RAPPORT TECHNIQUE

Some algorithms for matrix polynomials

G. Villard

RT157 **Part 2** **Mars 96**

Résumé

Pour une matrice polynomiale $P(x)$ de degré d dans $\mathcal{M}_{n,n}(K[x])$, où K est un corps commutatif, une réduction à la forme normale d'Hermite peut être calculée en $\tilde{O}(M(nd))$ opérations arithmétiques, si $M(n)$ est le coût du produit de deux matrices $n \times n$ sur K. De plus, une telle réduction peut être obtenue en temps parallèle $O(\log^{\lambda+1}(nd))$ en utilisant $O(L(nd))$ processeurs, si le problème de déterminer le premier sous-ensemble indépendant de l'ensemble des colonnes d'une matrice $nd \times nd$ sur K, peut être résolu en temps $O(\log^{\lambda}(nd))$ avec le même nombre de processeurs.
Ces résultats sont obtenus en généralisant au cas matriciel certaines techniques utilisées pour le pgcd de polynômes scalaires.

Mots clefs : algorithmes parallèles, $\mathcal{NC}_K^2$, forme normale d'Hermite, théorie des réalisations.

FAST MATRIX POLYNOMIAL COMPUTATIONS *

Gilles VILLARD †

Abstract. For a polynomial matrix $P(x)$ of degree d in $\mathcal{M}_{n,n}(K[x])$ where K is a commutative field, a reduction to the Hermite normal form can be computed in $\tilde{O}(M(nd))$ arithmetic operations if $M(n)$ is the time required to multiply two $n \times n$ matrices over K. Further, a reduction can be computed using $O(\log^{\lambda+1}(nd))$ parallel arithmetic steps and $O(L(nd))$ processors if the same processor bound holds with time $O(\log^{\lambda}(nd))$ for determining the lexicographically first maximal linearly independent subset of the set of the columns of an $nd \times nd$ matrix over K. These results are obtained by applying in the matrix case, the techniques used in the scalar case of the gcd of polynomials.

Keywords: Parallel algorithms, $\mathcal{NC}_K^2$, Hermite normal forms, theory of realizations.

Introduction

The problem of computing the *greatest common divisor* (gcd) of scalar polynomials in $K[x]$ (K is a commutative field) or of polynomial matrices in $\mathcal{M}_{m,n}(K[x])$ has attracted a lot of attention and has many applications in linear systems, control and realization theory, see [20] and references therein. The reverse approach has also being widely considered, that is the use of linear system theory and especially of the *theory of realizations* [21,15] for the computation of scalar polynomial gcd's [2,3]. This has led to various computational results: giving either efficient numerical procedures as for instance developped in [24] or both sequential and parallel complexity results [6,7].

From these latter points of view, the *multivariable case*, involving computations with polynomial matrices [26], is much less studied. Precisely, this paper aims at using standard results from linear systems theory to obtain new and better complexity bounds for the computation of the Hermite normal form via the polynomial matrix gcd. Our approach uses a generalization from the scalar case to the matrix case of certain algorithm for computing the gcd of polynomials.

Let $P(x)$ a matrix in $\mathcal{M}_{n,n}(K[x])$ of degree d. To not excessively burden our presentation, we assume the determinant of $P(x)$ to be non identically zero. The general case should be quite easily derived. Throughout the paper we use right (column) or left (row) equivalence. Two polynomial matrices are right (resp. left) equivalent is they differ by a right (resp. left) unimodular factor. In the same way, the different normal forms will be either obtained by right equivalence and be *column forms* or by left equivalence and be *row forms*. For each form we give a unique definition by column operations, the other one follows obviously.

There exists [29] a unique matrix $L_H(x)$ right equivalent to $P(x)$ and under (column) Hermite normal form:

$$L_H(x) = P(x)V(x)$$

with $V(x)$ unimodular in $\mathcal{M}_{n,n}(K[x])$. The Hermite form is an upper triangular form which diagonal entries are monic and such that in each row the entries following the diagonal entry are of lower degree.

---

*Manuscript submitted to ISSAC96.
†LMC-IMAG, 46 Av. F. Viallet, F-38031 Grenoble Cedex, `Gilles.Villard@imag.fr`.

• *Sequential point of view.* Over an abstract field K the Hermite form can be computed in sequential time $\tilde{O}(n^4 d)$ [18]. To consider a unique parameter, let μ be such that $n = O(\mu)$ and $d = O(\mu)$. The previous complexity is $\tilde{O}(\mu^5)$. The notation $\tilde{O}$ stands for a big "O" up to log terms, $\tilde{O}(n)$ is $O(n \log^\alpha n)$ for some positive integer α. Fast matrix multiplication techniques have not be applied to the computation of the Hermite form but only to the problem of finding a unimodular triangularization $R_T(x) = P(x)W(x)$ (then the Hermite form is obtained by reducing off-diagonal entries). Such a triangularization is computed in time $\tilde{O}(ndM(n))$ [16] if $M(n)$ is the complexity of matrix product. The same complexity holds for the best randomized solution [39].

Thus, a first problem was to apply fast matrix multiplication to the Hermite form computation. We will partially solve it by proposing a solution with deterministic sequential time $\tilde{O}(ndM(n) + M(nd))$. Since one can take $M(n) = n^{2.38}$, we thus slightly improve the asymptotic sequential complexity giving the new bound $\tilde{O}(\mu^{4.8})$. At this point, we may emphasize the two key ideas of the new algorithm. On the one hand, instead of directly computing the Hermite form of $P(x)$, it will be deduced from an intermediate form that is widely used in linear system theory – the *Popov form* [35,20]. On the other hand, the computation of this latter form for $P(x)$ will be reduced to the computation of a well chosen matrix polynomial gcd. This latter problem will be solved using block Hankel matrices (being inspired by the scalar case).

Using standard polynomial and matrix multiplications the best previous algorithms was requiring $O(n^5 d^2) = O(\mu^7)$ arithmetic operations for the elimination over polynomials in [18] or $O(n^4 d^3) = O(\mu^7)$ for the elimination over constants in [27]. The new deterministic one, of cost $\tilde{O}(n^3 d^3) = \tilde{O}(\mu^6)$ is thus also better and will match the cost of the best randomized solution [39] for a unimodular triangularization.

• *Parallel point of view.* Very few algorithms exist that compute the Hermite form fast using polynomially many processors. The solutions in [18,16] are elimination processes and are thus highly sequential. The problem has been shown to belong to the class $\mathcal{NC}$ in [22]. Nevertheless, this latter approach involves $O(n^2 d)$ structured linear systems of dimension $n^3 d$. Since quite prohibitive, the cost has not been precisely computed by the authors. For a number of parallel steps in $O(\log^3 \mu)$, one can evaluate the number of needed processors to be in $O(\mu^{11})$ over fields of characteristic zero and in $O(\mu^{18})$ over any fields. Another solution could be to compute the form by obtaining (in parallel) a row echelon form of a generalized Sylvester matrix – Toeplitz-like – as done in [8,27] (in sequential). For a time $O(\log^3 \mu)$, one can also evaluate the number of processors (this solution has not been investigated) to be in $O(\mu^9)$ over fields of characteristic zero and in $O(\mu^{16.5})$ over any fields. To the knowledge of the authors, no better deterministic bound is available.

The parallel version of our algorithm will substantially decrease these costs. An implementation will be possible in time $O(\log^4(nd))$ using $O\left(n^{3/2} dM(n) + (nd)^{1/2} M(nd)\right)$ or $O(\mu^{5.8})$ processors over fields of characteristic zero. Over any field the new bound is $\tilde{O}\left((nd)^4 M(nd)\right)$ or $O(\mu^{12.8})$.

