Università di Pisa

Dipartimento di Informatica

Technical Report : TR-97-15

Fast fraction-free triangularization of Bezoutians with applications to sub-resultant chain computation

D. A. Bini, L. Gemignani

September 05, 1997

ADDR: Corso Italia 40,56125 Pisa,Italy. TEL: +39-50-887111. FAX: +39-50-887226

Fast fraction-free triangularization of Bezoutians with applications to sub-resultant chain computation

Dario Andrea Bini

Dipartimento di Matematica, Università di Pisa,
via Buonarroti, 56100 Pisa, Italy

and

Luca Gemignani *

Dipartimento di Informatica, Università di Pisa,
Corso Italia 40, 56125 Pisa, Italy

ABSTRACT

An algorithm for the computation of the LU factorization over the integers of an $n \times n$ Bezoutian B is presented. The algorithm requires $O(n^2)$ arithmetic operations and involves integers having at most $O(n \log nc)$ bits, where c is an upper bound of the moduli of the integer entries of B. As an application, by using the correlations between Bezoutians and the Euclidean scheme, we devise a new division-free algorithm for the computation of the polynomial pseudo-remainder sequence of two polynomials of degree at most n in $O(n^2)$ arithmetic operations. The growth of the length of the integers involved in the computation is kept at the minimum value, i.e., $O(n \log nc)$ bits, no matter if the sequence is normal or not, where c is an upper bound of the moduli of the input coefficients. The algorithms, that rely on the Bareiss technique [2] and on the properties of the Schur complements of Bezoutians [4], improve the ones of [28], [8], [1].

---

1. INTRODUCTION

Let $u(x) = \sum_{i=0}^{n} u_i x^i \in \mathbf{Z}[x]$ and $v(x) = \sum_{i=0}^{m} v_i x^i \in \mathbf{Z}[x]$ be two polynomials of degree n and m, $n \geq m$, respectively, having integer coefficients.

*Supported by 40% funds of the Progetto Analisi numerica e matematica computazionale of MURST and by G.N.I.M. of C.N.R..

655 Avenue of the Americas, New York, NY 10010 0024- 3795/93/$6.00

The $n \times n$ Bezoutian $B = B(u,v) = (b_{i,j}) \in \mathbf{Z}^{n\times n}$ associated with $u(x)$ and $v(x)$ is defined by

$$\frac{u(x)v(y) - u(y)v(x)}{x-y} = \sum_{i,j=1}^{n} b_{i,j} x^{i-1} y^{j-1},$$

and can be represented in the following form [17],

$$\begin{bmatrix} u_1 & u_2 & \dots & u_n \\ u_2 & \cdot^{\cdot^{\cdot}} & \cdot^{\cdot^{\cdot}} & \\ \vdots & \cdot^{\cdot^{\cdot}} & \bigcirc & \\ u_n & & & \end{bmatrix} \begin{bmatrix} v_0 & v_1 & \dots & v_{n-1} \\ & v_0 & \ddots & \vdots \\ & \bigcirc & \ddots & v_1 \\ & & & v_0 \end{bmatrix} - \begin{bmatrix} v_1 & v_2 & \dots & v_n \\ v_2 & \cdot^{\cdot^{\cdot}} & \cdot^{\cdot^{\cdot}} & \\ \vdots & \cdot^{\cdot^{\cdot}} & \bigcirc & \\ v_n & & & \end{bmatrix} \begin{bmatrix} u_0 & u_1 & \dots & u_{n-1} \\ & u_0 & \ddots & \vdots \\ & \bigcirc & \ddots & u_1 \\ & & & u_0 \end{bmatrix}, \tag{1.1}$$

where we set $v_j = 0$ if $j > m$. Observe that the left-hand factors in the above matrix products are Hankel matrices, i.e., the entries in position (i,j) are functions of $i+j$, while the right-hand factors are Toeplitz matrices, i.e., the entries in position (i,j) are function of $j-i$. From this representation it is clear that the entries of $B(u,v)$ are bilinear functions in the coefficients of $u(x)$ and $v(x)$; furthermore, given the entries of a Bezoutian B, it is possible to compute the coefficients of two polynomials $u(x)$ and $v(x)$ such that $B = B(u,v)$ [5]. In particular, the last row of B, i.e., $u_n \begin{bmatrix} v_0 & \dots & v_{n-1} \end{bmatrix}$, provides the degree m as well as the coefficients of $v(x)$, up to the multiplicative factor u_n.

Bezoutians play an important role in many diverse fields of numerical and symbolic computing because of their connections to system-theoretic concepts like stability and signature-symmetric realizations (see [18], [10], [20] and [22]). These relations make possible to treat problems arising from signal processing and control theory as an integral part of linear algebra. In this way, the efficient recursive evaluation of a triangular factorization of B becomes a central computational issue.

Many fast algorithms, that is, requiring $O(n^2)$ arithmetic operations, have been developed for the triangularization of Bezoutians with entries over a field [22]. However, these methods generally suffer from potential instabilities which occur when ill-conditioned submatrices are encountered.

In this paper we address the stability problems by using exact integer arithmetic. That is, we develop a fast fraction-free procedure which reduces B into triangular form by means of ring operations and exact divisions only.

Our approach yields a symbolic method for the inversion of B that can be used for the development of fast inversion algorithms for other structured matrices such as Toeplitz and Hankel ones. Further, our procedure can be described as a matrix analogue of some variant of the Euclidean algorithm applied to $u(x)$ and $v(x)$:

$$\begin{array}{rcl} r_0(x) & = & u(x), \quad r_1(x) = v(x) \\ r_{i-1}(x) & = & r_i(x)q_i(x) - r_{i+1}(x) \quad i = 1, \ldots, L, \end{array} \tag{1.2}$$

where $-r_{i+1}(x)$ is the remainder of the division of $r_{i-1}(x)$ and $r_i(x)$, and $r_L(x)$ is the greatest common divisor (gcd) of $u(x)$ and $v(x)$. The sequence $\{r_i(x)\}_{i=0}^{L}$ is a polynomial remainder sequence (p.r.s.) of $u(x)$ and $v(x)$.

The sub-resultant theory (see [7], [9] and [24]) is the fundamental tool for computing any p.r.s. over an integral domain rather than on a field. The construction and the divisibility properties of sub-resultants are generally stated in terms of determinantal relations. Differently, we feel that it is extremely illuminating to see the sub-resultant chain computation as an effective tool for the fast symbolic recursive inversion of structured matrices. A similar matrix approach has been already considered in [1], [8], [12] and [28] but our method reaches better complexity estimates.

A first well-known noticeable property relating Bezoutians with Euclid's algorithm raises from the question of checking the relative primeness of two polynomials. Clearly, this can be done by computing the greatest common divisor by means of the celebrated Euclidean scheme (1.2). A different matrix approach that goes back to MacDuffee [25] is based on the fact that $u(x)$ and $v(x)$ are relatively prime if and only if $\det(v(F_u)) \neq 0$, where F_u denotes the Frobenius matrix associated with $u(x)$,

$$F_u = \frac{1}{u_n} \begin{bmatrix} 0 & u_n & & \\ & \ddots & \ddots & \\ & & 0 & u_n \\ -u_0 & \ldots & \ldots & -u_{n-1} \end{bmatrix}. \tag{1.3}$$

From Barnett's decomposition [18],

$$B(u, v) = B(u, 1)v(F_u) \tag{1.4}$$

it holds that MacDuffee's condition is equivalent to the non-singularity of the Bezoutian $B(u, v)$.

Another important property relating Bezoutians and the Euclidean scheme concerns the concept of *Schur complement.* Assume we are given a matrix A partitioned into four blocks $A_{i,j}$, $i, j = 1, 2$, where $A_{1,1}$ is $k \times k$ and non-singular. The Schur complement of $A_{1,1}$ in A is the matrix

$S = A_{2,2} - A_{2,1}A_{1,1}^{-1}A_{1,2}$ that is obtained in one step of block Gaussian elimination.

