ELSEVIER

Available online at www.sciencedirect.com

SCIENCE DIRECT®

Journal of Complexity 21 (2005) 72–86

Journal of COMPLEXITY

http://www.elsevier.com/locate/jco

Essentially optimal computation of the inverse of generic polynomial matrices

Claude-Pierre Jeannerod* and Gilles Villard

CNRS, INRIA, Laboratoire LIP, École Normale Supérieure de Lyon, 46, Allée d'Italie 69364 Lyon Cedex 07, France

Received 20 December 2002; accepted 15 March 2004

Available online 10 June 2004

Abstract

We present an inversion algorithm for nonsingular $n \times n$ matrices whose entries are degree d polynomials over a field. The algorithm is deterministic and, when n is a power of two, requires $O^{\sim}(n^3 d)$ field operations for a generic input; the soft-O notation $O^{\sim}$ indicates some missing $\log(nd)$ factors. Up to such logarithmic factors, this asymptotic complexity is of the same order as the number of distinct field elements necessary to represent the inverse matrix.

Keywords: Polynomial matrix; Matrix inversion; Minimal kernel basis

1. Introduction

Let K be an abstract commutative field and, for two positive integers n and d, consider a nonsingular $A \in \mathsf{K}[x]^{n \times n}$ of degree d. Since the determinant of A is a polynomial of degree up to nd, it follows from Cramer's rule that the number of field elements necessary to represent the inverse of A can be of the order of $n^3 d$. Assuming that n is a power of two, we present in this paper a deterministic inversion algorithm whose complexity is generically $O^{\sim}(n^3 d)$ field operations on an algebraic random access machine. Here and in the following, the $O^{\sim}$ notation indicates some missing

*Corresponding author.

E-mail addresses: claude-pierre.jeannerod@ens-lyon.fr (C.-P. Jeannerod), gilles.villard@ens-lyon.fr (G. Villard).

URLs: http://perso.ens-lyon.fr/claude-pierre.jeannerod/, http://perso.ens-lyon.fr/gilles.villard/.

0885-064X/$ - see front matter
doi:10.1016/j.jco.2004.03.005

$\log(nd)$ factors. By generically, we mean that the algorithm has the above asymptotic complexity for every $n \times n$ matrix polynomial of degree d whose coefficients do not form a point of a certain hypersurface of $\mathsf{K}^{n^2(d+1)}$.

The best previously known complexity estimate for computing the polynomial matrix inverse was $O^{\sim}(n^{\omega+1}d)$, where ω is the exponent for multiplying two $n \times n$ matrices over K [10, Chapter 1]. If $\omega > 2$, we thus improve the complexity for most $n \times n$ inputs with n a power of two; the improvement is by a factor of n when considering classical matrix multiplication ($\omega = 3$).

Let us recall how the above classical estimate $O^{\sim}(n^{\omega+1}d)$ for matrix inversion over $\mathsf{K}(x)$ is obtained. The determinant and the entries of the adjoint, whose degrees are bounded by nd, may be recovered for instance using evaluation/interpolation at $nd+1$ points [16, Section 5.5]. A randomized Las Vegas algorithm—A must be invertible at the $nd+1$ evaluation points—may thus rely on recursive matrix inversion over K in $O(n^{\omega})$ [9,28,32] and on a fast evaluation/interpolation scheme for univariate polynomials of degree nd in $O^{\sim}(\mathsf{M}(nd))$ [24], [16, Section 10]. Here and in the rest of the paper, $\mathsf{M}(d)$ is the number of operations in K sufficient for multiplying two polynomials of degree d in $\mathsf{K}[x]$. The method in [11] (over any ring) allows $\mathsf{M}(d) = O(d \log d \log\log d)$. Many other inversion approaches may be considered such as direct Gauss-Jordan elimination on truncated power series, Newton iteration [25], Hensel lifting à la Dixon [12], or linearization (see for instance [23] and the references therein). A deterministic $O^{\sim}(n^{\omega+1}d)$ algorithm is given in [29, Section 2]. This algorithm is a fraction-free version over $\mathsf{K}[x]$ (Bareiss' approach [1]) of the recursive inversion algorithms over K cited above. We see that none of these methods seems to reduce the complexity estimate over K below the order of $n^{\omega+1}d$. With classical matrix multiplication ($\omega = 3$) the cost of inversion was still about n times higher than the typical size of the inverse.

Our motivation for this work is the fact that some other basic linear algebra problems on polynomial matrices have much lower complexity estimates. It is known, since more than two decades, that a linear system can be solved exactly in $O^{\sim}(n^3 d)$ operations [12,25], and it has been shown more recently that the solution can be computed using fast matrix multiplication in $O^{\sim}(n^{\omega}d)$ operations [30,31]. Concerning the problem of computing the determinant, the classical techniques seen above also lead to the cost $O^{\sim}(n^{\omega+1}d)$. In the last years, this estimate has been reduced using rank perturbations by Eberly et al. [13], basis reduction by Mulders and Storjohann [26], or a Krylov–Lanczos approach by Kaltofen [19], Kaltofen and Villard [21,22]. By Hensel lifting with jumps to high order it is possible to compute the determinant in $O^{\sim}(n^{\omega}d)$ operations in K [30,31], and the same estimate is valid for the Smith normal form. An application of the latter method further gives an algorithm for column reduction in $O^{\sim}(n^{\omega}d)$ operations [17]. We may also point out that for $\omega = 3$, the approach of Kaltofen and Villard [22] gives an algorithm for computing the characteristic polynomial and the Frobenius normal form of a polynomial matrix in $O^{\sim}(n^{3+1/5}d)$ operations in K. Under the algebraic complexity model for matrices over an abstract field, the problems of computing the determinant, the characteristic polynomial and the inverse have the same exponent

(we refer for instance to the survey in [10, Chapter 16]). Nevertheless, in spite of the recent advances just mentioned, the same is not known in the polynomial case. The essentially optimal algorithm for inversion in the generic case that we propose here gives a new insight into the links between the problems.