The processor inefficiency of the algorithm, defined as the ratio of the parallel work over the sequential complexity is $O((nd)^{1/2}) = O(\mu)$ for fields of characteristic zero and $\tilde{O}(\mu^7)$ in the general case. These values essentially come from the inefficiency of known solutions [7] to the problem of computing the nullspace of a matrix, or more precisely, as we will see, to the problem of determining the lexicographically first *maximal linearly independent subset* of the set of the columns of a matrix.

These complexities does not express the boolean computational cost of the algorithms. We will not detail the corresponding studies in this paper. Further developments should be done [38] using the Bezoutian for matrix polynomials [4,37]. From results in [7] concerning the scalar polynomial gcd, very good bounds may be expected.

The paper is organized as follows: we first remind some results on the *polynomial matrix gcd* in §1 and on the *minimal realization problem* in §2. Then we will see in §3 how these suitable results leads to the algorithm for computing the *Popov form*. This is the first step toward the Hermite form. The Popov form is read-off the *nullspace of a block Hankel matrix* which entries are computed from the input matrix $P(x)$. Based on this result, we then devise a *sequential algorithm* and a *parallel algorithm* in §4 for computing the *Hermite normal form* of an invertible matrix from the previously computed Popov form. The cost analysis will be done in §5.

1 Polynomial matrix gcd

A *greatest common right divisor* (gcrd) of two polynomial matrices can be defined provided they have the same number of columns. In the following we will restrict ourselves to the case where $N(x)$ is in $\mathcal{M}_{m,n}(K[x])$ and $D(x)$ is nonsingular in $\mathcal{M}_{n,n}(K[x])$.

1.1 Definition

A gcrd of $N(x)$ and $D(x)$ is [29] any polynomial matrix $G(x)$ such that I) $G(x)$ is a right divisor of $N(x)$ and $D(x)$: for some matrices $\bar{N}(x)$ and $\bar{D}(x)$ we have

$$N(x) = \bar{N}(x)G(x), \quad D(x) = \bar{D}(x)G(x);$$

II) any other right divisor $G_1(x)$ of $N(x)$ and $D(x)$ is a right divisor of $G(x)$: $G(x) = T(x)G_1(x)$ for some polynomial matrix $T(x)$. Gcrd's are not unique, but if ${}^t[{}^tN(x), {}^tD(x)]$ is of rank n – here this is true by assumption – then all gcrd's of $N(x)$ and $D(x)$ must be nonsingular in $\mathcal{M}_{n,n}(K[x])$ and left equivalent. In order to define the gcrd uniquely, it is thus sufficient to consider any row normal form to represent the equivalence class of all the gcrd's. The row Hermite form is a possible choice; besides, one way to compute a gcrd [29] is to compute the Hermite form $L_H(x)$ of ${}^t[{}^tN(x), {}^tD(x)]$ by left equivalence:

$$U(x)\begin{bmatrix} N(x) \\ D(x) \end{bmatrix} = \begin{bmatrix} G(x) \\ 0 \end{bmatrix}. \tag{1}$$

And $G(x)$ is a gcrd of $N(x)$ and $D(x)$. But precisely, to devise efficient algorithms for the Hermite form, we are going to adopt the converse approach, first computing a gcrd and then obtaining the form itself.

1.2 The Popov form

Another choice to represent the gcrd's it to use the Popov form. This will be more judicious because the highest degree of the entries of the Popov form of a matrix, will be no greater than the degree of the matrix. This property being clearly false for the Hermite form. We define the column Popov form. For $G(x)$ in $\mathcal{M}_{n,n}(K[x])$ let d_j, $j \leq 1 \leq n$, the j-th column degree, be the degree of the j-th column of $G(x)$. The coefficient vector of

x^{d_j} is the j-th leading column coefficient vector. We let $[G(x)]_c$ be the matrix of these leading vectors. A matrix $G(x)$ is said to be *column reduced* (or column proper in [43]) if rank $[G(x)]_c =$ rank $G(x)$, it satisfies:

$$\deg \det G(x) = \Sigma_j^n d_j.$$

If, in addition, $G(x)$ satisfies the following properties, we shall say that $G(x)$ is in *Popov form*: I) the column degrees are increasigly ordered; II) the last entry of degree d_j in each column is monic, it is called the pivot of column j with row index r_j; III) if $d_j = d_k$ and $j < k$ then $r_j < r_k$; IV) all entries in a row containing a pivot element have degrees lower than that of the pivot element. The Popov form is normal in the usual sens and satisfies a degree property.

Proposition 1.1 ([35,20]) *Any two right (resp. left) equivalent matrices in $\mathcal{M}_{n,n}(K[x])$ of degree d_1 and d_2 have the same column (resp. row) Popov form of degree at most $min\{d_1, d_2\}$.*

Example 1. The following matrix $P(x)$ is not column reduced, rank $[P(x)]_c = 1$.

$$\begin{bmatrix} -2x - 2 + x^3 & 4x + 5 - x^3 \\ -x^2 + 2 + 4x & 2x^2 - 5 - 4x \end{bmatrix}.$$

The next one is right equivalent to $P(x)$ and is column reduced but not under Popov form since it does not obey requirement IV):

$$\begin{bmatrix} 2x + 3 & 4x + 5 - x^3 \\ x^2 - 3 & 2x^2 - 5 - 4x \end{bmatrix}.$$

The last matrix is also right equivalent to $P(x)$, now it is under Popov form:

$$\begin{bmatrix} 2x + 3 & 1 + x^3 \\ x^2 - 3 & -1 + 4x \end{bmatrix}$$

with $d_1 = 2, d_2 = 3$ and $r_1 = 2, r_2 = 1$. □

We may point out some related works. The univariate determinant computation in [30] may be viewed as the computation of a column reduced form. More generally, strong links are well knowns between column reduced forms and the determination of the finite eigenstructure of a polynomial matrix [41,40], and also with the factorization algorithm of Kublanovskaya [5].

As said previously the Popov form will be the first step toward the Hermite form. It will be computed in §3 by a polynomial matrix gcrd operation. Apart from using matrix equivalence and indentity (1), a gcrd can be computed from the generalized Sylvester resultant matrix in [8]. However this method seems to lead to the same conclusions than when using the Sylvester matrix in [27] and does not seem fully appropriate to parallelism. Indeed the two methods differ only by the form of the gcrd that is computed (Hermite *versus* Popov form) and suffer from the fact that the degree of the associated transformation matrix can be in $O(nd)$, even though the degree of the input matrix and of the Popov form of the gcrd is in $O(d)$.

A last method for polynomial gcrd is to compute an irreducible fraction. The right matrix fraction description

$$H(x) = N(x)D^{-1}(x) \tag{2}$$

is irreducible if $N(x)$ and $D(x)$ are right coprime or, equivalently, if any gcrd of $N(x)$ and $D(x)$ is a unimodular matrix. The same can be defined in a dual manner on the left with $H(x) = C^{-1}(x)L(x)$. If a right matrix fraction description is not irreducible, suppose we have a way to compute an irreducible description $\bar{N}(x)\bar{D}^{-1}(x)$ of $H(x)$, then a gcrd of $N(x)$ and $D(x)$ is computed as $G(x) = \bar{D}^{-1}(x)D(x)$ [8]. This is basically the approach used for the scalar polynomial gcd in [6,7]. We are going to remind in next section that the same can be done for matrices, and how we can compute an irreducible fraction.

2 The minimal realization problem

From a system-theoretical point of view, normal forms appear from the general *minimal (or partial) realization problem* [21,15].

