computed efficiently (Section 6 and Proposition 6.1). In particular, combining both contributions yields a fast and deterministic Popov form algorithm.

Theorem 1.1. *For a matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ of degree at most $d$ and with $m \leq n$, there is a deterministic algorithm which computes the Popov form of $\mathbf{M}$ using $O^\sim(m^{\omega-1}nd)$ operations in $\mathbb{K}$.*

The second contribution may of course be useful in situations where the pivot support is known for some reason. Yet, there are even general cases where it can be computed efficiently, namely when the shift has very unbalanced entries. This is typically the case of the Hermite form, for which the pivot support coincides with the column rank profile of $\mathbf{M}$. The latter can be efficiently obtained via an algorithm due to Zhou [25, Sec. 11], based on the kernel basis algorithm from [29]. This leads us to the next result.

Theorem 1.2. *Let $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ with full rank and $m < n$. There is a deterministic algorithm which computes the Hermite form of $\mathbf{M}$ using $O^\sim(m^{\omega-1}n\delta)$ operations in $\mathbb{K}$, where $\delta$ is any integer greater than the minimum of the sum of column degrees of $\mathbf{M}$ and of the sum of row degrees of $\mathbf{M}$.*

Using this quantity $\delta$, the mentioned cost for the kernel basis approach of [3] becomes $O^\sim((m+n)^{\omega-1}n^2\delta)$. Thus, when $n \in O(m)$ the cost in the above theorem already gains a factor $n$; this factor only increases when $n$ becomes large compared to $m$.

## 2 PRELIMINARIES

### 2.1 Basic notation

If $\mathbf{M}$ is an $m \times n$ matrix and $1 \leq j \leq n$, we denote by $\mathbf{M}_{*,j}$ the $j$th column of $\mathbf{M}$. If $J \subseteq \{1, \ldots, n\}$ is a set of column indices, $\mathbf{M}_{*,J}$ is the submatrix of $\mathbf{M}$ formed by the columns at the indices in $J$. We use analogous row-wise notation. Similarly, for a tuple $\boldsymbol{t} \in \mathbb{Z}^n$, then $\boldsymbol{t}_J$ is the subtuple of $\boldsymbol{t}$ formed by the entries at the indices in $J$.

When adding a constant to an integer tuple, for example $\boldsymbol{t} + 1$ for some $\boldsymbol{t} = (t_1, \ldots, t_m) \in \mathbb{Z}^m$, we really mean $(t_1 + 1, \ldots, t_m + 1)$; when comparing a tuple to a constant, for example $\boldsymbol{t} \leq 1$, we mean $\max(\boldsymbol{t}) \leq 1$. Two tuples of the same length will always be compared entrywise: $\boldsymbol{s} \leq \boldsymbol{t}$ stands for $s_i \leq t_i$ for all $i$. We use the notation $\mathrm{amp}(\boldsymbol{t}) = \max(\boldsymbol{t}) - \min(\boldsymbol{t})$, and $|\boldsymbol{t}| = t_1 + \ldots + t_m$ (note that the latter will mostly be used when $\boldsymbol{t}$ has nonnegative entries).

For a given nonnegative integer tuple $\boldsymbol{t} = (t_1, \ldots, t_m) \in \mathbb{Z}^m_{\geq 0}$, we denote by $\mathbf{x}^{\boldsymbol{t}}$ the diagonal matrix with entries $x^{t_1}, \ldots, x^{t_m}$.

### 2.2 Row spaces, kernels, and approximants

For a matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$, its *row space* is the $\mathbb{K}[x]$-module generated by its rows, that is, $\{\boldsymbol{\lambda}\mathbf{M}, \boldsymbol{\lambda} \in \mathbb{K}[x]^{1\times m}\}$. Then, a matrix $\mathbf{B} \in \mathbb{K}[x]^{r\times n}$ is a *row basis* of $\mathbf{M}$ if its rows form a basis of the row space of $\mathbf{M}$, in which case $r$ is the rank of $\mathbf{M}$.

The *left kernel* of $\mathbf{M}$ is the $\mathbb{K}[x]$-module $\{\mathbf{p} \in \mathbb{K}[x]^{1\times m} \mid \mathbf{pM} = \mathbf{0}\}$. A matrix $\mathbf{K} \in \mathbb{K}[x]^{k\times m}$ is a *left kernel basis of* $\mathbf{M}$ if its rows form a basis of this kernel, in which case $k = m - r$. Similarly, a *right* kernel basis of $\mathbf{M}$ is a matrix $\mathbf{K} \in \mathbb{K}[x]^{n\times(n-r)}$ whose *columns* form a basis of the right kernel of $\mathbf{M}$.

Given $\boldsymbol{d} = (d_1, \ldots, d_n) \in \mathbb{Z}^n_{>0}$, the set of *approximants for* $\mathbf{M}$ *at order* $\boldsymbol{d}$ is the $\mathbb{K}[x]$-module of rank $m$ defined as

$$\mathcal{A}_{\boldsymbol{d}}(\mathbf{M}) = \{\mathbf{p} \in \mathbb{K}[x]^{1\times m} \mid \mathbf{pM} = \mathbf{0} \bmod \mathbf{x}^{\boldsymbol{d}}\}.$$

The identity $\mathbf{pM} = \mathbf{0} \bmod \mathbf{x}^{\boldsymbol{d}}$ means that the $j$th entry of the vector $\mathbf{pM} \in \mathbb{K}[x]^{1\times n}$ is divisible by $x^{d_j}$, for all $j$.

Two $m \times n$ matrices $\mathbf{M}_1, \mathbf{M}_2$ have the same row space if and only if they are *unimodularly equivalent*, that is, there is a unimodular (invertible over $\mathbb{K}[x]$) matrix $\mathbf{U} \in \mathbb{K}[x]^{m\times m}$ such that $\mathbf{UM}_1 = \mathbf{M}_2$. For $\mathbf{M}_3 \in \mathbb{K}[x]^{r\times n}$ with $r \leq m$, then $\mathbf{M}_1$ and $\mathbf{M}_3$ have the same row

space exactly when $\mathbf{M}_3$ padded with $m-r$ zero rows is unimodularly equivalent to $\mathbf{M}_1$.

## 2.3 Row degrees and reduced forms

For a matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$, we denote by $\operatorname{rdeg}(\mathbf{M})$ the tuple of the degrees of its rows, that is, $(\deg(\mathbf{M}_{1,*}), \ldots, \deg(\mathbf{M}_{m,*}))$.

If $\mathbf{M}$ has no zero row, the *(row-wise) leading matrix* of $\mathbf{M}$, denoted by $\operatorname{lm}(\mathbf{M})$, is the matrix in $\mathbb{K}^{m\times n}$ whose entry $i,j$ is equal to the coefficient of degree $\deg(\mathbf{M}_{i,*})$ of the entry $i,j$ of $\mathbf{M}$.

For a matrix $\mathbf{R} \in \mathbb{K}[x]^{m\times n}$ with no zero row and $m \le n$, we say that $\mathbf{R}$ is *(row) reduced* if $\operatorname{lm}(\mathbf{R})$ has full rank. Thus, here a reduced matrix must have full rank (and no zero row), as in [5]. For more details about reduced matrices, we refer the reader to [3, 5, 12, 24]. In particular, we have the following characterizing properties:

- *Predictable degree property [5] [12, Thm. 6.3-13]:* we have
$$\deg(\boldsymbol{\lambda}\mathbf{R}) = \max\{\deg(\lambda_i) + \operatorname{rdeg}(\mathbf{R}_{i,*}),\ 1 \le i \le m\}$$
for any vector $\boldsymbol{\lambda} = [\lambda_i]_i \in \mathbb{K}[x]^{1\times m}$.
- *Minimality of the sum of row degrees [5]:* for any nonsingular matrix $\mathbf{U} \in \mathbb{K}[x]^{m\times m}$, we have $|\operatorname{rdeg}(\mathbf{UR})| \ge |\operatorname{rdeg}(\mathbf{R})|$.
- *Minimality of the tuple of row degrees [25, Sec. 2.7]:* for any nonsingular matrix $\mathbf{U} \in \mathbb{K}[x]^{m\times m}$, we have $\boldsymbol{s} \le \boldsymbol{t}$ where the tuples $\boldsymbol{s}$ and $\boldsymbol{t}$ are the row degrees of $\mathbf{R}$ and of $\mathbf{UR}$ sorted in nondecreasing order, respectively.

From the last item, it follows that two unimodularly equivalent reduced matrices have the same row degree up to permutation.

For a matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$, we call *reduced form of* $\mathbf{M}$ any reduced matrix $\mathbf{R} \in \mathbb{K}[x]^{r\times n}$ which is a row basis of $\mathbf{M}$. The third item above shows that $\deg(\mathbf{R}) \le \deg(\mathbf{M})$.

## 2.4 Pivots and Popov forms

For a nonzero vector $\mathbf{p} = [p_j]_j \in \mathbb{K}[x]^{1\times m}$, the *pivot index* of $\mathbf{p}$ is the largest index $j$ such that $\deg(p_j) = \deg(\mathbf{p})$ [12, Sec. 6.7.2]. In this case we call $p_j$ the *pivot entry* of $\mathbf{p}$. For the zero vector, we define its degree to be $-\infty$ and its pivot index to be 0. Further, the *pivot index* of a matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ is the tuple $(j_1, \ldots, j_m) \in \mathbb{Z}^m_{\ge 0}$ such that $j_i$ is the pivot index of $\mathbf{M}_{i,*}$. Note that we will only use the word "pivot" in this row-wise sense.