To our knowledge, the relations between the Schur complements of $B(u,v)$ and the polynomials generated by (1.2) have not been completely understood in the literature and no clear statement can be found until the paper [4] where the authors have shown that the computation of the Euclidean scheme can indeed be reduced to computing a block triangular factorization of the matrix $\hat{B} = J_n B(u,v) J_n$. Here and hereafter J_n is the $n \times n$ permutation matrix with unit anti-diagonal entries. Such reduction relies on the following:

PROPOSITION 1.1. *The $k \times k$ leading principal submatrix $\hat{B}_k$ of $\hat{B} = J_n B(u,v) J_n$ is non-singular if and only if there exists i, $1 \le i \le L$, such that $k = m_i = n - \deg r_i(x)$. Moreover, the Schur complement $S_{m_i} \in \mathbf{Q}^{(n-m_i)\times(n-m_i)}$ of $\hat{B}_{m_i}$ in $\hat{B}$ satisfies*

$$S_{m_i} = J_{n-m_i} B(r_i, r_{i+1}) J_{n-m_i}, \quad 1 \le m_i < n.$$

This result can be used for the efficient recursive evaluation over the quotient field $\mathbf{Q}$ of a block triangular factorization of both $J_n B(u,v) J_n$ and other related structured matrices (see [4], [11] and [14]). Moreover, it leads to the fastest parallel implementation of the Euclidean scheme over any field of constants [4], [13].

In this paper we develop a new method, based on proposition 1.1 and on the Bareiss fraction-free elimination scheme [2], for the triangularization of $\hat{B}$ over the integral domain $\mathbf{Z}$.

We prove that the structural invariance of the Bezoutian under Schur complementation still holds if the Bareiss variant of Gaussian elimination is applied. In this way we may devise an algorithm for the LU factorization of $\hat{B}$ that requires $O(n^2)$ arithmetic operations among integer numbers, where the length of the intermediate values of the computation grows at most as $O(n(\log nc))$, where c is an upper bound of the moduli of the coefficients of $u(x)$ and $v(x)$. Both the bounds on the arithmetic cost and on the length of the integers delivered by the algorithm are optimal.

As an application of our algorithm we present a new fraction-free implementation of the Euclidean scheme computation which delivers the (integer) coefficients of all the pseudo-remainder sequence, in $O(n^2)$ arithmetic operations between $O(n\log(nc))$-bit integers, no matter the polynomial remainder sequence is normal or not. Our approach can also be used for the sub-resultant chain computation since we show that the computed pseudo-remainders coincide with the sub-resultants up to a suitable scaling factor.

Our algorithm can be directly extended to polynomials with coefficients over more general integral domains [3]. Another extension is obtained by replacing the polynomials $u(x)$ and $v(x)$ with power series. In this case the

Bezoutian, as well as its Schur complements are infinite matrices associated with a power series remainder sequence, as defined in [8].

As a by-product, from the LU factorization of $\hat{B}$ we obtain a triangular factorization of the Sylvester matrix that can be used for certain polynomial computations.

The paper is organized as follows. In section 2 we revisit the Bareiss variant of Gaussian elimination and prepare the tools for the analysis of the LU factorization of the Bezoutian. In section 3 we describe and analyze our algorithm for the LU factorization over $\mathbf{Z}$ of the matrix $\hat{B}$. In section 4 the polynomial version of the algorithm for the computation of the polynomial remainder sequence is presented and analyzed. Section 5 contains some extensions and further applications.

2. PRELIMINARIES

Let us begin by reviewing the Bareiss variant of Gaussian elimination for matrices with entries over the integers [2].

Assume that $A \in \mathbf{Z}^{n\times n}$ is a strongly non-singular matrix, i.e., for any j such that $1 \leq j \leq n$, the $j \times j$ leading principal submatrix A_j of A is non-singular. It is well known from linear algebra that we may reduce the matrix $A = A^{(0)} = (a_{i,j}^{(0)})$ into upper triangular form by using Gaussian elimination. That is, we apply a sequence of transformations to $A^{(0)}$,

$$A^{(0)} \to A^{(1)} \to \ldots \to A^{(n-1)},$$

in such a way that $A^{(n-1)}$ is an upper triangular matrix. Each transformation requires elementary row operations only. A matrix description of the zeroing process involves certain elementary matrices called Gauss transformations. Specifically, we find that

$$A^{(k)} = M_k A^{(k-1)}, \tag{2.1}$$

where M_k is an elementary matrix (Gauss transformation) defined by

$$M_k = I_n - \boldsymbol{\tau}_k \mathbf{e}_k^T, \;\; \boldsymbol{\tau}_k^T = (a_{k,k}^{(k-1)})^{-1} \left[\begin{array}{cccc} \mathbf{0}^T & a_{k+1,k}^{(k-1)} & \ldots & a_{n,k}^{(k-1)} \end{array} \right], \;\; \mathbf{0} \in \mathbf{Z}^k.$$

Here and hereafter $\mathbf{e}_j$ denotes the $j-$th column of the identity matrix I_n of order n.

Equation (2.1) leads to the following rule for Gaussian elimination

$$a_{i,j}^{(k)} = a_{i,j}^{(k-1)} - \frac{a_{i,k}^{(k-1)}}{a_{k,k}^{(k-1)}} a_{k,j}^{(k-1)}, \quad k+1 \leq i \leq n, \; k+1 \leq j \leq n, \tag{2.2}$$

which is applied for $k = 1, 2, \ldots, n-1$. The entries $a_{k,k}^{(k-1)}$ are referred to as the pivots. This formula clearly shows that the elimination process produces fractions and, therefore, the entries of the matrix $A^{(k)}$ at the $k-$th step generally belong to the quotient field $\mathbf{Q}$.

However, the update equation (2.2) can be suitably modified in order to perform all the computations over the integral domain $\mathbf{Z}$ without moving to the quotient field. The resulting computational scheme is due to Bareiss [2] who also proposed extensions to more general integral domains [3].

Let us introduce the cumulative elimination matrix E_k, $k \geq 1$, such that

$$E_k = M_k E_{k-1} = (I_n - \boldsymbol{\tau}_k \mathbf{e}_k^T) E_{k-1}, \quad E_0 = I_n, \tag{2.3}$$

which satisfies

$$E_k A = A^{(k)}, \quad 1 \leq k \leq n-1. \tag{2.4}$$

In the view of the echelon form of $A^{(k)}$, we easily find that the vectors $\mathbf{e}_i^{(k)} \in \mathbf{Q}^i$, $1 \leq i \leq n-1$, consisting of the first i entries in the $(i+1)$-st row of E_k are such that

$$(\mathbf{e}_i^{(k)})^T A_{\min\{i,k\}} = -\left[\begin{array}{ccc} a_{i+1,1} & \ldots & a_{i+1,\min\{i,k\}} \end{array}\right]. \tag{2.5}$$

Since A is strongly non-singular, the rows of E_k are uniquely specified as the solutions of the above linear systems.

Replacing the solutions of (2.5) with the corresponding Cramer solutions

$$(\hat{\mathbf{e}}_i^{(k)})^T A_{\min\{i,k\}} = -\det(A_{\min\{i,k\}})\left[\begin{array}{ccc} a_{i+1,1} & \ldots & a_{i+1,\min\{i,k\}} \end{array}\right], \tag{2.6}$$

is the key to obtain an elimination procedure which works over $\mathbf{Z}$. The matrix $\hat{E}_k$ whose rows are defined by the vectors $\hat{\mathbf{e}}_i^{(k)}$ can be written as

$$\hat{E}_k = D_k E_k,$$

where

$$D_0 = I_n, \; D_k = \text{ Diag } [1, \det A_1, \ldots, \det A_{k-1}, \det A_k, \ldots, \det A_k].$$

The Bareiss triangularization scheme relies on computation of the sequence

$$A = \hat{A}^{(0)} \to \hat{A}^{(1)} \to \ldots \to \hat{A}^{(n-1)},$$

where

$$\hat{E}_k A = \hat{A}^{(k)} = (\hat{a}_{i,j}^{(k)}), \quad 1 \leq k \leq n-1, \tag{2.7}$$

For $1 \le k \le n-1$, from (2.3) there follows that

$$\hat{E}_k = (\ \mathrm{Diag}\ [I_k, \frac{\det A_k}{\det A_{k-1}}, \ldots, \frac{\det A_k}{\det A_{k-1}}] - D_k \boldsymbol{\tau}_k \mathbf{e}_k^T D_{k-1}^{-1}) \hat{E}_{k-1},$$

and, therefore,

$$\hat{E}_k = (\ \mathrm{Diag}\ [I_k, \frac{\det A_k}{\det A_{k-1}}, \ldots, \frac{\det A_k}{\det A_{k-1}}] - D_k \boldsymbol{\tau}_k \frac{\mathbf{e}_k^T}{\det A_{k-1}}) \hat{E}_{k-1}. \quad (2.8)$$