2.1 The problem

Consider $H(x) = N(x)D^{-1}(x)$ a strictly proper rational matrix:

Definition 2.1 *A fraction $u(x)/v(x)$ is strictly proper if deg $u <$ deg v. A matrix fraction in $\mathcal{M}_{m,n}(K(x))$ is strictly proper if all its entries are strictly proper.*

Thus $H(x)$ has a formal expansion at the infinity

$$H(x) = \Sigma_{i=1}^{\infty} H_i x^{-i} \tag{3}$$

where the H_i's are matrices in $\mathcal{M}_{m,n}(K)$. A triplet $\Sigma = (C, A, B)$ of matrices in $\mathcal{M}_{m,\nu}(K)$, $\mathcal{M}_{\nu,\nu}(K)$ and $\mathcal{M}_{\nu,n}(K)$ respectively such that:

$$CA^{i-1}B = H_i \ \text{ for } \ i = 1, \ldots \tag{4}$$

is called a *realization* of $H(x)$. Moreover Σ satisfies:

$$H(x) = C(xI - A)^{-1}B. \tag{5}$$

The realization is called *minimal* if the dimension ν is minimal, one say that $H(x)$ as transfer function have minimal realization Σ. Given $H(x)$, the problem is to find such a minimal realization. The H_i's are called the *Markov parameters*. In the scalar case $(n, m = 1)$, for a transfer function $h(x) = u(x)/v(x)$, a minimal realization gives a new representation of $h(x)$ as the quotient of two relatively prime polynomials:

$$h(x) = \frac{\bar{u}(x)}{\bar{v}(x)}.$$

So the reduction of the problem of computing the gcd of two polynomials $u(x)$ and $v(x)$ to the minimal realization problem is obvious. From $u(x)$ and $v(x)$ compute a minimal realization of the h_i's, this realization gives $\bar{u}(x)$ and $\bar{v}(x)$ and consequently the gcd. We refer to [6,7] for corresponding algorithms.

In the multivariable case, for a transfer matrix given as a matrix fraction $H(x) = N(x)D^{-1}(x)$, a minimal realization of $H(x)$ provides an irreducible fraction $\bar{N}(x)\bar{D}^{-1}(x)$ of $H(x)$ with $\bar{D}(x)$ under special form [26,20]. Thus the polynomial matrix gcd problem is also reduced to the minimal realization problem.

2.2 A minimal realization from the Markov parameters

For $H(x) = N(x)D^{-1}(x)$ strictly proper rational, we now look at the problem of finding a minimal realization matching the corresponding set of Markov parameters H_i, $i \geq 1$, given by (3) and (4). The problem can be solved by finding two right coprime matrices $\bar{N}(x)$ and $\bar{D}(x)$ such that

$$H(x) = N(x)D^{-1}(x) = \bar{N}(x)\bar{D}^{-1}(x) = \Sigma_{i=1}^{\infty} H_i x^{-i} \tag{6}$$

and such that $\bar{D}(x)$ is column reduced. For any integer $k \geq 1$, the H_i's define the block Hankel $km \times kn$ matrix $M(k)$ whose (i,j) block is H_{i+j-1}. From (6), the special structures of the denominator matrix $\bar{D}(x)$ are going to be reflected in special relations between the columns of $M(\nu+1)$.

Propositions below are very useful facts derived from [26,20]. They will directly lead, at next section, to a procedure for fraction reduction. Our scheme of proof is inspired from the scalar case in [6,7]. We give two propositions. Their two assertions correspond to the "if" and the "only if" part respectively of the following theorem which proof is omitted.

Theorem 2.1 (theorem 6.5-1, [20]). *A matrix fraction description is irreducible if and only if the determinantal degree of the denominator matrix is the dimension of any minimal realization of the fraction (right and left irreducible descriptions of the same fraction have the same denominator determinantal degree).*

The following result is derived from [26,20]. We state it by emphasizing the degree of the denominator matrix of the reduced fraction, this will be a relevant parameter to obtain good complexity bounds. The determinantal degree of the matrices plays the role of the polynomial degree during the euclidean division in the scalar case.

Proposition 2.1 *Let $H(x) = N(x)D^{-1}(x)$ be strictly proper in $\mathcal{M}_{m,n}(K(x))$ with $D(x)$ in $\mathcal{M}_{n,n}(K[x])$. Let ν be the dimension of a minimal realization of $H(x)$. If $H(x)$ has a right irreducible description $\bar{N}(x)\bar{D}^{-1}(x)$ and a left irreducible description $\bar{C}^{-1}(x)\bar{L}(x)$ such that the denominator matrices $\bar{D}(x)$ and $\bar{C}(x)$ have degrees bounded by δ, then rank $M(\delta) = \nu$.*

Proof (Compare proposition 2.1 in [6] for the scalar polynomial gcd). By minimality of ν we know that rank $M(\nu+i) =$ rank $M(\infty) = \nu$, $i \geq 0$ (lemma 6.5-7, [20]). We apply the "if" part of theorem 2.1. Since $\bar{N}(x)\bar{D}^{-1}(x)$ is irreducible, the determinantal degree of $\bar{D}(x)$ is ν. Using proposition 1.1, $\bar{D}(x)$ can be brought in Popov form $T(x)$ up to a right unimodular factor: $\bar{D}(x)V(x) = T(x)$. The degree of the entries of $T(x)$ are also bounded by δ the degree of $\bar{D}(x)$. We denote by T_i the i-th coefficient matrix of $T(x)$. Let $S(x) = \bar{N}(x)V(x)$, by (6), $S(x)T^{-1}(x) = \Sigma_{i=1}^{\infty} H_i x^{-i}$ or $(H_1 x^{-1} + H_2 x^{-2} + \ldots)(T_0 + T_1 x + \ldots T_\delta x^\delta) = S(x)$. Since $S(x)$ is a matrix polynomial, comparing the coefficient of x^{-i}, $i \geq 1$, on both side of the equation we obtain:

$$H_i T_0 + H_{i+1} T_1 + \ldots + H_{i+\delta} T_\delta = 0, \quad i \geq 1. \tag{7}$$

Thus there is a strong relationship between the entries of $T(x)$ and the dependency relations between the columns of $M(\nu+1)$. Indeed, let $T(x) = \{\Sigma_{k=0}^{\delta} t_{ljk} x^k\}$. From the definition of the Popov form, let d_j be the j-th column degree of $T(x)$ and let r_j be the index of the corresponding pivot entry. Identity (7) for $i \geq 1$ and for the j-th column of $T(x)$ gives:

$$M_{r_j}^{(d_j+1)} = \Sigma_{l=1}^{r_j-1} M_l^{(d_j+1)} t_{ljd_j} + \Sigma_{l=1}^{n} \Sigma_{k=0}^{d_j-1} M_l^{(k+1)} t_{ljk} \tag{8}$$

for $1 \leq j \leq n$ and with $M_l^{(k)}$ denoting the column l of the k-th block column of $M(\nu+1)$. For any l, only the columns $M_{r_l}^{(1)}, \ldots, M_{r_l}^{(d_j+1)}$ can appear in the right hand side of (8); the fact that $T(x)$ is in Popov form (requirement IV) implies that only the columns $M_{r_l}^{(1)}, \ldots, M_{r_l}^{(d_l)}$ can appear. So, if we look at all the identities (8) when j varies, $1 \leq j \leq n$, for each r_j, only the columns $M_{r_j}^{(1)}, \ldots, M_{r_j}^{(d_j)}$ can appear in some right hand side. By assumption there are $\Sigma_1^n d_j = \nu$ such columns. We denote $\bar{M}_\delta$ the submatrix of $M(\delta)$ built from these ν columns taken in the order they appear in $M(\delta)$. If any column degree is zero, say the j_0-th, then the columns $\bar{M}_{j_0}^{(k)}$, $k \geq 1$, are not involved in $\bar{M}_\delta$. We denote by ρ, $1 \leq \rho \leq n$, the number of nonzero column degrees. If π is an index permutation such that the $r_{\pi(j)}$ are increasingly ordered, $d_{\pi(j)} \neq 0$ for $1 \leq j \leq \rho$ and $d_{\pi(j)} = 0$ for $j \geq \rho + 1$,

