Popov form computation for matrices of Ore polynomials

Khochtali, Mohamed; Rosenkilde, Johan Sebastian Heesemann; Storjohann, Arne

Published in:
ISSAC 2017 - Proceedings of the 2017 ACM International Symposium on Symbolic and Algebraic Computation

Link to article, DOI:
10.1145/3087604.3087650

Publication date:
2017

Document Version
Peer reviewed version

Popov Form Computation for Matrices of Ore Polynomials

Mohamed Khochtali
University of Waterloo
Canada
mkhochta@uwaterloo.ca

Johan Rosenkilde, né Nielsen
Technical University of Denmark
Denmark
jsrn@jsrn.dk

Arne Storjohann
University of Waterloo
Canada
astorjoh@uwaterloo.ca

ABSTRACT

Let $\mathsf{F}[\partial;\sigma,\delta]$ be a ring of Ore polynomials over a field. We give a new deterministic algorithm for computing the Popov form P of a non-singular matrix $A \in \mathsf{F}[\partial;\sigma,\delta]^{n\times n}$. Our main focus is to ensure controlled growth in the size of coefficients from F in the case $\mathsf{F} = \mathsf{k}(z)$, and even $\mathsf{k} = \mathbb{Q}$. Our algorithms are based on constructing from A a linear system over F and performing a structured fraction-free Gaussian elimination. The algorithm is output sensitive, with a cost that depends on the orthogonality defect of the input matrix: the sum of the row degrees in A minus the sum of the row degrees in P. The resulting bit-complexity for the differential and shift polynomial case over $\mathbb{Q}(z)$ improves upon the previous best.

1 INTRODUCTION

Ore polynomial rings, also known as skew polynomial rings, are non-commutative generalizations of univariate polynomial rings, introduced by Ore [16]. They have a variety of applications, such as modeling recurrence relations and differential equations [16]. Row spaces of matrices over Ore polynomial rings arise in studying coupled systems of such equations. Computing normal forms of such matrices allows comparing systems and finding small or otherwise special elements in the spaces.

In this paper, we consider the computation of the Popov normal form of a non-singular matrix over an Ore polynomial ring (see the formal definition of Popov form in Section 2). Our focus is when the base field F is infinite so coefficient growth is a concern, in particular $\mathsf{F} = \mathsf{k}(z)$ where k is some field, possibly also with coefficient growth from k in mind, for example $\mathsf{k} = \mathbb{Q}$.

An Ore polynomial ring is given by a base field F, an automorphism σ of F, and a "derivation" of σ: this is a map $\delta : \mathsf{F} \to \mathsf{F}$ satisfying

$$\delta(a+b) = \delta(a) + \delta(b)$$
$$\delta(ab) = \sigma(a)\delta(b) + \delta(a)b \ .$$

The Ore ring $\mathsf{F}[\partial;\sigma,\delta]$ is then given as the set of finite formal sums $a_0 + a_1\partial + \ldots + a_d\partial^d$, with $a_i \in \mathsf{F}$. Addition of two Ore polynomials is the usual element-wise addition, while multiplication is given from the following non-commutative rule for multiplying an $a \in \mathsf{F}$ with ∂ on the right:

$$\partial a = \sigma(a)\partial + \delta(a) \ .$$

We mention two particularly important examples of Ore polynomial rings where $\mathsf{F} = \mathsf{k}(z)$:

- *Differential polynomials* where $\sigma(z) = z$ and $\delta(f(z)) = f'(z)$ is the usual derivative with respect to z.
- The *shift case*, or time-dependence, where $\sigma(f(z)) = f(z+1)$ is the shift automorphism and $\delta = 0$.

We refer to [2, 7] and the references therein for background on linear algebra with matrices of Ore polynomials. Here, we only mention that for matrices over Ore rings the notions of *rank* and *(non)-singularity* make sense: in particular, performing row or column operations on a matrix will not change its rank. Further, two matrices $M, M' \in \mathsf{F}[\partial;\sigma,\delta]^{n\times m}$ generate the same left row space if and only if there exists $U \in \mathrm{GL}_n(\mathsf{F}[\partial;\sigma,\delta])$ such that $M = UM'$, where $\mathrm{GL}_n(\mathsf{F}[\partial;\sigma,\delta])$ denotes the set of invertible $n \times n$ matrices over $\mathsf{F}[\partial;\sigma,\delta]$. Roughly speaking, the Popov form of a matrix A is the matrix that generates the same row space as A but with row degrees as small as possible.

Computing reduced matrices and normal forms of matrices over an Ore polynomial ring $\mathsf{F}[\partial;\sigma,\delta]$, while taking into account expression swell from F when F is infinite, has previously been considered. Beckermann, Cheng and Labahn [2] and Cheng and Labahn [5] compute row reduced bases using an "order basis" approach as known for matrices over $\mathsf{F}[x]$, and taking care of coefficient growth. Davies, Cheng and Labahn [17] show how computing the Popov form can be reduced to nullspace computation, a problem for which effective fraction-free techniques exist. Giesbrecht and Kim [7] compute the Hermite normal form of an Ore polynomial matrix by linearizing it to a larger matrix over F. The resulting problem can then be tackled completely by the usual approaches for matrices over F.

A common thread in the algorithms mentioned in the previous paragraph is that they either explicitly [7] or implicitly [2] consider a linearization over F of the input matrix: this allows to obtain bounds on the size of intermediate expressions. Although in a slightly different context, we remark that a linear algebra point for normal form computation can be found for the commutative case already in [13, Sections 6.7.1 and 6.72]. More recently, in [3] and [12, Section 7] it is shown that the shifted Popov basis of a $\mathsf{k}[x]$-module given by equations exactly corresponds to some specific rows

ISSAC '17, July 25-28, 2017, Kaiserslautern, Germany

ACM ISBN 978-1-4503-5064-8/17/07...$15.00
https://doi.org/10.1145/3087604.3087650

in a reduced row echelon form of the left nullspace of a constant matrix with much larger dimension than the original polynomial equations. The algorithm we present here is based on the linearization technique of Labhalla, Lombardi and Marlin [14].

Cost estimates for our algorithm are given in Section 6.1. We summarize some of these cost estimates here and compare with previous work. Consider computing the Popov form in the differential case when $\mathsf{F} = \mathsf{k}(z)$. We are given a non-singular input matrix $A \in \mathsf{k}(z)[\partial;\sigma,\delta]^{n\times n}$, that is, the entries of A are polynomials in ∂, the coefficients of which are rational functions from $\mathsf{k}(z)$. We can assume, without loss of generality, by clearing denominators, that A is over $\mathsf{k}[z][\partial;\sigma,\delta]$. A running time estimate in terms of operations from k thus involves three parameters: the dimension n; a bound d for $\deg_\partial A$; a bound e for $\deg_z A$. Our algorithm constructs from A a structured matrix of dimension $O(n^2d)\times O(n^2d)$ over $\mathsf{k}[z]$. We then perform a structured fraction-free Gaussian elimination to recover the Popov form. The cost of our algorithm is $O(n^{\omega+2}d^3\,\mathsf{M}(n^2de))$ operations from k. Here, ω is an exponent for matrix multiplication, and M is a multiplication time: two polynomials from $\mathsf{k}[z]$ of degree strictly less than t can be multiplied in $\mathsf{M}(t)$ operations from k. Assuming $\omega = 3$ and a pseudo-linear multiplication time, and ignoring logarithmic factors, the cost of our algorithm is then on the order of n^7d^4e operations from k. For comparison, the fraction-free algorithm supporting [2, Corollary 7.7] requires on the order of $n^9d^4e^2$ operations from k to produce a row reduced form of A, while the algorithm in [5, Theorem 6.2] requires on the order of $n^8d^4e + n^7d^3e^2$ operations from k.

Now consider the case $\mathsf{k} = \mathbb{Q}$. Like before, we assume our input matrix is over $\mathbb{Z}[z][\partial;\sigma,\delta]$. Ignoring logarithmic factors, and again assuming pseudo-linear integer arithmetic, our algorithm requires on the order of $n^9d^5e\log\beta$ bit operations. Here, β is a parameter that depends on the magnitude of integer coefficients in A (see Theorem 6.2). The modular algorithm supporting [17, Theorem 6.3] requires about $n^{10}d^5e\log\beta + n^9d^4e^2\log\beta$ bit operations.