Our approach, described in Section 2, consists in computing a nonsingular $U \in \mathsf{K}[x]^{n\times n}$ and a diagonal $B \in \mathsf{K}[x]^{n\times n}$ such that $UA = B$. The inverse of A is then recovered as $A^{-1} = B^{-1}U$. In order to achieve the announced $O^{\sim}(n^3 d)$ complexity, we shall make three remarks. First, with n a power of two, A can be diagonalized in $\log n$ block elimination steps, starting with

$$A = [A_{\mathrm{L}}\ A_{\mathrm{R}}] \to UA = \begin{bmatrix} \overline{U} \\ \underline{U} \end{bmatrix} [A_{\mathrm{L}}\ A_{\mathrm{R}}] = \begin{bmatrix} \overline{U}A_{\mathrm{L}} & \\ & \underline{U}A_{\mathrm{R}} \end{bmatrix}. \tag{1}$$

Here $A_{\mathrm{L}}, A_{\mathrm{R}}$ have dimensions $n \times n/2$ and $\overline{U}, \underline{U} \in \mathsf{K}[x]^{n/2\times n}$ are bases of the left kernels $\ker A_{\mathrm{R}}$, $\ker A_{\mathrm{L}}$ considered as $\mathsf{K}[x]$-submodules of $\mathsf{K}[x]^n$. The blank areas in matrices are assumed to be filled with zeros. We then observe in Section 3.1 that among all the possible kernel bases $\overline{U}$ and $\underline{U}$, those with rows of lowest degree typically have degree exactly d, the degree of A. Hence, choosing such *minimal bases* yields two square blocks of order $n/2$ and degree $2d$. The third and key point is that this property generically carries over from one step to the next one. In particular, we show in Section 3.2 that if the input matrix A of degree d is generic enough then all the minimal bases at step i of the computation of A^{-1} have degree exactly $2^{i-1}d$, regardless of the way these bases are computed. Therefore, the degree of the working polynomial matrices only doubles at each step, whereas their order is divided by two. As we shall finally see in Sections 4 and 5, combining deterministic $O^{\sim}(n^3 d)$ minimal basis computations with steps of type (1) eventually allows for A^{-1} to be computed in $O^{\sim}(n^3 d)$ field operations by using only classical matrix multiplication.

Notation: All matrix kernels are left kernels. We write K^* for $\mathsf{K}\backslash\{0\}$ and $|\mathsf{K}|$ for the cardinality of K. Also, for any real number y, $\lfloor y \rfloor$ (resp. $\lceil y \rceil$) is the greatest (resp. smallest) integer less than (resp. greater than) or equal to y. As already used in (1), if M is an $n \times m$ matrix then M_{L} is the $n \times \lfloor m/2 \rfloor$ matrix that consists of the leftmost $\lfloor m/2 \rfloor$ columns of M and M_{R} is the $n \times \lceil m/2 \rceil$ matrix that consists of the rightmost $\lceil m/2 \rceil$ columns of M. Submatrices $\overline{M}$ and $\underline{M}$ are defined similarly by considering top and bottom rows instead.

2. Inversion algorithm

Algorithm `Inverse` is described below. Here `MinimalKernelBasis` is any subroutine for computing a minimal basis of the left kernel of a polynomial matrix. (We give in Section 4 an example of such a subroutine that is appropriate to our complexity purposes.) Furthermore, when entering step i, the polynomial matrix B is block-diagonal with jth block $B_i^{(j)}$ of order $n/2^{i-1}$ for $1 \leqslant j \leqslant 2^{i-1}$.

Algorithm `Inverse`(A)
Input: $A \in \mathsf{K}[x]^{n\times n}$ of degree d
Output: A^{-1}
Condition: $\det A \neq 0$ and $n = 2^p$ with $p \in \mathbb{N}$
(a) $B \coloneqq \mathrm{copy}(A)$;
$U \coloneqq I_n$;
(b) **for** i **from** 1 **to** p **do** $\quad //B = \mathrm{diag}(B_i^{(1)}, \ldots, B_i^{(2^{i-1})})$
for j **from** 1 **to** 2^{i-1} **do**
$\underline{U}_i^{(j)} \coloneqq \mathtt{MinimalKernelBasis}(B_{i,\mathrm{L}}^{(j)})$; $\quad //\underline{U}_i^{(j)} B_{i,\mathrm{L}}^{(j)} = 0$
$\overline{U}_i^{(j)} \coloneqq \mathtt{MinimalKernelBasis}(B_{i,\mathrm{R}}^{(j)})$; $\quad //\overline{U}_i^{(j)} B_{i,\mathrm{R}}^{(j)} = 0$
od;
$U_i \coloneqq \mathrm{diag}(U_i^{(1)}, \ldots, U_i^{(2^{i-1})})$; $\quad //U_i^{(j)} = \begin{bmatrix} \overline{U}_i^{(j)} \\ \underline{U}_i^{(j)} \end{bmatrix}$
$B \coloneqq U_i B$;
$U \coloneqq U_i U$;
od;
(c) **return** $B^{-1}U$.

We now prove that algorithm `Inverse` is correct. For $i = 1$, it follows from $\det A \neq 0$ that $\underline{U}_1^{(1)}$ and $\overline{U}_1^{(1)}$ have full row rank. Additionally, $U_1 = U_1^{(1)}$ is nonsingular for otherwise $\ker A_{\mathrm{L}} \cap \ker A_{\mathrm{R}} \supsetneq \{0\}$ which contradicts $\det A \neq 0$. Therefore, the two blocks of order $n/2$ of $U_1 A$ are nonsingular. Repeating the argument for $i = 2, \ldots, p$, we see that the pth step of stage (b) produces a nonsingular $U \in \mathsf{K}[x]^{n\times n}$ and a diagonal $B \in \mathsf{K}[x]^{n\times n}$ such that $UA = B$. Correctness follows from identity $A^{-1} = B^{-1}U$.

In fact, the kernel bases need not be minimal for the algorithm to return A^{-1}. On the other hand, it is not hard to modify the algorithm so that it computes the inverse of *any* nonsingular polynomial matrix A: if n is not a power of two, the first step should yield two square blocks of respective orders $\lfloor n/2 \rfloor$ and $\lceil n/2 \rceil$ and so on. However, both minimality and $n = 2^p$ are necessary in our cost analysis of the algorithm when the input is *generic*. Indeed, the polynomial matrices $B_i^{(j)}$ and $U_i^{(j)}$ then have order $n/2^{i-1}$ and, as we shall prove in Section 3, minimality further implies that they typically have degree $2^{i-1}d$ for $1 \leqslant j \leqslant 2^{i-1}$. In other words, each of these polynomial matrices satisfies order $\times$ degree $= nd$.

3. Minimal kernel bases and genericity

We first recall in Section 3.1 the definition and some needed properties of minimal kernel bases of polynomial matrices. We also give an explicit formula for the construction of such bases in the generic case. This formula will then allow us

to characterize in Section 3.2 the degrees produced by algorithm `Inverse` for a generic input.