$$\bar{M}_\delta = \left[\bar{M}_{r_{\pi(1)}}^{(1)}, \ldots, \bar{M}_{r_{\pi(\rho)}}^{(1)}, \ldots\right] \in \mathcal{M}_{\delta m, \nu}(K)$$

with $\bar{M}_l^{(k)}$ denoting the column l of the k-th block column of $M(\delta)$. With this choice of columns in $M(\delta)$, it is natural to consider the corresponding ν rows of ${}^t[{}^tT_0, {}^tT_1, \ldots, {}^tT_{\delta-1}]$ as a $\nu \times n$ matrix:

$${}^t[{}^t\bar{T}_0, {}^t\bar{T}_1, \ldots, {}^t\bar{T}_{\delta-1}]$$

and the rows $d_{\pi(1)} + 1, \ldots, d_{\pi(n)} + 1$ form $\bar{T}_\delta$. The matrix $\bar{T}_\delta$ is a $n \times n$ permutation matrix since it is a submatrix of $[T(x)]_c$ and since the pivot entries in $T(x)$ are monic. Now, we may rewrite relations (7) and (8) in matrix form:

$$\begin{aligned} &\underbrace{\left[\bar{M}_{r_{\pi(1)}}^{(1)}, \ldots, \bar{M}_{r_{\pi(\rho)}}^{(1)}, \ldots\right]}_{\bar{M}_\delta} \begin{bmatrix} \bar{T}_0 \\ \bar{T}_1 \\ \vdots \\ \bar{T}_{\delta-1} \end{bmatrix} \bar{T}_\delta^{-1} \\ &\quad = \left[\bar{M}_{r_{\pi(1)}}^{(d_{\pi(1)}+1)}, \ldots, \bar{M}_{r_{\pi(\rho)}}^{(d_{\pi(\rho)}+1)}, \bar{M}_{r_{\pi(\rho+1)}}^{(1)}, \ldots\right]. \end{aligned} \tag{9}$$

We are going to show that $\bar{M}_\delta$ (and thus $M(\delta)$) is of rank ν. By contradiction, if it is not, we show that the fractions $\bar{N}(x)\bar{D}^{-1}(x)$ and $\bar{C}^{-1}(x)\bar{L}(x)$ was not irreducible. Indeed, if $\bar{M}_\delta$ has rank strictly lower than ν, the linear system (9) gives other solutions than $T(x)$. One such solution, $T'(x)$, can be constructed to have (at least) one column degree strictly lower than the corresponding column degree of $T(x)$, the others being lower or equal. Thus $T'(x)$ has determinantal degree strictly lower than ν. We also construct the associated numerator $S'(x)$ using [20]:

$$\begin{bmatrix} S'_{\delta-1} \\ \vdots \\ S'_0 \end{bmatrix} = \begin{bmatrix} H_1 & 0 & \ldots & 0 \\ H_2 & H_1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ H_\delta & \ldots & \ldots & H_1 \end{bmatrix} \begin{bmatrix} T'_\delta \\ \vdots \\ T'_1 \end{bmatrix} \tag{10}$$

Now, the expansions of $H(x)$ using $\bar{C}^{-1}(x)\bar{L}(x)$ and using $S'(x)(T'(x))^{-1}$ at infinity coincide up to the order 2δ:

$$\bar{C}^{-1}(x^{-1})\bar{L}(x^{-1})/x - S'(x^{-1})(T'(x^{-1}))^{-1}/x \equiv 0 \bmod x^{2\delta}$$

then there exists a polynomial matrix $Q(x)$ such that

$$\bar{L}(x^{-1})T'(x^{-1}) - \bar{C}(x^{-1})S'(x^{-1}) = x^{2\delta+1}\bar{C}(x^{-1})Q(x)T'(x^{-1}).$$

But $\bar{C}(x)$ and $T'(x)$ have degrees lower than δ, that is

$$\bar{L}(x^{-1})T'(x^{-1}) - \bar{C}(x^{-1})S'(x^{-1}) \equiv 0 \bmod x$$

and

$$\bar{L}(x)T'(x) - \bar{C}(x)S'(x) = 0.$$

Now, since $\bar{C}^{-1}(x)\bar{L}(x)$ is irreducible, the determinantal degree of $\bar{C}(x)$ must be (lemma 6.3-8, [20]) at most equal to the one of $T'(x)$ and thus strictly lower than ν. This contradicts the irreducibility of the fractions. □

Proposition 2.2 *Let $H(x) = N(x)D^{-1}(x)$ be as in proposition 2.1. If $\bar{D}(x) = D_\delta x^\delta + \ldots + D_0$ in $\mathcal{M}_{n,n}(K[x])$ is of determinantal degree ν and such that*

$$\underbrace{\begin{bmatrix} H_1 & H_2 & \ldots & H_{\delta+1} \\ H_2 & H_3 & \ldots & H_{\delta+2} \\ \vdots & & & \vdots \\ H_{\delta+1} & \ldots & \ldots & H_{2\delta+1} \end{bmatrix}}_{M(\delta+1)} \begin{bmatrix} D_0 \\ D_1 \\ \vdots \\ D_\delta \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{11}$$

then $D(x) = \bar{D}(x)G(x)$ where $G(x)$ is a gcrd of $N(x)$ and $D(x)$.

Proof (Compare proposition 3.6 in [6] for the scalar polynomial gcd). This is the “only if” part of theorem 2.1 and corresponds to the method proposed in [26] to compute a minimal realization. Indeed, once $\bar{D}(x)$ in Popov form is computed, a satisfying triplet (C, A, B) can be readily obtained [20]. If $\bar{D}(x)$ is nonsingular and satisfies (11) then one can built a description $\bar{N}(x)\bar{D}^{-1}(x)$ of $H(x)$ (using arguments similar to those used in the proof of proposition 2.1). And since the determinantal degree of the denominator is ν we know that the fraction is irreducible. Using for instance lemma 6.5-5 in [20], we get that $\bar{D}(x) = D(x)G^{-1}(x)$ where $G(x)$ is a gcrd of $N(x)$ and $D(x)$ (the fraction has been “simplified by” $G(x)$). □

In the rest of the paper, we will view the above propositions as giving a procedure to reduce a matrix fraction (or, equivalently, to compute a gcrd).

2.3 Reduction of a matrix fraction

We now relate the fraction reduction to the problem $\mathcal{L}(n)$ of finding the *lexicographically first maximal linearly independent subset of the set of the columns of a matrix in $\mathcal{M}_{n,n}(K)$*. Here, the problem is applied to a block Hankel matrix. We may (reasonably) assume that solving this problem is more expensive than matrix product or inversion.