On the one hand, we point out that the algorithms of [2, 5] solve a considerably more general problem than we do in this paper: they can be applied to input matrices of arbitrary shape and rank and thus compute the rank of the input matrix as well as a left nullspace. Although we hope to consider the rank deficient case in the future, our analysis currently assumes the input matrix is non-singular. On the other hand, the algorithms in [2, 5] only produce a row reduced form of A and not the canonical Popov forms.

Beyond the improved asymptotic worst case cost estimates we have reported above, our algorithm has two additional features. First, in the shift case, the worst case running times we have reported above are improved by a factor of n: the linearized system has a special shape in this case which the algorithm is able to exploit. Second, for inputs that are are not too far from being row reduced the running time is asymptotically faster. The *orthogonality defect* of A is the difference between the sum of the row degrees in A and the sum of the row degrees in its Popov form P, denoted by $\mathsf{OD}(A)$. Our algorithms are output sensitive in the parameter $\mathsf{OD}(A)$, which can be as small as 0 and as large as nd. If $n \le \mathsf{OD}(A) \le nd$ then the running times reported above are improved by a factor of $\mathsf{OD}(A)/(nd)$. For $\mathsf{OD}(A) < n$ further improvements are obtained. In the special case $\mathsf{OD}(A) = 0$, which means the input matrix is already reduced, the algorithm detects this and avoids the lion's share of the computation, instead applying a fast normalization to transform the input to Popov form.

The rest of this paper is organized as follows. In Section 2 we define some notation and recall some important facts about matrices of Ore polynomials that are established in [2, 7]. In Section 3 we recall the linearization method of Labhalla, Lombardi and Marlin [14] for Hermite form computation over $\mathsf{k}[x]$. Section 4 extends the method to compute the Popov form of a non-singular matrix of Ore polynomials. Section 5 gives the design and analysis of our algorithm for performing a structured block elimination of the linearized system. Section 6 shows how the elimination can be done in a fraction-free fashion and gives bounds on the sizes of intermediate expressions for some concrete cases of F, namely $\mathsf{F} = \mathsf{k}(z)$, $\mathsf{F} = \mathbb{Q}$ and $\mathsf{F} = \mathbb{Q}(z)$. Cost analysis for computing the Popov form over these rings is provided in Section 6.1. Section 7 concludes.

2 PRELIMINARIES

Let $\mathsf{F}[\partial;\sigma,\delta]$ be an Ore polynomial ring. The *degree* of a vector $\vec{v} \in \mathsf{F}[\partial;\sigma,\delta]^{1\times n}$ or matrix $A \in \mathsf{F}[\partial;\sigma,\delta]^{n\times m}$, denoted by $\deg\vec{v}$ and $\deg A$, respectively, is the maximal degree of the entries of $\vec{v}$ or A (we define $\deg 0 = -\infty$). If $\vec{v}$ is non-zero then by the *pivot* of $\vec{v}$, denoted $\mathrm{piv}(\vec{v})$, we mean the rightmost entry of $\vec{v}$ having $\deg\vec{v}$. The elements of $\vec{v}$ are denoted v_i for $i = 1,\ldots,n$. By $\mathrm{rdeg}A$ we mean the list $[d_1,d_2,\ldots,d_n]$ where $d_i = \deg \mathrm{row}_i A$, $1 \le i \le n$.

Definition 2.1. Given a non-singular $A \in \mathsf{F}[\partial;\sigma,\delta]^{n\times n}$ with $\mathrm{rdeg}A = [d_1,d_2,\ldots,d_n]$, the *leading matrix* of A, denoted $\mathsf{LM}(A) \in \mathsf{F}^{n\times n}$, is the matrix whose (i,j) entry is the coefficient of ∂^{d_i} of $A_{i,j}$. A is said to be *row reduced* if $\mathrm{rank}(\mathsf{LM}(A)) = n$.

A canonical row reduced basis is provided by the Popov form. Although the Popov form can be defined for a matrix of arbitrary shape and rank, in this paper we focus on non-singular matrices.

Definition 2.2. A non-singular matrix $P \in \mathsf{F}[\partial;\sigma,\delta]^{n\times n}$ is in *Popov form* if $\mathrm{LM}(P)$ is unit lower triangular and the degrees of off-diagonal entries of P are strictly less than the degree of the diagonal entry in the same column.

Note that the definition of Popov form implies the pivot index of row i is i, $1 \le i \le n$.

A matrix in Popov form is row reduced but the converse is not true. This is classical for matrices over $\mathsf{F}[x]$, see [13, Section 6.3.2]. For the extension to matrices over $\mathsf{F}[\partial;\sigma,\delta]$, see [2, Lemma A.1 (a)]. The last item is often called the Predictable Degree Property.

THEOREM 2.3. *Let $A \in \mathsf{F}[\partial;\sigma,\delta]^{n\times n}$ be non-singular. Then the following are equivalent:*

(1) *A is row reduced.*
(2) *Among all matrices that are left equivalent to A, the list of row degrees of A, when sorted in non-decreasing order, will be lexicographically minimal.*
(3) *For any $\vec{v} \in \mathsf{F}[\partial;\sigma,\delta]^{1\times n}$, we have*

$$\deg(\vec{v}A) = \max_{i=1,\ldots,n} (\deg \operatorname{row}_i A + \deg v_i) \ .$$

LEMMA 2.4. *If $A, U, P \in \mathsf{F}[\partial;\sigma,\delta]^{n\times n}$, all non-singular and U invertible and P in Popov form such that $UA = P$, then $\deg U \le (n-1)\deg A$.*

PROOF. By Item 3 of Theorem 2.3, then $\deg U^{-1} \le \deg A$ since the degree of each row of P is non-negative. By [7, Corollary 3.3] then $U = (U^{-1})^{-1}$ has degree at most $(n-1)\deg A$. □

The following notion is a measure for how far A is from being row reduced:

Definition 2.5. Let $A \in \mathsf{F}[\partial;\sigma,\delta]$ and non-singular. The *orthogonality defect* of A, denoted $\mathsf{OD}(A)$, is given as $\sum \operatorname{rdeg} A - \sum \operatorname{rdeg} P$, where P is the Popov form of A.

LEMMA 2.6. *If A is row reduced then $\mathsf{OD}(A) = 0$.*

PROOF. Follows immediately from Theorem 2.3, Item 2. □

3 WARM-UP: HERMITE FORM OF MATRICES OVER $\mathsf{F}[X]$ VIA LINEARIZATION

Let $A \in \mathsf{F}[x]^{n\times n}$ be non-singular with $\deg A = d$. The Hermite form of A is the unique upper trinagular matrix that is left equivalent to A, has monic diagonal entries, and has degrees of of off-diagonal entries strictly less than the degree of the diagonal entry in the same column. Labhalla, Lombardi and Marlin [14] show how to recover H by transforming a matrix over F of dimension $(n^2d + n - dn) \times (n^2d + n)$ to reduced row echelon form.

Example 3.1. Let $\mathsf{F} = \mathbb{Z}/(7)$. The input matrix

$$A = \begin{bmatrix} 3x^2+6x+6 & 5x^2+3x+3 & 6x^2+x \\ 5x^2+5 & 6x^2+x & 5x^2+2x+6 \\ 3x+5 & 4x+5 & 5x^2+2x+1 \end{bmatrix} \in \mathsf{F}[x]^{3\times 3}$$

has Hermite form

$$H = \begin{bmatrix} 1 & 4 & 0 \\ & x+1 & 6 \\ & & x^2+6x \end{bmatrix} \in \mathsf{F}[x]^{3\times 3}.$$