3.1. Definition, degree characterization and explicit construction

For a positive integer m, let $M \in \mathsf{K}[x]^{2m\times m}$ with rank m and let $U \in \mathsf{K}[x]^{m\times 2m}$ with rows forming a basis of the $\mathsf{K}[x]$-submodule $\ker M$. It is sufficient for our purpose to restrict ourselves to matrices having twice as many rows as columns. We further denote by d_i the ith row degree of U, that is, the highest degree of all the entries of the ith row of U. The polynomial matrix U is a *minimal basis* of $\ker M$ when $\sum_{i=1}^{m} d_i$ is minimal among all the polynomial bases of $\ker M$ [14]. Here we shall use only two properties of minimal kernel bases but we refer to Forney [14] and Kailath [18, Section 6] for a comprehensive treatment. First, although minimal kernel bases are not unique, their row degrees $\{d_i\}_{1\leqslant i\leqslant m}$ are unique up to ordering; such indices are usually called the *minimal row degrees of* $\ker M$ or the *left Kronecker indices of* M, for when M has degree one they coincide with some block dimensions in the Kronecker canonical form of M [18, Section 6.5.4]. Second, if d is the degree of M, one has the upper bound

$$\sum_{i=1}^{m} d_i \leqslant md. \tag{2}$$

Some minimal row degrees can thus be of the order of md. However, in most cases, all of them are equal to d. To verify this typical behaviour, let us associate with $M = \sum_{i=0}^{d} M_i x^i$ the block-Toeplitz matrix

$$\mathcal{T}(M) = \begin{bmatrix} M_0 & M_1 & \cdots & M_d & & \\ & \ddots & \ddots & \vdots & \ddots & \\ & & M_0 & M_1 & \cdots & M_d \end{bmatrix} \in \mathsf{K}^{2md\times 2md}. \tag{3}$$

To any nonzero vector $u = \sum_{i=0}^{d-1} u_i x^i$ in $\ker M$ of degree less than d corresponds the nonzero vector $[u_0^T, \ldots, u_{d-1}^T]^T$ in $\ker \mathcal{T}(M)$. Thus if $\det \mathcal{T}(M) \neq 0$ then $d_i \geqslant d$ for $1 \leqslant i \leqslant m$ and, using (2), $d_i = d$. It is not hard to verify that $\det \mathcal{T}(M)$ is a nonzero polynomial in the $2m^2(d+1)$ coefficients of the entries of M; therefore the minimal kernel bases of M generically have degree d.

For algorithm `Inverse` with generic input A, this means that the minimal bases $\underline{U}_1^{(1)}$, $\overline{U}_1^{(1)}$ at the first step both have degree d. In addition, by uniqueness of the minimal degrees, the latter is true independently of the way the bases are computed. Now what about the minimal basis degrees at the remaining steps? In order to show in Section 3.2 below that in general the degrees at step i are $2^{i-1}d$, we shall further use the following explicit construction of a minimal kernel basis of M when $\det \mathcal{T}(M) \neq 0$. Indeed, it follows from identifying the matrix coefficients in both sides of polynomial matrix equation $UM = 0$ that a minimal kernel basis is given by any

of the $m \times 2m$ matrices $N = \sum_{i=0}^{d} N_i x^i$ such that

$$N_d \text{ is a basis of } \ker M_d, \tag{4a}$$

$$[N_0|\cdots|N_{d-1}] = -N_d[O|M_0|\cdots|M_{d-1}]\mathcal{T}(M)^{-1}. \tag{4b}$$

The fact below is an immediate consequence of the block structure of $\mathcal{T}(M)$ which we shall use to prove Proposition 3.

Fact 1. *If* $\det \mathcal{T}(M) \neq 0$ *then matrices* M *and* N *as in* (4) *have degree* d *exactly, and their leading matrix coefficients* M_d *and* N_d *have full rank.*

One can take for N_d in (4a) the particular kernel basis obtained by applying Gaussian elimination with pivoting (GEP) to the rows of M_d. Here, by pivoting—at the start of the ith stage of elimination—we mean exchanging rows i and k where $k \geqslant i$ is the smallest index such that the (k, i) entry is nonzero. The main point is that when replacing (4a) with

$$N_d \text{ is the basis of } \ker M_d \text{ computed by GEP}, \tag{4c}$$

the entries of $N_0, \ldots, N_d$ given by (4b and c) are now uniquely defined as rational functions over K of the entries of $M_0, \ldots, M_d$ (see for example [15, Section 2.4]).

3.2. Typical degrees of minimal kernel bases during inversion

Consider $n^2(d+1)$ indeterminates $\alpha_{i,j,k}$ for $1 \leqslant i, j \leqslant n$, $0 \leqslant k \leqslant d$, and let $A \in \mathsf{K}[\alpha_{1,1,0}, \ldots, \alpha_{i,j,k}, \ldots, \alpha_{n,n,d}][x]^{n \times n}$ have its (i,j) entry equal to $\sum_{k=0}^{d} \alpha_{i,j,k} x^k$. Recall that $n = 2^p$ for some $p \in \mathbb{N}$ and let

$$v_i = n/2^{i-1} \quad \text{and} \quad \delta_i = 2^{i-1} d \quad \text{for } 1 \leqslant i \leqslant p. \tag{5}$$

First, assume that algorithm `Inverse` is runned formally with subroutine `MinimalKernelBasis` replaced with minimal basis formula (4b and c). We show in Lemma 2 below that this construction leads to successive block-Toeplitz matrices as in (3) that are invertible. We link the invertibility of these matrices to a well defined and nonzero rational function Φ in the $\alpha_{i,j,k}$'s. This means that (4b and c) with $(m, d) = (v_i, \delta_i)$ reflects the degrees of the matrices computed at the ith step of the algorithm in the generic case. As a consequence of the uniqueness of the minimal degrees, we then show in Proposition 3 that, if Φ is well defined and nonzero for a given input A, these degrees are still δ_i for *any* choice of minimal bases.