Proposition 2.3 *Let $H(x) = N(x)D^{-1}(x)$ in $\mathcal{M}_{m,n}(K(x))$, $m \le n$, be as in proposition 2.1, in particular there exist left and right irreducible fraction description of $H(x)$ whose denominator matrices have degrees bounded by δ. A reduced fraction description $H(x) = \bar{N}(x)\bar{D}^{-1}(x)$ with $\bar{D}(x)$ under Popov form can be computed in two solutions of the problem $\mathcal{L}(n\delta)$ for a block Hankel matrix with $m \times n$ blocks.*

Proof. We first compute the matrix $M(\delta+1)$. This is done by computing for each entry $h_{i,j}(x)$ of $H(x)$ the Taylor expansion of $h_{i,j}(x^{-1})/x$ (this is clearly of lower cost than the rest). Now, from the linear system (11) we compute the linear system (9). Applying $\mathcal{L}(n\delta)$ to $M(\delta+1)$ gives by construction the submatrix $\bar{M}_\delta$. Since we choose the lexicographically first set of columns, the corresponding dependencies give a column reduced matrix $T(x)$ that is under Popov form up to a column permutation. This matrix is obtained by solving the system (9), for instance by isolating a square invertible submatrix of $\bar{M}_\delta$ using a second solution of $\mathcal{L}(n\delta)$ as done in [9] for the nullspace computation. Then $\bar{D}(x)$ is computed under Popov form simply by reordering the columns of $T(x)$ with respect to their degrees, and $\bar{N}(x)$ is deduced from identity (10). By construction, $\bar{D}(x)$ is of determinantal degree ν, proposition 2.2 tells us that $\bar{N}(x)\bar{D}^{-1}(x)$ is irreducible. □

Costs will be detailed in §5. Before concluding, let us remind the links with the extended Euclidean algorithm (or Berlekamp-Massey algorithm) used in [10] to solve sparse linear equations. The corresponding problem and algorithm to find a generator of a matrix sequence correspond to the solution of linear systems (11) and (9). Given the sequence of the H_i's, the minimum generator computed there may be viewed as one of the column vectors of the denominator matrix $\bar{D}(x)$ of a well chosen transfer function.

3 Computing the Popov form

Applying above results, we now compute the column Popov form $T(x)$ of a matrix $P(x)$.

3.1 Direct approach

Since Popov forms appear at the denominator of irreducible fractions, the problem is to find a suitable transfer function $H(x)$ associated to $P(x)$ *i.e.* such that the Popov form of the denominator of an irreducible description of $H(x)$ is $T(x)$. But, as emphasized by [14], to study a nonsingular matrix polynomial $P(x)$ it is natural to study systems for which the transfer function is $H^*(x) = P^{-1}(x)$. Precisely we are going to see that $H^*(x)$ leads to a suitable choice $H(x)$ to compute $T(x)$. We denote by $\mathcal{I}(n,d)$ the problem of inverting a matrix of degree d in $\mathcal{M}_{n,n}(K[x])$.

Proposition 3.1 *The column Popov form of a nonsingular matrix $P(x)$ of degree d in $\mathcal{M}_{n,n}(K[x])$ can be computed in one matrix inversion $\mathcal{I}(n,d)$ and two computations of independent columns $\mathcal{L}(nd)$ of a block Hankel matrix.*

Proof. Consider $H^*(x) = P^{-1}(x)$. If we denote by $P^*(x)$ the adjoint matrix of $P(x)$ and $\Delta(x)$ the diagonal matrix of dimension n whose nonzero entries are the determinant of $P(x)$:

$$H^*(x) = P^{-1}(x) = P^*(x)\Delta^{-1}(x).$$

In the general case, $H^*(x)$ is not strictly proper. However, since we are going to focus on the reduced fraction, we can take:

$$H(x) = N(x)\Delta^{-1}(x) = (P^*(x) \bmod \Delta(x))\,\Delta^{-1}(x)$$

with the obvious definition of the *modulo* $\Delta(x)$. This new transfer function is strictly proper and $G(x)$ is a gcrd of $P^*(x)$ and $\Delta(x)$ if and only if it is a gcrd of $N(x)$ and $\Delta(x)$. Now, $P^*(x)$ is such a gcrd thus every gcrd $G(x)$ of $N(x)$ and $\Delta(x)$ is given by

$$G(x) = U(x)P^*(x)$$

with $U(x)$ unimodular. Thus any irreducible right fraction description $\bar{N}(x)\bar{\Delta}^{-1}(x)$ of $H(x)$, obtained by "extracting" a gcrd from $N(x)$ and $\Delta(x)$ satisfies

$$\bar{\Delta}(x) = \Delta(x)G^{-1}(x) = \Delta(x)(P^*(x))^{-1}U^{-1}(x) = P(x)V(x)$$

with $V(x)$ unimodular. From the uniqueness of the Popov form, to compute the Popov form $T(x)$ of $P(x)$ thus reduces to compute an irreducible fraction description of $H(x)$ with denominator $\bar{D}(x)$ under Popov form: $T(x) = \bar{D}(x)$. Since the column and the row Popov forms of $P(x)$ are at most of degree d, there exist irreducible right and left fraction descriptions of $H(x)$ with denominators of degrees at most d, we conclude by applying proposition 2.3 to $H(x)$. One additional matrix inversion is needed to build $M(d+1)$. □

3.2 Dual approach

Clearly, it is also possible to exactly match the scalar case for the polynomial gcd [6]. For $u(x)$ and $v(x)$, the authors of this paper proceed in two steps: first computing a reduced fraction $\bar{v}(x)/\bar{u}(x)$ then obtaining the gcd as $u(x)/\bar{u}(x)$. In the same way, we may consider $H(x) = A(x)\Delta^{-1}(x)$. The reduction of the fraction now gives as denominator the Popov form $\bar{P}^*(x)$ of the adjoint matrix of $P(x)$, $T(x)$ is easily deduced (row Popov form with this dual approach). This methods compares favourably with the above one in the sens that it can be easily adapted for singular matrices ($\Delta(x)$ can be build from a well chosen nonzero minor of $P(x)$) [38]. But its main drawback is that the degree of $\bar{P}^*(x)$ can be as large as nd. Thus one has to take $\delta = nd$ and to deal with $M(\delta) = M(nd)$ which is of dimensions $mnd \times n^2d$.

This dual point of view, when applied to the computation of the gcd of n polynomials, resembles the method in [24] for this latter problem which is a particular case of the Popov or of the Hermite normal form for a $1 \times n$ row matrix. Adopting the system theoretic language, $M(\delta+1)$ in proposition 2.2 has to be viewed as a *controllability matrix* of an implicitly defined linear system $S(A, B)$. In the direct approach (resp. the dual approach), the controllable modes (the uncontrollable modes) of $S(A, B)$ provide the Popov form.

4 Computing the Hermite form

By proposition 3.1 it remains to focus on the computation of the column Hermite normal form $L_H(x)$ of $P(x)$ from the Popov one $T(x)$. We may use two approaches that are in fact equivalent. Let us look at both of them. The first one works in a way analogous to the computation of the Popov form we have presented. It just consider the fraction descriptions in a form that gain from the Popov form. We also briefly expose, in spite of the repetition, an abstract point of view using modules. This will emphasize the links, both theoretically and practically, between normal forms of matrix polynomials and normal forms of matrices over a field under similarity.