A matrix $\mathbf{P} \in \mathbb{K}[x]^{m\times n}$ is in *weak Popov form* if it has no zero row and the entries of the pivot index of $\mathbf{P}$ are all distinct [15]; a weak Popov form is further called *ordered* if its pivot index is in (strictly) increasing order. A weak Popov matrix is also reduced.

The (ordered) weak Popov form is not canonical: a given row space may have many (ordered) weak Popov forms. The Popov form adds a normalization property, yielding a canonical form; we use the definition from [2, Def. 3.3]:

A matrix $\mathbf{P} \in \mathbb{K}[x]^{m\times n}$ is in *Popov form* if it is in ordered weak Popov form, the corresponding pivot entries are monic, and in each column of $\mathbf{P}$ which contains a (row-wise) pivot the other entries have degree less than this pivot entry.

For a matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ of rank $r$, there exists a unique $\mathbf{P} \in \mathbb{K}[x]^{r\times n}$ which is in Popov form and has the same row space as $\mathbf{M}$ [3, Thm. 2.7]. We call $\mathbf{P}$ *the Popov form of* $\mathbf{M}$. For a more detailed treatment of Popov forms, see [2, 3, 12].

For example, consider the unimodularly equivalent matrices
$$\begin{bmatrix} x^2 & x+1 & 2 \\ 2x+2 & 2x & 2 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} x^2-x-1 & 1 & 1 \\ x+1 & x & 1 \end{bmatrix},$$
defined over $\mathbb{F}_7[x]$; the first one is in $\mathbf{0}$-weak Popov form and the second one is its $\mathbf{0}$-Popov form. Note that any deterministic rule for ordering the rows would lead to a canonical form; we use that of [2, 3], while that of [12, 15] sorts the rows by degrees and would consider the second matrix not to be normalized.

Going back to the general case, we denote by $\boldsymbol{\pi}(\mathbf{M}) \in \mathbb{Z}^r_{>0}$ the pivot index of the Popov form of $\mathbf{M}$, called the *pivot support* of $\mathbf{M}$. In most cases, $\boldsymbol{\pi}(\mathbf{M})$ differs from the pivot index of $\mathbf{M}$. We have the following important properties:

- The pivot index of $\mathbf{M}$ is equal to its pivot support $\boldsymbol{\pi}(\mathbf{M})$ if and only if $\mathbf{M}$ is in ordered weak Popov form.
- For any nonzero $\boldsymbol{\lambda} \in \mathbb{K}[x]^{1\times m}$, the pivot index of $\boldsymbol{\lambda}\mathbf{M}$ appears in the pivot support $\boldsymbol{\pi}(\mathbf{M})$; in particular each nonzero entry of the pivot index of $\mathbf{M}$ is in $\boldsymbol{\pi}(\mathbf{M})$.

For the first item, we refer to [3, Sec. 2] (in this reference, the set formed by the entries of the pivot support is called "pivot set" and ordered weak Popov forms are called quasi-Popov forms). The second item is a simple extension of the predictable degree property (see for example [16, Lem. 1.17] for a proof).

## 2.5 Computational Tools

We will rely on the following result from [29, Cor. 4.6 and Thm. 3.4] about the computation of kernel bases in reduced form. Note that a matrix is *column reduced* if its transpose is reduced.

Theorem 2.1 ([29]). *There is an algorithm* MinimalKernelBasis *which, given a matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ with $m \le n$, returns a right kernel basis $\mathbf{K} \in \mathbb{K}[x]^{m\times(n-r)}$ of $\mathbf{M}$ in column reduced form using*
$$\tilde{O}\left(n^\omega \lceil m \deg(\mathbf{M})/n \rceil\right) \subseteq \tilde{O}\left(n^\omega \deg(\mathbf{M})\right)$$
*operations in $\mathbb{K}$. Furthermore, $|\operatorname{cdeg}(\mathbf{K})| \le r \deg(\mathbf{M})$.*

For the computation of normal forms of square, nonsingular matrices, we use the following result (shifts will be introduced in Section 5; setting $\boldsymbol{s} = 0$ corresponds to the unshifted case).

Theorem 2.2 ([18]). *There is an algorithm* NonsingularPopov *which, given a nonsingular matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times m}$ and a shift $\boldsymbol{s} \in \mathbb{Z}^m$, returns the $\boldsymbol{s}$-Popov form of $\mathbf{M}$ using*
$$\tilde{O}\left(m^\omega \lceil |\operatorname{rdeg}(\mathbf{M})|/m \rceil\right) \subseteq \tilde{O}\left(m^\omega \deg(\mathbf{M})\right)$$
*operations in $\mathbb{K}$.*

This is [18, Thm. 1.3] with a minor modification: we have replaced the "generic determinant bound" measure by the upper bound of sum of row degrees, since this is sufficient for our needs here.

# 3 POPOV FORM VIA COMPLETION INTO A SQUARE AND NONSINGULAR MATRIX

We now present a new Las Vegas algorithm for computing the (unshifted) Popov form $\mathbf{P}$ of a rectangular matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ with full rank and $m < n$, relying on algorithms for the case of square, nonsingular matrices. We obtain the satisfactory cost $\tilde{O}(m^\omega \deg(\mathbf{M}))$ in the case $n \in O(m)$. While this was also obtained by the Las Vegas algorithm of Sarkar [21], ours has the advantage of

being faster if the *average* row degree of $\mathbf{M}$ is significantly smaller than $m \deg(\mathbf{M})$.

The idea is to find a matrix $\mathbf{C} \in \mathbb{K}[x]^{(n-m)\times n}$ such that the Popov form of $[\mathbf{M}^\mathsf{T} \ \ \mathbf{C}^\mathsf{T}]^\mathsf{T}$ contains $\mathbf{P}$ as an identifiable subset of its rows. We show that if $\mathbf{C}$ is drawn randomly of sufficiently high degree, then this is true with high probability. Formally:

*Definition 3.1.* Let $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ have full rank with $m < n$ and let $\mathbf{P} \in \mathbb{K}[x]^{m\times n}$ be the Popov form of $\mathbf{M}$. A *completion of* $\mathbf{M}$ is any matrix $\mathbf{C} \in \mathbb{K}[x]^{(n-m)\times n}$ such that:

$$\min(\mathrm{rdeg}(\mathbf{C})) > \deg(\mathbf{P}) \text{ and } \begin{bmatrix}\mathbf{P}\\ \mathbf{C}\end{bmatrix} \text{ is row reduced .}$$

The next lemma shows that: 1) if $\mathbf{C}$ is a completion, then $\mathbf{P}$ will appear as a submatrix of the Popov form of $[\mathbf{M}^\mathsf{T} \ \ \mathbf{C}^\mathsf{T}]^\mathsf{T}$; and 2) we can easily check from that Popov form whether $\mathbf{C}$ is a completion or not. The latter is essential for a Las Vegas algorithm.

Lemma 3.2. *Let $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ have full rank with $m < n$ with Popov form $\mathbf{P}$, and let $\mathbf{C} \in \mathbb{K}[x]^{(n-m)\times n}$ be such that $[\mathbf{M}^\mathsf{T} \ \ \mathbf{C}^\mathsf{T}]^\mathsf{T}$ has full rank and $\min(\mathrm{rdeg}(\mathbf{C})) > \deg(\mathbf{P})$. Then, $\mathbf{C}$ is a completion of $\mathbf{M}$ if and only if $\mathrm{rdeg}(\hat{\mathbf{P}})$ contains a permutation of $\mathrm{rdeg}(\mathbf{C})$, where $\hat{\mathbf{P}}$ is the Popov form of $[\mathbf{M}^\mathsf{T} \ \ \mathbf{C}^\mathsf{T}]^\mathsf{T}$. In this case, $\mathbf{P}$ is the submatrix of $\hat{\mathbf{P}}$ formed by its rows of degree less than $\min(\mathrm{rdeg}(\mathbf{C}))$.*

Proof. First, we assume that $\mathbf{C}$ is a completion of $\mathbf{M}$. Then $[\mathbf{P}^\mathsf{T} \ \ \mathbf{C}^\mathsf{T}]^\mathsf{T}$ is reduced, and therefore it has the same row degree as its Popov form $\hat{\mathbf{P}}$ up to permutation. Hence, in particular, $\mathrm{rdeg}(\hat{\mathbf{P}})$ contains a permutation of $\mathrm{rdeg}(\mathbf{C})$.