For $1 \leqslant i \leqslant p$, $1 \leqslant j \leqslant 2^{i-1}$, let the matrices $A_i^{(j)}, N_i^{(j)} \in \mathsf{K}[x]^{v_i \times v_i}$ be of degree δ_i and such that: $A_1^{(1)} = A$; $\overline{N}_i^{(j)}$ is the minimal basis of $\ker A_{i,\mathrm{R}}^{(j)}$ of the form (4b and c); $\underline{N}_i^{(j)}$ is the minimal basis of $\ker A_{i,\mathrm{L}}^{(j)}$ of the form (4b and c); and

$$\begin{bmatrix} A_{i+1}^{(2j-1)} & \\ & A_{i+1}^{(2j)} \end{bmatrix} = \begin{bmatrix} \overline{N}_i^{(j)} \\ \underline{N}_i^{(j)} \end{bmatrix} \begin{bmatrix} A_{i,\mathrm{L}}^{(j)} & A_{i,\mathrm{R}}^{(j)} \end{bmatrix} \quad \text{for } 1 \leqslant j \leqslant 2^{i-1}, \quad 1 \leqslant i < p. \tag{6}$$

Then let $\Phi = \prod_{i=1}^{p} \prod_{j=1}^{2^{i-1}} \det \mathcal{T}(A_{i,\mathrm{L}}^{(j)}) \det \mathcal{T}(A_{i,\mathrm{R}}^{(j)})$.

Lemma 2. *For $n \geqslant 2$, Φ is a nonzero element of $\mathsf{K}(\alpha_{1,1,0}, \ldots, \alpha_{i,j,k}, \ldots, \alpha_{n,n,d})$.*

Proof. We prove the statement by recurrence on the ith stage of the construction. To prove both the existence of Φ—matrix inversions in (4b)—and the fact that $\Phi \neq 0$, it suffices to show that the successive determinants $\det \mathcal{T}(A_{i,\mathrm{L}}^{(j)})$ and $\det \mathcal{T}(A_{i,\mathrm{R}}^{(j)})$ are nonzero for a particular matrix A over $\mathsf{K}[x]$; we shall denote this particular matrix by $A_{n,d}$ and define it as follows. For n a power of two, let

$$A_{n,d} = x^d I_n - J_n, \quad \text{where} \quad J_n = \begin{bmatrix} & J_{n/2} \\ I_{n/2} & \end{bmatrix} \text{ if } n \geqslant 2 \quad \text{and} \quad J_1 = 1.$$

Let also $N_{n,d} = x^d I_n + J_n$. We show that the determinants used to define Φ are nonzero by proving that, when starting with $A_1^{(1)} = A_{n,d}$, construction (6) yields

$$A_i^{(j)} = A_{v_i,\delta_i} \quad \text{and} \quad N_i^{(j)} = N_{v_i,\delta_i}. \tag{7}$$

For $i = 1$ one can verify by inspection that $\mathcal{T}(A_{1,\mathrm{L}}^{(1)})$ and $\mathcal{T}(A_{1,\mathrm{R}}^{(1)})$ are invertible, of determinant ± 1; for example,

$$\mathcal{T}(A_{1,\mathrm{R}}^{(1)}) = \begin{bmatrix} \begin{bmatrix} -J_{n/2} \\ O \end{bmatrix} & & & \begin{bmatrix} O \\ I_{n/2} \end{bmatrix} & & \\ & \ddots & & & \ddots & \\ & & \begin{bmatrix} -J_{n/2} \\ O \end{bmatrix} & & & \begin{bmatrix} O \\ I_{n/2} \end{bmatrix} \end{bmatrix} \in \mathsf{K}^{nd \times nd}.$$

To obtain $N_1^{(1)} = N_{n,d}$, notice further that applying (4c) to the leading term $[O \ I_{n/2}]^T$ of $A_{1,\mathrm{R}}^{(1)}$ yields for $\overline{N}_1^{(1)}$ the leading term $[I_{n/2} \ O]$. Notice also that $\mathcal{T}(A_{1,\mathrm{R}}^{(1)})^{-1}$ is equal

to the transpose of $\mathcal{T}([-J_{n/2}^{-1} \mid x^d I_{n/2}]^T)$:

$$\mathcal{T}(A_{1,\mathrm{R}}^{(1)})^{-1} = \begin{bmatrix} \begin{bmatrix} -J_{n/2}^{-1} & O \end{bmatrix} & & & \\ & \ddots & & \\ & & \begin{bmatrix} -J_{n/2}^{-1} & O \end{bmatrix} & \\ \begin{bmatrix} O & I_{n/2} \end{bmatrix} & & & \\ & \ddots & & \\ & & \begin{bmatrix} O & I_{n/2} \end{bmatrix} & \end{bmatrix} \in \mathsf{K}^{nd \times nd}.$$

It then follows from (4b) that $\overline{N}_1^{(1)} = \overline{N}_{n,d}$; similarly, $\underline{N}_1^{(1)} = \underline{N}_{n,d}$ because (4c) gives the leading term $[O\ I_{n/2}]$ and $\mathcal{T}(A_{1,\mathrm{L}}^{(1)})^{-1}$ equals the transpose of $\mathcal{T}([x^d I_{n/2} \mid -I_{n/2}]^T)$. Hence, $N_1^{(1)} = N_{n,d}$ and (7) holds for $i = 1$. Now, if (7) holds for $i \in \{1, \ldots, p-1\}$, this is still true for $i+1$. Indeed, the block-diagonalization scheme (6) and identity

$$N_{n,d} A_{n,d} = \begin{bmatrix} A_{n/2,2d} & \\ & A_{n/2,2d} \end{bmatrix}$$

imply that $A_{i+1}^{(j)} = A_{v_{i+1},\delta_{i+1}}$ for $1 \leqslant j \leqslant 2^i$. We further obtain $\det \mathcal{T}(A_{i+1,\mathrm{L}}^{(j)}) \neq 0$, $\det \mathcal{T}(A_{i+1,\mathrm{R}}^{(j)}) \neq 0$ and $N_{i+1}^{(j)} = N_{v_{i+1},\delta_{i+1}}$ in the same way as for $i = 1$. □

Proposition 3. *Let $A \in \mathsf{K}[x]^{n \times n}$ be nonsingular of degree d. If $\Phi(A) \in \mathsf{K}^*$ then the matrices $B_{i,\mathrm{L}}^{(j)}$, $B_{i,\mathrm{R}}^{(j)}$, $\underline{U}_i^{(j)}$, $\overline{U}_i^{(j)}$ in* `Inverse`(A) *have degree δ_i.*

Proof. Let $B_i^{(j)}$, $U_i^{(j)}$ be the polynomial matrices involved during stage (b) of algorithm `Inverse`(A) and, since $\Phi(A) \in \mathsf{K}^*$, consider $A_i^{(j)}$, $N_i^{(j)}$ as in (6).