Comparing (2.4) and (2.7) says that

$$\hat{a}_{i,j}^{(k-1)} = \det A_{k-1} a_{i,j}^{(k-1)}, \quad k \le i, j \le n.$$

In particular, regarding at the pivot entries, we have

$$\hat{a}_{k,k}^{(k-1)} = \det A_{k-1} a_{k,k}^{(k-1)} = \det A_{k-1} \frac{\det A_k}{\det A_{k-1}} = \det A_k.$$

These equations allow us to express the vector $D_k \boldsymbol{\tau}_k$ of (2.8) in terms of the entries of $\hat{A}^{(k-1)}$,

$$D_k \boldsymbol{\tau}_k = \begin{bmatrix} \mathbf{0}^T & \hat{a}_{k+1,k}^{(k-1)} & \ldots & \hat{a}_{n,k}^{(k-1)} \end{bmatrix}^T, \quad \mathbf{0} \in \mathbf{Z}^k.$$

In this way we arrive at the following update equation

$$\hat{a}_{k-1,k-1}^{(k-2)} \hat{E}_k = (\ \mathrm{Diag}\ [\hat{a}_{k-1,k-1}^{(k-2)} I_k, \hat{a}_{k,k}^{(k-1)} I_{n-k}] - \begin{bmatrix} \mathbf{0} \\ \hat{a}_{k+1,k}^{(k-1)} \\ \vdots \\ \hat{a}_{n,k}^{(k-1)} \end{bmatrix} \mathbf{e}_k^T) \hat{E}_{k-1}.$$

This gives the fraction-free elimination scheme:

$$\hat{a}_{i,j}^{(k)} = \frac{\hat{a}_{k,k}^{(k-1)} \hat{a}_{i,j}^{(k-1)} - \hat{a}_{i,k}^{(k-1)} \hat{a}_{k,j}^{(k-1)}}{\hat{a}_{k-1,k-1}^{(k-2)}}, \quad k+1 \le i \le n,\ k+1 \le j \le n, \ (2.9)$$

which is applied for $k = 1, 2, \ldots, n-1$, with the initial condition $\hat{a}_{0,0}^{(-1)} = 1$.

As we know, the assumption about the non-singularity of the leading principal submatrices of A can be removed. In practice, for a generic matrix A, the rule (2.9) should be complemented with a suitable pivoting strategy which has to be applied whenever $\hat{a}_{k,k}^{(k-1)} = 0$. The resulting algorithm will reduce a permuted version of A into upper triangular form by means of ring operations and exact divisions only.

The arithmetic cost of Bareiss' algorithm is the same as the one of Gaussian elimination, i.e., $O(n^3)$ arithmetic operations. We may easily compute an upper bound to the Boolean cost if we estimate the bit-length of the integers involved in the computation. In fact, if $\nu(m)$ denotes the Boolean cost of the multiplication of m-bit integers and $m(n)$ is the maximum length of the integers involved in the computation, then $O(n^3\nu(m(n)))$ is an upper bound to the Boolean complexity of the Bareiss variant of Gaussian elimination. Now, m-bit integer multiplication can be performed with the customary algorithm at the cost $\nu(m) = O(m^2)$, or by means of Karatsuba algorithm [6] at the cost $O(m^{\log_2 3})$. The algorithm of Schönhage-Strassen [29], based on FFT, reaches the best known complexity bound $\nu(m) = O(m \log m \log\log m)$, but with a larger overhead factor. In the current multi-precision packages like GMP, the GNU Multi Precision package, all of the three algorithms are implemented and applied according to the length of the operands.

Relation (2.6) is the key to analyze the growth of the length of $\hat{a}_{i,j}^{(k)}$ and consequently, the Boolean cost of Bareiss' algorithm. In fact, using Hadamard's inequality, we see that the length of the entries of $\hat{E}_k$ is bounded by $O(k \log kc)$ where c is an upper bound on the magnitude of the entries of the starting matrix A. From (2.7) there follows that the same estimates hold for the entries of $\hat{A}^{(k)}$, whence we have $m(n) = O(n \log nc)$.

In conclusion, we are able to reduce a generic matrix $A \in \mathbf{Z}^{n\times n}$ into its upper triangular form using the Bareiss integer-preserving transformation algorithm at the overall cost of $O(n^3\nu(n \log nc))$ Boolean operations. In the next section we will explain how it is possible to improve this complexity bound in the case where the matrix A has the structure of a Bezoutian of two polynomials with integer coefficients.

3. TRIANGULARIZATION OF BEZOUTIANS

In this section we introduce and analyze a division-free algorithm for computing a triangular factorization of the permuted Bezoutian $\hat{B} = J_n B J_n$ over the integers. For this purpose we need a different description of $B = B(u, v)$.

By combining the Barnett factorization (1.4), applied with $v(x) = x$, with the symmetry of the Bezoutian we obtain the interesting commuting property

$$B(u, 1)F_u = F_u^T B(u, 1) \tag{3.1}$$

that implies the following characterization of the entries of B.

PROPOSITION 3.1. *The rows $\mathbf{w}_i^T = \mathbf{e}_i^T \hat{B}$ of the matrix $\hat{B}$ can be constructed according to the recurrence:*

$$\mathbf{w}_{i+1}^T = \mathbf{w}_i^T J_n F_u J_n + \frac{u_{n-i}}{u_n} \mathbf{w}_1^T, \quad 1 \le i \le n-1.$$

where we have $\mathbf{w}_1^T = u_n \begin{bmatrix} v_0 & \dots & v_{n-1} \end{bmatrix}$, $v_j = 0$ if $j > m =$ $\deg(v(x))$, and F_u is the Frobenius matrix (1.3) associated with $u(x)$.

Proof. From (1.4), using the symmetry of $B(u,v)$, we obtain that

$$\hat{B} = J_n v(F_u^T) J_n J_n B(u,1) J_n = v(J_n F_u^T J_n) J_n B(u,1) J_n.$$

The rows of the matrix $J_n F_u^T J_n$ satisfy

$$\mathbf{e}_i^T J_n F_u^T J_n = \mathbf{e}_{i+1}^T - \frac{u_{n-i}}{u_n} \mathbf{e}_1^T, \quad 1 \le i \le n,$$

where we set $\mathbf{e}_{n+1} = \mathbf{0}$. There follows that

$$(\mathbf{e}_i^T J_n F_u^T J_n - \mathbf{e}_{i+1}^T + \frac{u_{n-i}}{u_n} \mathbf{e}_1^T) v(J_n F_u^T J_n) = 0, \quad 1 \le i \le n.$$

Since $J_n F_u^T J_n$ and $v(J_n F_u^T J_n)$ commute, then, for $1 \le i \le n-1$, we find that

$$\mathbf{e}_i^T v(J_n F_u^T J_n) J_n F_u^T J_n + \frac{u_{n-i}}{u_n} \mathbf{e}_1^T v(J_n F_u^T J_n) = \mathbf{e}_{i+1}^T v(J_n F_u^T J_n).$$

Such recurrences allow the fast evaluation of the rows of $v(J_n F_u^T J_n)$ once we know the first one. Moreover, in the view of (3.1), by multiplying them on the right by $J_n B(u,1) J_n$, we obtain similar recursions for the rows of $\hat{B}$, namely,

$$\mathbf{e}_i^T \hat{B} J_n F_u J_n + \frac{u_{n-i}}{u_n} \mathbf{e}_1^T \hat{B} = \mathbf{e}_{i+1}^T \hat{B}, \quad 1 \le i \le n-1,$$

which establishes the proposition. ∎

Now we adjust the Bareiss integer-preserving transformation algorithm applied to $\hat{B}$ in order to exploit the structural properties of this matrix. We give an inductive description of the procedure, where the structure and the properties of the matrices involved are described step by step. Denote with

$$\hat{B}^{(0)} = \hat{B} \to \hat{B}^{(1)} \to \cdots \to \hat{B}^{(n-1)}$$

the sequence of matrices generated by our algorithm, where $B^{(n-1)}$ is upper triangular. In particular we focus our attention on the construction of the transformation that maps $\hat{B}^{(m_i)}$ to $\hat{B}^{(m_{i+1})}$, where m_i, $i = 0, \dots, L$,

($m_0 = 0, S_{m_0} = \hat{B}$), are defined as in proposition 1.1. This reduction is accomplished by means of single steps of the Bareiss scheme complemented with a suitable pivoting strategy.