Now, we assume that $\mathrm{rdeg}(\hat{\mathbf{P}})$ contains a permutation of $\mathrm{rdeg}(\mathbf{C})$ and our goal is to show that $[\mathbf{P}^\mathsf{T} \ \ \mathbf{C}^\mathsf{T}]^\mathsf{T}$ is reduced and $\hat{\mathbf{P}}$ contains $\mathbf{P}$ as a submatrix. Write $\hat{\mathbf{P}}$ as $[\hat{\mathbf{P}}_1^\mathsf{T} \ \ \hat{\mathbf{P}}_2^\mathsf{T}]^\mathsf{T}$ up to row permutations, where $\hat{\mathbf{P}}_1$ are the rows of degree less than $\min(\mathrm{rdeg}(\mathbf{C}))$. By assumption, $\hat{\mathbf{P}}_2$ must have at least $n-m$ rows and $\hat{\mathbf{P}}_1$ at most $m$. Then there is a unimodular transformation such that

$$\begin{bmatrix}\mathbf{U}_{11} & \mathbf{U}_{12}\\ \mathbf{U}_{21} & \mathbf{U}_{22}\end{bmatrix}\begin{bmatrix}\hat{\mathbf{P}}_1\\ \hat{\mathbf{P}}_2\end{bmatrix} = \begin{bmatrix}\mathbf{P}\\ \mathbf{C}\end{bmatrix}, \tag{1}$$

By the predictable degree property then $\mathbf{U}_{12} = 0$, so since $\mathbf{P}$ has full rank $m$, then $\hat{\mathbf{P}}_1$ must have exactly $m$ rows and $\mathbf{U}_{11}$ is unimodular. Therefore $\hat{\mathbf{P}}_1 = \mathbf{P}$ since both matrices are in Popov form. But then $\mathrm{rdeg}(\hat{\mathbf{P}})$ must be a permutation of $(\mathrm{rdeg}(\mathbf{P}), \mathrm{rdeg}(\mathbf{C}))$ as sought. □

Lemma 3.3. *Let $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ have full rank with $m < n$ and $d = \deg(\mathbf{M})$. Let $S \subseteq \mathbb{K}$ be a finite subset of cardinality $q$ and let $\mathbf{L} \in \mathbb{K}^{(n-m)\times n}$ with entries chosen uniformly at random from $S$. Then $x^{d+1}\mathbf{L}$ is a completion of $\mathbf{M}$ with probability at least $\prod_{i=1}^{n-m}(1-q^{-i})$ if $\mathbb{K}$ is finite and $S = \mathbb{K}$, and at least $1 - \frac{n-m}{q}$ otherwise.*

Proof. We first note that for $x^{d+1}\mathbf{L}$ to be a completion of $\mathbf{M}$, it is enough that the matrix

$$\mathrm{lm}\left(\begin{bmatrix}\mathbf{P}\\ \mathbf{C}\end{bmatrix}\right) = \begin{bmatrix}\mathrm{lm}(\mathbf{P})\\ \mathrm{lm}(\mathbf{C})\end{bmatrix} = \begin{bmatrix}\mathrm{lm}(\mathbf{P})\\ \mathbf{L}\end{bmatrix} \in \mathbb{K}^{n\times n}$$

be invertible. Indeed, this implies first that $[\mathbf{P}^\mathsf{T} \ \ \mathbf{C}^\mathsf{T}]^\mathsf{T}$ is reduced; and second, that $\mathbf{C}$ has no zero row, hence $\mathrm{rdeg}(\mathbf{C}) = (d+1, \ldots, d+1)$ and $\min(\mathrm{rdeg}(\mathbf{C})) = d+1 > \deg(\mathbf{M})$.

In the case of a finite field $\mathbb{K}$ with $q$ elements, the probability that the above matrix is invertible is $\prod_{i=1}^{n-m}(1-q^{-i})$. If $\mathbb{K}$ is infinite or of cardinality $\geq q$, the Schwartz-Zippel lemma implies that the probability that the above matrix is singular is at most $(n-m)/q$. □

Thus, if $\mathbb{K}$ is infinite, it is sufficient to take $S$ of cardinality at least $2(n-m)$ to ensure that $x^{d+1}\mathbf{L}$ is a completion with probability at least $1/2$. On the other hand, if $\mathbb{K}$ is finite of cardinality $q$, we have the following bounds on the probability:

$$\prod_{i=1}^{n-m}(1-q^{-i}) > \begin{cases} 0.28 & \text{if } q = 2,\\ 0.55 & \text{if } q = 3,\\ 0.75 & \text{if } q > 5.\end{cases}$$

In Algorithm 1, we first test the nonsingularity of $\mathbf{N} = [\mathbf{M}^\mathsf{T} \ \ \mathbf{C}^\mathsf{T}]^\mathsf{T}$ before computing $\hat{\mathbf{P}}$, since the fastest known Popov form algorithms in the square case do not support singular matrices. Over a field with at least $2n\deg(\mathbf{N}) + 1$ elements, a simple Monte Carlo test for this is to evaluate the polynomial matrix at a random $\alpha \in \mathbb{K}$ and testing the resulting scalar matrix for nonsingularity; this falsely reports singularity only if $\det(\mathbf{N})$ is divisible by $(x-\alpha)$. Alternatively, a deterministic check is as follows. First, apply the partial linearization of [8, Sec. 6], yielding a matrix $\overline{\mathbf{N}} \in \mathbb{K}[x]^{\overline{n}\times\overline{n}}$ such that $\overline{\mathbf{N}}$ is nonsingular if and only if $\mathbf{N}$ is nonsingular; $\overline{n} \in O(n)$; and $\deg(\overline{\mathbf{N}}) \leq \lceil|\mathrm{rdeg}(\mathbf{N})|/n\rceil$. This does not involve arithmetic operations. Since $\overline{\mathbf{N}}$ is nonsingular if and only if its kernel is trivial, compute its minimal basis kernel using the algorithm from [26] using $O^\sim(n^\omega \deg(\overline{\mathbf{N}})) \subseteq O^\sim(n^\omega \lceil|\mathrm{rdeg}(\mathbf{N})|/n\rceil)$ operations in $\mathbb{K}$. Instead of computing a kernel basis, one could also test the nonsingularity of $\overline{\mathbf{N}}$ using algorithms from [8], as explained in [21, p. 24].

*Algorithm 1:* RandomCompletionPopov
*Input:* matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ with full rank and $m < n$. Set $S \subseteq \mathbb{K}$ of cardinality $q$.
*Output:* the Popov form of $\mathbf{M}$, or *failure*.

1. $\mathbf{L} \leftarrow$ matrix in $\mathbb{K}^{(n-m)\times n}$ with entries chosen uniformly and independently at random from $S$.
2. $\mathbf{C} \leftarrow x^{\deg(\mathbf{M})+1}\mathbf{L}$
3. *If* $[\mathbf{M}^\mathsf{T} \ \ \mathbf{C}^\mathsf{T}]^\mathsf{T}$ is singular *then return failure*
4. $\hat{\mathbf{P}} \leftarrow$ NonsingularPopov($[\mathbf{M}^\mathsf{T} \ \ \mathbf{C}^\mathsf{T}]^\mathsf{T}$)
5. *If* $\mathrm{rdeg}(\hat{\mathbf{P}})$ does not contain a permutation of $\mathrm{rdeg}(\mathbf{C})$ *then return failure*
6. *Return* the submatrix of $\hat{\mathbf{P}}$ formed by its rows of degree less than $\min(\mathrm{rdeg}(\mathbf{C}))$

Proposition 3.4. *Algorithm 1 is correct and the probability that a failure is reported at Step 3 or Step 5 is as indicated in Lemma 3.3. If* NonsingularPopov *is the algorithm of [18], Algorithm 1 uses*

$$\tilde{O}\left(n^\omega\left\lceil\frac{|\mathrm{rdeg}(\mathbf{M})| + (n-m)\deg(\mathbf{M})}{n}\right\rceil\right) \subseteq \tilde{O}\left(n^\omega \deg(\mathbf{M})\right)$$

*operations in $\mathbb{K}$.*

Indeed, from Theorem 2.2, Step 4 uses $O^\sim(n^\omega\lceil\Delta/n\rceil)$ operations where $\Delta = |\mathrm{rdeg}([\mathbf{M}^\mathsf{T} \ \ \mathbf{C}^\mathsf{T}]^\mathsf{T})| = |\mathrm{rdeg}(\mathbf{M})| + (n-m)(\deg(\mathbf{M})+1)$.

While other Popov form algorithms could be used, that of [18] allows us to take into account the average row degree of $\mathbf{M}$. Indeed, if $|\mathrm{rdeg}(\mathbf{M})| \ll m\deg(\mathbf{M})$ and $n - m \ll n$, the cost bound above is asymptotically better than $O^\sim(n^\omega \deg(\mathbf{M}))$.

*Remark 1:* As we mentioned in Section 2.4, the pivot index of $\mathbf{M}$ is a subset of $\boldsymbol{\pi}(\mathbf{M})$. Therefore, one can let $\mathbf{L}$ be zero at all columns where $\mathbf{M}$ has a pivot, or indices one otherwise knows appear in $\boldsymbol{\pi}(\mathbf{M})$. If $\mathbf{M}$ has uneven degrees (e.g. if $\mathbf{M} = \mathbf{M}'\mathbf{x}^{\boldsymbol{s}}$ for some shift $\boldsymbol{s}$, see Section 5.1), then this can be particularly worthwhile. In the case where for some reason we know $\boldsymbol{\pi}(\mathbf{M})$, then $\mathbf{L}$ can simply be taken such that $\mathbf{L}_{*,\{1,\ldots,n\}\setminus\boldsymbol{\pi}(\mathbf{M})}$ is the identity matrix. In that case, Algorithm 1 becomes deterministic.

# 4 COMPUTING THE PIVOT SUPPORT

We now consider a matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ with $m < n$, possibly rank-deficient, and we focus on the computation of its pivot support $\boldsymbol{\pi}(\mathbf{M})$. In Section 4.1, we give a deterministic algorithm which is efficient when $n \in O(m)$. In Section 4.2 we explain how this can be used iteratively to efficiently find the pivot support when $m \ll n$.