It suffices to show that there exists invertible constant matrices $C_i^{(j)}$ of order $v_i = n/2^{i-1}$ such that, for $1 \leqslant j \leqslant 2^{i-1}$ and $1 \leqslant i \leqslant p$,

$$B_i^{(j)} = C_i^{(j)} A_i^{(j)} \quad \text{and, if } i < p, \quad U_i^{(j)} C_i^{(j)} = \begin{bmatrix} C_{i+1}^{(2j-1)} & \\ & C_{i+1}^{(2j)} \end{bmatrix} N_i^{(j)}. \tag{8}$$

Indeed, the target degree bound δ_i for $B_i^{(j)}$ and $U_i^{(j)}$ then follows from $\Phi(A) \in \mathsf{K}^*$ and Fact 1 which gives the bound δ_i for $A_i^{(j)}$ and $N_i^{(j)}$.

We proceed by recurrence on i. When $i = 1$, $B_1^{(1)} = A_1^{(1)} = A$ and $C_1^{(1)} = I_n$. To see why $U_1^{(1)}$ is an invertible constant multiple of $N_1^{(1)}$, notice that both $\overline{U}_1^{(1)}$ and $\overline{N}_1^{(1)}$ are minimal kernel bases of $B_{1,\mathrm{R}}^{(1)} = A_{1,\mathrm{R}}^{(1)} = A_{\mathrm{R}}$. Hence there exists a unimodular $D \in \mathsf{K}[x]^{n/2 \times n/2}$ such that $\overline{U}_1^{(1)} = D\overline{N}_1^{(1)}$. Now, it follows from the assumption on Φ and from Fact 1 applied to $A_{1,\mathrm{R}}^{(1)}, \overline{N}_1^{(1)}$ that D must have degree zero. We can thus take

$C_2^{(1)} = D$. Similarly, $\underline{U}_1^{(1)} = C_2^{(2)} \underline{N}_1^{(1)}$ for some invertible constant matrix $C_2^{(2)}$ and (8) holds for $i = 1$.

Assume that (8) holds for $i < p$. It follows from $B_{i+1}^{(2j-1)} = \overline{U}_i^{(j)} B_{i,\mathrm{L}}^{(j)}$, (6) and (8) that $B_{i+1}^{(2j-1)} = C_{i+1}^{(2j-1)} A_{i+1}^{(2j-1)}$ (and similarly for $B_{i+1}^{(2j)}$). Now consider $\overline{U}_{i+1}^{(2j-1)}$ and, for simplicity, let X stand for $X_{i+1}^{(2j-1)}$ for $X \in \{A, B, C, N, U\}$ and let δ stand for δ_{i+1}. To show that $\overline{U}C = D\overline{N}$ for some invertible constant matrix D, recall first that $\overline{U}$ is a minimal kernel basis for $B_{\mathrm{R}} = CA_{\mathrm{R}}$. Therefore, $\overline{U}C$ is a kernel basis for A_{R}; it is further minimal, as we explain now. By assumption on Φ, $\det \mathcal{T}(B_{\mathrm{R}}) = (\det C)^{\delta} \times \det \mathcal{T}(A_{\mathrm{R}}) \neq 0$ and then all the rows of $\overline{U}$ have degree δ (see Section 3.1). Consequently, $\overline{U}C$ is a kernel basis for A_{R} of degree at most δ and, since the minimal row degrees of $\ker A_{\mathrm{R}}$ are $\delta, \ldots, \delta$, $\overline{U}C$ is minimal. We conclude as for $i = 1$ that $\overline{U}C = D\overline{N}$ for some invertible constant matrix D. Using similar arguments, we obtain $\underline{U}C = D'\underline{N}$ for another invertible constant matrix D' and (8) therefore holds for $i + 1$. □

The zeros of numerator$(\Phi) \times$ denominator(Φ) define a hypersurface of $\mathsf{K}^{n^2(d+1)}$. By identifying the matrix set $\{A \in \mathsf{K}[x]^{n \times n}:\ \deg A \leqslant d\}$ with $\mathsf{K}^{n^2(d+1)}$, we therefore get the following corollary.

Corollary 4. *The matrices* $B_{i,\mathrm{L}}^{(j)}$, $B_{i,\mathrm{R}}^{(j)}$, $\underline{U}_i^{(j)}$, $\overline{U}_i^{(j)}$ *in* `Inverse`(A) *have degree* δ_i *for all nonsingular* $A \in \mathsf{K}[x]^{n \times n}$ *of degree* d *except those in a certain hypersurface of* $\mathsf{K}^{n^2(d+1)}$.

Again, the typical degrees δ_i in Proposition 3 and Corollary 4 are independent of the way minimal kernel bases are computed. The next section deals with the cost of computing such bases.

4. Minimal kernel basis computation

In algorithm `Inverse` the degrees of the successive minimal bases are not known in advance. To get a low complexity estimate in the favorable cases where the bases actually have small degrees (the generic case), we thus use a minimal basis algorithm whose cost is sensitive to these degrees. In particular, for a $2m \times m$ input matrix M of degree d, the algorithm detects whether the genericity condition $\det \mathcal{T}(M) \neq 0$ is satisfied. If so, a minimal basis of $\ker M$ is returned in $O^{\sim}(m^3 d)$ operations in K.

Several approaches exist for computing minimal polynomial bases of matrix polynomial kernels. Most of them are based on matrix pencil normal forms, see for example [5,6] and references therein, but it is unclear whether they lead to the target complexity estimate $O^{\sim}(m^3 d)$. Following the characterization (4), another possibility is structured linear system solving. In particular, the block-Toeplitz linear system (4b) can be solved in $O^{\sim}(m^3 d)$ field operations [20]. Such a fast

structured solver uses preconditioning with random matrices in order to prevent some particular minors to vanish during recursions [20, Appendix A]. However, for the whole inversion algorithm, this should amount to replacing condition $\Phi(A)\in\mathsf{K}^*$ with another generically satisfied condition of the same nature, say, $\Phi(A)\Psi(A)\in\mathsf{K}^*$.

