Article:

Bourne, M., Winkler, J.R. and Yi, S. (2017) The computation of the degree of an approximate greatest common divisor of two Bernstein polynomials. Applied Numerical Mathematics, 111. pp. 17-35. ISSN 0168-9274

https://doi.org/10.1016/j.apnum.2016.08.005

The computation of the degree of an approximate greatest common divisor of two Bernstein polynomials

Martin Bourne[a,1], Joab R. Winkler[a,*], Su Yi[b]

[a] *The University of Sheffield, Department of Computer Science, Regent Court, 211 Portobello, Sheffield S1 4DP, United Kingdom*
[b] *Institute of High Performance Computing, 1 Fusionopolis Way, 16-16 Connexis North, Singapore 138632*

Abstract

This paper considers the computation of the degree t of an approximate greatest common divisor $d(y)$ of two Bernstein polynomials $f(y)$ and $g(y)$, which are of degrees m and n respectively. The value of t is computed from the QR decomposition of the Sylvester resultant matrix $S(f,g)$ and its subresultant matrices $S_k(f,g), k = 2, \ldots, \min(m,n)$, where $S_1(f,g) = S(f,g)$. It is shown that the computation of t is significantly more complicated than its equivalent for two power basis polynomials because (a) $S_k(f,g)$ can be written in several forms that differ in the complexity of the computation of their entries, (b) different forms of $S_k(f,g)$ may yield different values of t, and (c) the binomial terms in the entries of $S_k(f,g)$ may cause the ratio of its entry of maximum magnitude to its entry of minimum magnitude to be large, which may lead to numerical problems. It is shown that the QR decomposition and singular value decomposition (SVD) of the Sylvester matrix and its subresultant matrices yield better results than the SVD of the Bézout matrix, and that $f(y)$ and $g(y)$ must be processed before computations are performed on these resultant and subresultant matrices in order to obtain good results.

Keywords: Sylvester resultant matrix, Bernstein polynomials, approximate greatest common divisor

*Corresponding author

[1] Martin Bourne was supported by a studentship from The Agency for Science, Technology and Research, (A*), Singapore, and The University of Sheffield.

2010 MSC: 12Y05, 65F99

1. Introduction

The computation of the greatest common divisor (GCD) of two polynomials occurs in several applications, including image processing, control systems, robotics and the computation of intersections of Bézier curves and surfaces in computer aided geometric design [4]. The GCD is defined for exact polynomials only, but practical problems yield inexact polynomials because their coefficients are corrupted by noise. It is therefore necessary to consider an approximate greatest common divisor (AGCD) of noisy forms $f(y)$ and $g(y)$ of, respectively, the exact polynomials $\hat{f}(y)$ and $\hat{g}(y)$. The GCD of $\hat{f}(y)$ and $\hat{g}(y)$ is unique up to an arbitrary non-zero constant, but an AGCD of $f(y)$ and $g(y)$ is not unique because it can be defined in several ways. Furthermore, each AGCD may be considered to be the GCD of polynomials that lie in neighbourhoods of $f(y)$ and $g(y)$, and these AGCDs may not be unique, apart from scaling.

An AGCD of two polynomials and methods for its computation are discussed in Section 2, and it is shown in Section 3 that the degree of the GCD of $\hat{f}(y)$ and $\hat{g}(y)$, which are of degrees m and n respectively, can be calculated from the rank of their Sylvester matrix $S(\hat{f},\hat{g})$ and its subresultant matrices $S_k(\hat{f},\hat{g})$, $k = 2,\ldots,\min(m,n)$, where $S_1(\hat{f},\hat{g}) = S(\hat{f},\hat{g})$. Consideration of the entries of these matrices shows it is convenient to rearrange them in order to reduce the computational complexity of their evaluation. This rearrangement leads to Section 4, where an equation that allows $S_k(\hat{f},\hat{g})$ to be computed from $S_j(\hat{f},\hat{g})$, $j < k$, is developed.

It is shown in Section 5 that $S_k(\hat{f},\hat{g})$, $k = 1,\ldots,\min(m,n)$, must be processed by three operations before computations are performed on these matrices in order to minimise numerical problems that may arise. Methods for the computation of the degree of an AGCD of $f(y)$ and $g(y)$ are discussed in Section 6, and Section 7 contains examples of this computation. The contents of the paper are summarised in Section 8.

2. An approximate greatest common divisor

The following definition of an AGCD of two polynomials is used by Bini and Boito [2]. It involves concepts of the nearness of two polynomials, the maximum degree of a polynomial from the set of polynomials that satisfy the nearness condition, and a measure of the distance between two polynomials.

Definition 2.1. *Let $f(y)$ and $g(y)$ be polynomials of degrees m and n respectively. A polynomial $d(y)$ is an ϵ-divisor of $f(y)$ and $g(y)$ if there exist polynomials $\tilde{f}(y)$ and $\tilde{g}(y)$, of degrees m and n respectively, such that*

$$\left\| f(y) - \tilde{f}(y) \right\| \leq \epsilon \left\| f(y) \right\| \qquad \text{and} \qquad \left\| g(y) - \tilde{g}(y) \right\| \leq \epsilon \left\| g(y) \right\|,$$

and $d(y)$ divides $\tilde{f}(y)$ and $\tilde{g}(y)$. If $d(y)$ is an ϵ-divisor, of maximum degree, of $f(y)$ and $g(y)$, then it is called an ϵ-GCD, or AGCD, of $f(y)$ and $g(y)$. The polynomials $u(y) = \tilde{f}(y)/d(y)$ and $v(y) = \tilde{g}(y)/d(y)$ are called ϵ-cofactors.