## 4.1 Deterministic pivot support computation via column basis factorization

We will compute $\boldsymbol{\pi}(\mathbf{M})$ by inspecting the left kernel space of a right kernel of $\mathbf{M}$: unlike the case of matrices over fields, this does not just yield the row space of $\mathbf{M}$, but rather a superset of it: consider for example the matrix $\mathrm{diag}(x,\ldots,x)$ which has an empty right kernel, whose left kernel is therefore spanned by the identity matrix. We rely on the following factorization:

Lemma 4.1 ([27, Sec. 3]). *Let $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ have rank $r \in \mathbb{Z}_{>0}$, let $\mathbf{K} \in \mathbb{K}[x]^{n\times(n-r)}$ be a right kernel basis of $\mathbf{M}$, and let $\mathbf{F} \in \mathbb{K}[x]^{r\times n}$ be a left kernel basis of $\mathbf{K}$. Then, we have $\mathbf{M} = \mathbf{C}\mathbf{F}$ for some column basis $\mathbf{C} \in \mathbb{K}[x]^{m\times r}$ of $\mathbf{M}$.*

More precisely, we will use the following consequence.

Lemma 4.2. *The matrices $\mathbf{M}$ and $\mathbf{F}$ in Lemma 4.1 have the same pivot support, that is, $\boldsymbol{\pi}(\mathbf{M}) = \boldsymbol{\pi}(\mathbf{F})$.*

Proof. Since $\mathbf{M} = \mathbf{C}\mathbf{F}$, the row space of $\mathbf{M}$ is contained in that of $\mathbf{F}$. Hence, by the properties at the end of Section 2.4, $\boldsymbol{\pi}(\mathbf{M}) \subseteq \boldsymbol{\pi}(\mathbf{F})$ as sets. But since $\mathbf{M}$ and $\mathbf{F}$ both have rank $r$, both pivot supports have exactly $r$ different elements, and must be equal. □

We will read off $\boldsymbol{\pi}(\mathbf{F})$ from $\mathbf{F}$ by ensuring that this matrix is in ordered weak Popov form. First, we obtain a column reduced right kernel basis $\mathbf{K}$ of $\mathbf{M}$ using MinimalKernelBasis (see Theorem 2.1). However, the degree profile of $\mathbf{K}$ prevents us from using the same algorithm to compute a left kernel basis $\mathbf{F}$ efficiently, since the average row degree of $\mathbf{K}$ could be as large as $r\deg(\mathbf{M})$. To circumvent this issue, we combine the observations that $\deg(\mathbf{F})$ is bounded and that $\mathbf{K}$ has small average *column* degree to conclude that $\mathbf{F}$ can be efficiently obtained via an approximant basis (see Section 2).

Lemma 4.3. *Let $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ have rank $r \in \mathbb{Z}_{>0}$ and let $\mathbf{K} \in \mathbb{K}[x]^{n\times(n-r)}$ be a right kernel basis of $\mathbf{M}$. Then, any left kernel basis of $\mathbf{K}$ which is in reduced form must have degree at most $d = \deg(\mathbf{M})$. As a consequence, if $\hat{\mathbf{P}} \in \mathbb{K}[x]^{n\times n}$ is a reduced basis of $\mathcal{A}_{\boldsymbol{d}}(\mathbf{K})$, where $\boldsymbol{d} = \mathrm{cdeg}(\mathbf{K}) + d + 1 \in \mathbb{Z}^{n-r}$, then the submatrix $\mathbf{P}$ of $\hat{\mathbf{P}}$ formed by its rows of degree at most $d$ is a reduced left kernel basis of $\mathbf{K}$.*

Proof. Let $\mathbf{F} \in \mathbb{K}[x]^{r\times n}$ be a left kernel basis of $\mathbf{K}$ in reduced form. By Lemma 4.1, $\mathbf{M} = \mathbf{C}\mathbf{F}$ for some matrix $\mathbf{C} \in \mathbb{K}[x]^{m\times r}$. Then, the predictable degree property implies that $\deg(\mathbf{F}) \le \deg(\mathbf{C}\mathbf{F}) = d$.

For the second claim (which is a particular case of [27, Lem. 4.2]), note that $\mathbf{P}$ is reduced as a subset of the rows of a reduced matrix. Besides, $\mathrm{cdeg}(\mathbf{P}\mathbf{K}) < \boldsymbol{d}$ by construction, hence $\mathbf{P}\mathbf{K} = \mathbf{0} \bmod \mathbf{x}^{\boldsymbol{d}}$ implies $\mathbf{P}\mathbf{K} = \mathbf{0}$. It remains to show that $\mathbf{P}$ generates the left kernel of $\mathbf{K}$: any vector of degree at most $d$ in this kernel is in particular in $\mathcal{A}_{\boldsymbol{d}}(\mathbf{K})$ and therefore is a combination of the rows of $\hat{\mathbf{P}}$; using the predictable degree property, we obtain that this combination only involves rows from the submatrix $\mathbf{P}$. □

If we compute $\hat{\mathbf{P}}$ in ordered weak Popov form, then the submatrix $\mathbf{P}$ is in ordered weak Popov form as well, and therefore $\boldsymbol{\pi}(\mathbf{M})$ can be directly read off. The computation of an approximant basis in ordered weak Popov form can be done via the algorithm of [11], which returns one in Popov form.

*Algorithm 2:* PivotSupportViaFactor
*Input:* matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ with $m \le n$.
*Output:* the pivot support $\boldsymbol{\pi}(\mathbf{M})$ of $\mathbf{M}$.

1. *If* $\mathbf{M} = \mathbf{0}$ *then return* the empty tuple $() \in \mathbb{Z}^0_{>0}$
2. $\mathbf{K} \in \mathbb{K}[x]^{n\times(n-r)} \leftarrow$ MinimalKernelBasis($\mathbf{M}$)
3. $\hat{\mathbf{P}} \in \mathbb{K}[x]^{n\times n} \leftarrow$ ordered weak Popov basis of $\mathcal{A}_{\boldsymbol{d}}(\mathbf{K})$, with $\boldsymbol{d} = \mathrm{cdeg}(\mathbf{K}) + (\deg(\mathbf{M}) + 1) \in \mathbb{Z}^{n-r}$
4. $\mathbf{F} \in \mathbb{K}[x]^{r\times n} \leftarrow$ the rows of $\hat{\mathbf{P}}$ of degree at most $d$
5. *Return* the pivot index of $\mathbf{F}$

Proposition 4.4. *Algorithm 2 is correct and uses $O\tilde{}(n^\omega \deg(\mathbf{M}))$ operations in $\mathbb{K}$.*

Proof. Note that we compute the rank of $\mathbf{M}$ as $r$ by the indirect assignment at Step 2. Besides, $\mathbf{F}$ is in ordered weak Popov form since it is a submatrix formed by rows of $\hat{\mathbf{P}}$ itself in ordered weak Popov form. This implies that $\boldsymbol{\pi}$ is indeed the pivot support of $\mathbf{F}$. Then, the correctness directly follows from Lemmas 4.2 and 4.3.

By Theorem 2.1, Step 2 costs $O\tilde{}(n^\omega d)$, where $d = \deg(\mathbf{M})$, and $|\mathrm{cdeg}(\mathbf{K})| \le rd$. Thus, the sum of the approximation order defined at Step 3 is $|\boldsymbol{d}| = |\mathrm{cdeg}(\mathbf{K})| + (n-r)(d+1) < n(d+1)$. Then, this step uses $O\tilde{}(n^{\omega-1}|\boldsymbol{d}|) \subseteq O\tilde{}(n^\omega d)$ operations [11, Thm. 1.4]. □

Note that in this algorithm we do not require $\mathbf{M}$ to have full rank. The only reason why we assume $m \le n$ is because the cost bound for the computation of a kernel basis at Step 2 is not clear to us in the case $m > n$ (the same assumption is made in [29]).

Here, it seems more difficult to take average degrees into account than in Algorithm 1. While the average degree of the $m$ columns of $\mathbf{M}$ with largest degree could be taken into account by the kernel basis algorithm of [29], it seems that the computation of $\mathbf{F}$ via an approximant basis remains in $O\tilde{}(n^\omega d)$ nevertheless.

## 4.2 The case of wide matrices

In this section we will deal with pivots of submatrices $\mathbf{M}_{*,J}$, where $J = \{j_1 < \ldots < j_k\} \subseteq \{1,\ldots,n\}$. To use column indices of $\mathbf{M}_{*,J}$ in $\mathbf{M}$, we introduce for any such $J$ the operator $\phi_J : \{1,\ldots,k\} \to \{1,\ldots,n\}$ satisfying $\phi_J(i) = j_i$. We abuse notation by applying $\phi_J$ element-wise to tuples, such as in $\phi_J(\boldsymbol{\pi}(\mathbf{M}_{*,J}))$.