We now recall in detail another deterministic $O^{\sim}(m^3d)$ approach that relies only on rational function Φ. It is based on matrix Hermite–Padé approximation. We compute a minimal basis of $\ker M$ as a submatrix of a suitable minimal approximant basis for M, called a σ-basis in [2]. This follows the idea of [27, Chapter 4] as applied in [3,4]. Intuitively, a left minimal approximant basis for $M\in\mathsf{K}[x]^{2m\times m}$ is a nonsingular $V\in\mathsf{K}[x]^{2m\times 2m}$ such that

$$VM\equiv 0 \bmod x^{\tau} \quad \text{for some } \tau\in\mathbb{N} \tag{9}$$

and whose row degrees are as small as possible among all such approximants. More precisely, let $\sigma\in\mathbb{N}$ and, following [2, p. 809], let for $v\in\mathsf{K}[x]^{2m}$

$$\operatorname{ord} v = \sup\{\tau\in\mathbb{N}\colon\ v^T(x^m)\cdot M(x^m)\cdot[1,x,\ldots,x^{m-1}]^T\equiv 0 \bmod x^{\tau}\}.$$

Denote further by $\deg v$ the highest degree of all the entries of v. A *σ-basis for the rows of M* is a matrix $V\in\mathsf{K}[x]^{2m\times 2m}$ such that

(I) for $1\leqslant i\leqslant 2m$, $\operatorname{ord} V^{(i,*)}\geqslant\sigma$, where $V^{(i,*)}$ is the ith row of V;
(II) every polynomial vector $v\in\mathsf{K}[x]^{2m}$ such that $\operatorname{ord} v\geqslant\sigma$ admits a unique decomposition $v^T=\sum_{i=1}^{2m} c^{(i)}V^{(i,*)}$ where, for $1\leqslant i\leqslant 2m$, $c^{(i)}\in\mathsf{K}[x]$ and $\deg c^{(i)}+\deg V^{(i,*)}\leqslant\deg v$.

This definition coincides with [2, Definition 3.2] when the m components of the multiindex in [2] are the same. Also, approximation (9) follows from (I) by taking $\sigma=m\tau$, and regularity and minimality of V follow from (II). Note that, in particular, V has degree no more than τ. Proposition 5 below shows that if the approximation order $\sigma=m\tau$ is large enough compared to the minimal row degrees of $\ker M$ then there are exactly m rows of V forming a minimal basis for $\ker M$. Although a more general version not restricted to the $2m\times m$ case can be found in [27], we give a proof here for the sake of completeness.

Proposition 5. *Let $M\in\mathsf{K}[x]^{2m\times m}$ with rank m, degree d and left Kronecker indices $\{d_i\}_{1\leqslant i\leqslant m}$, and let V be a σ-basis for the rows of M. If $\sigma\geqslant m(\max_i d_i+d+1)$ then the m rows of V with smallest degrees form a minimal basis of* $\ker M$.

Proof. For $1\leqslant i\leqslant 2m$, $V^{(i,*)}(x^m)\cdot M(x^m)\cdot[1,x,\ldots,x^{m-1}]^T\equiv 0 \bmod x^{\sigma}$ where the left-hand side is a polynomial of degree at most

$$m(\deg V^{(i,*)}+d+1)-1. \tag{10}$$

It thus follows from (10) and from $\sigma\geqslant m(\max_i d_i+d+1)$ that a row of V whose degree is no more than $\max_i d_i$ is a vector of $\ker M$. Let us now show that V has m rows of respective degrees $d_1,\ldots,d_m$. By definition, a vector u_1 of $\ker M$ of degree d_1

can be written as $u_1 = \sum_{h=1}^{2m} c_1^{(h)} V^{(h,*)}$ with $\deg c_1^{(h)} + \deg V^{(h,*)} \leqslant d_1$. Hence, there exists h_1 such that $\deg V^{(h_1,*)} \leqslant d_1$. Now assume that V has $i-1$ rows $V^{(h_1,*)}, \ldots, V^{(h_{i-1},*)}$ of respective degrees $d_1, \ldots, d_{i-1}$ and let u_i be a vector of $\ker M$ that does not belong to the submodule generated by these $i-1$ rows and such that $\deg u_i = d_i$. As for $i = 1$, there exists $h_i \notin \{h_1, \ldots, h_{i-1}\}$ such that $\deg V^{(h_i,*)} \leqslant d_i$. Therefore, V contains m distinct rows (indexed by $h_1, \ldots, h_m$) such that the h_ith row belongs to $\ker M$ and has degree at most d_i. These m rows are linearly independent in $\ker M$ and, since $\sum_{i=1}^{m} d_i$ is minimal for any such set of rows, they must form a minimal basis. Notice that the remaining m rows of V cannot belong to $\ker M$ and therefore have degrees greater than $\max_i d_i$. The choice of the m rows with smallest degrees in the statement of the proposition is thus well defined. □

Because of the bound (2) on the Kronecker indices, Proposition 5 implies that every σ-basis for the rows of M such that $\sigma \geqslant m(md + d + 1)$ contains a minimal kernel basis for M. However, notice that if $\max_i d_i$ is known to be no more than d then σ can be decreased to $m(2d + 1)$. We make the algorithm sensitive to the output degree in the following straightforward way. If a first attempt with the approximation order $m(2d + 1)$ is sufficient then we stop and output the basis. Otherwise we increase the order. We remark that this test on the order could be included in the approximation algorithm itself.

Algorithm `MinimalKernelBasis`(M)
Input: $M \in \mathsf{K}[x]^{2m \times m}$ of degree d
Output: a minimal basis $U \in \mathsf{K}[x]^{m \times 2m}$ of $\ker M$
$\sigma := m(2d + 1)$;
($\bigstar$) $V :=$ *a* σ-basis for the rows of M;
$U :=$ the m rows of V with smallest degrees;
if $\sigma = m(2d + 1)$ and $UM \neq 0$ **then** go to ($\bigstar$) with $\sigma = m(md + d + 1)$ **fi**;
return U.

In algorithm `MinimalKernelBasis` above, σ-bases can be computed deterministically with the method of [2] or its counterpart using fast matrix multiplication [17]. For our generic inversion purposes, it is sufficient to show that the algorithm has cost $O^{\sim}(m^3 d)$ when U has degree no more than d, i.e. when taking $\sigma = m(2d + 1)$ is enough to get $UM = 0$.

For $\sigma = m(2d + 1)$, the matrix U has degree $O(d)$ and testing for "$UM \neq 0$" therefore costs $O(\mathsf{MM}(m, d))$, where $\mathsf{MM}(m, d)$ is the complexity of multiplying two $m \times m$ polynomial matrices of degree $O(d)$. Recall that ω is the exponent for multiplying two $m \times m$ matrices over K and that multiplying two degree d polynomials in $\mathsf{K}[x]$ can be done in $\mathsf{M}(d) = O(d \log d \log\log d)$ operations in K. From [11] we have $\mathsf{MM}(m, d) = O(m^{\omega} \cdot \mathsf{M}(d))$, thus

$$\mathsf{MM}(m, d) = O(m^{\omega} d \log d \log\log d). \tag{11}$$

When the field K has at least $2d+1$ elements, polynomial matrix multiplication can be done by multipoint evaluation/interpolation; in this case, the best known estimate is from [7,8] and gives

$$\mathsf{MM}(m,d) = O(m^\omega d + m^2 d \log d \log\log d) \quad \text{if } |\mathsf{K}| > 2d. \tag{12}$$

Let $\mathsf{B}(m,d)$ be the complexity of computing a σ-basis for the rows of M such that the approximation order $\sigma = m(2d+1)$ is sufficient. Theorem 2.4 of [17] gives the estimate

$$\mathsf{B}(m,d) = O\left(\sum_{i=0}^{\log\delta} 2^i \mathsf{MM}(m, 2^{-i}\delta)\right), \tag{13}$$

where δ is the smallest integer power of two such that $\delta \geqslant 2d+1$. Since $\delta = O(d)$, it follows from (11) and (13) that $\mathsf{B}(m,d) = O(m^\omega d \log^2 d \log\log d)$ and, for large enough fields, $\mathsf{B}(m,d) = O(m^\omega d \log d) + O^\sim(m^2 d)$ follows from (12) and (13). Using classical matrix multiplication only ($\omega = 3$), we thus obtain the following consequence of Proposition 5.