Indeed, the reduced row echeleon form of the Sylvester matrix

$$\left[\begin{array}{ccccccccc|ccccccccc|ccccccccc}
3&6&6&&&&&&&5&3&3&&&&&&&6&1&&&&&&&\\
&3&6&6&&&&&&&5&3&3&&&&&&&6&1&&&&&&\\
&&3&6&6&&&&&&&5&3&3&&&&&&&6&1&&&&&\\
&&&3&6&6&&&&&&&5&3&3&&&&&&&6&1&&&&\\
&&&&3&6&6&&&&&&&5&3&3&&&&&&&6&1&&&\\
\hline
5&0&5&&&&&&&6&1&&&&&&&&5&2&6&&&&&&&\\
&5&0&5&&&&&&&6&1&&&&&&&&5&2&6&&&&&&\\
&&5&0&5&&&&&&&6&1&&&&&&&&5&2&6&&&&&\\
&&&5&0&5&&&&&&&6&1&&&&&&&&5&2&6&&&&\\
&&&&5&0&5&&&&&&&6&1&&&&&&&&5&2&6&&&\\
\hline
&3&5&&&&&&&&4&5&&&&&&&&5&2&1&&&&&&\\
&&3&5&&&&&&&&4&5&&&&&&&&5&2&1&&&&&\\
&&&3&5&&&&&&&&4&5&&&&&&&&5&2&1&&&&\\
&&&&3&5&&&&&&&&4&5&&&&&&&&5&2&1&&&\\
&&&&&3&5&&&&&&&&4&5&&&&&&&&5&2&1&&\\
\hline
&&&&&&&&&&&&&&&&&&&&&&&&&&&\\
&&&&&&&&&&&&&&&&&&&&&&&&&&&\\
&&&&&&&&&&&&&&&&&&&&&&&&&&&\\
&&&&&&&&&&&&&&&&&&&&&&&&&&&\\
&&&&&&&&&&&&&&&&&&&&&&&&&&&
\end{array}\right] \in \mathsf{F}^{15\times 21}$$

is

$$\left[\begin{array}{ccccccccc|ccccccccc|ccccccccc}
1&&&&&&&&&4&&&&&&&&&0&2&1&&&&&&&4&0\\
&1&&&&&&&&0&&&&&&&&&3&6&1&&&&&&&0&4\\
&&1&&&&&&&0&&&&&&&&&4&0&6&&&&&&&5&3\\
&&&1&&&&&&0&&&&&&&&&3&0&0&&&&&&&0&4\\
&&&&1&&&&&0&&&&&&&&&4&0&0&&&&&&&4&3\\
&&&&&1&&&&0&&&&&&&&&3&0&0&&&&&&&0&4\\
&&&&&&1&&&0&&&&&&&&&4&0&0&&&&&&&0&0\\
\hline
&&&&&&&1&&&&&&&&&&&1&2&2&&&&&&&3&6\\
&&&&&&&&1&&&&&&&&&&6&0&2&&&&&&&4&1\\
&&&&&&&&&1&&&&&&&&&1&0&0&&&&&&&0&6\\
&&&&&&&&&&1&&&&&&&&6&0&0&&&&&&&6&1\\
&&&&&&&&&&&1&1&&&&&&0&0&&&&&&&0&6\\
\hline
&&&&&&&&&&&&&&&&&&&&&1&&&&&&&6&0\\
&&&&&&&&&&&&&&&&&&&&&&1&&&&&&6&0\\
&&&&&&&&&&&&&&&&&&&&&&&1&6&0\\
&&&&&&&&&&&&&&&&&&&&&&&&&&&&&\\
&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
\end{array}\right] \in \mathsf{F}^{15\times 21}.$$

The submatrix comprised of the last nonzero row in each horizontal slice, namely rows 7, 12 and 15, corresponds to a linearization of the Hermite form.

4 POPOV FORM OF MATRICES OVER $\mathsf{F}[\partial;\sigma,\delta]$ VIA LINEARIZATION

In this section we apply the linearization technique to compute the Popov form of $\mathsf{F}[\partial;\sigma,\delta]$ polynomials. Here we deal only with the overall correctness of the approach and its structural properties, while the latter sections give precise algorithms for the steps as well as cost bounds.

Define the linearization $\phi_P : \mathsf{F}[\partial;\sigma,\delta]^{*\times n} \mapsto \mathsf{F}^{*\times(n^2d+n)}$ by

$$\begin{aligned}\phi_P(\vec{v}) &= \phi_P \begin{bmatrix} v_1 & \cdots & v_n \end{bmatrix} \\ &= \left[\begin{array}{ccc|c|ccc} [v_n]_{nd} & \cdots & [v_1]_{nd} & \cdots & [v_n]_0 & \cdots & [v_1]_0 \end{array}\right]\end{aligned}$$

where $[v_i]_k$ denotes the coefficient of ∂^k of $v_i \in \mathsf{F}[\partial;\sigma,\delta]^{*\times 1}$.

Example 4.1. Consider the following example over $\mathbb{Z}_7(z)[\partial;\sigma,\delta]$ with $n = d = 2$:

$$\vec{v} = \begin{bmatrix} \partial^2 + (3+z)\partial + z^3 & 2\partial + (1+2z) \end{bmatrix}$$
$$\phi_P(\vec{v}) = [\ 0\ \ 0\ \mid\ 0\ \ 0\ \mid\ 0\ \ 1\ \mid\ 2\ \ (3+z)\ \mid\ (1+2z)\ \ z^3\ ]$$

Let now A_{lin} be given as the ϕ_P-image of the vectors

$$\partial^j \operatorname{row}_i(A) \qquad \text{for } i = 1,\ldots,n \text{ and } j = 0,\ldots,nd - \deg \operatorname{row}_i(A)\,,$$

ordered by descending degrees and breaking ties by the i index. Then we can write A_{lin} uniquely in block upper triangular form as

$$A_{\mathsf{lin}} = \begin{bmatrix} B_{nd} \\ \hline B_{nd-1} \\ \hline \vdots \\ \hline B_0 \end{bmatrix} = \begin{bmatrix} C_{nd} & * & \cdots & * \\ & C_{nd-1} & \cdots & * \\ & & \ddots & \vdots \\ & & & C_0 \end{bmatrix}, \qquad (1)$$

where each $C_* \in \mathsf{F}^{*\times n}$ has no zero rows. Note that the row space of A_{lin} is in one-to-one correspondence, through ϕ_P, with the set

$$\left\{ \sum_{i=1}^{n} u_i \operatorname{row}_i A \;\middle|\; u_i \in \mathsf{F}[\partial;\sigma,\delta],\ \deg u_i \le nd - \deg \operatorname{row}_i A \right\}. \tag{2}$$

LEMMA 4.2. *If A is non-singular, then A_{lin} has full row rank and row dimension $n^2d+n-\sum \operatorname{rdeg} A$. For $d \le t \le nd$, B_t (and C_t) has exactly n rows.*

PROOF. A_{lin} has full row rank, since any F-linear relation between rows of A_{lin} maps to an $\mathsf{F}[\partial;\sigma,\delta]$-linear relation between rows of A through ϕ_P and by the definition of A_{lin}. No such relation exists since A is non-singular. The i'th row of A is represented in exactly $nd - \deg \operatorname{row}_i(A) + 1$ of the B_t, so the row dimension of A_{lin} becomes as claimed. This also shows that B_t has n rows when $t \ge d$ since every row of A is represented. □

We say the pivot of a vector $\vec{v} \in \mathsf{F}^{1\times(n^2d+n)}$ is the index of the left-most non-zero element of $\vec{v}$ (not to be confused with the pivot of a vector over $\mathsf{F}[\partial;\sigma,\delta]$, see Section 2). Define $\eta : \{1,\dots,n\} \times \mathbb{Z}_{\ge 0} \to \{1,\dots,n^2d+n\}$ as the map between (pivot, degree) of vectors from $\mathsf{F}[\partial;\sigma,\delta]^{1\times n}$ and the pivot in vectors from $\mathsf{F}^{1\times(n^2d+n)}$ induced by ϕ_P, that is,

$$\eta(i,d') = n(nd-d') + n + 1 - i\ .$$

For a vector $\vec{v} \in \mathsf{F}^{1\times(n^2d+n)}$ we say that the *P-pivot* and *P-degree* of $\vec{v}$ are the first and second components of the 2-tuple $\eta^{-1}(i)$, where i is $\vec{v}$'s pivot.

Now let R_{lin} be the reduced row echelon form of A_{lin}. Then R_{lin} can also be written uniquely in block upper triangular form as

$$R_{\mathsf{lin}} = \begin{bmatrix} T_{nd} & * & \cdots & * \\ & T_{nd-1} & \cdots & * \\ & & \ddots & \vdots \\ & & & T_0 \end{bmatrix}, \tag{3}$$

where each $T_* \in \mathsf{F}^{*\times n}$ has no zero rows. Note that those rows in R_{lin} with P-degree t are contained in the submatrix of R_{lin} occupied by T_t. Because R_{lin} is in echelon form, for any given degree t and pivot i, there is at most one row in R_{lin} with P-degree t and P-pivot i, and any row in the row space of R_{lin} with P-degree t and P-pivot i will be a linear combination of this row, and possibly rows below it.