Assume that after k steps, $k = m_i$, $k \geq 0$, the computed matrix $\hat{B}^{(k)} = (\hat{b}_{i,j}^{(k)})$ is such that

$$\hat{E}_k \hat{B} = \hat{B}^{(k)} = \begin{bmatrix} T^{(k)} & W^{(k)} \\ 0 & Q^{(k)} \end{bmatrix}, \tag{3.2}$$

where $T^{(k)} \in \mathbf{Z}^{k\times k}$ is upper triangular and $Q^{(k)} \in \mathbf{Z}^{(n-k)\times(n-k)}$. Further, let the elimination matrix $\hat{E}_k$ have the following structure

$$\hat{E}_k = \begin{bmatrix} E^{(k)} & 0 \\ D^{(k)} & \hat{b}_{k,k}^{(k-1)} I_{n-k} \end{bmatrix}, \quad E^{(k)} \in \mathbf{Z}^{k\times k},$$

where $\hat{b}_{k,k}^{(k-1)} \neq 0$ ($\hat{b}_{0,0}^{(-1)} = 1$) denotes the pivot at the $k-$th step.

Partition the matrix $\hat{B}$ as

$$\hat{B} = \hat{B}^{(0)} = \begin{bmatrix} \hat{B}_k & G_k \\ G_k^T & C_k \end{bmatrix}, \quad C_k \in \mathbf{Z}^{(n-k)\times(n-k)}.$$

From (3.2) we obtain the following two matrix equations

$$D^{(k)} \hat{B}_k + \hat{b}_{k,k}^{(k-1)} G^T = 0$$

and

$$D^{(k)} G + \hat{b}_{k,k}^{(k-1)} C_k = Q^{(k)}.$$

Since $\hat{B}_k = \hat{B}_{m_i}$ is non-singular in the light of proposition 1.1, the former equation yields

$$D^{(k)} = -\hat{b}_{k,k}^{(k-1)} G^T \hat{B}_k^{-1},$$

and replacing this expression in the latter provides

$$\hat{b}_{k,k}^{(k-1)} (C_k - G^T \hat{B}_k^{-1} G) = Q^{(k)}.$$

In other words for $k > 0$, $Q^{(k)}$ coincides with the Schur complement S_{m_i} up to the multiplicative factor $\hat{b}_{k,k}^{(k-1)}$, i.e.,

$$Q^{(k)} = \hat{b}_{k,k}^{(k-1)} S_{m_i}. \tag{3.3}$$

In the view of proposition 1.1, this means that $Q^{(k)}$ has the structure of a permuted Bezoutian associated with the polynomials $r_i(x)$ and $r_{i+1}(x)$ of degree $n-m_i$ and $n-m_{i+1}$, respectively, defined in (1.2). In particular, from (1.1) there follows that the $(m_{i+1}-m_i)\times(m_{i+1}-m_i)$ leading principal submatrix of $Q^{(k)}$ is a non-singular lower triangular (with respect to the anti-diagonal) Hankel matrix.

Now, consider the problem of determining $\hat{B}^{(m_{i+1})}$ from $\hat{B}^{(m_i)}$. This issue can be addressed by using scaled Gauss transformations applied to the matrix $Q^{(k)}$ only. In other words we focus our attention on the integer preserving transformation which yields $Q^{(m_{i+1})}$ from $Q^{(m_i)}$. To be more specific, let $\hat{Q}^{(k)}$ denote the $(n-k)\times(n-k)$ matrix obtained by permuting the rows of $Q^{(k)}$ in the following way

$$\hat{Q}^{(k)} = \begin{bmatrix} J_{m_{i+1}-m_i} & 0 \\ 0 & I_{n-m_{i+1}} \end{bmatrix} Q^{(k)}. \tag{3.4}$$

It can be easily seen that the $(m_{i+1}-m_i)\times(m_{i+1}-m_i)$ leading principal submatrix $(\hat{Q}^{(k)})_{m_{i+1}-m_i}$ of $\hat{Q}^{(k)}$ is a non-singular upper triangular Toeplitz matrix. This fact strongly simplifies the triangularization process. Moreover, the permutation $J_{m_{i+1}-m_i}$ introduced in this way does not affect the sought LU factorization. In fact, the Schur complement of $(\hat{Q}^{(k)})_{m_{i+1}-m_i}$ in $\hat{Q}^{(k)}$ coincides with $S_{m_{i+1}}$ up to a suitable scaling factor. This is the key observation that allows us to apply the integer preserving transformation keeping unchanged the Bezout structure of the Schur complement. In this way, by virtue of proposition 1.1, we obtain that $Q^{(m_{i+1})}$ is specified by its first row or column.

Let us analyze more closely the transformation that provides the first row of $Q^{(m_{i+1})}$ from that of $Q^{(m_i)}$. The modified Gauss elementary matrices $\hat{M}_{m_i+1}, \hat{M}_{m_i+2}, \ldots, \hat{M}_{m_{i+1}}$, involved in the sequence of transformations

$$\begin{bmatrix} I_{m_i} & 0 & 0 \\ 0 & J_{m_{i+1}-m_i} & 0 \\ 0 & 0 & I_{n-m_{i+1}} \end{bmatrix} \hat{B}^{(m_i)} \to \hat{B}^{(m_i+1)} \to \ldots \to \hat{B}^{(m_{i+1})},$$

are given by

$$\hat{M}_{m_i+1} = \begin{bmatrix} I_{m_i+1} & 0 \\ 0 & \alpha_{m_i+1} I_{n-m_i-1} \end{bmatrix} - \begin{bmatrix} \mathbf{0} \\ \hat{b}^{(m_i)}_{m_{i+1}+1,m_i+1} \\ \vdots \\ \hat{b}^{(m_i)}_{n,m_i+1} \end{bmatrix} \frac{\mathbf{e}^T_{m_i+1}}{\hat{b}^{(m_i-1)}_{m_i,m_i}},$$

where

$$\alpha_{m_i+1} = \frac{\hat{b}^{(m_i)}_{m_{i+1},m_i+1}}{\hat{b}^{(m_i-1)}_{m_i,m_i}}, \qquad \mathbf{0} \in \mathbf{Z}^{m_{i+1}},$$

and, for $2 \leq j \leq m_{i+1} - m_i$, by

$$\hat{M}_{m_i+j} = \begin{bmatrix} I_{m_i+j} & 0 \\ 0 & \alpha_{m_i+j} I_{n-m_i-j} \end{bmatrix} - \begin{bmatrix} \mathbf{0} \\ \hat{b}^{(m_i+j-1)}_{m_{i+1}+1,m_i+j} \\ \vdots \\ \hat{b}^{(m_i+j-1)}_{n,m_i+j} \end{bmatrix} \frac{\mathbf{e}^T_{m_i+j}}{\hat{b}^{(m_i+j-2)}_{m_i+j-1,m_i+j-1}},$$

with

$$\alpha_{m_i+j} = \frac{\hat{b}^{(m_i+j-1)}_{m_i+j,m_i+j}}{\hat{b}^{(m_i+j-2)}_{m_i+j-1,m_i+j-1}}, \qquad \mathbf{0} \in \mathbf{Z}^{m_{i+1}}, .$$

If we set

$$\hat{E}_{m_{i+1}} = M_{m_{i+1}} \cdots \hat{M}_{m_i+1} \begin{bmatrix} I_{m_i} & 0 & 0 \\ 0 & J_{m_{i+1}-m_i} & 0 \\ 0 & 0 & I_{n-m_{i+1}} \end{bmatrix} \hat{E}_{m_i}$$

then we find that

$$\hat{E}_{m_{i+1}} \hat{B} = \hat{B}^{(m_i+1)}.$$

Moreover, the matrix $\hat{E}_{m_{i+1}}$ has the same structure as $\hat{E}_{m_i}$, and the diagonal entries β_{m_i+j}, $1 \leq j \leq m_{i+1} - 1$, in position $m_i + j$ of the matrix $\hat{E}_{m_{i+1}}$, are such that

$$\beta_{m_i+j} = (\hat{E}_{m_i+1})_{m_i+j,m_i+j} = \alpha^j_{m_i+1} \hat{b}^{(m_i-1)}_{m_i,m_i} \in \mathbf{Z}. \tag{3.5}$$