This definition of an AGCD of $f(y)$ and $g(y)$ is a function of ϵ, the maximum value of the upper bound of the relative error between $f(y)$ and $\tilde{f}(y)$, and $g(y)$ and $\tilde{g}(y)$. The value of ϵ may not be known, or it may only be known approximately, in which case this definition of an AGCD may not be appropriate. Another definition, which uses subresultant matrices of the Sylvester matrix of $f(y)$ and $g(y)$, is therefore considered in Section 3.

Previous work on the computation of an AGCD of two power basis polynomials has used the QR decomposition [7, 20] and the singular value decomposition (SVD) [6, 9] of the Sylvester matrix. Also, optimisation methods [5, 21] and methods that exploit the structure of the Sylvester matrix [1, 2, 11, 12, 21] have been used. The methods described in these papers require that the threshold ϵ be specified, and common divisors of degree k, $k = \min(m, n)$, $\min(m, n) - 1$, $\min(m, n) - 2, \ldots, 2, 1$, are computed and an error measure is calculated for each value of k. The procedure terminates at the first (largest) value of k for which the error measure is less than ϵ.

It was noted above that ϵ may not be known in practical problems, or it may only be known approximately. Previous work has shown, however, that if

$f(y)$ and $g(y)$ are preprocessed, then computations on the Sylvester matrix and its subresultant matrices enable the degree t of an AGCD to be computed, even when the value of ϵ, or bounds on its value, are not known [17]. This method for the determination of t has been used for the computation of a structured low rank approximation of the Sylvester matrix [16] and multiple roots of a polynomial [14, 18].

The computation of an AGCD of two Bernstein polynomials is considered in [3, 19], and the work described in this paper extends the work in these two papers. The application of Euclid's algorithm to the computation of the GCD of two Bernstein polynomials is considered in [13], but unsatisfactory results are obtained and the need for robust methods for this computations is emphasized.

3. The Sylvester matrix and the degree of the GCD

This section considers the calculation of the degree of the GCD of $\hat{f}(y)$ and $\hat{g}(y)$, and it is shown that it reduces to the computation of the rank of each matrix $S_k(\hat{f}, \hat{g})$, $k = 1, \ldots, \min(m, n)$. The discussion in this section is brief, and more details are in [19].

If the degree of the GCD of $\hat{f}(y)$ and $\hat{g}(y)$ is $\hat{t}$, then, for each value of $k = 1, \ldots, \hat{t} - 1$, $\hat{f}(y)$ and $\hat{g}(y)$ have more than one common divisor of degree k, and they have only one common divisor, to within an arbitrary non-zero scalar multiplier, of degree $\hat{t}$. It follows that if $\hat{d}_k(y)$ is a common divisor of degree k, there exist quotient polynomials $\hat{u}_k(y)$ and $\hat{v}_k(y)$, which are of degrees $m - k$ and $n - k$ respectively, such that

$$\hat{f}(y) = \hat{u}_k(y)\hat{d}_k(y) \qquad \text{and} \qquad \hat{g}(y) = \hat{v}_k(y)\hat{d}_k(y).$$

It is shown in [19] that these equations can be expressed in matrix form as

$$D_k^{-1} T_k(\hat{f}, \hat{g}) Q_k \begin{bmatrix} \hat{v}_k \\ -\hat{u}_k \end{bmatrix} = 0, \qquad k = 1, \ldots, \min(m, n), \tag{1}$$

where $D_k^{-1} \in \mathbb{R}^{(m+n-k+1)\times(m+n-k+1)}$ is given by

$$D_k^{-1} = \text{diag} \begin{bmatrix} \frac{1}{\binom{m+n-k}{0}} & \frac{1}{\binom{m+n-k}{1}} & \cdots & \frac{1}{\binom{m+n-k}{m+n-k}} \end{bmatrix},$$

$T_k(\hat{f},\hat{g}) \in \mathbb{R}^{(m+n-k+1)\times(m+n-2k+2)}$ is given by

$$T_k(\hat{f},\hat{g}) = \left[\begin{array}{cccccc} \hat{a}_0\binom{m}{0} & & & \hat{b}_0\binom{n}{0} & & \\ \hat{a}_1\binom{m}{1} & \ddots & & \hat{b}_1\binom{n}{1} & \ddots & \\ \vdots & \ddots & \hat{a}_0\binom{m}{0} & \vdots & \ddots & \hat{b}_0\binom{n}{0} \\ \vdots & \ddots & \hat{a}_1\binom{m}{1} & \vdots & \ddots & \hat{b}_1\binom{n}{1} \\ \hat{a}_m\binom{m}{m} & \ddots & \vdots & \hat{b}_n\binom{n}{n} & \ddots & \vdots \\ & \ddots & \vdots & & \ddots & \vdots \\ & & \hat{a}_m\binom{m}{m} & & & \hat{b}_n\binom{n}{n} \end{array}\right], \tag{2}$$

$Q_k \in \mathbb{R}^{(m+n-2k+2)\times(m+n-2k+2)}$ contains the binomial terms of the quotient polynomials $\hat{u}_k(y)$ and $\hat{v}_k(y)$,

$$Q_k = \operatorname{diag}\left[\begin{array}{cccccc} \binom{n-k}{0} & \cdots & \binom{n-k}{n-k} & \binom{m-k}{0} & \cdots & \binom{m-k}{m-k} \end{array}\right], \tag{3}$$

and $\hat{u}_k \in \mathbb{R}^{m-k+1}$ and $\hat{v}_k \in \mathbb{R}^{n-k+1}$ contain the coefficients of $\hat{u}_k(y)$ and $\hat{v}_k(y)$ respectively. The matrix $S_k(\hat{f},\hat{g}) \in \mathbb{R}^{(m+n-k+1)\times(m+n-2k+2)}$ is a modified form of the kth Sylvester subresultant matrix,

$$S_k(\hat{f},\hat{g}) = D_k^{-1}T_k(\hat{f},\hat{g})Q_k, \tag{4}$$

because the standard form of this matrix is $D_k^{-1}T_k(\hat{f},\hat{g})$. It is shown in the sequel that the modified form $S_k(\hat{f},\hat{g})$ of the Sylvester matrix and its subresultant matrices has computational advantages [19].