THEOREM 4.3. *Let A be non-singular, and let R_{lin} be the reduced row echelon form of A_{lin}. Then the Popov form P of A is the matrix whose i'th row is the ϕ_P^{-1}-image of the row of R_{lin} with minimal P-degree having P-pivot i.*

PROOF. The unique unimodular matrix $U \in \mathsf{F}[\partial;\sigma,\delta]^{n\times n}$ with $UA = P$ has $\deg U \le (n-1)d$ by Lemma 2.4, and therefore the ϕ_P-linearized rows of P are contained in the row space of R_{lin}. We will prove that they in fact appear directly as rows of R_{lin}. By the minimality of the row degrees of the Popov form, Item 2 of Theorem 2.3, the rows chosen as in the theorem must therefore be exactly those rows of R_{lin}.

So for $1 \le i \le n$, consider the i'th row $\vec{p}$ of P, which has pivot i. Since $\phi_P(\vec{p})$ is in the row space of R_{lin}, there must be exactly one row $\vec{r}_k$ of R_{lin} with the same pivot, with row index k. If $\vec{w}$ is the unique vector over F satisfying $\vec{w}R_{\mathsf{lin}} = \phi_P(\vec{p})$ then clearly $w_k = 1$ and $w_j = 0$ for $j < k$. We claim $w_j = 0$ also for $j > k$ in which case $\vec{r}_k = \phi_P(\vec{p})$ as we wanted to prove. Suppose, to arrive at a contradiction, that $w_j \ne 0$ for some $j > k$, and let $\vec{r}_j$ be the j'th row of R_{lin}. Since all other rows of R_{lin} are zero at the pivot position of $\vec{r}_j$, that means $\deg p_{i,j'} \ge d'$, where j', d' are the P-pivot respectively P-degree of $\vec{r}_j$. On the other hand, since the $\phi_P^{-1}(\vec{r}_j)$ is in the row space of A and has pivot j', the minimality of the degrees of the Popov form implies $d' \ge \deg p_{j',j'}$. But then $\deg p_{i,j'} \ge \deg p_{j',j'}$, which contradicts that P is in Popov form. We conclude that $w_j = 0$ for $j > k$, and hence $\phi_P(\vec{p}) = \vec{r}_k$. □

Example 4.4. For clarity, we exemplify the approach with a usual polynomial ring, i.e. $\sigma = \mathsf{id}$ and $\delta = 0$. Consider the input $A \in \mathsf{F}[x]^{3\times 3}$ from Example 3.1, $\mathsf{F} = \mathbb{Z}/(7)$. Then

$$A_{\mathsf{lin}} = \begin{bmatrix}
6&5&3&1&3&6&0&3&6 & & & & & & & & & & & & \\
5&6&5&2&1&0&6&0&5 & & & & & & & & & & & & \\
5&0&0&2&4&3&1&5&5 & & & & & & & & & & & & \\
 & & &6&5&3&1&3&6&0&3&6 & & & & & & & & & \\
 & & &5&6&5&2&1&0&6&0&5 & & & & & & & & & \\
 & & &5&0&0&2&4&3&1&5&5 & & & & & & & & & \\
 & & & & & &6&5&3&1&3&6&0&3&6 & & & & & & \\
 & & & & & &5&6&5&2&1&0&6&0&5 & & & & & & \\
 & & & & & &5&0&0&2&4&3&1&5&5 & & & & & & \\
 & & & & & & & & &6&5&3&1&3&6&0&3&6 & & & \\
 & & & & & & & & &5&6&5&2&1&0&6&0&5 & & & \\
 & & & & & & & & &5&0&0&2&4&3&1&5&5 & & & \\
 & & & & & & & & & & & & & & & & & & & & \\
 & & & & & & & & & & & & & & & & & & & & \\
 & & & & & & & & & & & & & & & & & & & & \\
\end{bmatrix} \in \mathsf{F}^{15\times 21}. \tag{4}$$

The row reduced echelon form of A_{lin} is

$$R_{\mathsf{lin}} = \left[\begin{array}{cc|cc|ccc|cc}
1 & & 6 & 6 & & & 2 & 0 & 0 \\
 1\ 2 & & 4 & 6 & & & 3 & 6 & 5 \\
\hline
 & 1\ 0 & & 6 & & & 2 & 4 & 1 \\
 & 1\ 2 & & 4 & & & 2 & 5 & 3 \\
\hline
 & & 1\ 0 & & & & 6 & 0 & 0 \\
 & & 1\ 2 & & & & 0 & 0 & 0 \\
\hline
 & & & & 1 & & 6 & 0 & 0 \\
 & & & & 1 & & 0 & 6 & 5 \\
 & & & & & 1 & 0 & 4 & 1 \\
\hline
 & & & & & 1 & 6 & 0 & 0 \\
 & & & & & 1 & 6 & 1 & 2 \\
 & & & & & 1\ 4 & 3 & 6 & \\
\hline
 & & & & & & 1 & 6 & 5 \\
 & & & & & & 1\ 4 & 1 & \\
\hline
 & & & & & & & 1\ 2 &
\end{array}\right]. \tag{4}$$

We pick out the the rows of R_{lin} with minimal degree having P-pivot 1, 2, and 3, respectively. The Popov form of A is thus

$$\phi_P^{-1}\left[\begin{array}{cc|cc|ccc|cc}
 & & & & & & 1 & 4 & 1 \\
 & & & & & & & 1\ 2 & \\
\hline
 & & & & & & & & \\
\hline
 & & & & 1 & 6 & 0 & 0 &
\end{array}\right] = \begin{bmatrix} x+1 & 0 & 4 \\ 2 & 1 & 0 \\ 0 & 0 & x^2+6x \end{bmatrix}.$$

We now consider some structural properties of A_{lin} and R_{lin}. Recall that we have written A_{lin} as a block upper triangular matrix

$$A_{\mathsf{lin}} = \left[\begin{array}{ccc|ccc} C_{nd} & \cdots & * & * & \cdots & * \\ & \ddots & \vdots & \vdots & \ddots & \vdots \\ & & C_{k+1} & * & \cdots & * \\ \hline & & & C_k & \cdots & * \\ & & & & \ddots & \vdots \\ & & & & & C_0 \end{array}\right]$$

with each C_* of column dimension n and with no zero rows. Let $\{a_1, a_2, \ldots, a_n\}$ be the multi-set of row degrees of A. The following lemma follows from the definition of A_{lin}.

LEMMA 4.5. *For $k = 0, 1, \ldots, nd$, the trailing submatrix*

$$\begin{bmatrix} C_k & \cdots & * \\ & \ddots & \vdots \\ & & C_0 \end{bmatrix}$$

of A_{lin} has row dimension $(k+1)n - \sum_{i=1}^{n} \min(a_i, k+1)$.

We have also written R_{lin} in a block upper triangular form as

$$R_{\mathsf{lin}} = \left[\begin{array}{ccc|ccc} T_{nd} & \cdots & * & * & \cdots & * \\ & \ddots & \vdots & \vdots & \ddots & \vdots \\ & & T_{k+1} & * & \cdots & * \\ \hline & & & T_k & \cdots & * \\ & & & & \ddots & \vdots \\ & & & & & T_0 \end{array}\right]$$

where each $T_* \in \mathsf{F}^{*\times n}$ has no zero rows. Let $\{p_1, p_2, \ldots, p_n\}$ be the multi-set of row degrees in the Popov form of A. The following lemma follows as a corollary of Theorem 4.3.

LEMMA 4.6. *For $k = 0, 1, \ldots, nd$, the trailing submatrix*

$$\begin{bmatrix} T_k & \cdots & * \\ & \ddots & \vdots \\ & & T_0 \end{bmatrix} \tag{5}$$

of R_{lin} has row dimension at most $(k+1)n - \sum_{i=1}^{n} \min(p_i, k+1)$.

For $k = 0, 1, \ldots, nd$, define OD_k to be the row dimension of the left nullspace of the principal submatrix

$$\begin{bmatrix} C_{nd} & \cdots & * \\ & \ddots & \vdots \\ & & C_{k+1} \end{bmatrix} \tag{6}$$

of A_{lin}. Recall that $\mathsf{OD}(A) := \sum \mathrm{rdeg} A - \sum \mathrm{rdeg} P$.