From the structure of each Gauss transformation, we also deduce the following key property: *the computation of the first row of* $Q^{(m_{i+1})}$ *can be carried out by determining both the first and the* $(m_{i+1} - m_i - j + 1)$*-st row of the matrix* $Q^{(k+j)}$, $1 \leq j \leq m_{i+1} - m_i$. Now, the first row of $Q^{(k+j)}$ is given by the $(j+1)$-st row of $\hat{Q}^{(k)}$ multiplied by

$$\alpha^j_{m_i+1} = \left(\frac{\hat{b}^{(m_i)}_{m_{i+1},m_i+1}}{\hat{b}^{(m_i-1)}_{m_i,m_i}} \right)^j. \tag{3.6}$$

Hence, the computation of the first row of $Q^{(k+j)}$ can be efficiently performed over the integers in two steps by means of the following

rule 1:
firstly, multiply the $(j+1)$-st row of $\hat{Q}^{(k)}$ by β_{m_i+j} and, secondly, divide the result by $\hat{b}^{(m_i-1)}_{m_i,m_i}$.

For the evaluation of the entries $\hat{b}^{(m_i+j)}_{m_{i+1}+1,h}$, $m_i + j \leq h \leq n$, in the $(m_{i+1} - m_i - j + 1)$-st row of the matrix $Q^{(m_i+j)}$, the following formulae can be used:

rule 2:
for $j = 1$, compute

$$\hat{b}^{(m_i+j)}_{m_{i+1}+1,h} = \frac{\hat{b}^{(m_i)}_{m_{i+1},m_i+1}\hat{b}^{(m_i)}_{m_{i+1}+1,h} - \hat{b}^{(m_i)}_{m_{i+1}+1,m_i+1}\hat{b}^{(m_i)}_{m_{i+1},h}}{\hat{b}^{(m_i-1)}_{m_i,m_i}};$$

otherwise, for $j = 2, \ldots, m_{i+1} - m_i$, compute

$$\hat{b}^{(m_i+j)}_{m_{i+1}+1,h} = \frac{\hat{b}^{(m_i+j-1)}_{m_i+j,m_i+j}\hat{b}^{(m_i+j-1)}_{m_{i+1}+1,h} - \hat{b}^{(m_i+j-1)}_{m_{i+1}+1,m_i+j}\hat{b}^{(m_i+j-1)}_{m_i+j,h}}{\hat{b}^{(m_i+j-2)}_{m_i+j-1,m_i+j-1}}.$$

These formulae can be proved by observing that the entries $\hat{b}^{(m_i+j-1)}_{m_i+j,h}$, $m_i + j \leq h \leq n$, are located on the first row of $Q^{(m_i+j)}$.

In this way, we finally obtain the following fraction-free algorithm that transforms a permuted Bezoutian $\hat{B}$ into its upper triangular form by using a sort of structured Gaussian elimination.

Algorithm. 1 *Fraction-free triangularization of Bezoutians.*

Input: the entries of (the first $n - m$ rows of) the permuted Bezoutian $\hat{B} = \hat{B}^{(0)} = JB(u,v)J \in \mathbf{Z}^{n \times n}$, where $u(x)$ and $v(x)$ are two polynomials with integer coefficients of degree n and m, $n \geq m$, respectively. Alternatively, the coefficients of the polynomials $u(x)$ and $v(x)$.

Output: the integer entries of an upper triangular matrix $\hat{B}^{(n-1)} \in \mathbf{Z}^{n \times n}$ obtained from $\hat{B}$ by row operations only.

- By means of a direct inspection of the first row of $\hat{B}^{(0)}$, find the integer coefficients of $f_1(x) = u_n v(x)$ as well as its degree m. Set $m_0 = 0$ and $m_1 = n - m$. Initialize $\hat{b}^{(m_0-1)}_{m_0,m_0} = 1$, $i = 0$.

- While $m_{i+1} < n$ do:

1. Let the $(m_i + 1)$-st row of $\hat{B}^{(n-1)}$ be equal to the first row of $\hat{Q}^{(m_i)}$ (compare (3.2)). If $m_{i+1} \geq m_i + 2$, then evaluate the rows $m_i + 2$ through m_{i+1} of $\hat{B}^{(n-1)}$ by recursively applying *rule 1* for the computation of the first row of $Q^{(m_i+j)}$ with $1 \leq j \leq m_{i+1} - 1$.

2. Determine the first row of $Q^{(m_{i+1})}$ by means of *rule 2*. Let $f_{i+2}(x) \in \mathbf{Z}[x]$ be the polynomial whose coefficients are given by the entries of such row.

3. If $f_{i+2}(x) = 0$ or $\deg(f_{i+2}(x)) = 0$, then set $m_{i+2} = n$. The rows $m_{i+1} + 1$ through m_{i+2} of $\hat{B}^{(n-1)}$ are defined by those of $\hat{Q}^{(m_{i+1})}$.

4. Otherwise, if $\deg(f_{i+2}(x)) > 0$, then set $m_{i+2} = n - \deg(f_{i+2}(x))$ and denote as $\mathrm{lc}(f_{i+2}(x)) \neq 0$ the leading coefficient of $f_{i+2}(x)$. Construct the row 1 through $m_{i+2} - m_{i+1} + 1$ of

$$Q^{(m_{i+1})} = \frac{1}{\mathrm{lc}(f_{i+2}(x))} J_{n-m_{i+1}} B(f_{i+1}, f_{i+2}) J_{n-m_{i+1}},$$

by using proposition 3.1.

5. Set $i = i + 1$.

Concerning the cost analysis of this algorithm, we first observe that the computation of the rows $m_i + 1$ through m_{i+1} of $\hat{B}^{(n-1)}$ requires $O((m_{i+1} - m_i)n)$ arithmetic operations. This yields the overall arithmetic cost of $O(n^2)$ operations and, therefore, our algorithm is asymptotically optimal in the arithmetic model of computation. Moreover, if c is an upper bound on the magnitude of the entries of B, then the length of the integers involved in the computation is $O(n(\log nc))$ bits. Hence, the overall Boolean cost is $O(n^2\nu(n \log nc))$, where $\nu(m)$ denotes the Boolean cost of multiplication of m-bit integers.

4. SUB-RESULTANT CHAIN COMPUTATION

So far we have seen how a permuted Bezoutian $JB(u,v)J$ can efficiently be reduced into upper triangular form by means of a fraction-free elimination scheme. In this section we make use of this result in order to develop a new algorithm for the computation of a polynomial remainder sequence, generated by $u(x)$ and by $v(x)$, over the integral domain $\mathbf{Z}$.

A sequence of polynomials $\{f_i(x)\}_{i=0}^{L}$ is a polynomial remainder sequence (p.r.s.) of $f_0(x) \neq 0$ and $f_1(x) \neq 0$ over the integral domain $\mathbf{D}$ if $L \geq 1$, $f_i(x) \in \mathbf{D}[x]$ and

$$\rho_{i-1} f_{i-1}(x) = t_i(x) f_i(x) - \gamma_{i-1} f_{i+1}(x), \quad 1 \leq i \leq L, \tag{4.1}$$

where

$$t_i(x) \in \mathbf{D}[x],\ \deg(f_i(x)) = \eta_i > \eta_{i+1},\ \rho_i, \gamma_i \in \mathbf{D} \setminus \{0\},\ f_{L+1}(x) = 0.$$

If $f_0(x),\ f_1(x) \in \mathbf{Z}[x]$, then, setting $\rho_{i-1} = (\mathrm{lc}(f_i(x)))^{\eta_{i-1} - \eta_i + 1}$ and $\gamma_{i+1} = -1$, where $\mathrm{lc}(f(x))$ denotes the leading coefficient of $f(x)$, we define the process of pseudo-divisions (4.1), which always yields a pseudo-quotient $t_i(x)$ and a pseudo-remainder $f_{i+1}(x)$ with integer coefficients. It is well known that the sequence generated in this way may suffer from an exponential growth of the length of the coefficients. This may make the algorithm unusable for the large Boolean cost.