The following simple lemma is the crux of the algorithm:

Lemma 4.5. *Let* $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$*, and consider any set of indices* $J \subseteq \{1,\dots,n\}$*. Then* $(\boldsymbol{\pi}(\mathbf{M}) \cap J) \subseteq \phi_J(\boldsymbol{\pi}(\mathbf{M}_{*,J}))$ *with equality whenever* $\boldsymbol{\pi}(\mathbf{M}) \subseteq J$*.*

Proof. If a vector $\mathbf{v} \in \mathbb{K}[x]^{1\times n}$ in the row space of $\mathbf{M}$ has $\boldsymbol{\pi}(\mathbf{v}) \in J$, then also $\boldsymbol{\pi}(\mathbf{v}) = \phi_J(\boldsymbol{\pi}(\mathbf{v}_{*,J}))$. This implies $(\boldsymbol{\pi}(\mathbf{M}) \cap J) \subseteq \phi_J(\boldsymbol{\pi}(\mathbf{M}_{*,J}))$ since the pivot index of any vector in the row space of $\mathbf{M}$ (resp. $\mathbf{M}_{*,J}$) appears in $\boldsymbol{\pi}(\mathbf{M})$ (resp. $\boldsymbol{\pi}(\mathbf{M}_{*,J})$), see Section 2.4. It also immediately implies the equality whenever $\boldsymbol{\pi}(\mathbf{M}) \subseteq J$. □

These properties lead to a fast method for computing the pivot support when $n \gg m$, relying on a black box PivotSupport which efficiently finds the pivot support when $n \in O(m)$: one first considers the $2m$ left columns $\mathbf{M}_{*,\{1,\dots,2m\}}$ and uses PivotSupport to compute their pivot support $\boldsymbol{\pi}_1$. Then, Lemma 4.5 suggests to discard all columns of $\mathbf{M}$ not in $\boldsymbol{\pi}_1$, thus obtaining a matrix $\mathbf{M}_1$. Then, we repeat the same process to obtain $\mathbf{M}_2, \mathbf{M}_3$, etc.

*Algorithm 3:* WideMatrixPivotSupport
*Input:* matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ with $m \le n$.
*Output:* the pivot support $\boldsymbol{\pi}(\mathbf{M})$ of $\mathbf{M}$.
*Assumption:* The algorithm PivotSupport takes as input $\mathbf{M}$ and returns $\boldsymbol{\pi}(\mathbf{M})$.

1. *If* $n \le 2m$ *then return* PivotSupport($\mathbf{M}$)
2. $\boldsymbol{\pi}_0 \leftarrow$ PivotSupport($\mathbf{M}_{*,\{1,\dots,2m\}}$)
3. $\hat{\mathbf{M}} \leftarrow [\mathbf{M}_{*,\boldsymbol{\pi}_0} \quad \mathbf{M}_{*,\{2m+1,\dots,n\}}]$
4. $[\boldsymbol{\pi}_1 \quad \boldsymbol{\pi}_2] \leftarrow$ WideMatrixPivotSupport($\hat{\mathbf{M}}$), such that $\max(\boldsymbol{\pi}_1) \le \#\boldsymbol{\pi}_0$ and $\min(\boldsymbol{\pi}_2) > \#\boldsymbol{\pi}_0$.
5. *Return* $\left[\phi_{\boldsymbol{\pi}_0}(\boldsymbol{\pi}_1) \quad \phi_{\{2m+1,\dots,n\}}(\boldsymbol{\pi}_2)\right]$

Proposition 4.6. *Algorithm 3 is correct. It uses at most* $\lceil n/m \rceil$ *calls to* PivotSupport*, each with a* $m \times k$ *submatrix of* $\mathbf{M}$ *as input, where* $k \le 2m$*. If* $n \ge m$ *and* PivotSupport *is Algorithm 2, then Algorithm 3 is deterministic and uses*

$$\tilde{O}\left(m^{\omega-1} n \deg(\mathbf{M})\right)$$

*operations in* $\mathbb{K}$*.*

Proof. The correctness follows from Lemma 4.5, and the operation count is obvious. If using Algorithm 2 for PivotSupport, the correctness and cost bound follow from Proposition 4.4. □

# 5 PRELIMINARIES ON SHIFTED FORMS

## 5.1 Shifted forms

The notions of reduced and Popov forms presented in Sections 2.3 and 2.4 can be extended by introducing additive integer weights in the degree measure for vectors, following [23, Sec. 3]: a *shift* is a tuple $\boldsymbol{s} = (s_1,\dots,s_n) \in \mathbb{Z}^n$, and the *shifted degree* of a row vector $\mathbf{p} = [p_1 \ \cdots \ p_n] \in \mathbb{K}[x]^{1\times n}$ is

$$\operatorname{rdeg}_{\boldsymbol{s}}(\mathbf{p}) = \max(\deg(p_1)+s_1,\dots,\deg(p_n)+s_n) = \operatorname{rdeg}(\mathbf{p}\mathbf{x}^{\boldsymbol{s}}),$$

where $\mathbf{x}^{\boldsymbol{s}} = \operatorname{diag}(x^{s_1},\dots,x^{s_n})$. Note that here $\mathbf{p}\mathbf{x}^{\boldsymbol{s}}$ may be over the ring of Laurent polynomials if $\min(\boldsymbol{s}) < 0$; below, actual computations will always remain over $\mathbb{K}[x]$. Note that $\boldsymbol{s} = \mathbf{0}$ yields the usual degree notion.

This leads to shifted reduced forms for cases where one is interested in matrices whose rows minimise the $\boldsymbol{s}$-degree, instead of the usual degree. The generalized definitions from Section 2 can be concisely described as follows. For a matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$, its $\boldsymbol{s}$-row degree is $\operatorname{rdeg}_{\boldsymbol{s}}(\mathbf{M}) = \operatorname{rdeg}(\mathbf{M}\mathbf{x}^{\boldsymbol{s}})$. If $\mathbf{M}$ has no zero row, its $\boldsymbol{s}$-leading matrix is $\operatorname{lm}_{\boldsymbol{s}}(\mathbf{M}) = \operatorname{lm}(\mathbf{M}\mathbf{x}^{\boldsymbol{s}})$, and the $\boldsymbol{s}$-pivot index and entries of $\mathbf{M}$ are the pivot index and entries of $\mathbf{M}\mathbf{x}^{\boldsymbol{s}}$. The *$\boldsymbol{s}$-pivot degree* of $\mathbf{M}$ is the tuple of the degrees of its $\boldsymbol{s}$-pivot entries; this is equal to $\operatorname{rdeg}_{\boldsymbol{s}}(\mathbf{M}) - \boldsymbol{s}_J$, where $J$ is the $\boldsymbol{s}$-pivot index of $\mathbf{M}$ and $\boldsymbol{s}_J$ the corresponding subshift.

If $\mathbf{M}$ has no zero row and $m \le n$, then $\mathbf{M}$ is in $\boldsymbol{s}$-reduced, $\boldsymbol{s}$-weak Popov, $\boldsymbol{s}$-ordered weak Popov or $\boldsymbol{s}$-Popov form if $\mathbf{M}\mathbf{x}^{\boldsymbol{s}}$ has the respective un-shifted form, whenever $\min(\boldsymbol{s}) \ge 0$. Since adding a constant to all the entries of $\boldsymbol{s}$ simply shifts the $\boldsymbol{s}$-degree of any vector by the same constant, this does not affect the $\boldsymbol{s}$-leading matrix or the $\boldsymbol{s}$-pivots, and therefore does not change any of the shifted forms: we can therefore extend the definition of these to cover also $\boldsymbol{s}$ with negative indices, or we may conversely assume $\min(\boldsymbol{s}) = 0$ without loss of generality.

For a matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ of rank $r$, its $\boldsymbol{s}$-Popov form is the unique row basis of $\mathbf{M}$ which is in $\boldsymbol{s}$-Popov form. The $\boldsymbol{s}$-pivot index of the $\boldsymbol{s}$-Popov form of $\mathbf{M}$ is denoted by $\boldsymbol{\pi}_{\boldsymbol{s}}(\mathbf{M}) \in \mathbb{Z}^r_{>0}$ and is called the $\boldsymbol{s}$-pivot support of $\mathbf{M}$. For more details about shifted forms, we refer to [3].

Computationally, it is well known that finding the shifted Popov form can be reduced to the unshifted case very easily: given a matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ and a nonnegative shift $\boldsymbol{s} \in \mathbb{Z}^n$, simply compute the unshifted Popov form $\hat{\mathbf{P}}$ of $\mathbf{M}\mathbf{x}^{\boldsymbol{s}}$. Then $\hat{\mathbf{P}}$ will have the form $\hat{\mathbf{P}} = \mathbf{P}\mathbf{x}^{\boldsymbol{s}}$, and then $\mathbf{P}$ is the $\boldsymbol{s}$-Popov form of $\mathbf{M}$.

If $m < n$ and the unshifted Popov form computation can be carried out in complexity $O^{\sim}(m^{\omega-1} n \deg(\mathbf{M}))$, then this strategy computes the $\boldsymbol{s}$-Popov form in $O^{\sim}(m^{\omega-1} n(\deg(\mathbf{M}) + \operatorname{amp}(\boldsymbol{s})))$. Whenever $\operatorname{amp}(\boldsymbol{s}) \in O(\deg(\mathbf{M}))$ this cost is satisfactory, but especially when $\operatorname{amp}(\boldsymbol{s}) > m\deg(\mathbf{M})$ we could hope for a better cost: Lemma 5.1, Eq. (3) shows that $\deg(\mathbf{P}) \le m\deg(\mathbf{M})$ and hence we could hope to compute $\mathbf{P}$ in time $O^{\sim}(m^{\omega} n \deg(\mathbf{M}))$.