THEOREM 4.7. *For $k = 0, 1, \ldots, nd$ we have $\mathsf{OD}_k \leq \mathsf{OD}(A)$.*

PROOF. Let P be a permutation, and U be a unit lower triangular non-singular matrix over F that such that premultiplying (6) by UP transforms it to echelon form

$$\left[\begin{array}{c} R_{k+1} \\ \hline \end{array}\right]$$

with OD_k zero rows. Then applying $\mathrm{diag}(U, I)$ to A_{lin} yields

$$\left[\begin{array}{c|c} U & \\ \hline & I \end{array}\right] \left[\begin{array}{ccc|ccc} C_{nd} & \cdots & * & * & \cdots & * \\ & \ddots & \vdots & \vdots & \ddots & \vdots \\ & & C_{k+1} & * & \cdots & * \\ \hline & & & C_k & \cdots & * \\ & & & & \ddots & \vdots \\ & & & & & C_0 \end{array}\right] = \left[\begin{array}{c|ccc} R_{k+1} & * & \cdots & * \\ \hline & E_k & \cdots & * \\ & C_k & \cdots & * \\ & & \ddots & \vdots \\ & & & C_0 \end{array}\right], \tag{7}$$

where $E_k \in \mathsf{F}^{\mathsf{OD}_k \times n}$. Considering that A_{lin} has full row rank, the row dimension of the submatrix

$$\begin{bmatrix} E_k & \cdots & * \\ C_k & \cdots & * \\ & \ddots & \vdots \\ & & C_0 \end{bmatrix} \tag{8}$$

of the matrix on the right of (7) will be equal to the row dimension of the trailing submatrix (5) of R_{lin}. Lemmas 4.5 and 4.6 now give

$$\begin{aligned} \mathsf{OD}_k &\leq \sum_{i=1}^{n} \min(a_i, k+1) - \sum_{i=1}^{n} \min(p_i, k+1) \\ &= \sum_{i=1}^{n} (\min(a_i, k+1) - \min(p_i, k+1)). \end{aligned}$$

Assume now that $a_1 \leq a_2 \leq \cdots \leq a_n$ and $p_1 \leq p_2 \leq \cdots p_n$. Then $a_i - p_i \geq 0$ for $i = 1, 2, \ldots, n$ by Item 2 of Theorem 2.3, and

$$\min(a_i, k+1) - \min(p_i, k+1) \begin{cases} = a_i - p_i \text{ if } a_i \leq k+1 \\ = 0 \qquad \text{if } a_i > k+1 \text{ an } p_i \geq k+1 \\ < a_i - p_i \text{ if } a_i > k+1 \text{ and } p_i < k+1 \end{cases}.$$

Thus $\min(a_i, k+1) - \min(p_i, k+1) \leq a_i - p_i$ in all cases, establishing the result. □

5 BLOCK ELIMINATION

Let $A \in \mathsf{F}[\partial; \sigma, \delta]^{n\times n}$ be non-singular with $\deg A \leq d$. In this section we show how to perform a structured Gaussian elimination of the linearized system A_{lin} over F. We first consider in Section 5.1 the problem of transforming A to Popov form when A is already row reduced. Then we consider the general case in Section 5.2.

5.1 Normalization if already row reduced

We can detect if A is row reduced by testing its leading coefficient matrix for non-singularity. Suppose A is already row reduced. Let U be the unique matrix such that $UA = P$ is in Popov form. By the predictable degree property (Theorem 2.3, item 3) then $\deg \mathrm{col}_i U \leq d - \deg \mathrm{row}_i A$ ($\deg P \leqslant d$), $1 \leq i \leq n$. Consider the following submatrix of A_{lin} (1) comprised of the last $n(d+1) - \sum \mathrm{rdeg} A$ rows:

$$\bar{A}_{\mathsf{lin}} = \begin{bmatrix} B_d \\ \vdots \\ \dot{B}_0 \end{bmatrix} = \left[\begin{array}{ccc|c} C_d & \cdots & * & \\ & \ddots & \vdots & \\ & & \dot{C}_0 & \end{array}\right]. \tag{9}$$

Note that the row space of $\bar{A}_{\mathsf{lin}}$ is in one-to-one correspondence, through ϕ_P, with the set

$$\left\{ \sum_{i=1}^{n} u_i \,\mathrm{row}_i A \;\middle|\; u_i \in \mathsf{F}[\partial; \sigma, \delta],\ \deg u_i \leq d - \deg \mathrm{row}_i A \right\}.$$

As a corollary of Lemma 4.2 we have that $\bar{A}_{\mathsf{lin}}$ has full row rank $n(d+1) - \sum \mathrm{rdeg} A$. The next theorem is a corollary of Theorem 4.3.

THEOREM 5.1. *Let A be non-singular and row reduced, and let $\bar{R}_{\mathsf{lin}}$ be the reduced row echelon form of $\bar{A}_{\mathsf{lin}}$. Then the Popov form P of A is the matrix S whose i'th row is the ϕ_P^{-1}-image of the row of $\bar{R}_{\mathsf{lin}}$ with minimal degree having P-pivot i.*

Because A is row reduced, by Lemma 2.6 we have $\mathsf{OD}(A) = 0$, so by Theorem 4.7 each block C_* in $\bar{A}_{\mathsf{lin}}$ will have full row rank. Since the right block of $\bar{A}_{\mathsf{lin}}$ has column dimension $n(d+1)$, performing standard Gauss Jordan elimination

would cost $O((nd)^3)$ operations from F to produce $\bar{R}_{\mathsf{lin}}$ in its entirety. We can save a factor of d by avoiding the complete computation of $\bar{R}_{\mathsf{lin}}$. Instead, first compute an echelon form of $\bar{A}_{\mathsf{lin}}$ by applying Gaussian elimination to each full rank slice B_*. Gaussian elimination of a single B_* has cost $O(n^3 d)$, yielding a total cost for all slices of $O(n^3 d^2)$ operations in F. Then use back substitution to reduce the n rows whose ϕ_P^{-1}-image has minimal degree and P-pivot i, $1 \le i \le n$. This costs an additional $O(n^3 d^2)$. Finally, scale these n rows so their pivots are equal to one. We obtain the following result.

THEOREM 5.2. *Let $A \in \mathsf{F}[\partial;\sigma,\delta]^{n\times n}$ be non-singular and row reduced, with $\deg A \le d$. The Popov form of A can be computed within the following cost:*

- *Computing $\partial^k \operatorname{row}_i A$ for $1 \le k \le d - \deg \operatorname{row}_i A$, $1 \le i \le n$.*
- *An additional $O(n^3 d^2)$ operations from F.*

5.2 General case

Now assume that A is not already row reduced, so that $\mathsf{OD} := \mathsf{OD}(A) > 0$. The key observation is that since $\deg P \le d$, the ϕ_P-linearization of the rows of P will be contained in the row space of the trailing submatrix (5) of R_{lin} for $k = d$. This implies that the rows of R_{lin} occupied by $T_{nd}, T_{nd-1}, \cdots, T_{d+1}$ are not required.

Our algorithm for performing the elimination of A_{lin} has three phases. The first phase computes the matrix (8) for $k = d$, whose row space is equal to that of (5) for $k = d$. The second phase transforms this matrix to row echelon form. The third phase performs back substitution to reduce the n rows whose ϕ_P^{-1}-image has minimal degree and P-pivot i, $1 \le i \le n$.

Our main computation tool is the Gauss transform [18, Section 2.3]. Given as input a matrix

$$\begin{bmatrix} E_k \\ C_k \end{bmatrix} \in \mathsf{F}^{O(\mathsf{OD}+n)\times n}, \tag{10}$$

the so called Gauss transform algorithm [18, Algorithm 2.14] can be used to produce a permutation matrix P_k and unit lower triangular matrix U_k such that

$$\overbrace{\left[\begin{array}{c|c} F_k & \\ \hline N_k & I \end{array}\right]}^{U_k} P_k \begin{bmatrix} E_k \\ C_k \end{bmatrix} = \left[\begin{array}{c} G_k \\ \hline \end{array}\right],$$

where $\begin{bmatrix} N_k \mid I \end{bmatrix} P_k$ and G_k are the left nullspace basis and a row echelon form, respectively, of the input matrix (10).

Phase 1: For convenience, let E_0 be the $0 \times n$ matrix. We will compute a Gauss transform as described above for $k = nd, nd-1, \ldots, d+1$. At the start of stage k we are exactly in the situation shown in (7). The key observation is that no entries in the rows occupied by R are required, and so the computation of these rows can be avoided. To go from stage k to $k+1$ we can thus apply only the nullspace to the next slice and obtain

$$\begin{bmatrix} N_k \mid I \end{bmatrix} P_k \left[\begin{array}{c|ccc} E_k & * & \cdots & * \\ C_k & * & \cdots & * \end{array}\right] = \begin{bmatrix} \ & E_{k-1} & \cdots & * \end{bmatrix}.$$

Continue this for $k = nd, nd-1, \ldots, d+1$.