Equation (1) has a non-zero solution for $k = 1,\ldots,\hat{t}$, and the coefficient matrix $S_k(\hat{f},\hat{g})$ is therefore singular for these values of k. Since $\hat{f}(y)$ and $\hat{g}(y)$ do not possess a common divisor of degree k, $k = \hat{t}+1,\ldots,\min(m,n)$, it follows that

$$\begin{array}{rcll} \text{rank } S_k(\hat{f},\hat{g}) & < & m+n-2k+2, & k = 1,\ldots,\hat{t}, \\ \text{rank } S_k(\hat{f},\hat{g}) & = & m+n-2k+2, & k = \hat{t}+1,\ldots,\min(m,n). \end{array} \tag{5}$$

The value of $\hat{t}$ is therefore equal to the largest integer k such that $S_k(\hat{f},\hat{g})$ is singular, and thus the computation of $\hat{t}$ reduces to the determination of the rank of each matrix $S_k(\hat{f},\hat{g})$.

An AGCD of $f(y)$ and $g(y)$ was defined in Definition 2.1, but it may not be suitable for the solution of practical problems because it is a function of ϵ, which may not be known. Equation (5) allows a definition of the degree of an AGCD of $f(y)$ and $g(y)$ that is independent of ϵ to be given. This definition of the degree of an AGCD is posed in terms of the condition number $\kappa(S_k(f,g))$ of $S_k(f,g)$ because $\kappa(S_k(f,g))$ is large if minor perturbations in $f(y)$ and/or $g(y)$ induce singularity in $S_k(f,g)$, but $\kappa(S_k(f,g))$ is small if $S_k(f,g)$ is distant from singularity. The ratio of this condition number for two successive values of k can be used to calculate the degree of an AGCD of $f(y)$ and $g(y)$ because it allows the change from singularity to non-singularity of $S_k(f,g)$, with respect to a unit change in the value of k, to be determined.[2]

Definition 3.1. *Let $f(y)$ and $g(y)$ be polynomials of degrees m and n respectively, and let $S_k(f,g), k = 1,\ldots,\min(m,n)$, be the kth Sylvester subresultant matrix, where $S_1(f,g) = S(f,g)$. If $\kappa(S_k(f,g))$ is the condition number of $S_k(f,g)$, then the degree t of an AGCD of $f(y)$ and $g(y)$ is equal to the value of k for which the ratio $\kappa(S_k(f,g))/\kappa(S_{k+1}(f,g)), k = 1,\ldots,m+n-2k+1$, attains its maximum value,*

$$t = \arg \max_{k=1,\ldots,m+n-2k+1} \frac{\kappa(S_k(f,g))}{\kappa(S_{k+1}(f,g))}.$$

It follows from (4) that

$$\text{rank } S_k(\hat{f},\hat{g}) = \text{rank } D_k^{-1} T_k(\hat{f},\hat{g}) Q_k, \tag{6}$$

and since D_k^{-1} and Q_k are non-singular, it also follows that

$$\text{rank } S_k(\hat{f},\hat{g}) = \text{rank } T_k(\hat{f},\hat{g}) = \text{rank } D_k^{-1} T_k(\hat{f},\hat{g}) = \text{rank } T_k(\hat{f},\hat{g}) Q_k, \tag{7}$$

and thus the second, third and fourth expressions can, in principle, be used for the determination of the rank of $S_k(\hat{f},\hat{g})$. It may be thought it is easiest to use

[2] The degree and coefficients of an AGCD are defined in Definition 2.1, but only the degree of an AGCD is defined in Definition 3.1. This restriction in Definition 3.1 is justified because this paper is concerned with the computation of the degree of an GCD, and not its coefficients.

$T_k(\hat{f},\hat{g})$ because of the simplicity of the formation of its entries, but it follows from (2) that even if $a_i, b_j = O(1)$, these entries may range over several orders of magnitude because of the binomial terms, and this may cause numerical problems [8, 10]. This consideration also shows that numerical problems may arise when $T_k(\hat{f},\hat{g})Q_k$ is used to determine the degree of the GCD of $\hat{f}(y)$ and $\hat{g}(y)$ because each non-zero entry of this matrix contains the product of two binomial terms.

Similar problems occur when $D_k^{-1}T_k(\hat{f},\hat{g})$ is used, but they manifest themselves slightly differently because its entries contain the binomial terms

$$\frac{\binom{m}{i}}{\binom{m+n-k}{p}}, \qquad i = 0,\ldots,m, \qquad p = i,\ldots,n-k+i,$$

and

$$\frac{\binom{n}{j}}{\binom{m+n-k}{q}}, \qquad j = 0,\ldots,n, \qquad q = j,\ldots,m-k+j,$$

and thus the range of the magnitude of the terms in the denominators is significantly larger than the range of the magnitude of the terms in the numerators, even for moderate values of m and n. It therefore follows that even if $a_i, b_j = O(1)$, the numerical problems stated above may arise when $D_k^{-1}T_k(\hat{f},\hat{g})$ is used to determine the degree of the GCD of $\hat{f}(y)$ and $\hat{g}(y)$. The problems associated with the second, third and fourth expressions in (7) show, therefore, that it is necessary to consider the modified Sylvester matrix and its subresultant matrices, which are defined in (6) and whose binomial terms are

$$\frac{\binom{m}{i}\binom{n-k}{p-i}}{\binom{m+n-k}{p}}, \qquad i = 0,\ldots,m, \qquad p = i,\ldots,n-k+i,$$

and

$$\frac{\binom{n}{j}\binom{m-k}{q-j}}{\binom{m+n-k}{q}}, \qquad j = 0,\ldots,n, \qquad q = j,\ldots,m-k+j.$$