Throughout this section we will assume that $f_0(x) = u(x) \in \mathbf{Z}[x]$ and $f_1(x) = v(x) \in \mathbf{Z}[x]$, that is, we are considering p.r.s. of $u(x)$ and $v(x)$. By choosing $\rho_i = \gamma_i = 1$, we obtain the Euclidean scheme (1.2) for the computation of a p.r.s. $\{r_i(x)\}_{i=0}^{L}$ over the quotient field $\mathbf{D} = \mathbf{Q}$. Other different selections of the rational values of ρ_i and γ_i, give polynomial remainder sequences $u(x), v(x), f_2(x), \ldots, f_L(x)$ in which the previous one can be embedded up to similarity. Specifically, let $f(x),\ g(x) \in \mathbf{D}[x]$, $f(x)$ and $g(x)$ are called similar ($f(x) \sim g(x)$) iff there exist $a,\ b \in \mathbf{D} \setminus \{0\}$ such that $af(x) = bg(x)$. As a special case $f(x)$ and $g(x)$ are associates ($f(x) \simeq g(x)$) iff they are similar and the coefficients of similarity are units, i.e., elements of $\mathbf{D}$ with a multiplicative inverse in $\mathbf{D}$. Then, it can be easily seen that $r_i(x) \sim f_i(x)$ and, therefore, $f_L(x)$ is a scalar multiple of the gcd of $u(x)$ and $v(x)$ computed over $\mathbf{Q}$.

The problem we are mainly interested is the fast computation of a p.r.s. of $u(x)$ and $v(x)$ over the integral domain $\mathbf{D} = \mathbf{Z}$. The resulting Boolean cost clearly depends on the coefficient growth which can be kept down by removing greatest common divisors. However, gcd computations are generally expensive and so they should be avoided. An alternative approach which does not require gcd computations consists of performing pseudo-divisions whereby factors known to exist are systematically removed from the coefficients of the pseudo-remainders.

The fundamental theorem of sub-resultants (see [7], [9] and [24]) relates by similarity a p.r.s. of $u(x)$ and $v(x)$ with the sub-resultant chain of $u(x)$ and $v(x)$ defined as follows. The $(n+m) \times (n+m)$ resultant or Sylvester matrix of $u(x)$ and $v(x)$ is given by

$$\mathcal{S}_0 = \begin{bmatrix} u_n & & \ldots & & u_0 & & \\ & \ddots & & & & \ddots & \\ & & u_n & & \ldots & & u_0 \\ v_m & & \ldots & v_0 & & & \\ & \ddots & & & \ddots & & \\ & & \ddots & & & \ddots & \\ & & & v_m & \ldots & \ldots & v_0 \end{bmatrix} = \begin{bmatrix} U \\ V \end{bmatrix}, \tag{4.2}$$

where $U \in \mathbf{Z}^{m \times (n+m)}$ and $V \in \mathbf{Z}^{n \times (n+m)}$. For k, $0 \leq k < m$, let $\mathcal{S}_{i,k}$ be the submatrix of $\mathcal{S}_0$ obtained by deleting the last k of the m rows of U, the last k of the n rows of V, and the last $2k+1$ columns excepting column

$m+n-k-i$. The polynomial $s_k(x) \in \mathbf{Z}[x]$ defined by

$$s_k(x) = \sum_{i=0}^{k} \det(\mathcal{S}_{i,k}) x^i,$$

is called the k-th sub-resultant of $u(x)$ and $v(x)$.

Consider the sub-resultant chain $u(x)$, $v(x)$, $s_{m-1}(x)$, ..., $s_0(x)$. The fundamental theorem of sub-resultants establishes for any p.r.s. $\{f_i(x)\}_{i=0}^{L}$ of $u(x)$ and $v(x)$ over $\mathbf{Q}$ with $\deg(f_i(x)) = \eta_i$ and $c_i = \mathrm{lc}(f_i(x))$, that

$$f_i(x) \sim s_{\eta_i}(x) \sim s_{\eta_{i-1}-1}(x), \quad 2 \le i \le L \tag{4.3}$$

and

$$s_k(x) = 0, \quad 0 \le k \le \eta_L \text{ or } \eta_i < k < \eta_{i-1} - 1, \quad 2 \le i \le L.$$

Moreover, the coefficients of similarity in (4.3) are given explicitly in terms of ρ_i and γ_i of (4.1). Specifically, for $2 \le j \le L$, we have [24],

$$\{\prod_{i=0}^{j-2} \rho_i^{\eta_{i+1}-\eta_j}\} s_{\eta_j}(x) = \{\prod_{i=0}^{j-2} (-1)^{\eta_i-\eta_j} \gamma_i^{\eta_{i+1}-\eta_j} c_{i+1}^{\eta_i-\eta_{i+2}}\} \rho_j^{\eta_{j-1}-\eta_j-1} f_j(x),$$

and

$$\{\prod_{i=0}^{j-2} \rho_i^{\eta_{i+1}-\eta_{j-1}+1}\} s_{\eta_{j-1}-1}(x) = \{\prod_{i=0}^{j-2} (-1)^{\eta_i-\eta_{j-1}+1} \gamma_i^{\eta_{i+1}-\eta_{j-1}+1} c_{i+1}^{\eta_i-\eta_{i+2}}\} \times c_{j-1}^{-\eta_{j-1}+\eta_j+1} f_j(x).$$

By applying these results to the Euclidean p.r.s. (1.2), we easily find that

$$s_{\eta_j}(x) = \{\prod_{i=0}^{j-2} (-1)^{\eta_i-\eta_j} (\mathrm{lc}(r_{i+1}(x))^{\eta_i-\eta_{i+2}}\} r_j(x), \tag{4.4}$$

and

$$s_{\eta_{j-1}-1}(x) = \{\prod_{i=0}^{j-2} (-1)^{\eta_i-\eta_{j-1}+1} (\mathrm{lc}(r_{i+1}(x))^{\eta_i-\eta_{i+2}}\} \times (\mathrm{lc}(r_{j-1}(x))^{-\eta_{j-1}+\eta_j+1} r_j(x), \tag{4.5}$$

where

$$\eta_i = n - m_i, \quad 0 \le i \le L. \tag{4.6}$$

Similar relations also hold for the polynomial sequence provided by algorithm 1 for reducing $JB(u,v)J$ into an upper triangular form.

From now on $\{f_i(x)\}_{i=0}^{L}$ is the polynomial sequence where $f_0(x) = u(x)$, $f_1(x) = v(x)$ and $f_i(x) \in \mathbf{Z}[x]$, $2 \le i \le L$, is the polynomial of degree $n - m_i$ whose coefficients are determined by the entries of the first row of $Q^{(m_{i-1})}$ (compare with step 2 of the algorithm 1). In the view of (3.3), we have

$$f_{i+1}(x) = \hat{b}_{m_i,m_i}^{(m_i-1)} \mathrm{lc}(r_i(x)) r_{i+1}(x) \tag{4.7}$$

Moreover, by using (3.3), (3.6) and (3.5), we obtain that

$$Q^{(m_i+1)} = \left(\frac{\hat{b}_{m_{i+1},m_i+1}^{(m_i)}}{\hat{b}_{m_i,m_i}^{(m_i+1)}}\right)^{m_{i-1}-m_i} \hat{b}_{m_i,m_i}^{(m_i-1)} S_{m_{i+1}} = \hat{b}_{m_{i+1},m_{i+1}}^{(m_{i+1}-1)} S_{m_{i+1}},$$

which implies

$$\hat{b}_{m_{i+1},m_{i+1}}^{(m_{i+1}-1)} = (\mathrm{lc}(r_i(x))\mathrm{lc}(r_{i+1}(x)))^{m_{i+1}-m_i} \hat{b}_{m_i,m_i}^{(m_i-1)}.$$

By induction, we then find that

$$\hat{b}_{m_i,m_i}^{(m_i-1)} = u_n^{n-m} \{\prod_{j=0}^{i-2} (\mathrm{lc}(r_{j+1}(x)))^{m_{j+2}-m_j}\} (\mathrm{lc}(r_i(x)))^{m_i-m_{i-1}}.$$

Replacing this expression into relation (4.7), this results in

$$\frac{f_{i+1}(x)}{u_n^{n-m}} = \{\prod_{j=0}^{i-2} (\mathrm{lc}(r_{j+1}(x)))^{m_{j+2}-m_j}\} (\mathrm{lc}(r_i(x)))^{m_i-m_{i-1}+1} r_{i+1}(x), \tag{4.8}$$

for $1 \le i \le L-1$. Putting (4.5), (4.6) and (4.8) together, we finally arrive at the next proposition.