Phase 2: For $k = d, d-1, \ldots, 0$, we apply the complete Gauss transform to the work matrix:

$$U_k P_k \left[\begin{array}{c|ccc} E_k & * & \cdots & * \\ C_k & * & \cdots & * \end{array}\right] = \left[\begin{array}{c|ccc} G_k & * & \cdots & * \\ & E_{k-1} & \cdots & * \end{array}\right].$$

Repeating this for $k = d, d-1, \ldots, 0$, we have computed the row echelon form

$$G = \begin{bmatrix} G_d & \cdots & * \\ & \ddots & \vdots \\ & & G_0 \end{bmatrix} \in \mathsf{F}^{*\times n(d+1)}.$$

Phase 3: Identify for $i = n, n-1, \ldots, 1$, the row in G whose ϕ_P^{-1}-image has minimal degree and P-pivot i, and use back substitution to zero out the entries in this row which are above a pivot, similar to how we proceeded in Section 5.1. Finally, scale these n rows to make their pivots equal to one.

Example 5.3. Consider the input matrix

$$A = \begin{bmatrix} 7x^2+3x+8 & 9x^2+7x+4 & x^2+2x+2 \\ 3x^2+4 & 7x^2+6x+8 & 5x^2+10x \\ 3x^2+2x+5 & 7x^2+5x+1 & 4x^2+8x+5 \end{bmatrix} \in \mathbb{Z}/(11)^{3\times 3}.$$

Then

$$A_{\mathsf{lin}} = \begin{bmatrix} C_6 & * & * & * & * & * \\ & C_5 & * & * & * & * \\ & & C_4 & * & * & * \\ & & & C_3 & * & * \\ & & & & C_2 & * \end{bmatrix}$$

$$= \begin{bmatrix} 1&9&7&2&7&3&2&4&8 \\ 5&7&3&10&6&0&0&8&4 \\ 4&7&3&8&5&2&5&1&5 \\ & & &1&9&7&2&7&3&2&4&8 \\ & & &5&7&3&10&6&0&0&8&4 \\ & & &4&7&3&8&5&2&5&1&5 \\ & & & & & &1&9&7&2&7&3&2&4&8 \\ & & & & & &5&7&3&10&6&0&0&8&4 \\ & & & & & &4&7&3&8&5&2&5&1&5 \end{bmatrix}$$

For Phase 1, step $k = 6$ we compute and apply the nullspace of C_6 to obtain

$$\left[\begin{array}{c} E_5 \\ \hline C_5 \\ & C_4 \\ & & C_3 \\ & & & C_2 \end{array}\right] = \left[\begin{array}{ccccccccc} 0&0&0&0&4&8 & & & \\ \hline 1&9&7&2&7&3&2&4&8 \\ 5&7&3&10&6&0&0&8&4 \\ 4&7&3&8&5&2&5&1&5 \\ & & &1&9&7&2&7&3&2&4&8 \\ & & &5&7&3&10&6&0&0&8&4 \\ & & &4&7&3&8&5&2&5&1&5 \\ & & & & & & & & & & & \\ & & & & & & & & & & & \end{array}\right]$$

Phase 1, step $k = 5$ we compute and apply the nullspace of the 4×3 matrix occupied by E_5 and C_5 to obtain

$$\left[\begin{array}{c} E_4 \\ \hline C_4 \\ & C_3 \\ & & C_2 \end{array}\right] = \left[\begin{array}{ccccccccc} 0&4&8 & & & & & & \\ 0&0&0&0&4&8 & & & \\ \hline 1&9&7&2&7&3&2&4&8 \\ 5&7&3&10&6&0&0&8&4 \\ 4&7&3&8&5&2&5&1&5 \\ & & &1&9&7&2&7&3&2&4&8 \\ & & &5&7&3&10&6&0&0&8&4 \\ & & &4&7&3&8&5&2&5&1&5 \\ & & & & & & & & & & & \\ & & & & & & & & & & & \end{array}\right]$$

We continue Phase 1 steps $k = 4, 3$ with nullspace application. Then we switch to Phase 2. For Phase 2 steps $k = 2, 1, 0$ we

apply the entire Gauss transforms, yielding the echelon form

$$\left[\begin{array}{c|c|c|c} & G_2 & * & * \\ \hline & & G_1 & * \\ \hline & & & G_0 \end{array}\right] = \left[\begin{array}{ccc|ccc|ccc} & & & 1 & 9 & 7 & 2 & 7 & 3 & 2 & 4 & 8 \\ & & & & 1 & 10 & 3 & 8 & 2 & 0 & 0 & 0 \\ & & & & & 1 & 10 & 1 & 3 & 0 & 0 & 0 \\ \hline & & & & & & & & 1 & 2 & 0 & 0 & 0 \\ & & & & & & & & & 1 & 10 & 1 & 3 \\ \hline & & & & & & & & & & & 1 & 2 \end{array}\right].$$

In Phase 3, we identify the Popov rows (rows 1, 5 and 6 in this example) and then do back substitution:

$$\left[\begin{array}{ccc|ccc|ccc} & & & 1 & & & 2 & & & 2 & & 0 \\ & & & & 1 & 10 & 3 & 8 & 2 & 0 & 0 & 0 \\ & & & & & 1 & 10 & 1 & 3 & 0 & 0 & 0 \\ \hline & & & & & & & & 1 & 2 & 0 & 0 & 0 \\ & & & & & & & & & 1 & 10 & & 1 \\ \hline & & & & & & & & & & & 1 & 2 \end{array}\right].$$

The Popov form of A is thus

$$\phi_P^{-1}\left[\begin{array}{c|c|c} & & 1 \\ \hline & & \\ \hline 1 & 2 & 2 \end{array}\right]\left[\begin{array}{cc|cc|cc} & 1 & 10 & & 1 & 0 \\ & & 2 & & 1 & 2 \\ \hline & & 2 & & 0 & 0 \end{array}\right] = \left[\begin{array}{cc|cc|cc} x+1 & 0 & & 10 & & \\ 2 & 1 & & 0 & & \\ \hline 0 & 0 & x^2+2x+2 & & & \\ & & & & & \\ \hline & & & & & \end{array}\right].$$

The next theorem gives a cost analysis of the algorithm just described in terms of operations from F. The theorem gives three cost estimates. First, we give an unconditional cost estimate based only on the input parameters n and d. Second, we give a refined cost estimate in terms of OD. Third, we consider the case of special Ore rings (such as the shift case) for which the matrix A_{lin} may have the shape shown in Example 4.4, that is, with a large block upper triangular submatrix of zeroes in the northeast corner: the cost estimates are improved by a factor of n in this case.

THEOREM 5.4. *Let $A \in \mathsf{F}[\partial;\sigma,\delta]^{n\times n}$ be non-singular with $\deg A \le d$. The Popov form of A can be computed within the following costs.*

(1) *General case:*
   - *Computing $\partial^k \operatorname{row}_i A$ for $1 \le k \le nd - \deg \operatorname{row}_i A$, $1 \le i \le n$.*
   - *Additional $O(n^{\omega+2}d^3)$ field operations from F.*

(2) *A more refined cost is obtained by considering the parameter OD. Assume that A is not already row reduced, so that $\mathsf{OD} := \mathsf{OD}(A) > 0$. Then the number of additional operations is reduced to:*
   - *$O(\mathsf{OD}^{\omega-2}n^4d^2)$ if $\mathsf{OD} < n$*
   - *$O(\mathsf{OD}\, n^{\omega+1}d^2)$ if $\mathsf{OD} \ge n$*

(3) *Finally, suppose that the Ore ring $\mathsf{F}[\partial;\sigma,\delta]$ has the property that for any nonzero element $f \in \mathsf{F}[\partial;\sigma,\delta]$, the trailing degree of ∂f is at least one more than the trailing degree of f. Then the O-estimate in part 1 is reduced by a factor of n, and the O-estimates in part 2 become*
   - *$O(n^{\omega+1}d + \mathsf{OD}^{\omega-2}n^3d^2)$ if $\mathsf{OD} < n$*
   - *$O(\mathsf{OD}\, n^{\omega}d^2)$ if $\mathsf{OD} \ge n$*

PROOF. We first establish part 2 of the theorem. By Theorem 4.7, the row dimension of each nullspace N_* is bounded by OD. Instead of considering the three phases separately, we will partition the computational work done as follows. The nullspace N_k is applied for all k, $0 \le k \le nd$, but the unit lower triangular block F_k is applied only for $0 \le k \le d$. Also note that for $k \le d$, the column dimension of the slice to which F_k is being applied to is bounded by $(d+1)n$. The application of the permutations P_* do not dominate the cost. We can thus partition the computational work as follows.