4.1 First formulation

For the irreducible right fraction description with denominator $T(x)$ under Popov form, that we have computed at proposition 3.1, one can readily construct a minimal realisation $\Sigma = (C, A, B)$ (as defined in §2.1). The matrix A in $\mathcal{M}_{\nu,\nu}(K)$ can be chosen in *shift form* [35,20]:

$$A = \left[\begin{array}{ccc|ccc|c|cccc}
 & & \times & 0 & & \times & \cdots & & & & \times \\
 & & \times & 1 & & \times & \cdots & & & & \times \\
 & & \times & & 1 & \times & \cdots & & & & \times \\ \hline
0 & & \times & & & \times & \cdots & & & & \times \\
1 & & \times & & & \times & \cdots & & & & \times \\
 & 1 & \times & & & \times & \cdots & & & & \times \\ \hline
 & & \times & & & \times & \cdots & 0 & & & \times \\
 & & \times & & & \times & \cdots & 1 & & & \times \\
 & & \times & & & \times & \cdots & & 1 & & \times \\
 & & \times & & & \times & \cdots & & & 1 & \times
\end{array}\right] \in \mathcal{M}_{\nu,\nu}(K).$$

$$B = \begin{array}{c}\left[\begin{array}{cccc c ccc}
1 & 0 & \vdots & 0 & & \times & \vdots & \times \\
0 & 0 & & 0 & & \times & & \times \\
0 & 0 & & 0 & & \times & & \times \\ \hline
0 & 1 & & 0 & & \times & & \times \\
0 & 0 & & 0 & & \times & & \times \\
0 & 0 & & 0 & & \times & & \times \\ \hline
\vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\ \hline
0 & 0 & & 1 & & \times & & \times \\
0 & 0 & & 0 & & \times & & \times \\
0 & 0 & & 0 & & \times & & \times \\
0 & 0 & & 0 & & \times & & \times
\end{array}\right] \\
\underbrace{\hphantom{1\quad 0\quad \vdots\quad 0}}_{\rho}\qquad\underbrace{\hphantom{\times\quad \vdots\quad \times}}_{n-\rho}
\end{array} \in \mathcal{M}_{\nu,n}(K).$$

To construct A we map $T(x)$ onto the above structure. Only the columns j, $1 \le j \le \rho$ of $T(x)$ with nonzero column degree d_j are relevant here. From the definition of the Popov form, in each row containing a zero degree pivot, all the other entries are zero, let us call them non relevant. Up to column and row permutations we may assume that the leading $\rho \times \rho$ submatrix of $T(x)$ gather together all the relevant entries of $T(x)$. Row operation are not allowed: they will be reflected back in the end matrix. The matrix A is a $\rho \times \rho$ block matrix. The block (r_j, j) is the companion block of minimal polynomial the pivot of the j-th column of $T(x)$. The other blocks (r_j, l), $l \neq j$, have nonzero entries only in their last column. This column is given by the coefficients of the corresponding entry $T_{r_j,l}(x) = \Sigma_{k=0}^{d_j-1} = t_{r_jlk}x^k$ of degree at most $d_j - 1$ (requirement IV) of the Popov form), it is:

$$^t[-t_{r_jl0}, -t_{r_jl1}, \ldots, -t_{r_jl(d_j-1)}]. \tag{12}$$

We omit the matrix C since it is not involved in the computation. The matrix B is an element of $\mathcal{M}_{\nu,n}(K)$. It is constructed as two blocks of columns vectors. The ρ first column vectors of B are canonical vectors of K^ν. Let π be the index permutation considered during the proof of proposition 2.1 such that the $r_{\pi(j)}$ are increasingly ordered and $d_{\pi(j)} \neq 0$, $1 \le j \le \rho$. The first column vector of B is the first canonical vector and the l-th column vector of B, $2 \le l \le \rho$, is the $(d_{\pi(1)} + \ldots + d_{\pi(l-1)} + 1)$-th canonical vector of K^ν. The $n - \rho$ last column vectors of B are given by the columns with pivot equal to one. They express dependencies with respect to the previous ones. For the l-th column vector of B, $\rho+1 \leq l \leq n$, these are given by the coefficients of the entries of column $\pi(l)$ of $T(x)$, as done above at (12) for the nonpivot entries of the relevant part.

Now, we proceed as for the Popov form during the proof of proposition 2.1 since the realization and the transfer matrix satisfy identity (5). There is a strong relationship between the entries of the Hermite normal form of the denominator matrix $T(x)$ and the columns of the block Krylov matrix $M(A,B,\nu)$:

$$[B_1, AB_1, \ldots, A^\nu B_1, \ldots, B_n, \ldots, A^\nu B_n] \in \mathcal{M}_{\nu,n\nu} \tag{13}$$

where B_j denotes the j-th column of B for $1 \leq j \leq n$.

Proposition 4.1 *The column Hermite normal form $L_H(x)$ of $T(x)$ in Popov form as given by proposition 3.1, can be computed in $O(\log(nd))$ solutions of $\mathcal{L}(nd)$.*

Proof. We do not detail the proof, it is based on classical results in [20]. By construction (§6.4.4, [20]),

$$H(x) = C(xI-A)^{-1}B = \bar{N}(x)T^{-1}(x).$$

Since (C,A,B) is a minimal realization of $H(x)$ then [20] there exists a matrix $\Psi(x)$ in $\mathcal{M}_{\nu,n}(K[x])$ right coprime with $D(x)$ such that

$$(xI-A)^{-1}B = \Psi(x)T^{-1}(x).$$

Now, $L_H(x)$ is right equivalent to $T(x)$ thus the same holds for it and for a matrix $\Phi(x)$:

$$(xI-A)^{-1}B = \Phi(x)L_H^{-1}(x).$$

We keep the same reasoning than for proposition 2.1 with $M(A,B,\nu)$ playing the role of matrix $M(\delta+1)$ (indeed, from a system-theoretic point of view they are both *controllability matrices*). We get that the entries of $L_H(x)$ are read-off the nullspace of $M(A,B,\nu)$. We may write a linear system which solutions gives the entries of $L_H(x)$. This linear system is computed by first isolating the ν lexicographically first independent columns of $M(A,B,\nu)$. Since the matrix is block Krylov, we can restrict ourselves to $\nu \times 2\nu$ matrices as done in [25]. The cost is that of $O(\log(\nu))$ solutions of $\mathcal{L}(\nu)$. The fact that $\nu \leq nd$ terminates the proof. □

4.2 Modules over a P.I.D

These results can be expressed using the *language of modules*. Well known from a theoretical point of view [19], this approach has not been often use to derive practical algorithms for computing matrix normal forms [28,1,31]. In [31] the author computes transformations for normal forms of matrices over a field (under similarity) from transformations for corresponding normal forms of matrix polynomials (under unimodular equivalence). We refer to this paper and to [42] for the reverse procedure (from a system-theoretic point of view this is *Nerode's equivalence* to compute special forms of realizations [33]). For $T(x)$ in Popov form, the quotient $K[x]$-module

$$\mathcal{Y} = K^n[x]/{}^tT(x)K^n[x]$$

is a $K[x]$-module and also a vector space of dimension ν over K. A concrete representation of the vector space as $K[x]$-module is $(K^\nu,\ .x = .A)$ where A is the matrix in shift form constructed from $T(x)$, the following isomorphisms diagram commutes:

$$
\begin{array}{ccc}
K^n[x]/{}^tT(x)K^n[x] & \stackrel{p}{\rightarrow} & K^\nu \\
\downarrow x & & \downarrow A \\
K^n[x]/{}^tT(x)K^n[x] & \stackrel{p}{\rightarrow} & K^\nu
\end{array}
\qquad (14)
$$

where p denote the coordinate function according to the natural basis of $\mathcal{Y}$ and to the ρ nonzero column degrees of $T(x)$ denoted by $d^{(1)}, \ldots, d^{(\rho)}$:

$$
\begin{array}{l}
\overline{(1,0,\ldots,0)}, \overline{(x,0,\ldots,0)}, \ldots, \overline{(x^{d^{(1)}-1},0,\ldots,0)}, \ldots \\
\quad \ldots, \overline{(0,0,\ldots,1)}, \overline{(0,0,\ldots,x)}, \ldots, \overline{(0,0,\ldots,x^{d^{(\rho)}-1})}.
\end{array}
$$