Proposition 6. *Let $M \in \mathsf{K}[x]^{2m\times m}$ with rank m and degree d. If the left Kronecker indices of M are bounded by d, then Algorithm* `MinimalKernelBasis` *returns a minimal basis of* $\ker M$ *in $O(m^3 d \log^2 d \log\log d)$ field operations; if $|\mathsf{K}| > 2d$, this bound becomes $O(m^3 d \log d) + O^\sim(m^2 d)$.*

If Algorithm `MinimalKernelBasis` does not detect that $\sigma = m(2d+1)$ is sufficient, then a higher approximation order $m(md+d+1)$ is used. Hence, a minimal kernel basis can always be computed in $O^\sim(m^{\omega+1}d)$ operations in K.

5. Cost analysis of inversion for a generic input

We study the cost of algorithm `Inverse` when the input matrix A is such that $\Phi(A) \in \mathsf{K}^*$. In particular, v_i and δ_i are as in (5). We assume that `MinimalKernelBasis` implements the method of the previous section, and the matrix polynomial multiplication complexity $\mathsf{MM}(m,d)$ is as in (11) and (12).

The asymptotic complexity of algorithm `Inverse` can be bounded as follows. First, Propositions 3 and 6 imply that the 2^i minimal bases at step i can be computed at cost $2^i \times O(v_i^3 \delta_i \log^2 \delta_i \log\log \delta_i)$, that is

$$O(2^{-i} n^3 d \log^2(nd) \log\log(nd)). \tag{14}$$

This becomes $O(2^{-i} n^3 d \log(nd)) + O^\sim(n^2 d)$ if $|\mathsf{K}| > 2d$. The update $B := U_i B$ consists in multiplying two block-diagonal matrices, each of them having 2^{i-1} diagonal blocks of order v_i and degree δ_i. This costs $2^{i-1} \times O(\mathsf{MM}(v_i, \delta_i))$. To update the dense matrix U, we update each of its 2^{i-1} block-rows with 2^{i-1} matrix products of order v_i and degree δ_i. This costs $2^{i-1} \times O(2^{i-1}\mathsf{MM}(v_i, \delta_i))$. The total cost of matrix updates at

step i is thus bounded by this latter quantity; with (11) and $\omega = 3$, this gives

$$O(n^3 d \log(nd) \log\log(nd)). \tag{15}$$

With (12) and $\omega = 3$, this gives instead

$$O(n^3 d + 2^i n^2 d \log(nd) \log\log(nd)) \quad \text{if } |\mathsf{K}| > 2d. \tag{16}$$

The total costs induced by (14)–(16) follow from $\sum_{i=1}^{\log n} 2^{-i} = O(1)$ and $\sum_{i=1}^{\log n} 2^i = O(n)$; hence the cost of stage (b) of algorithm `Inverse` is bounded by $O(n^3 d \log^2(nd) \log\log(nd))$, and if $|\mathsf{K}| > 2d$, this reduces to $O(n^3 d \log(nd) \log\log(nd)) + O^{\sim}(n^2 d)$. Stage (c) consists in reducing n^2 fractions whose numerators and denominators have degrees bounded by $nd - d$ and nd, respectively; this can be done by $O(n^3 d \log^2(nd) \log\log(nd))$ field operations as well. We give the conclusion of this analysis in the theorem below.

Theorem 7. *Let $A \in \mathsf{K}[x]^{n \times n}$ be nonsingular of degree d, with n a power of 2. If $\Phi(A) \in \mathsf{K}^*$, algorithm* `Inverse` *computes A^{-1} in $O(n^3 d \log^2(nd) \log\log(nd)) + O^{\sim}(n^2 d)$ field operations.*

Corollary 8. *Algorithm* `Inverse` *computes A^{-1} in $O^{\sim}(n^3 d)$ field operations for all nonsingular $A \in \mathsf{K}[x]^{n \times n}$ of degree d and with n a power of 2, except those in a certain hypersurface of $\mathsf{K}^{n^2(d+1)}$.*

When ignoring logarithmic factors but assuming fast matrix multiplication over K, (14) and (15) become, respectively, $O^{\sim}(2^{(2-\omega)i} n^{\omega} d)$ and $O^{\sim}(2^{(3-\omega)i} n^{\omega} d)$. When i ranges from 1 to $\log n$, the cost of computing minimal kernel bases therefore decreases from $O^{\sim}(n^{\omega} d)$ to $O^{\sim}(n^2 d)$; simultaneously, the cost of matrix updates increases from $O^{\sim}(n^{\omega} d)$ to $O^{\sim}(n^3 d)$. Hence, asymptotically and regardless of logarithmic factors, basis computations dominate at early stages of the algorithm whereas matrix updates dominate at the end.

Clearly, it remains to remove the assumption that n is a power of two. The assumption is used in Proposition 3 and thus Theorem 7. For general dimensions, similar results should follow from inverting

$$\mathcal{A} = \begin{bmatrix} A & \\ X & I_{2^p - n} \end{bmatrix}, \quad 2^{p-1} < n < 2^p,$$

with X a generic polynomial matrix of degree d. If A is generic, then the degrees of the minimal kernel bases in `Inverse`$(\mathcal{A})$ should still be bounded by the δ_i's, although not equal to them anymore. It also remains to get rid of the genericity condition $\Phi(A) \in \mathsf{K}^*$, and to develop a method for handling minimal kernel bases with possibly unbalanced degrees.