## 5.2 Hermite form

A matrix $\mathbf{H} = [h_{i,j}] \in \mathbb{K}[x]^{r\times n}$ with $r \le n$ is in *Hermite form* [10, 14, 19] if there are indices $1 \le j_1 < \cdots < j_r \le n$ such that:

- $h_{i,j} = 0$ for $1 \le j < j_i$ and $1 \le i \le r$,
- $h_{i,j_i}$ is monic (therefore nonzero) for $1 \le i \le r$,
- $\deg(h_{i',j_i}) < \deg(M_{i,j_i})$ for $1 \le i' < i \le r$.

We call $(j_1,\dots,j_r)$ the *Hermite pivot index* of $\mathbf{H}$; note that it is precisely the column rank profile of $\mathbf{H}$.

For a matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$, its Hermite form $\mathbf{H} \in \mathbb{K}[x]^{r\times n}$ is the unique row basis of $\mathbf{M}$ which is in Hermite form. We call *Hermite pivot support* of $\mathbf{M}$ the Hermite pivot index of $\mathbf{H}$. Note that this is also the column rank profile of $\mathbf{M}$, since $\mathbf{M}$ is unimodularly equivalent to $\mathbf{H}$ (up to padding $\mathbf{H}$ with zero rows).

For a given $\mathbf{M}$, the Hermite form can be seen as a specific shifted Popov form: defining the shift $\boldsymbol{h} = (nt,\dots,2t,t)$ for any $t > \deg(\mathbf{H})$, the $\boldsymbol{h}$-Popov form of $\mathbf{M}$ coincides with its Hermite form [3, Lem. 2.6]. Besides, the $\boldsymbol{h}$-pivot index of $\mathbf{H}$ is $(j_1,\dots,j_r)$; in other words, the Hermite pivot support $\boldsymbol{\pi}_{\boldsymbol{h}}(\mathbf{M})$ is the column rank profile of $\mathbf{M}$.

### 5.3 Degree bounds for shifted Popov forms

The next result states that the unimodular transformation $\mathbf{U}$ between $\mathbf{M}$ and its $\boldsymbol{s}$-Popov form $\mathbf{P}$ only depends on the submatrices of $\mathbf{M}$ and $\mathbf{P}$ formed by the columns in the $\boldsymbol{s}$-pivot support. It also gives useful degree bounds for the matrices $\mathbf{U}$ and $\mathbf{P}$; for a more general study of such bounds, we refer to [3, Sec. 5].

Lemma 5.1. *Let $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ have full rank with $m \le n$, let $\boldsymbol{s} \in \mathbb{Z}^n$, let $\mathbf{P} \in \mathbb{K}[x]^{m\times n}$ be the $\boldsymbol{s}$-Popov form of $\mathbf{M}$, and let $\boldsymbol{\pi} = \boldsymbol{\pi}_{\boldsymbol{s}}(\mathbf{M})$ be the $\boldsymbol{s}$-pivot index of $\mathbf{P}$. Then $\mathbf{M}_{*,\boldsymbol{\pi}} \in \mathbb{K}[x]^{m\times m}$ is nonsingular, $\mathbf{P}_{*,\boldsymbol{\pi}}$ is its $\boldsymbol{s}_{\boldsymbol{\pi}}$-Popov form, and $\mathbf{U} = \mathbf{P}_{*,\boldsymbol{\pi}}\mathbf{M}_{*,\boldsymbol{\pi}}^{-1} \in \mathbb{K}[x]^{m\times m}$ is the unique unimodular matrix such that $\mathbf{UM} = \mathbf{P}$.*

*Furthermore, we have the following degree bounds:*

$$\deg(\mathbf{P}) \le \deg(\mathbf{M}) + \operatorname{amp}(\boldsymbol{s}), \tag{2}$$

$$\operatorname{cdeg}(\mathbf{U}_{*,i}) \le |\operatorname{rdeg}(\mathbf{M})| - \operatorname{rdeg}(\mathbf{M}_{i,*}) \quad \text{for } 1 \le i \le m, \tag{3}$$

$$\deg \mathbf{U} \le |\operatorname{cdeg}(\mathbf{M}_{*,\boldsymbol{\pi}})|, \tag{4}$$

$$\deg(\mathbf{P}) \le \min(|\operatorname{rdeg}(\mathbf{M})|, |\operatorname{cdeg}(\mathbf{M}')|) \le m\deg(\mathbf{M}) \quad \text{where } \mathbf{M}' \text{ is } \mathbf{M} \text{ with its zero columns removed.} \tag{5}$$

Proof. Let $\hat{\mathbf{P}} = \mathbf{M}_{*,\boldsymbol{\pi}}$, $\hat{\mathbf{M}} = \mathbf{M}_{*,\boldsymbol{\pi}}$, and $\hat{\boldsymbol{s}} = \boldsymbol{s}_{\boldsymbol{\pi}}$. Note first that $\hat{\mathbf{P}}$ is nonsingular and in $\hat{\boldsymbol{s}}$-Popov form. Let $\mathbf{V} \in \mathbb{K}[x]^{m\times m}$ be any unimodular matrix such that $\mathbf{VM} = \mathbf{P}$. Then in particular $\mathbf{V}\hat{\mathbf{M}} = \hat{\mathbf{P}}$, hence $\hat{\mathbf{M}}$ is nonsingular and unimodularly equivalent to $\hat{\mathbf{P}}$, which is therefore the $\hat{\boldsymbol{s}}$-Popov form of $\hat{\mathbf{M}}$. Besides, we have $\mathbf{V} = \hat{\mathbf{P}}\hat{\mathbf{M}}^{-1} = \mathbf{U}$.

It remains to prove the degree bounds. The first one comes from the minimality of $\mathbf{P}$. Indeed, since $\mathbf{P}$ is an $\boldsymbol{s}$-reduced form of $\mathbf{M}$ we have $\max(\operatorname{rdeg}_{\boldsymbol{s}}(\mathbf{P})) \le \max(\operatorname{rdeg}_{\boldsymbol{s}}(\mathbf{M}))$. It suffices to remark that the left-hand side of this inequality is at least $\deg(\mathbf{P}) + \min(\boldsymbol{s})$ while the right-hand side is at most $\deg(\mathbf{M}) + \max(\boldsymbol{s})$.

Let $\boldsymbol{\delta} \in \mathbb{Z}_{\ge 0}^m$ be the $\boldsymbol{s}$-pivot degree of $\mathbf{P}$. Then, $\hat{\mathbf{P}}$ is in $(-\boldsymbol{\delta})$-Popov form with $\operatorname{rdeg}_{-\boldsymbol{\delta}}(\hat{\mathbf{P}}) = \mathbf{0}$ and $\operatorname{cdeg}(\hat{\mathbf{P}}) = \boldsymbol{\delta}$ [11, Lem. 4.1]. Besides, $\hat{\mathbf{P}}$ is column reduced and thus $|\operatorname{cdeg}(\hat{\mathbf{P}})| = \deg(\det(\hat{\mathbf{P}}))$ [12, Sec. 6.3.2], hence $|\boldsymbol{\delta}| = \deg(\det(\hat{\mathbf{M}}))$.

Let $\boldsymbol{t} = (t_1, \ldots, t_m) = \operatorname{rdeg}(\mathbf{U}^{-1})$. We obtain $\operatorname{rdeg}_{-\boldsymbol{\delta}}(\hat{\mathbf{M}}) = \operatorname{rdeg}_{-\boldsymbol{\delta}}(\mathbf{U}^{-1}\hat{\mathbf{P}}) = \operatorname{rdeg}_{\mathbf{0}}(\mathbf{U}^{-1}) = \boldsymbol{t}$ by the predictable degree property (with shifts, see e.g. [25, Lem. 2.17]). Now, $\mathbf{U}$ being the transpose of the matrix of cofactors of $\mathbf{U}^{-1}$ divided by the constant $\det(\mathbf{U}^{-1}) \in \mathbb{K} \setminus \{0\}$, we obtain $\operatorname{cdeg}(\mathbf{U}_{*,i}) \le |\boldsymbol{t}| - t_i$ for $1 \le i \le m$. Since $-\boldsymbol{\delta} \le \mathbf{0}$ we have $\boldsymbol{t} = \operatorname{rdeg}_{-\boldsymbol{\delta}}(\hat{\mathbf{M}}) \le \operatorname{rdeg}(\mathbf{M})$, hence $|\boldsymbol{t}| - t_i \le |\operatorname{rdeg}(\mathbf{M})| - \operatorname{rdeg}(\mathbf{M}_{i,*})$. This proves (3).

Every entry of the adjugate of $\hat{\mathbf{M}}$ has degree at most $|\operatorname{cdeg}(\hat{\mathbf{M}})|$. Then, $\mathbf{U} = \hat{\mathbf{P}}\hat{\mathbf{M}}^{-1}$ gives $\deg(\mathbf{U}) \le \deg(\hat{\mathbf{P}}) - \deg(\det(\hat{\mathbf{M}})) + |\operatorname{cdeg}(\hat{\mathbf{M}})|$. This yields (4) since $\deg(\hat{\mathbf{P}}) = \max(\boldsymbol{\delta}) \le |\boldsymbol{\delta}| = \deg(\det(\hat{\mathbf{M}}))$.