Computational experiments in Section 4 show that $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$ is the best form of the Sylvester matrix and its subresultant matrices, with respect to the complexity of the computation of the entries of the matrix, the condition number and the minimisation of the ratio of the entry of maximum magnitude to the entry of minimum magnitude, to use for the rank test (5). This form appears to be the most complicated of the four forms in (7), but it is shown in the sequel that its advantages with respect to the other three forms are significant.

4. An alternative form for the entries of $S_k(\hat{f}, \hat{g})$

Each entry of $S_k(\hat{f}, \hat{g}) = D_k^{-1} T_k(\hat{f}, \hat{g}) Q_k$ requires the evaluation of three binomial terms, but it is shown in this section it can be rearranged, such that only two binomial terms need be evaluated, which leads to more efficient computations. It is then shown that $S_k(\hat{f}, \hat{g})$ can be obtained from $S_j(\hat{f}, \hat{g})$, $j < k$, by a series of matrix multiplications. This method for the computation of $S_k(\hat{f}, \hat{g})$ from $S_j(\hat{f}, \hat{g})$ differs from the method in [19], where each matrix $S_k(\hat{f}, \hat{g})$ is computed using the matrix multiplications in (4).

4.1. Rearrangement of the entries of $S_k(\hat{f}, \hat{g})$

The matrix $S_k(\hat{f}, \hat{g})$ has the form

$$\left[\begin{array}{cccccc}
\frac{\hat{a}_0 \binom{m}{0}\binom{n-k}{0}}{\binom{m+n-k}{0}} & & & \frac{\hat{b}_0 \binom{n}{0}\binom{m-k}{0}}{\binom{m+n-k}{0}} & & \\
\frac{\hat{a}_1 \binom{m}{1}\binom{n-k}{0}}{\binom{m+n-k}{1}} & \ddots & & \frac{\hat{b}_1 \binom{n}{1}\binom{m-k}{0}}{\binom{m+n-k}{1}} & \ddots & \\
\vdots & \ddots & \frac{\hat{a}_0 \binom{m}{0}\binom{n-k}{n-k}}{\binom{m+n-k}{n-k}} & \vdots & \ddots & \frac{\hat{b}_0 \binom{n}{0}\binom{m-k}{m-k}}{\binom{m+n-k}{m-k}} \\
\vdots & \ddots & \frac{\hat{a}_1 \binom{m}{1}\binom{n-k}{n-k}}{\binom{m+n-k}{n-k+1}} & \vdots & \ddots & \frac{\hat{b}_1 \binom{n}{1}\binom{m-k}{m-k}}{\binom{m+n-k}{m-k+1}} \\
\frac{\hat{a}_m \binom{m}{m}\binom{n-k}{0}}{\binom{m+n-k}{m}} & \ddots & \vdots & \frac{\hat{b}_n \binom{n}{n}\binom{m-k}{0}}{\binom{m+n-k}{n}} & \ddots & \vdots \\
 & \ddots & \vdots & & \ddots & \vdots \\
 & & \frac{\hat{a}_m \binom{m}{m}\binom{n-k}{n-k}}{\binom{m+n-k}{m+n-k}} & & & \frac{\hat{b}_n \binom{n}{n}\binom{m-k}{m-k}}{\binom{m+n-k}{m+n-k}}
\end{array}\right], \tag{8}$$

and its entries are functions of k, which implies they differ from the entries of $S_{k-1}(\hat{f}, \hat{g})$. It follows from (8) that $S_k(\hat{f}, \hat{g})$ can be partitioned as

$$S_k(\hat{f}, \hat{g}) = \left[\begin{array}{cc} C_k(\hat{f}) & C_k(\hat{g}) \end{array}\right], \tag{9}$$

where $C_k(\hat{f}) \in \mathbb{R}^{(m+n-k+1)\times(n-k+1)}$ and $C_k(\hat{g}) \in \mathbb{R}^{(m+n-k+1)\times(m-k+1)}$ contain the coefficients of $\hat{f}(y)$ and $\hat{g}(y)$ respectively. Entry (i,j) of $C_k(\hat{f})$ is given by

$$C_k(\hat{f})_{(i,j)} = \begin{cases} \frac{\hat{a}_{i-j}\binom{m}{i-j}\binom{n-k}{j-1}}{\binom{m+n-k}{i-1}}, & \begin{array}{l} j = 1, \dots, n-k+1, \\ i = j, \dots, m+j, \end{array} \\ 0, & \text{otherwise}, \end{cases} \tag{10}$$

and similarly, entry (i,j) of $C_k(\hat{g})$ is given by

$$C_k(\hat{g})_{(i,j)} = \begin{cases} \frac{\hat{b}_{i-j}\binom{n}{i-j}\binom{m-k}{j-1}}{\binom{m+n-k}{i-1}}, & \begin{array}{l} j = 1, \dots, m-k+1, \\ i = j, \dots, n+j, \end{array} \\ 0, & \text{otherwise}. \end{cases} \tag{11}$$

The binomial terms in these expressions are functions of i or j, and it is shown they can be simplified, such that two, and not three, binomial terms need be evaluated for each value of k. These modified expressions have many advantages, including the simplification of the calculation of the geometric means of the non-zero entries of $C_k(\hat{f})$ and $C_k(\hat{g})$, and the removal of constant column multipliers from these matrices. Also, they allow the development of an equation in which $S_k(\hat{f}, \hat{g})$ is expressed in terms of $S_j(\hat{f}, \hat{g})$, $j < k$.