Proposition 4.1. *For the polynomial sequence* $\{f_i(x)\}_{i=0}^{L}$, $f_0(x) = u(x)$, $f_1(x) = v(x)$, *generated by algorithm 1 we have*

$$f_{i+1}(x) \simeq u_n^{n-m} s_{\eta_i-1}(x), \quad 1 \le i \le L-1.$$

In conclusion, our fraction-free scheme for Bezoutian triangularization yields a fast method to compute the sub-resultant chain of $u(x)$ and $v(x)$ over the integral domain $\mathbf{Z}$. The overall cost is $O(n^2\nu(n \log nc))$ Boolean operations, where c here denotes an upper bound on the magnitude of the coefficients of the input polynomials. For comparison, the sub-resultant p.r.s. method by A. Akritas [1], based on the triangularization of the resultant matrix, requires $O(n^3\nu(n \log nc))$ Boolean operations since it is based on an unstructured version of Gaussian elimination. Differently, the sub-resultant method of [8] uses a fast elimination procedure but, however, a quadratic coefficient growth can occur so slowing down the overall Boolean cost. The same criticism applies to the algorithms of [15].

We have seen that the Bareiss variant of Gaussian elimination can be applied to factoring a permuted Bezoutian keeping unchanged the structural properties of the Schur complements. This provides a fast division-free algorithm for computing a polynomial remainder sequence of two polynomials with c-bit integer coefficients within the Boolean complexity bound $O(n^2\nu(n \log nc))$.

The same result trivially holds for polynomials having coefficients over a general integral domain, say $\mathbf{Z}[\mathbf{y}]$, for $\mathbf{y} = (y_1, \ldots, y_q)$ being a set of indeterminates.

We observe that, due to the permutation matrices $J_{m_{i+1}-m_i}$ used at the general stage of the algorithm, the sequence of polynomials $f_i(x)$ that is obtained in this way is not a Sturm sequence. In other words, because of the permutations, the matrix factorization cannot be transformed into an LDL^T one.

In order to obtain a Sturm sequence we have to change the sign of $f_i(x)$ according to the values of $m_{i+1} - m_i$. In fact, $\det J_k = -1$ for $k = 2 \bmod 4$ or $k = 3 \bmod 4$, while $\det J_k = 1$ otherwise. Differently, by means of a slight change in the permutation strategy, it is possible to obtain right away a Sturm sequence without taking care about the sign of the permutations. The idea relies on applying a suitable permutation both to the rows and to the columns of JBJ so that the Schur complement obtained in this way is structurally unchanged. This strategy also leads to a method for computing the LDL^T factorization of a permuted Bezoutian that is a useful tool in the solution of root-localization problems [22]. More precisely, without loss of generality assume that $q = m_{i+1} - m_i > 1$ and consider the $q \times q$ leading principal submatrix $Q_q^{(m_i)}$ of $Q^{(m_i)}$ which is a lower triangular (with respect to the anti-diagonal) Hankel matrix. Unlike algorithm 1, where we apply the row permutation J_q to the matrix $Q_q^{(m_i)}$ in order to find the triangular matrix $J_q Q_q^{(m_i)}$, here we apply the circulant permutation

$$C_q = \begin{bmatrix} 0 & \ldots & 0 & 1 \\ 1 & 0 & \ddots & 0 \\ & \ddots & \ddots & \vdots \\ 0 & & 1 & 0 \end{bmatrix}$$

both to the columns and to the rows of $Q_q^{(m_i)}$, that is, we consider the matrix $C_q Q_q^{(m_i)} C_q^T$ having the structure

$$C_q Q_q^{(m_i)} C_q^T = \left[\begin{array}{ccccc} a_q & a_1 & a_2 & \dots & a_{q-1} \\ a_1 & 0 & 0 & \dots & 0 \\ a_2 & 0 & 0 & \dots & a_1 \\ \vdots & \vdots & \vdots & .\cdot\cdot & \vdots \\ a_{q-1} & 0 & a_1 & \dots & a_{q-2} \end{array} \right],$$

where the $(q-2) \times (q-2)$ trailing principal submatrix is a lower triangular Hankel matrix.

The 2×2 leading principal submatrix of the above matrix is nonsingular, therefore it is possible to apply one step of 2×2 block Gaussian elimination in the Bareiss version [2]. It is immediate to verify that the Schur complement obtained in this way coincides with the $(q-2) \times (q-2)$ trailing principal submatrix and no arithmetic operation is needed in this computation. The permutation policy, together with the Schur complementation by means of block Gaussian elimination in the Bareiss version, can be recursively applied to the whole matrix $Q^{(m_i)}$. In this way we arrive at the LDL^T factorization of a permuted Bezoutian, where the same permutation acts both on the rows and on the columns. Bounds to the growth of the lengths of the intermediate integers generated by this scheme can be given similarly to the ones of section 3.

An interesting application of the LU factorization of the permuted Bezoutian comes from the following formula (see [21]) that relates Bezoutians with the resultant or Sylvester matrix $\mathcal{S}_0$ of (4.2)

$$\mathcal{S}_0 = \left[\begin{array}{cc} J & O \\ -W(v) & W(u) \end{array} \right]^{-1} \left[\begin{array}{cc} I & O \\ O & B(u,v) \end{array} \right] \left[\begin{array}{cc} \hat{W}(u) & W(u) \\ I & O \end{array} \right]$$

where

$$W(u) = \left[\begin{array}{ccc} u_1 & \dots & u_n \\ \vdots & .\cdot\cdot & \\ u_n & & O \end{array} \right], \quad \hat{W}(u) = \left[\begin{array}{ccc} O & & u_0 \\ & .\cdot\cdot & \vdots \\ u_0 & \dots & u_{n-1} \end{array} \right].$$

Given the LU decomposition of a permuted Bezoutian over the integers, the above formula enable us to solve a Sylvester system in $O(n^2)$ arithmetic operations by avoiding explicit divisions within the overall Boolean cost $O(n^2\nu(n \log nc))$. This fact finds applications in the solution of polynomial congruences like $u(x)w(x)+v(x)z(x) = r(x)$ that can be reduced to solving Sylvester systems.

Another extension of our algorithm concerns computations with power series remainder sequences and Padé fractions. Let us recall the following definitions from [8].

Definition 5.1. Let s be a non-negative integer. The power series $q(x)$, $R(x)$, where $R(0) \neq 0$ or $R(x) = 0$ are defined to be a power series pseudo-quotient and a pseudo-remainder, on power series pseudo-division of $u(x)$ and $v(x)$ with respect to s, if

$$\alpha u(x) + q(x)v(x) = x^{s+t}\beta R(x) \tag{5.1}$$

for some integer $t > 0$ and some non-zero $\alpha, \beta \in \mathbf{Z}$.

Definition 5.2. Given power series $u(x)$, $v(x)$, and a nonnegative integer s, a power series remainder sequence is the sequence $\{s_i, R_i(x)\}_{i=-1,0,\ldots}$, where $s_{-1} = 1$, $s_0 = s$, $R_{-1}(x) = u(x)$, $R_0(x) = v(x)$, and

$$\alpha_{i+1}R_{i-1}(x) + q_{i+1}(x)R_i(x) = x^{s_i+s_{i+1}}\beta_{i+1}R_{i+1}(x), \quad i = 0, \ldots \tag{5.2}$$

where $q_i(x)$ and $R_{i+1}(x)$ and s_{i+1} are respectively a power series pseudo-quotient, a pseudo-remainder and the scaling factor on pseudo-division of $R_{i-1}(x)$ by $R_i(x)$ with respect to s_i.

Let us first point out the following:

Remark 5.1. From the definition of Bezoutian it holds $JB(u,v)J = B(\tilde{u},\tilde{v})$, where $\tilde{u}(x) = x^n u(x^{-1})$, $\tilde{v}(x) = x^n v(x^{-1})$. Hence, in the light of proposition 1.1 the block LU factorization of the unpermuted Bezoutian $B(u,v)$ corresponds to applying the Euclidean scheme to $\tilde{u}(x)$ and $\tilde{v}(x)$.