The second inequality in (5) is implied by $|\operatorname{rdeg}(\mathbf{M})| \le m\deg(\mathbf{M})$. Besides, from $\mathbf{P} = \mathbf{UM} = \sum_{i=1}^m \mathbf{U}_{*,i}\mathbf{M}_{i,*}$ we see that (3) implies $\deg(\mathbf{P}) \le |\operatorname{rdeg}(\mathbf{M})|$. For $j \in \boldsymbol{\pi}$ we have $\operatorname{cdeg}(\mathbf{P}_{*,j}) \le |\operatorname{cdeg}(\hat{\mathbf{P}})| = \deg(\det(\hat{\mathbf{M}})) \le |\operatorname{cdeg}(\mathbf{M}')|$. Now, let $j \in \{1, \ldots, n\} \setminus \boldsymbol{\pi}$: if $\mathbf{M}_{*,j} = \mathbf{0}$ then $\mathbf{P}_{*,j} = \mathbf{0}$, and otherwise it follows from (4) that $\operatorname{cdeg}(\mathbf{P}_{*,j}) = \deg(\mathbf{UM}_{*,j}) \le |\operatorname{cdeg}(\hat{\mathbf{M}})| + \operatorname{cdeg}(\mathbf{M}_{*,j}) \le |\operatorname{cdeg}(\mathbf{M}')|$. □

## 6 SHIFTED POPOV FORM WHEN THE PIVOT SUPPORT IS KNOWN

Here, we focus on the efficient computation of the $\boldsymbol{s}$-Popov form $\mathbf{P}$ of $\mathbf{M}$ when the $\boldsymbol{s}$-pivot support $\boldsymbol{\pi} = \boldsymbol{\pi}_{\boldsymbol{s}}(\mathbf{M})$ is known.

A first approach of using the knowledge of $\boldsymbol{\pi} = \boldsymbol{\pi}_{\boldsymbol{s}}(\mathbf{M})$ follows the remark of Section 3: in Algorithm 1 we take $\mathbf{L}$ to be identity matrix at the columns $\boldsymbol{\pi}$ and zero elsewhere. Then it is easy to verify that $\mathbf{C} = \mathbf{L}\mathbf{x}^{\max(\operatorname{rdeg}_{\boldsymbol{s}}(\mathbf{M}))-\boldsymbol{s}}$ is a completion for $\hat{\mathbf{M}} = \mathbf{M}\mathbf{x}^{\boldsymbol{s}}$, and hence Algorithm 1 will return $\hat{\mathbf{P}}$, the Popov form of $\hat{\mathbf{M}}$, and hence the $\boldsymbol{s}$-Popov form of $\mathbf{M}$ is $\hat{\mathbf{P}}\mathbf{x}^{-\boldsymbol{s}}$. This will remove the randomness of Algorithm 1, but will cost $O\tilde{}(n^\omega(\deg(\mathbf{M}) + \operatorname{amp}(\boldsymbol{s})))$. This seems unsatisfactory when $n \gg m$: for example when $\boldsymbol{s} = \mathbf{0}$, the size of the input and output is in $O(mn\deg(\mathbf{M}))$, so a more satisfactory cost bound for that case would be $O\tilde{}(m^{\omega-1}n\deg(\mathbf{M}))$.

We achieve this with our second approach which works in three steps, and which is formalised as Algorithm 4. First, we compute the $\boldsymbol{s}_{\boldsymbol{\pi}}$-Popov form of the submatrix $\mathbf{M}_{*,\boldsymbol{\pi}}$, which can be done efficiently since this submatrix is square and nonsingular. Then, we use polynomial matrix division to obtain the unimodular transformation $\mathbf{U} \in \mathbb{K}[x]^{m\times m}$ such that $\mathbf{M}_{*,\boldsymbol{\pi}_{\boldsymbol{s}}(\mathbf{M})} = \mathbf{U}\mathbf{P}_{*,\boldsymbol{\pi}_{\boldsymbol{s}}(\mathbf{M})}$. Lastly, we compute the remaining part of the $\boldsymbol{s}$-Popov form of $\mathbf{M}$ as $\mathbf{U}^{-1}\mathbf{M}_{*,\{1,\ldots,n\}\setminus\boldsymbol{\pi}}$. Note that, even for $\boldsymbol{s} = \mathbf{0}$, all entries of $\mathbf{U}^{-1}$ may have degree in $\Theta(m\deg(\mathbf{M}))$; we can avoid handling such large degrees by computing this product truncated at precision $x^{\delta+1}$, where $\delta$ is an upper bound on the degree of the $\boldsymbol{s}$-Popov form. For example, if $\boldsymbol{s} = \mathbf{0}$ then we can take $\delta = \deg(\mathbf{M})$.

*Algorithm 4:* KnownSupportPopov
*Input:*
- matrix $\mathbf{M} \in \mathbb{K}[x]^{m\times n}$ with full rank and $m < n$,
- shift $\boldsymbol{s} \in \mathbb{Z}^n$,
- the $\boldsymbol{s}$-pivot support $\boldsymbol{\pi} = \boldsymbol{\pi}_{\boldsymbol{s}}(\mathbf{M})$ of $\mathbf{M}$,
- bound $\delta \in \mathbb{Z}_{>0}$ on the degree of the $\boldsymbol{s}$-Popov form of $\mathbf{M}$.

*Default:* $\delta = 1 + \min(|\operatorname{rdeg}(\mathbf{M})|, |\operatorname{cdeg}(\mathbf{M}')|, \deg(\mathbf{M}) + \operatorname{amp}(\boldsymbol{s}))$, where $\mathbf{M}'$ is $\mathbf{M}$ with zero columns removed.
*Output:* the $\boldsymbol{s}$-Popov form of $\mathbf{M}$.

1. $\mathbf{P} \leftarrow$ zero matrix in $\mathbb{K}[x]^{m\times n}$
2. $\mathbf{P}_{*,\boldsymbol{\pi}} \leftarrow$ NonsingularPopov$(\mathbf{M}_{*,\boldsymbol{\pi}}, \boldsymbol{s}_{\boldsymbol{\pi}})$
3. $\mathbf{U} \leftarrow \mathbf{M}_{*,\boldsymbol{\pi}}\mathbf{P}_{*,\boldsymbol{\pi}}^{-1} \in \mathbb{K}[x]^{m\times m}$
4. $\delta \leftarrow \min(\delta,\ 1 + \max(\operatorname{rdeg}_{\boldsymbol{s}_{\boldsymbol{\pi}}}(\mathbf{P}_{*,\boldsymbol{\pi}})) - \min(\boldsymbol{s}_{(1,\ldots,n)\setminus\boldsymbol{\pi}})$
5. $\mathbf{P}_{*,\{1,\ldots,n\}\setminus\boldsymbol{\pi}} \leftarrow \mathbf{U}^{-1}\,\mathbf{M}_{*,\{1,\ldots,n\}\setminus\boldsymbol{\pi}} \bmod x^{\delta}$
6. *Return* $\mathbf{P}$

Proposition 6.1. *Algorithm 4 is correct and uses $O\tilde{}(m^{\omega-1}n\delta)$ operations in $\mathbb{K}$, where*

$$\delta = 1 + \min(|\operatorname{rdeg}(\mathbf{M})|, |\operatorname{cdeg}(\mathbf{M}')|, \deg(\mathbf{M}) + \operatorname{amp}(\boldsymbol{s})),$$

*and $\mathbf{M}'$ is $\mathbf{M}$ with zero columns removed.*

Proof. Let $\mathbf{Q} \in \mathbb{K}[x]^{m\times n}$ be the $\boldsymbol{s}$-Popov form of $\mathbf{M}$. For correctness we prove that $\mathbf{P} = \mathbf{Q}$. The first part of Lemma 5.1 shows that indeed $\mathbf{Q}_{*,\boldsymbol{\pi}} = \mathbf{P}_{*,\boldsymbol{\pi}}$, and that $\mathbf{U} = \mathbf{M}_{*,\boldsymbol{\pi}}\mathbf{P}_{*,\boldsymbol{\pi}}^{-1} = \mathbf{M}_{*,\boldsymbol{\pi}}\mathbf{Q}_{*,\boldsymbol{\pi}}^{-1}$ computed at Step 3 is the unimodular matrix such that $\mathbf{M} = \mathbf{UQ}$.

The last item of Lemma 5.1 proves that the input default value of $\delta$ is more than $\deg(\mathbf{Q})$. Besides, by definition of $\boldsymbol{s}$-pivots and $\boldsymbol{s}$-Popov form, the column $j$ of $\mathbf{Q}$ has degree at most

$$\max(\operatorname{rdeg}_{\boldsymbol{s}_{\boldsymbol{\pi}}}(\mathbf{Q}_{*,\boldsymbol{\pi}})) - s_j = \max(\operatorname{rdeg}_{\boldsymbol{s}_{\boldsymbol{\pi}}}(\mathbf{P}_{*,\boldsymbol{\pi}})) - s_j.$$

It follows that $\delta > \deg(\mathbf{Q}_{*,\{1,\ldots,n\}\setminus\boldsymbol{\pi}})$ after Step 4, and therefore the submatrix $\mathbf{Q}_{*,\{1,\ldots,n\}\setminus\boldsymbol{\pi}}$ is equal to the truncated product $\mathbf{U}^{-1}\,\mathbf{M}_{*,\{1,\ldots,n\}\setminus\boldsymbol{\pi}} \bmod x^{\delta}$ computed at Step 5. Thus, $\mathbf{Q} = \mathbf{P}$.