The same can be done for the Hermite normal form $L_H(x)$. Following the converse process of the one used to construct A from $T(x)$, from $L_H(x)$ we can contruct a matrix A'. Since $L_H(x)$ is triangular, A' is Hessenberg, it is said to be either in *shift-Hessenberg form* [1] or in *polycyclic form* [34]. In addition, A' is similar to A. We can also define

$$\mathcal{Y}' = K^n[x]/{}^tL_H(x)K^n[x]$$

and we get a commuting diagram similar to (14) with a coordinate function p'. Merging the two finally leads to:

$$
\begin{array}{ccc}
K[x]^n/{}^tT(x)K^n[x] & \rightarrow & K[x]^n/{}^tL_H(x)K^n[x] \\
\downarrow p & & \downarrow p' \\
K^\nu, A & \rightarrow & K^\nu, A'
\end{array}
\qquad (15)
$$

As shown in [31,42], given A, this diagram allows to develop algorithms to compute A' if $T(x)$ and $L_H(x)$ are known. Conversely, given $T(x)$ it is possible to compute its Hermite form if A and A' are known. Results for the computation of a shift-Hessenberg form of a matrix, such that the algorithm in [25], could thus be used for another proof of proposition 4.1. This would lead to the same complexity bounds.

5 Cost analysis

By using the results of previous sections we establish new cost bounds for the computation of the Hermite normal form of a matrix polynomial of dimension n and of degree d. We let $n = O(\mu)$ and $d = O(\mu)$. For a survey of basic sequential and parallel complexities, *e.g.* for matrix or polynomial multiplication, we refer to [7].

Proposition 5.1 *A reduction to the Hermite normal form for a non singular matrix in $\mathcal{M}_{n,n}(K[x])$ of degree d can be computed in deterministic sequential time $\tilde{O}(ndM(n) + M(nd)) = \tilde{O}(\mu^{4.8})$ over K using fast arithmetic, or, $\tilde{O}(\mu^6)$ using standard arithmetic.*

Proof. The first step consists in computing the Popov form of the matrix. By proposition 3.1, the corresponding cost is that of matrix polynomial inversion. Indeed, the solution of problem $\mathcal{L}(nd)$ for a block Hankel matrix is of lower cost. The matrix is of dimension n thus the cost is $O(M(n))$ operations over univariate polynomials of degree $O(nd)$. This gives $\tilde{O}(ndM(n))$ over K. Then by proposition 4.1 the remaining cost is $O(\log(nd))$ times the cost of a solution of $\mathcal{L}(nd)$. We know it is $\tilde{O}(M(nd))$ using the algorihm in [25]. The complexity given in [11] for matrix multiplication leads to the announced cost $\tilde{O}((nd)^{2.4})$. Using standard arithmetic, unlike the eliminations in [18,16] but as the probabilistic triangularization in [39], our algorithm is susceptable to a evaluation/interpolation scheme. Since two polynomials of degree d are multiplied in $O(n^2d^2)$, the cost of matrix inversion is $O(n^4d^2)$. With $M(n) = n^3$ we get $\tilde{O}(n^4d^2+n^3d^3) = \tilde{O}(\mu^6)$ for the whole computation. Clearly, the unique transformation matrix is obtained with the same bounds. $\square$

Proposition 5.2 *A reduction to the Hermite normal form for a non singular matrix in* $\mathcal{M}_{n,n}(K[x])$ *of degree d can be computed in* $O(\log^4(nd))$ *arithmetic parallel steps using* $O(n^{3/2}dM(n) + (nd)^{1/2}M(nd)) = O(\mu^{5.8})$ *processors if K is of characteristic zero. The computation can be performed in time* $O(\log^3(nd))$ *using* $\tilde{O}((nd)^4M(nd)) = O(\mu^{12.8})$ *processors over an arbitrary field.*

Proof. Over a field of characteristic zero, we refer to [36] for matrix inversion. The number of processors required is $O(n^{1/2}M(n))$ for $O(\log^2 n)$ arithmetic steps. Together with the cost of the operations on polynomials of degree nd, in time $O(\log(nd))$ using $\tilde{O}(nd)$ processors, this gives $\tilde{O}(n^{3/2}dM(n))$. Then, we follow results of [17] for the rank, of [23] for the nullspace ($O(\log^2 n)$ steps) and of [12] for the subset of independent columns ($\times O(\log n)$ steps), and we use, as proposed in [13], the parallel handling of [25] for the block Krylov system involved in proposition 4.1 ($\times O(\log n)$ steps). The Hermite form is computed from the Popov form in time $O(\log^4(nd))$ using $O((nd)^{1/2}M(nd))$ processors.

Over an arbitrary field, the costs are dominated by the solution of $\mathcal{L}(nd)$. As for the processor bound of the solution of the nullspace problem in [9] this leads to n times [7] the processor bound for the rank. This latter problem is solved using the algorithm in [32]. The number of steps if $O(\log^3(nd))$ using $\tilde{O}((nd)^4M(nd))$ processors. The unique transformation matrix is obtained with the same bounds. $\square$

Conclusion

We have given new complexity bounds for some operations on matrix polynomials. These have been obtained using a strong relationship (never exploited for practical purposes in matrix computer algebra) with linear system theory in control. Pursuing this approach, many technical work remains to be done. But the questions we did not tackled here (*e.g* the use of the Bezoutian, the boolean complexity, the case of singular matrices, links with Euclid's algorithm) should be solved as it has been done for scalar polynomials.

References

1. D. Augot and P. Camion. The minimal polynomials, characteristic subspaces, normal bases and the Frobenius form. Technical Report 2006, INRIA France, August 1993.
2. S. Barnett. Greatest common divisor of two polynomials. *Linear Algebra and its Applications*, 3:7–9, 1970.
3. S. Barnett. Greatest common divisor of several polynomials. *Proc. Cambridge Philos. Society*, 70:263–268, 1971.