The binomial terms in entry (i,j) of $C_k(\hat{f})$, which is defined in (10), can be simplified

$$\frac{\binom{m}{i-j}\binom{n-k}{j}}{\binom{m+n-k}{i}} = \frac{\binom{m+n-k-i}{n-k-j}\binom{i}{j}}{\binom{m+n-k}{n-k}},$$

and thus (10) and (11) can be written as

$$C_k(\hat{f})_{(i,j)} = \begin{cases} \frac{\hat{a}_{i-j}\binom{m+n-k-i+1}{n-k-j+1}\binom{i-1}{j-1}}{\binom{m+n-k}{n-k}}, & \begin{array}{l} j = 1, \dots, n-k+1, \\ i = j, \dots, m+j, \end{array} \\ 0, & \text{otherwise}, \end{cases} \tag{12}$$

and

$$C_k(\hat{g})_{(i,j)} = \begin{cases} \frac{\hat{b}_{i-j}\binom{m+n-k-i+1}{m-k-j+1}\binom{i-1}{j-1}}{\binom{m+n-k}{m-k}}, & \begin{array}{l} j = 1, \dots, m-k+1, \\ i = j, \dots, n+j, \end{array} \\ 0, & \text{otherwise}, \end{cases} \tag{13}$$

respectively. The forms of $C_k(\hat{f})$ and $C_k(\hat{g})$ in (10) and (11) show that the denominators must be evaluated $m+n-k+1$ times for each value of k, but their forms in (12) and (13) are computationally more efficient because they require only two evaluations for each value of k. These improved expressions for the entries of $C_k(\hat{f})$ and $C_k(\hat{g})$ require that the modified Sylvester matrix and its subresultant matrices (4) be used, rather than the standard form $D_k^{-1}T_k(\hat{f},\hat{g})$.

The denominators can be removed from $C_k(\hat{f})$ and $C_k(\hat{g})$ because they do not change the properties of $S_k(\hat{f},\hat{g})$ for the computation of the degree of the GCD of $\hat{f}(y)$ and $\hat{g}(y)$. In particular, their removal is equivalent to scaling $\hat{f}(y)$ and $\hat{g}(y)$ by, respectively, $\binom{m+n-k}{n-k}$ and $\binom{m+n-k}{m-k}$ for each value of k, and thus if the modified Sylvester matrix and its subresultant matrices from which the denominators are omitted are denoted by $\tilde{S}_k(\hat{f},\hat{g})$,

$$\tilde{S}_k(\hat{f},\hat{g}) = \left[\begin{array}{cc} \tilde{C}_k(\hat{f}) & \tilde{C}_k(\hat{g}) \end{array}\right], \qquad k=1,\ldots,\min(m,n), \tag{14}$$

then

$$\tilde{C}_k(\hat{f}) = \binom{m+n-k}{n-k} C_k(\hat{f}) \quad \text{and} \quad \tilde{C}_k(\hat{g}) = \binom{m+n-k}{m-k} C_k(\hat{g}). \tag{15}$$

The simplifications (12) and (13) of the entries of $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$ are one advantage of this modified form of the Sylvester matrix and its subresultant matrices, and its other advantages arise from consideration of the condition numbers, and the magnitudes of the entries, of $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$, $D_k^{-1}T_k(\hat{f},\hat{g})$, $T_k(\hat{f},\hat{g})$ and $\tilde{S}_k(\hat{f},\hat{g})$, which is defined in (14). These advantages are considered in Example 4.1.

Example 4.1. Let the degree and coefficients of $\hat{f}(y)$ be $m=50$ and $\hat{a}_i=1, i=0,\ldots,50$, and let the degree and coefficients of $\hat{g}(y)$ be $n=5$ and $\hat{b}_j=1, j=0,\ldots,5$. The restriction that the coefficients be equal to one makes it easier to isolate the effects of the binomial coefficients. Figures 1 and 2 show the variation of $\log_{10}\kappa(D_k^{-1}T_k(\hat{f},\hat{g})Q_k)$ and $\log_{10}\kappa(\tilde{S}_k(\hat{f},\hat{g}))$, and $\log_{10}\kappa(D_k^{-1}T_k(\hat{f},\hat{g})Q_k)$, $\log_{10}\kappa(D_k^{-1}T_k(\hat{f},\hat{g}))$ and $\log_{10}\kappa(T_k(\hat{f},\hat{g}))$, respectively, with k, where $\kappa(X)$ denotes the condition number of X. Figure 3 shows the ratios, on a logarithmic scale, of the entry of maximum magnitude to the entry of minimum magnitude of $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$, $D_k^{-1}T_k(\hat{f},\hat{g})$ and $T_k(\hat{f},\hat{g})$.