A Gauss transform: at most $nd+1$ times, compute a Gauss transform of a matrix bounded in dimension $O(\mathsf{OD}+n)\times n$.

B Nullspace application: at most $nd+1$ times, multiply an $O(\mathsf{OD})\times n$ matrix by an $n \times O(n^2d)$ matrix.

C Computing the echelon form: at most $d+1$ times, multiply an $O(n)\times O(n)$ matrix by a $O(n)\times O(nd)$ matrix.

D Back substitution: $O(n^3d^2)$ operations.

Since the rank of the input matrix (10) is bounded by its column dimension n, the cost of computing (U_k, P_k) for a given k is bounded by $O((\mathsf{OD}+n)n^{\omega-1})$ by [18, Proposition 2.15]. This gives a total cost for (A) of $O((\mathsf{OD}+n)n^{\omega}d)$. Using an obvious block decomposition, (C) can be done in time $O(n^{\omega}d^2)$.

It remains to bound the cost of (B). There are two cases, depending on whether $\mathsf{OD} \le n$ or $\mathsf{OD} > n$. Using an obvious block decomposition, a single nullspace application has cost $O(\mathsf{OD}^{\omega-2}n^3d)$ if $\mathsf{OD} \le n$ and $O(\mathsf{OD}\, n^{\omega}d)$ otherwise. The total cost for (B) is thus $O(\mathsf{OD}^{\omega-2}n^4d^2)$ if $\mathsf{OD} \le n$ and $O(\mathsf{OD}\, n^{\omega+1}d^2)$ if $\mathsf{OD} > n$. In both cases these upper bounds for the cost of (B) dominate the cost bounds for (A), (C) and (D).

Part 1 of the theorem follows by substituting the a priori upper bound $\mathsf{OD} \le nd$ into the O-bound in part 2 for the case $\mathsf{OD} \ge n$.

For part 3, note that the (nonzero part) of the slice to which the nullspace is applied will now have dimension $O(nd)$ instead of $O(n^2d)$. The cost estimates for the work in part (B) are thus reduced by a factor of n, but this still dominates the cost of parts (A), (C) and (D) in case $\mathsf{OD} \ge n$. If $\mathsf{OD} < n$ then part (A) costs $O(n^{\omega+1}d)$ which might dominate the cost of part (B). □

6 FRACTION FREE BLOCK ELIMINATION

Now consider the case when all entries in A_{lin} are coming from an integral domain, for example $\mathsf{F} = \mathsf{k}(z)$ for a field k but all entries are in $\mathsf{F}[z]$, or even $\mathbb{Z}[z]$ when $\mathsf{k} = \mathbb{Q}$. It is desirable in this setting to keep all intermediate quantities in the computation integral, while at the same time controlling their growth. The classic technology for this purpose in the linear algebra setting is fraction-free Gaussian elimination [1, 6]. To take advantage of the block upper triangular shape of A_{lin} we could directly apply a sparse variant of fraction-free Gaussian elimination [15]. In this section we show how to incoporate matrix multiplication. The Gauss transform algorithm [18, Algorithm 2.14] is actually designed to do fraction-free Gaussian elimination, and because of its column recursive formulation, is well suited to the elimination of A_{lin}.

The incorporation of fraction-free techniques into the algorithm supporting Theorem 5.4 is straightforward. For $k = nd, nd-1, \ldots, -1$, the fraction-free Gauss transform

algorithm also computes Δ_k, the minor of R_k in (7) comprised of its rank profiles columns. To start the process set $\Delta_{nd+1} = 1$ since R_{nd+1} is the 0×0 matrix, and recall that E_{nd} is the $0 \times n$ matrix. At step k we have the scaled matrix

$$\Delta_{k+1} \left[\begin{array}{c|ccc} E_k & * & \cdots & * \end{array} \right]$$

from the previous step, together with Δ_{k+1}. The rows of A_{lin} occupied by C_k are premultiplied by Δ_{k+1} to form the next slice

$$\Delta_{k+1} \left[\begin{array}{c|ccc} E_k & * & \cdots & * \\ C_k & * & \cdots & * \end{array} \right], \tag{11}$$

which will be fraction-free, that is, all entries are minors of A_{lin} of dimension bounded by one plus the rank of R_{k+1}. (We remark that only scaling the rows of A_{lin} that will be involved in the next elimination step is important for the complexity, and similar to [15].) At stage k, the fraction-free Gauss transform takes as input (11), together with Δ_{k+1}, and returns as output the permutation P_k and the scaled matrix

$$\bar{U}_k = \left[\begin{array}{c|c} \bar{F}_k & \\ \hline \Delta_k N_k & \Delta_k I \end{array} \right], \tag{12}$$

together with Δ_k. The matrix $\bar{F}_k$ is equal to the unit lower triangular F_k from before but with each row scaled by a certain minor of A_{lin} which is known a priori to clear any denominators. The output (12) is also fraction-free, that is, all entries are minors of A_{lin} of dimension bounded by the rank of R_k. The nullspace applications in Phase 1 can now be done in a fraction-free fashion as

$$\frac{1}{\Delta_{k+1}} \left(\left(\Delta_k \left[\begin{array}{c|cc} N_k & I \end{array} \right] P_k \right) \left(\Delta_{k+1} \left[\begin{array}{c|ccc} E_k & * & \cdots & * \\ C_k & * & \cdots & * \end{array} \right] \right) \right)$$

yielding

$$\Delta_k \left[\begin{array}{c|ccc} & E_{k-1} & \cdots & * \end{array} \right].$$

Similarly, in Phase 2 the entire Gauss transform is applied. The back substitution in Phase 3 can be done iteratively in a fraction-free fashion also [1].

Using the fraction-free approach, all intermediate quantities arising during the elimination (i.e., the entries of (11) and (12)) will thus be minors of A_{lin}. We recall some well known a priori bounds for the size of these minors for some common cases. We will use size and $\overline{\mathsf{size}}$ for the bounds for A_{lin} and $\bar{A}_{\mathsf{lin}}$ (9) respectively.

- $\mathsf{F} = \mathsf{k}[z]$ with $\deg_z A_{\mathsf{lin}}, \deg_z \bar{A}_{\mathsf{lin}} \leq e$. Multiplying the row dimension of A_{lin} and $\bar{A}_{\mathsf{lin}}$ by e gives explicit bounds for the degrees $\mathsf{size}_{\mathsf{k}[z]}$ and $\overline{\mathsf{size}}_{\mathsf{k}[z]}$ of minors of A_{lin} and $\bar{A}_{\mathsf{lin}}$ that satisfy

  $$\mathsf{size}_{\mathsf{k}[z]} \in O(n^2 de)$$

  and

  $$\overline{\mathsf{size}}_{\mathsf{k}[z]} \in O(nde).$$

- $\mathsf{F} = \mathbb{Z}$ with the magnitude of entries of A_{lin} and $\bar{A}_{\mathsf{lin}}$ bounded by β. Hadamard's inequality [11, Corollary 7.82] gives an explicit bound $2^{\mathsf{size}_{\mathbb{Z}}}$ and $2^{\overline{\mathsf{size}}_{\mathbb{Z}}}$ for the magnitudes of minors of A_{lin} and $\bar{A}_{\mathsf{lin}}$ that satisfy

  $$\mathsf{size}_{\mathbb{Z}} \in O(n^2 d \log(nd\beta))$$

  and

  $$\overline{\mathsf{size}}_{\mathbb{Z}} \in O(nd \log(nd\beta)).$$

- $\mathsf{F} = \mathbb{Z}[z]$ with $\deg_z A \leq e$, and with the magnitude of integer coefficients of entries of A_{lin} and $\bar{A}_{\mathsf{lin}}$ bounded by β. Multiplying the determinant degree bound above with the logarithm base 2 of an explicit magnitude bound for the coefficients [8] gives

  $$\mathsf{size}_{\mathbb{Z}[z]} \in O(n^4 d^2 e \log(nde\beta))$$

  and

  $$\overline{\mathsf{size}}_{\mathbb{Z}[z]} \in O(n^2 d^2 e \log(nde\beta)).$$

Now let M be a multiplication time for $\mathsf{k}[z]$, that is, two polynomials from $\mathsf{k}[z]$ with degree strictly less than t can be multiplied in $\mathsf{M}(t)$ operations from k. Then over $\mathsf{k}[z]$ a cost estimate in terms of operations from k is obtained by multiplying the algebraic cost estimates of Theorem 5.4 by $\mathsf{M}(\mathsf{size}_{\mathsf{k}[z]})$. Note that the polynomial multiplication can be done modulo z^p for $p = 2\,\mathsf{size}_{\mathsf{k}[z]} + 1$ to control degrees during the fast matrix multiplications.