We have noticed in [17] that Algorithm `Inverse` may be specialized for computing the determinant of a generic matrix A in $O^{\sim}(n^{\omega} d)$ operations. In the generic case this yields an alternative approach to the determinant algorithm in [30,31]. These two different methods for the determinant are respectively based on Hermite–Padé

approximation and Newton–Hensel lifting: how do they compare? As we have seen, recent advances show that several problems on polynomial matrices can be solved in $O^{\sim}(n^{\omega}d)$ operations. The latter is also the cost of polynomial matrix multiplication. For inversion we get an algorithm whose cost is essentially the size of the output. The extension of the list of polynomial matrix problems that can be solved in asymptotically $O^{\sim}(n^{\omega}d)$ algebraic operations plus the input/output size should be pursued.

Acknowledgments

We thank Erich Kaltofen for his numerous helpful comments in the course of this work.

References

[1] E.H. Bareiss, Computational solution of matrix problems over an integral domain, J. Inst. Math. Appl. 10 (1972) 68–104.
[2] B. Beckermann, G. Labahn, A uniform approach for the fast computation of matrix-type Padé approximants, SIAM J. Matrix Anal. Appl. 15 (1994) 804–823.
[3] B. Beckermann, G. Labahn, G. Villard, Shifted normal forms of polynomial matrices, in: Proceedings of International Symposium on Symbolic and Algebraic Computation, Vancouver, Canada, ACM Press, New York, 1999, pp. 189–196.
[4] B. Beckermann, G. Labahn, G. Villard, Normal forms for general polynomial matrices, J. Symbolic Comput., to appear.
[5] Th. Beelen, G.J. van den Hurk, C. Praagman, A new method for computing a column reduced polynomial matrix, Systems Control Lett. 10 (1988) 217–224.
[6] Th. Beelen, P. Van Dooren, An improved algorithm for the computation of Kronecker's canonical form of a singular pencil, Linear Algebra Appl. 105 (1988) 9–65.
[7] A. Bostan, Algorithmique efficace pour des opérations de base en calcul formel, Thèse de Doctorat, École Polytechnique, Palaiseau, France, Décembre 2003.
[8] A. Bostan, E. Schost, Polynomial evaluation and interpolation on special sets of points, Preprint, Laboratoire STIX, École Polytechnique, Palaiseau, France, September 2003.
[9] J. Bunch, J. Hopcroft, Triangular factorization and inversion by fast matrix multiplication, Math. Comput. 28 (1974) 231–236.
[10] P. Bürgisser, M. Clausen, M.A. Shokrollahi, Algebraic Complexity Theory, Vol. 315, Grundlehren der Mathematischen Wissenschaften, Springer, Berlin, 1997.
[11] D.G. Cantor, E. Kaltofen, On fast multiplication of polynomials over arbitrary algebras, Acta Inform. 28 (7) (1991) 693–701.
[12] J.D. Dixon, Exact solutions of linear equations using p-adic expansions, Numer. Math. 40 (1982) 137–141.
[13] W. Eberly, M. Giesbrecht, G. Villard, Computing the determinant and Smith form of an integer matrix, In: Proceedings of 41st Annual IEEE Symposium on Foundations of Computer Science, Redondo Beach, CA, USA, IEEE Computer Society Press, Silver Spring, MD, November 2000, pp. 675–685.
[14] G.D. Forney Jr., Minimal bases of rational vector spaces, with applications to multivariable linear systems, SIAM J. Control 13 (1975) 493–520.
[15] F.R. Gantmacher, Théorie des matrices, Editions Jacques Gabay, 1990.

[16] J. von zur Gathen, J. Gerhard, Modern Computer Algebra, Cambridge University Press, Cambridge, 1999.
[17] P. Giorgi, C.-P. Jeannerod, G. Villard, On the complexity of polynomial matrix computations, in: Proceedings of International Symposium on Symbolic and Algebraic Computation, ACM Press, Philadelphia, PE, USA, 2003, pp. 135–142.
[18] T. Kailath, Linear Systems, Prentice-Hall, Englewood Cliffs, NJ, 1980.
[19] E. Kaltofen, On computing determinants without divisions, in: Proceedings of International Symposium on Symbolic and Algebraic Computation, Berkeley, ACM Press, CA, USA, 1992, pp. 342–349.
[20] E. Kaltofen, Analysis of Coppersmith's block Wiedemann algorithm for the parallel solution of sparse linear systems, Math. Comput. 64 (210) (1995) 777–806.
[21] E. Kaltofen, G. Villard, On the complexity of computing determinants, in: Proceedings of Fifth Asian Symposium on Computer Mathematics, Lecture Notes Series on Computing, Vol. 9, World Scientific, Singapore, 2001, pp. 13–27. Extended abstract, invited contribution (Kaltofen), journal version in [22].
[22] E. Kaltofen, G. Villard, On the complexity of computing determinants, Research Report LIP 2003-36, Laboratoire LIP, ENS Lyon, France, July 2003.
[23] C.-A. Lin, C.-W. Yang, T.-F. Hsieh, An algorithm for inverting rational matrices, Systems Control Lett. 27 (1996) 47–53.
[24] J.D. Lipson, Chinese remainder and interpolation algorithms, in: Proceedings of the Second ACM Symposium on Symbolic and Algebraic Manipulation, ACM Press, Los Angeles, CA, USA, 1971, pp. 372–391.
[25] R.T. Moenck, J.H. Carter, Approximate algorithms to derive exact solutions to systems of linear equations, in: Proceedings of EUROSAM, Lecture Notes in Computer Science, Vol. 72, Springer, Berlin, 1979, pp. 63–73.
[26] T. Mulders, A. Storjohann, On lattice reduction for polynomial matrices, J. Symbolic Comput. 35 (4) (2003) 377–401.
[27] M.-P. Quéré-Stuchlik, Algorithmique des faisceaux linéaires de matrices—Application à la théorie des systèmes linéaires et à la résolution d'équations algébro-différentielles, Thèse de Doctorat, Université Paris 6, Paris, France, Juin 1997.
[28] A. Schönhage, Unitäre Transformationen grosser Matrizen, Numer. Math. 20 (1973) 409–417.
[29] A. Storjohann, Algorithms for matrix canonical forms, Ph.D. Thesis, ETH—Swiss Federal Institute of Technology, Zürich, December 2000.
[30] A. Storjohann, High-order lifting, in: Proceedings of International Symposium on Symbolic and Algebraic Computation, Lille, France, ACM Press, New York, 2002, pp. 246–254.
[31] A. Storjohann, High-order lifting and integrality certification, J. Symbolic Comput. 36 (3–4) (2003) 613–648.
[32] V. Strassen, Gaussian elimination is not optimal, Numer. Math. 13 (1969) 354–356.