The degree of $\hat{g}(y)$ was then changed to $n = 25$ and $n = 45$, and the results are shown in Figures 4, 5 and 6, and Figures 7, 8 and 9, respectively. The graphs in the figures and other results show the advantages of performing all the GCD computations using the modified Sylvester matrix and its subresultant matrices $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$, $k = 1,\ldots,\min(m,n)$:

1. Figures 1, 4 and 7 show that $\kappa(D_k^{-1}T_k(\hat{f},\hat{g})Q_k)$ is several orders of magnitude smaller than $\kappa(\bar{S}(\hat{f},\hat{g}))$, apart from for small values of k, when they are approximately equal.
2. Figures 2, 5 and 8 show that
$$\kappa(D_k^{-1}T_k(\hat{f},\hat{g})Q_k) < \kappa(D_k^{-1}T_k(\hat{f},\hat{g})), \kappa(T_k(\hat{f},\hat{g})),$$
for all values of k, and the maximum ratio between these condition numbers occurs for small values of k.
3. Figures 3, 6 and 9 show the ratios τ_1 and τ_2 of the entry of maximum magnitude to the entry of minimum magnitude of, respectively, $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$ and $D_k^{-1}T_k(\hat{f},\hat{g})$. The value of this ratio for $T_k(\hat{f},\hat{g})$ is also shown, and it is constant because its entries are functions of m and n only, and they are independent of k. It is seen that $\tau_1/\tau_2 \ll 1$ if $m(n)$ is much larger than $n(m)$, and it increases as $m \to n$. Even though the coefficients of $\hat{f}(y)$ and $\hat{g}(y)$ are equal to one, the ratios τ_1 and τ_2 may be very large for moderate and large values of m and n, and these large values can cause numerical problems in polynomial computations [8, 10].

The graphs in Figures 1 - 9 are typical of the results obtained with other values of m, n and k, but there are combinations of m, n and k for which the trends in these figures are not observed and slightly different results are obtained. ■

The results of Example 4.1 suggest it is better to use the modified form $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$ of the Sylvester matrix and its subresultant matrices than the standard form $D_k^{-1}T_k(\hat{f},\hat{g})$ because of its smaller condition number for all values

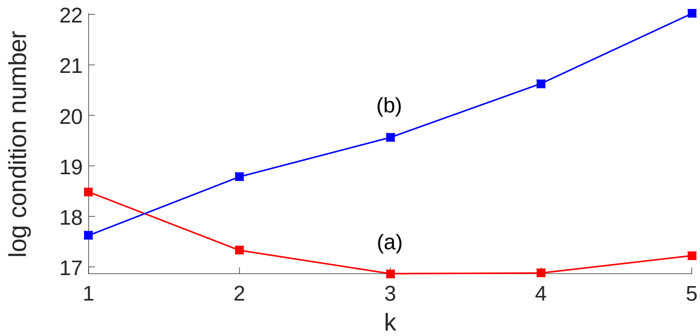

Figure 1: The condition numbers of (a) $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$ and (b) $\tilde{S}_k(\hat{f},\hat{g})$, against k, for $m = 50$ and $n = 5$.

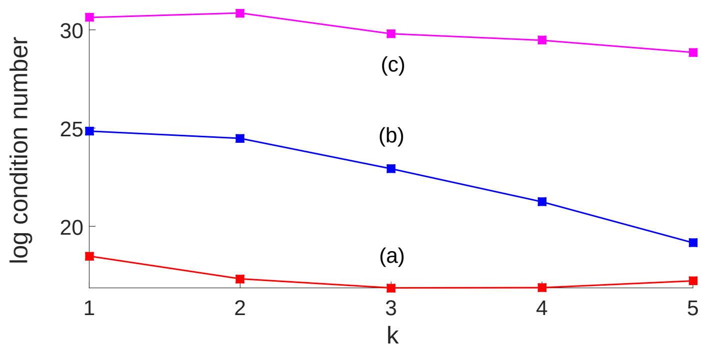

Figure 2: The condition numbers of (a) $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$, (b) $D_k^{-1}T_k(\hat{f},\hat{g})$ and (c) $T_k(\hat{f},\hat{g})$, against k, for $m = 50$ and $n = 5$.

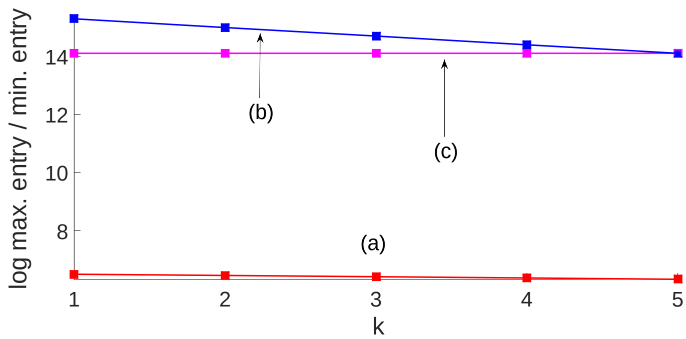

Figure 3: The ratios of the entry of maximum magnitude to the entry of minimum magnitude of (a) $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$, (b) $D_k^{-1}T_k(\hat{f},\hat{g})$ and (c) $T_k(\hat{f},\hat{g})$, against k, for $m = 50$ and $n = 5$.

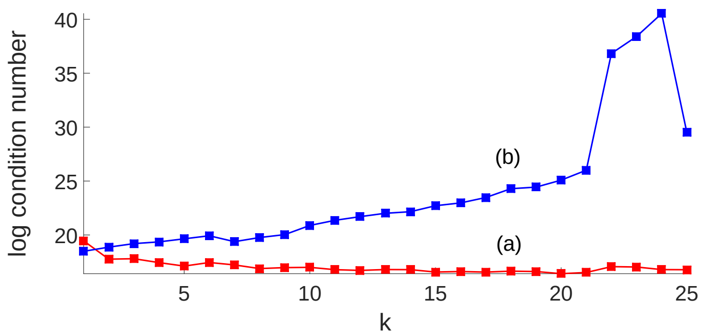

Figure 4: The condition numbers of (a) $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$ and (b) $\tilde{S}_k(\hat{f},\hat{g})$, against k, for $m = 50$ and $n = 25$.