If M is a multiplication time for $\mathbb{Z}$, that is, two integers with bit-length bounded by t can be multiplied with $\mathsf{M}(t)$ bit operations, then a cost estimate in terms of bit operations for the cases $\mathbb{Z}$ and $\mathbb{Z}[z]$ are obtained by multiplying the algebraic cost estimates by $\mathsf{M}(\mathsf{size}_{\mathbb{Z}})$ and $\mathsf{M}(\mathsf{size}_{\mathbb{Z}[z]})$. Note that for the $\mathbb{Z}[z]$ case we can use Kronecker substitution [9] to reduce the integer polynomial multiplication to integer multiplication. Similar to the case $\mathsf{k}[z]$, the multiplication can be done modulo 2^p for an appropriate $p \in O(\mathsf{size}_{\mathbb{Z}[z]})$. We remark that faster algorithms for polynomial matrix [4] and integer matrix [10] multiplication are avaiable.

6.1 Cost analysis for some common Ore rings

Let $A \in \mathsf{F}[\partial; \sigma, \delta]^{n \times n}$ be non-singular with degree d. First consider the case $\mathsf{F} = \mathsf{k}(z)$. We will assume that A has entries over $\mathsf{k}[z]$. This can be achieved by clearing denominators, if necessary. As in [7], we will assume that $\sigma(z) \in \mathsf{k}[z]$ and $\deg_z \delta(z) \leq 1$. Then $\partial z = \sigma(z)\partial + \delta(z) \in \mathsf{k}[z][\partial]$ and the degree in z and in ∂ remains unchanged. The linearized systems will thus be over $\mathsf{k}[z]$, allowing application of our fraction-free algorithm.

THEOREM 6.1. *Let $A \in \mathsf{k}[z][\partial; \sigma, \delta]^{n \times n}$ be non-singular with $\deg A \leq d$ and $\deg_z A \leq e$. If A is row reduced, this can be detected and the Popov form of A computed in $O(n^3 d^2\, \mathsf{M}(nde))$ operations from k. If A is not row reduced, then cost estimates in term of operations from k for computing the Popov form are obtained by multiplying the O-estimates from Theorem 5.4 by $\mathsf{M}(n^2 de)$.*

PROOF. To test if A is row reduced we can check if its leading coefficient matrix is non-singular. This check will not dominate the cost. The theorem now follows from Theorems 5.1 and 5.4 and the estimates for $\overline{\mathsf{size}}_{\mathsf{k}[z]}$ and $\mathsf{size}_{\mathsf{k}[z]}$. □

Now consider the case $\mathsf{F} = \mathbb{Q}(z)$. As before, we will assume that A has entries over $\mathbb{Z}[z]$. Then A has entries polynomials

in ∂ whose coefficients are polynomials in $\mathbb{Z}[z]$; let $||A||_\infty$ denote the largest in absolute value of any (integer) coefficient of any of these $\mathbb{Z}[z]$ coefficients.

THEOREM 6.2. *Let $A \in \mathbb{Z}[z][\partial;\sigma,\delta]^{n\times n}$ be non-singular with $\deg A \leq d$ and $\deg_z A \leq e$. Suppose our Ore ring is either the differential polynomials (where $\sigma(z) = z$, $\delta(z) = 1$) or the shift polynomials (where $\sigma(z) = z+1$, $\delta(z) = 0$). If A is row reduced, this can be detected and the Popov form of A can be computed in $O(n^3d^2)$ operations with integers bounded in length by $O\tilde{}(n^2d^2e(\log||A||_\infty + e))$ bits. If A is not row reduced, then cost estimates for comuting the Popov form in terms of operations on integers bounded in length by $O\tilde{}(n^4d^2e(\log||A||_\infty + e))$ bits are given by Theorem 5.4.*

PROOF. As mentioned in Section 6, we can use Kronecker substitution to reduce arithmetic operations from $\mathbb{Z}[z]$ to integer arithmetic. As before, we can test if A is row reduced by checking if its leading coefficient matrix is non-singular using fraction-free gaussian elimination. From the proof of [7, Corollary 5.9] we have that $\log\beta := \log||A_{\mathsf{lin}}||_\infty \in O(\log||A|| + e\log(nd))$. The theorem now follows from Theorems 5.1 and 5.4 and the estimates for $\overline{\mathsf{size}}_{\mathbb{Z}[z]}$ and $\mathsf{size}_{\mathbb{Z}[z]}$. □

7 CONCLUSION

For the shift and differential Ore rings we bound the bit complexity of our algorithm when the input matrix is over $\mathbb{Z}[z]$. Our algorithm is faster than the row reduction algorithm of [5], which however handles the important case of non-square and singular inputs which we do not. Our approach still works for rectangular and rank deficient matrices (of any dimensions) but the complexity could be much higher. Making the algorithm as efficient for rectangular and rank deficient matrices is left for future work.

REFERENCES

[1] E. H. Bareiss. 1968. Sylvester's Identity and Multistep Integer-Preserving Gaussian Elimination. *Math. Comp.* 22, 103 (1968), 565–578.

[2] Bernhard Beckermann, Howard Cheng, and George Labahn. 2006. Fraction-Free Row Reduction of Matrices of Ore Polynomials. *J. Symb. Comp.* 41, 5 (2006), 513–543.

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

[4] A. Bostan and E. Schost. 2005. Polynomial evaluation and interpolation on special sets of points. *Journal of Complexity* 21, 4 (2005), 420–446. Festschrift for the 70th Birthday of Arnold Schönhage.

[5] H. Cheng and G. Labahn. 2007. Modular computation for matrices of Ore polynomials. *Computer Algebra 2006: Latest Advances in Symbolic Algorithms* (2007), 43–66.

[6] J. Edmonds. 1967. On Systems of Distinct Linear Representative. *J. Res. Nat. Bur. Standards* 71B (1967), 241–245.

[7] M. Giesbrecht and M. S. Kim. 2012. Computing the Hermite form of a matrix of Ore polynomials. *Journal of Algebra* 376 (2012), 341–362.

[8] A. J. Goldstein and R. L. Graham. 1974. A Hadamard-type bound on the coefficients of a determinant of polynomials. *SIAM Review* 16 (1974), 394–395.

[9] D. Harvey. 3009. Faster polynomial multipication via multipoint Kronecker substitution. *Journal of Symbolic Computation* 44, 10 (3009), 1502–1510.

[10] D. Harvey and Joris van der Hoeven. 2014. On the commplexity of integer matrix multiplication. (2014). https://hal.archives-ouvertes.fr/hal-01071191.

[11] R. A. Horn and C. R. Johnson. 1985. *Matrix Analysis*. Cambridge University Press, Cambridge, UK.

[12] C. P. Jeannerod, V. Neiger, É. Schost, and G. Villard. 2017. Computing minimal interpolation bases. *Journal of Symbolic Computation* 83 (2017), 272–314.

[13] T. Kailath. 1980. *Linear Systems*. Prentice Hall, Englewood Cliffs, N.J.

[14] S. E. Labhalla, H. Lombardi, and R. Marlin. 1996. Algorithmes de Calcul de la Réduction d'Hermite d'une Matrice à Coefficients Polynomiaux. *Theoretical Computer Science* 161, 1&2 (1996), 69–92.

[15] H. R. Lee and B. D. Saunders. 1995. Fraction free Gaussian elimination for sparse matrices. *Journal of Symbolic Computation* 19, 5 (1995), 393–402.

[16] O. Ore. 1933. Theory of non-commutative polynomials. *Annals of Mathematics* 34 (1933), 480–508.

[17] G. Labahn P. Davies, H. Cheng. 2008. Computing Popov form of general Ore polynomial matrices. In *Proceedings of Milestones in Computer Algebra*. Scarborough, Tobago, 149–156.

[18] A. Storjohann. 2000. *Algorithms for Matrix Canonical Forms*. Ph.D. Dissertation. Swiss Federal Institute of Technology, ETH–Zurich.