Given two power series $u(x) = \sum_{i=0}^{+\infty} u_i x^i$, $u_0 \neq 0$, $v(x) = \sum_{i=0}^{+\infty} v_i x^i$, we may define the Bezoutian associated with $u(x)$ and $v(x)$ as the infinite matrix $B = (b_{i,j})_{i,j=1,2,\ldots}$ such that

$$\frac{u(x)v(y) - u(y)v(x)}{x-y} = \sum_{i,j=1}^{+\infty} b_{i,j}x^{i-1}y^{j-1},$$

Moreover, the matrix equation (1.1) can be easily extended to the case of power series, where the four matrices involved are infinite, i.e.,

$$\begin{aligned} B(u,v) = &\begin{bmatrix} u_1 & u_2 & \cdots \\ u_2 & \iddots & \iddots \\ \vdots & \iddots & \iddots \end{bmatrix} \begin{bmatrix} v_0 & v_1 & \cdots \\ & v_0 & \ddots \\ & \bigcirc & \ddots \end{bmatrix} \\ - &\begin{bmatrix} v_1 & v_2 & \cdots \\ v_2 & \iddots & \iddots \\ \vdots & \iddots & \iddots \end{bmatrix} \begin{bmatrix} u_0 & u_1 & \cdots \\ & u_0 & \ddots \\ & \bigcirc & \ddots \end{bmatrix}, \end{aligned} \tag{5.3}$$

Observe also that from the bilinearity of the Bezoutian $B(u,v)$ associated with the power series $u(x)$ and $v(x)$, we have $B(u,v) = B(u,\tilde{v})$, where $\tilde{v}(x) = v(x) - \frac{v_0}{u_0}u(x) = \sum_{i=0}^{+\infty} \tilde{v}_i x^i$, is such that $\tilde{v}_0 = 0$. Thus, without loss of generality we may assume that $v_0 = 0$.

The following proposition can be easily obtained by rephrasing in terms of power series the proof of the analogous result given in [4] in terms of polynomials, in the light of remark 5.1.

PROPOSITION 5.1. *Consider the power series $u(x)$, $v(x)$ such that $u_0 \neq 0$, $v_0 = \ldots = v_{s_0-1} = 0$, $v_{s_0} \neq 0$. Then the leading principal submatrix $(B(u,v))_{s_0}$ of $B(u,v)$ is nonsingular and the Schur complement S of $(B(u,v))_{s_0}$ in $B(u,v)$ is such that $S = S(x^{-s_0}v, w)$ where $w(x)$ is the pseudo-remainder on power series pseudo-division of $u(x)$ and $v(x)$ with respect to s_0.*

The recursive application of the above proposition leads to the following result.

PROPOSITION 5.2. *Let $u(x)$ and $v(x)$ be the power series defined in proposition 5.1 and $\{s_i, R_i(x)\}_{i=-1,0,\ldots}$ the power series remainder sequence of definition 2, where $s_{-1} = 1$. Then, the $k \times k$ leading principal submatrix of $B = B(u,v)$ is non-singular if and only if there exists an integer $i \geq 0$ such that $k = s_0 + \ldots + s_i$. Moreover, the Schur complement S_k of B_k in B satisfies*

$$S_k = B(x^{-s_i}R_i, R_{i+1}).$$

Thus, from the above result, it follows that the LU (block) factorization of $B(u,v)$ corresponds to compute the power series remainder sequence $\{s_i, R_i(x)\}_{i=-1,0,\ldots}$, of $u(x)$ and $v(x)$, and our algorithm of section 3 can be used to compute, for any n and m, the first n coefficients of R_i, $i = 1, \ldots, m$.

REFERENCES

1 A.G. Akritas, *A new method for computing greatest common divisor and polynomial remainder sequences, Numer. Math.,* **52** (1988), pp.119-127.

2 E. Bareiss, *Sylvester's identity and multistep integer-preserving Gaussian elimination,* Math. Comp., **22** (1968), pp. 565-578.

3 E. Bareiss, *Computational solution of matrix problems over an integral domain,* J. Inst. Maths Applcs., **10** (1972), pp. 68-104.

4 D. Bini, L. Gemignani, *Fast parallel computation of the polynomial remainder sequence via Bezout and Hankel matrices,* SIAM J. Comput., **24** (1995), pp. 63-77.

5 D. Bini, L. Gemignani, *Erratum: Fast parallel computation of the polynomial remainder sequence via Bezout and Hankel matrices*, SIAM J. Comput., **25** (1996), p. 1358.

6 D. Bini, V. Pan, *Polynomial and matrix computations, volume 1: Fundamental algorithms,* Birkhäuser, Boston, MA, 1994.

7 W. S. Brown, *On Euclid's algorithm and the computation of polynomial greatest common divisors*, J. ACM, **18** (1971), pp. 478-504.

8 S. Cabay, P. Kossowski, *Power series remainder sequences and Padé fractions over an integral domain*, J. Symbolic Computation, **10** (1990), pp. 139-163.

9 G.E. Collins, *Subresultants and reduced polynomial remainder sequences, J. Assoc. Comput. Mach.,* **14** (1967), pp. 128-142.

10 P. A. Fuhrmann, *A polynomial approach to linear algebra,* Springer-Verlag, New York, NY, 1996.

11 L. Gemignani, *Fast inversion of Hankel and Toeplitz matrices*, Information Processing Letters, **41** (1992), pp. 119-123.

12 L. Gemignani, *Solving Hankel systems over the integers*, J. Symbolic Computation, **18** (1994), pp. 573-584.

13 L. Gemignani, *Computationally efficient applications of the Euclidean algorithm to zero location*, Linear Algebra Appl., **249** (1996), pp. 79-91.

14 L. Gemignani, *Schur complements of Bezoutians and the inversion of block Hankel and Toeplitz matrices*, Linear Algebra Appl., **253** (1997), pp. 39-59.

15 L. Gemignani, *GCD of polynomials and Bezout matrices*, TR 97-03, Computer Science Dep., University of Pisa. To appear on the proceedings of ISSAC97.

16 W. B. Gragg, A. Lindquist, *On the partial realization problem*, Linear Algebra Appl., **50** (1983), pp. 277-319.

17 G. Heinig, K. Rost, *Algebraic methods for Toeplitz-like matrices and operators,* Birkhäuser, Boston, MA, (1984).

18 U. Helmke, P. A. Fuhrmann, *Bezoutians*, Linear Algebra Appl., **122** (1989), pp. 1039-1097.

19 I. S. Iohvidov, *Hankel and Toeplitz Matrices and Forms*, Birkhäuser, Boston, MA, (1984).

20 T. Kailath, *Linear systems*, Prentice-Hall Inc., Englewood Cliffs, N.J., 1980.

21 P. Lancaster, M. Tismenetsky, *The Theory of Matrices, Second Edition, with Applications* Academic Press, Orlando 1985.

22 H. Lev-Ari, Y. Bistritz, T. Kailath, *Generalized Bezoutians and families of efficient zero-location procedures*, IEEE Transactions on Circuits and Systems, **38** (1991), pp. 170-186.

23 T. Lickteig, M. F. Roy, *Cauchy index computation*, CALCOLO, to appear.

24 R. Loos, *Generalized polynomial remainder sequences*, Computer algebra symbolic and algebraic computations, B. Buchberger, G . E. Collins, R. Loos eds., Springer (1982), pp. 115-137.

25 C. C. MacDuffee, *Some applications of matrices in the theory of equations,* Am. Math. Mon., **57** (1930), pp. 154-161.

26 R. T. Moenck, *Practical fast polynomial multiplication*, in Proc. SYMSAC'76, ed. R. D. Jenks, ACM Press (1976), pp. 166- 176.

27 M. F. Roy, *Basic algorithms in real algebraic geometry and their complexity: from Sturm's theorem to the existential theory of reals,* Lectures on Real Geometry in memoriam of Mario Raimondo, de Gruyter Expositions in Mathematics, to appear.

28 J. R. Sendra, J. Llovet, *GCD of polynomials and Hankel matrices*, J. Symbolic Computation, **13** (1992), pp. 25-39.

29 A. Schönage, V. Strassen, *Schnelle multiplikation grosse Zahlen*, Computing, **7** (1971), pp. 281-292.