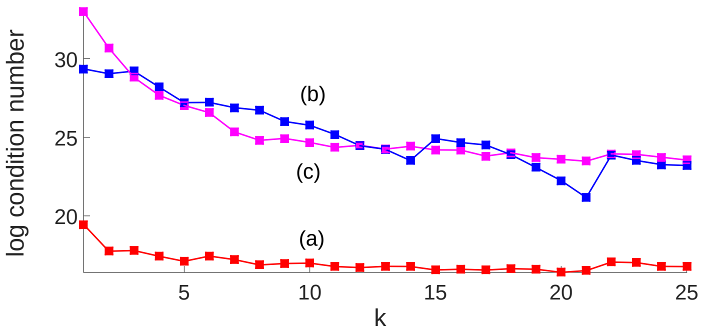

Figure 5: The condition numbers of (a) $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$, (b) $D_k^{-1}T_k(\hat{f},\hat{g})$ and (c) $T_k(\hat{f},\hat{g})$, against k, for $m = 50$ and $n = 25$.

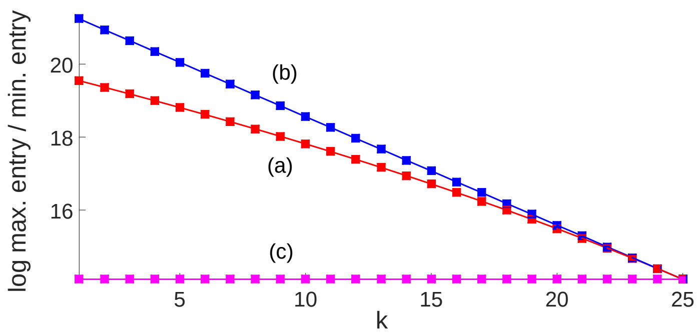

Figure 6: The ratios of the entry of maximum magnitude to the entry of minimum magnitude of (a) $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$, (b) $D_k^{-1}T_k(\hat{f},\hat{g})$ and (c) $T_k(\hat{f},\hat{g})$, against k, for $m = 50$ and $n = 25$.

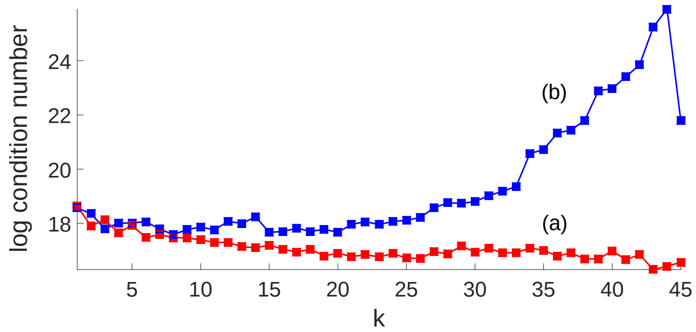

Figure 7: The condition numbers of (a) $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$ and (b) $\tilde{S}_k(\hat{f},\hat{g})$, against k, for $m = 50$ and $n = 45$.

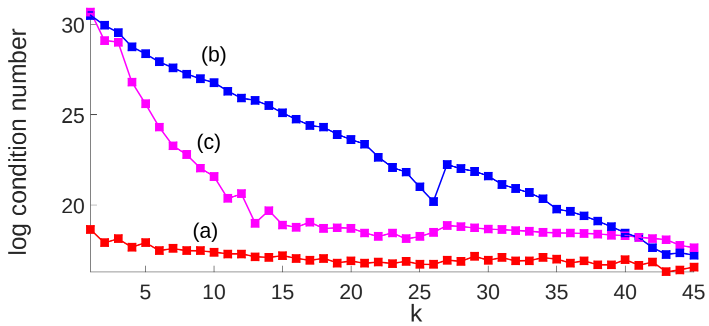

Figure 8: The condition numbers of (a) $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$, (b) $D_k^{-1}T_k(\hat{f},\hat{g})$ and (c) $T_k(\hat{f},\hat{g})$, against k, for $m = 50$ and $n = 45$.

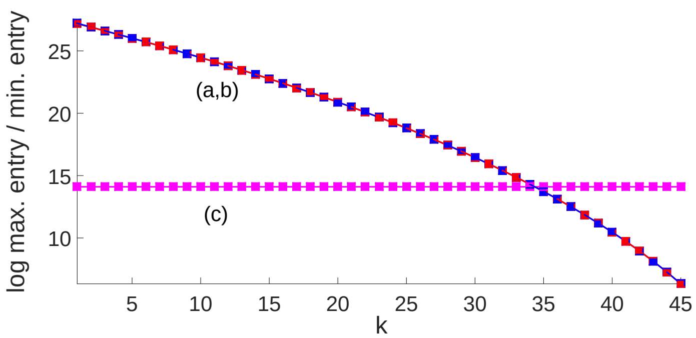

Figure 9: The ratios of the entry of maximum magnitude to the entry of minimum magnitude of (a) $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$, (b) $D_k^{-1}T_k(\hat{f},\hat{g})$ and (c) $T_k(\hat{f},\hat{g})$, against k, for $m = 50$ and $n = 45$.

of k, and $\tau_1 \leq \tau_2$. Figures 6 and 9 show that the ratio τ_1 may be large, but it is shown in Section 5 that $f(y)$ and $g(y)$ are processed before an AGCD is computed, such that τ_1 is minimised. The range of the magnitudes of the entries of $T_k(\hat{f},\hat{g})$ may be more or less than the range of the magnitudes of the entries of $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$, depending on the values of m, n and k, but Figures 2, 5 and 8 show that its condition number is larger than the condition number of $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$. It is important to note that these observations arise from computational experiments, and they are not theoretical derivations.