Now we detail the cost bound. Step 2 uses $O(m^\omega \deg(\mathbf{M}_{*,\boldsymbol{\pi}}))$ operations, by Theorem 2.2. Step 3 has the same cost by Lemma 6.2; note that $\mathbf{P}_{*,\boldsymbol{\pi}}$ is in $\boldsymbol{s}_{\boldsymbol{\pi}}$-Popov form and thus column reduced. This is within the announced bound since

$$O(m^\omega \deg(\mathbf{M}_{*,\boldsymbol{\pi}})) \subseteq O(m^{\omega-1} n \deg(\mathbf{M}))$$

and $\deg(\mathbf{M}) \leq \delta$ holds by definition of $\delta$.

Finally, Step 5 costs $O\tilde{}(m^{\omega-1}n\delta)$ operations in $\mathbb{K}$: since $\mathbf{U}(0) \in \mathbb{K}^{m\times m}$ is invertible, the truncated inverse of $\mathbf{U}$ is computed by Newton iteration in time $O\tilde{}(m^\omega\delta)$; then, the truncated product uses $O\tilde{}(m^\omega \lceil (n-m)/m \rceil \delta)$ operations. □

At Step 3, we compute a product of the form $\mathbf{B}\mathbf{A}^{-1}$, knowing that it has polynomial entries and that $\mathbf{A}$ is column reduced; in particular, $\deg(\mathbf{B}\mathbf{A}^{-1}) \leq \deg(\mathbf{B})$ [18, Lem. 3.1]. Then, it is customary to obtain $\mathbf{B}\mathbf{A}^{-1}$ via a Newton iteration on the "reversed matrices" (see e.g. [21, Chap. 5] and [25, Chap. 10]).

Lemma 6.2. *For a column reduced matrix $\mathbf{A} \in \mathbb{K}[x]^{m\times m}$ and a matrix $\mathbf{B} \in \mathbb{K}[x]^{m\times m}$ which is a left multiple of $\mathbf{A}$, the quotient $\mathbf{B}\mathbf{A}^{-1}$ can be computed using $O\tilde{}(m^\omega \deg(\mathbf{B}))$ operations in $\mathbb{K}$.*

Proof. We follow Steps 1 and 2 of the algorithm PM-QuoRem from [18], on input $\mathbf{A}$, $\mathbf{B}$, and $d = \deg(\mathbf{B})+1$; hence the requirement $\operatorname{cdeg}(\mathbf{B}) < \operatorname{cdeg}(\mathbf{A}) + (d,\dots,d)$ is satisfied. It is proved in [18, Prop. 3.4] that these steps correctly compute the quotient $\mathbf{B}\mathbf{A}^{-1}$; yet we do a different cost analysis since the assumptions on parameters in [18, Prop. 3.4] might not be satisfied here.

Step 1 of PM-QuoRem computes reversals $\hat{\mathbf{A}} = \mathbf{A}(x^{-1})\mathbf{x}^{\operatorname{cdeg}(\mathbf{A})}$ and $\hat{\mathbf{B}} = \mathbf{B}(x^{-1})\mathbf{x}^{\operatorname{cdeg}(\mathbf{B})}$ of the matrices: this uses no arithmetic operation. These matrices also have dimensions $m \times m$ and the constant coefficient of $\hat{\mathbf{A}}$ is invertible since $\mathbf{A}$ is column reduced. Step 2 computes the truncated product $\hat{\mathbf{A}}\hat{\mathbf{B}}^{-1} \bmod x^{d+1}$, which can be done via Newton iteration in $O\tilde{}(m^\omega d)$ operations in $\mathbb{K}$. □

Since Algorithm 4 works for an arbitrary shift, it allows us in particular to find the Hermite form of $\mathbf{M}$ when its Hermite pivot support is known. It turns out that the latter can be computed efficiently via a column rank profile algorithm from [25].

Proof of Theorem 1.2. Take any integer $\delta$ as in Theorem 1.2:

$$\delta > \min(|\operatorname{rdeg}(\mathbf{M})|, |\operatorname{cdeg}(\mathbf{M}')|),$$

where $\mathbf{M}'$ is $\mathbf{M}$ with zero columns removed.

Let $\boldsymbol{h} = (n\delta,\dots,2\delta,\delta)$. By Lemma 5.1, $\delta$ is more than the degree of the Hermite form of $\mathbf{M}$; therefore the $\boldsymbol{h}$-Popov form of $\mathbf{M}$ is also its Hermite form (see Section 5.2). Thus, up to the knowledge of the Hermite pivot support $\boldsymbol{\pi}_{\boldsymbol{h}}(\mathbf{M})$ of $\mathbf{M}$, we can compute the Hermite form of $\mathbf{M}$ using $O\tilde{}(m^{\omega-1}n\delta)$ operations via Algorithm 4.

As mentioned in Section 5.2, $\boldsymbol{\pi}_{\boldsymbol{h}}(\mathbf{M})$ is also the column rank profile of $\mathbf{M}$. It is shown in [25, Sec. 11.2] how to use row basis and left kernel basis computations to obtain this rank profile in $O\tilde{}(m^{\omega-1}n\sigma)$ operations, where $\sigma = \lceil |\operatorname{rdeg}(\mathbf{M})|/m \rceil$ is roughly the average row degree of $\mathbf{M}$. We have $\sigma \leq 1 + \min(|\operatorname{rdeg}(\mathbf{M})|)$ by definition, and one can easily verify that $|\operatorname{rdeg}(\mathbf{M})|/m \leq |\operatorname{cdeg}(\mathbf{M}')|$, hence $\sigma \leq \delta$. □

# REFERENCES


[1] B. Beckermann and G. Labahn. 2000. Fraction-Free Computation of Matrix Rational Interpolants and Matrix GCDs. *SIAM J. Matrix Anal. Appl.* 22, 1 (2000), 114–144.

[2] B. Beckermann, G. Labahn, and G. Villard. 1999. Shifted Normal Forms of Polynomial Matrices. In *ISSAC'99*. ACM, 189–196.

[3] B. Beckermann, G. Labahn, and G. Villard. 2006. Normal forms for general polynomial matrices. *J. Symbolic Comput.* 41, 6 (2006), 708–737.

[4] D. G. Cantor and E. Kaltofen. 1991. On fast multiplication of polynomials over arbitrary algebras. *Acta Inform.* 28, 7 (1991), 693–701.

[5] 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.

[6] P. Giorgi, C.-P. Jeannerod, and G. Villard. 2003. On the complexity of polynomial matrix computations. In *ISSAC'03*. ACM, 135–142.

[7] S. Gupta. 2011. *Hermite forms of polynomial matrices.* Master's thesis. University of Waterloo, Canada.

[8] 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.

[9] S. Gupta and A. Storjohann. 2011. Computing Hermite Forms of Polynomial Matrices. In *ISSAC'11*. ACM, 155–162.

[10] C. Hermite. 1851. Sur l'introduction des variables continues dans la théorie des nombres. *Journal für die reine und angewandte Mathematik* 41 (1851), 191–216.

[11] 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.

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

[13] G. Labahn, V. Neiger, and W. Zhou. 2017. Fast, deterministic computation of the Hermite normal form and determinant of a polynomial matrix. *J. Complexity* (in press) (2017).

[14] C. C. MacDuffee. 1933. *The Theory of Matrices.* Springer-Verlag Berlin Heidelberg. https://doi.org/10.1007/978-3-642-99234-6

[15] T. Mulders and A. Storjohann. 2003. On lattice reduction for polynomial matrices. *J. Symbolic Comput.* 35 (2003), 377–401. Issue 4.

[16] 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

[17] V. Neiger. 2016. Fast computation of shifted Popov forms of polynomial matrices via systems of modular polynomial equations. In *ISSAC'16*. ACM, 365–372.

[18] V. Neiger and T. X. Vu. 2017. Computing canonical bases of modules of univariate relations. In *ISSAC'17*. ACM.

[19] M. Newman. 1972. *Integral Matrices.* Number v. 45 in Integral matrices. Academic Press.

[20] V. M. Popov. 1972. Invariant Description of Linear, Time-Invariant Controllable Systems. *SIAM Journal on Control* 10, 2 (1972), 252–264.

[21] S. Sarkar. 2011. *Computing Popov Forms of Polynomial Matrices.* Master's thesis. University of Waterloo, Canada.

[22] S. Sarkar and A. Storjohann. 2011. Normalization of row reduced matrices. In *ISSAC'11*. ACM, 297–304.

[23] 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.

[24] W. A. Wolovich. 1974. *Linear Multivariable Systems.* Applied Mathematical Sciences, Vol. 11. Springer-Verlag New-York.

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

[26] W. Zhou and G. Labahn. 2012. Efficient Algorithms for Order Basis Computation. *J. Symbolic Comput.* 47, 7 (2012), 793–819.

[27] W. Zhou and G. Labahn. 2013. Computing Column Bases of Polynomial Matrices. In *ISSAC'13*. ACM, 379–386.

[28] W. Zhou and G. Labahn. 2014. Unimodular Completion of Polynomial Matrices. In *ISSAC'14*. ACM, 413–420.

[29] W. Zhou, G. Labahn, and A. Storjohann. 2012. Computing Minimal Nullspace Bases. In *ISSAC'12*. ACM, 366–373.