4. S. Barnett and P. Lancaster. Some properties of the bezoutian for polynomial matrices. *Linear Algebra and Multilinear Algebra*, 9:99–110, 1980.
5. V.A. Belyĭ, V.B. Khazanov, and V.N. Kublanovskaya. Spectral problems for matrix pencils. Methods and algorithms III. *Soviet J. Numer. Anal. Math. Model.*, 4(1):19–51, 1989.
6. D. Bini and L. Gemignani. Fast parallel computation of the polynomial remainder sequence via Bezout and Hankel matrices. *SIAM J. Comput.*, 24(1):63–77, February 1995.
7. D. Bini and V. Pan. *Polynomial and matrix computations.* Birkhäuser, 1994.
8. R.R. Bitmead, S.Y. Kung, B.D.O. Anderson, and T. Kailath. Greatest common divisors via generalized Sylvester and Bezout matrices. *IEEE Trans. Automat. Control.*, 23(6):1043–1047, 1978.
9. A. Borodin, J. von zur Gathen, and J. Hopcroft. Fast parallel matrix and gcd computations. *Information and Control*, 52:241–256, 1982.
10. D. Coppersmith. Solving homogeneous linear equations over GF(2) via block Wiedemann algorithm. *Mathematics of Computation*, 62(205):333–350, 1994.
11. D. Coppersmith and S. Winograd. Matrix multiplication via arithmetic progressions. In *19 th. Annual ACM Symp. Theory Comp.*, pages 1–6, 1987.
12. W. Eberly. Parallel independent subsets and matrix factorizations. In *Proc. 3rd IEEE Conference on Parallel and Distributed Processing.* IEEE Computer Society Press, 1991.
13. M. Giesbrecht. Nearly optimal algorithms for canonical matrix forms. *SIAM Journal on Computing*, 1994. To appear.
14. I. Gohberg, P. Lancaster, and L. Rodman. *Matrix polynomials.* Academic Press, New-York, 1982.
15. W.B. Gragg and A. Lindquist. On the partial realization problem. *Linear Algebra and its Applications*, 50:277–319, 1983.
16. J.L. Hafner and K.S. Mc Curley. Asymptotically fast triangularization of matrices over rings. *SIAM J. Comput.*, 20(6):1068–1083, 1991.
17. O. Ibarra, S. Moran, and L.E. Rosier. A note on the parallel complexity of computing the rank of order n matrices. *Information Processing Letters*, 11:162, 1980.
18. C.S. Iliopoulos. Worst-case complexity bounds on algorithms for computing the canonical structure of finite abelian groups and the Hermite and Smith normal forms of an integer matrix. *SIAM J. Comput.*, 18(4):658–669, 1989.
19. N. Jacobson. *Lectures in Abstract Algebra II, Linear Algebra.* Springer-Verlag, 1953.
20. T. Kailath. *Linear systems.* Prentice Hall, 1980.
21. R.E. Kalman, P. Falb, and M. Arbib. *Topics in mathematical system theory.* McGraw-Hill, New-York, 1969.
22. E. Kaltofen, M.S. Krishnamoorthy, and B.D. Saunders. Fast parallel computation of Hermite and Smith forms of polynomials matrices. *SIAM J. Alg. Disc. Meth.*, 8 4, pp 683-690, 1987.
23. E. Kaltofen and V. Pan. Processor efficient parallel solution of linear systems over an abstract field. In *Proc. 3rd Annual ACM Symposium on Parallel Algorithms and Architecture.* ACM-Press, 1991.
24. N. Karcanias and M. Mitrouli. A matrix pencil based numerical method for the computation of the gcd of polynomials. *IEEE Trans. Autom. Contr.*, 39(5):977–981, 1994.
25. W. Keller-Gehrig. Fast algorithms for the characteristic polynomial. *Theoretical Computer Science*, 36:309–317, 1985.
26. S.Y. Kung. *Multivariable and multidimensional systems: analysis and design.* PhD thesis, Dpt. of Electrical Engineering, Stanford University, June 1977.

27. S.E. Labhalla, H. Lombardi, and R. Marlin. Algorithmes de calcul de la réduction d'Hermite d'une matrice à coefficients polynomiaux. *Theoretical Computer Science*, 1995. To appear.

28. H. Lüneburg. *On rational form of endomorphims : a primer to constructive algebra.* Wissenschaftsverlag, Mannheim, 1987.

29. C.C. MacDuffee. *The theory of matrices.* Chelsea, New-York, 1956.

30. D. Manocha and J.F. Canny. Multipolynomial resultants and linear algebra. In *International Symposium on Symbolic and Algebraic Computation, Berkeley California USA.* ACM Press, July 1992.

31. T. Mulders. Computation of normal forms for matrices. In Algoritmen In De Algebra, A Seminar on Algebraic Algorithms, 1993. University of Nijmegen, Netherlands.

32. K. Mulmuley. A fast parallel algorithm to compute the rank of a matrix over an arbitrary field. *Combinatorica*, 7(1):101–104, 1987.

33. A. Nerode. Linear automaton transformations. *Proc. American Math. Soc.*, 9:541–544, 1958.

34. P. Ozello. *Calcul exact des formes de Jordan et de Frobenius d'une matrice.* PhD thesis, Université Scientifique et Médicale de Grenoble, France, 1987.

35. V.M. Popov. Invariant description of linear, time-invariant controllable systems. *SIAM J. Contr.*, 10:252–264, May 1972.

36. F.P. Preparata and D.V. Sarwate. An improved parallel processor bound in fast matrix inversion. *Inform. Proc. Lett.*, 7:148–150, 1978.

37. V. Pták and H.K. Wimmer. On the Bezoutian for polynomial matrices. *Linear Algebra and its Applications*, 71:267–272, 1985.

38. M.P. Quéré and G. Villard. Matrices quasi-Hankel et Bezoutiennes pour le calcul de la forme normale d'Hermite de matrices polynomiales. Preprint IMAG, Grenoble France, 1996.

39. A. Storjohann. Computation of Hermite and Smith normal forms of matrices. Master's thesis, 1994. University of Waterloo, Canada.

40. P. Van Dooren and P. Dewilde. The eigenstructure of an arbitrary polynomial matrix: computational aspects. *Linear Algebra and its Applications*, 50:545–579, 1983.

41. P.M. Van Dooren, P. Dewilde, and J. Vandewalle. On the determination of the Smith-Macmillan form of a rational matrix from its Laurent expansion. *IEEE Trans. Cir. Sys.*, 26(3):180–189, March 1979.

42. G. Villard. Fast parallel algorithms for matrix reduction to normal forms. Technical Report APACHE 16, IMAG Grenoble France, 1995.

43. W.A. Wolovich. *Linear multivariable systems.* Springer-Verlag, New-York, 1974.

APACHE: List of Research Reports

All these reports are available via ftp: `ftp.imag.fr:/pub/APACHE/RAPPORT`

1: Présentation du projet APACHE (en français)
B. Plateau et al. - October 1993 (revised version November 94)

2: Matrix Transpose for Block Allocations on Processor Networks
C. Calvin, and D. Trystram - November 1993

3: Optimal Scheduling of UECT trees on two identical processors
F. Guinand and D. Trystram - November 1993

4: Some Ergodic Results on Stochastic Iterative DED
J-M. Vincent - December 1993

5: Optimal parallel execution of complete binary trees and grids into most popular interconnection networks
E. Bampis, J-C. Konig and D. Trystram - December 1993

6: The numerical solution of stochastic automata networks
W. Stewart, K. Atif and B. Plateau - November 1993

7: Parallel algebraic computations: about the Jordan normal form
J-L. Roch and G. Villard - February 1994

8: Parallel algebraic computations: about the Smith normal form
G. Villard - February 1994

9: Gossiping in Torus with Circuit-Switched Routing
C. Calvin, S. Perennes and D. Trystram - April 1994

10: Parallel Matrix-vector Product on Rings with a minimum of Communications
L. Colombet, P. Michallon and D. Trystram - June 1994

11: ATHAPASCAN-0a sur PVM 3 : définition et mode d'emploi (en français)
M. Christaller - June 1994

12: Introduction au modèle de programmation par processus communicants : deux exemples PVM et MPI (en français)
C. Calvin and L. Colombet - July 1994

13: Analysis of gossiping algorithms in torus with restricted bufferization capabilities
J-C. Konig, P.S. Rao and D. Trystram - November 1994

14: PAC++ v2.0 User and developer manual
T. Gautier, J.L. Roch and G. Villard - November 1994

15: Performance Evaluation of Parallel Systems: the Scope Extensible Interactive Environment
Yves Arrouye, December 1994

16: Fast parallel algorithms for matrix reduction to normal forms
G. Villard - June 1995 (Updated Feb. 96)

17: Worst Case Analysis of Lawler's Algorithm for Scheduling Trees with Communication Delays
F. Guinand, C. Rapine and D. Trystram - July 1995

18: Optimized execution replay mechanism for RPC-based parallel programming models
A. Fagot, J. Chassin de Kergommeaux - July 1995

19: Fast matrix polynomial computations
G. Villard - February 96