Equations (12) and (13) show that the entries of $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$ can be computed more efficiently than the entries of $D_k^{-1}T_k(\hat{f},\hat{g})$, and the next section shows that the entries of $D_k^{-1}T_k(\hat{f},\hat{g})Q_k$ can also be computed recursively.

4.2. The computation of $S_k(\hat{f},\hat{g})$ from $S_{k-1}(\hat{f},\hat{g})$

An equation that allows the entries of $S_k(\hat{f},\hat{g})$ to be computed from the entries of $S_{k-1}(\hat{f},\hat{g})$ is developed in this section. The partitioned form of $S_k(\hat{f},\hat{g})$ is shown in (9), and it is adequate to establish the relationship between $C_k(\hat{f})$ and $C_{k-1}(\hat{f})$ because an identical relationship is valid between $C_k(\hat{g})$ and $C_{k-1}(\hat{g})$, with m replaced by n.

Theorem 4.1. *The matrices $C_k(\hat{f})$ and $C_{k-1}(\hat{f})$ satisfy*

$$C_k(\hat{f}) = \lambda_{k-1} A_{k-1}(\hat{f},\hat{g}) C_{k-1}(\hat{f}) B_{k-1}(\hat{g}), \tag{16}$$

where $A_{k-1} = A_{k-1}(\hat{f},\hat{g})$ is given by

$$A_{k-1} = \begin{bmatrix} 0 & \frac{1}{1} & & & \\ 0 & & \frac{1}{2} & & \\ & & & \ddots & \\ 0 & & & & \frac{1}{m+n-(k-1)} \end{bmatrix} \in \mathbb{R}^{(m+n-k+1)\times(m+n-k+2)}, \tag{17}$$

$$B_{k-1}(\hat{g}) = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 1 & & & \\ & 2 & & \\ & & \ddots & \\ & & & n-(k-1) \end{bmatrix} \in \mathbb{R}^{(n-k+2)\times(n-k+1)}, \tag{18}$$

$$\lambda_{k-1} = \frac{\binom{m+n-k+1}{n-k+1}}{\binom{m+n-k}{n-k}} = \frac{m+n-k+1}{n-k+1},$$

and the only non-zero entries of $A_{k-1}(\hat{f},\hat{g})$ and $B_{k-1}(\hat{g})$ are on their super-diagonal and sub-diagonal, respectively.

Proof Use the definitions of $C_k(\hat{f}), A_{k-1}(\hat{f},\hat{g}), B_{k-1}(\hat{g})$ and λ_{k-1} to establish the equivalence of the left and right hand sides of (16). ■

The application of Theorem 4.1 to $C_k(\hat{g})$ shows that

$$\begin{aligned} S_k(\hat{f},\hat{g}) &= \begin{bmatrix} C_k(\hat{f}) & C_k(\hat{g}) \end{bmatrix} \\ &= A_{k-1}(\hat{f},\hat{g}) \begin{bmatrix} \lambda_{k-1} C_{k-1}(\hat{f}) B_{k-1}(\hat{g}) & \mu_{k-1} C_{k-1}(\hat{g}) B_{k-1}(\hat{f}) \end{bmatrix}, \end{aligned} \tag{19}$$

where $B_{k-1}(\hat{f}) \in \mathbb{R}^{(m-k+2)\times(m-k+1)}$ is given by

$$B_{k-1}(\hat{f}) = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 1 & & & \\ & 2 & & \\ & & \ddots & \\ & & & m-(k-1) \end{bmatrix} \quad \text{and} \quad \mu_{k-1} = \frac{m+n-k+1}{m-k+1},$$

and it therefore follows from (14) and (15) that (19) can be written in a form in which the constants λ_{k-1} and μ_{k-1} are omitted,

$$\tilde{S}_k(\hat{f},\hat{g}) = A_{k-1}(\hat{f},\hat{g}) \begin{bmatrix} \tilde{C}_{k-1}(\hat{f}) B_{k-1}(\hat{g}) & \tilde{C}_{k-1}(\hat{g}) B_{k-1}(\hat{f}) \end{bmatrix}. \tag{20}$$

Equations (19) and (20) show that the kth subresultant matrix can be computed from the $(k-1)$th subresultant matrix, and they can be extended to the equations between $S_k(\hat{f},\hat{g})$ and $S_j(\hat{f},\hat{g})$, and $\tilde{S}_k(\hat{f},\hat{g})$ and $\tilde{S}_j(\hat{f},\hat{g})$, respectively, $j<k$. For simplicity, only the relationship between $\tilde{S}_k(\hat{f},\hat{g})$ and $\tilde{S}_j(\hat{f},\hat{g})$ is derived, and this is considered in Theorem 4.2.

Theorem 4.2. *The matrices $\tilde{C}_k(\hat{f})$ and $\tilde{C}_j(\hat{f})$ satisfy*

$$\tilde{C}_k(\hat{f}) = A_{k-1}\cdots\left(A_{j+1}\big(A_j\tilde{C}_j(\hat{f})B_j\big)B_{j+1}\right)\cdots B_{k-1}, \tag{21}$$

where $A_p = A_p(\hat{f},\hat{g})$, $B_q = B_q(\hat{g})$, and $A_{k-1}(\hat{f},\hat{g})$ and $B_{k-1}(\hat{g})$ are defined in (17) and (18) respectively. The product $A_{k-1}\cdots A_j$ yields the matrix $A_{k,j}(\hat{f},\hat{g}) \in \mathbb{R}^{(m+n-k+1)\times(m+n-j+1)}$,

$$A_{k,j}(\hat{f},\hat{g}) = \prod_{i=j}^{k-1} A_i(\hat{f},\hat{g}) \;=\; \left[\begin{array}{ccc|ccc} 0 & \cdots & 0 & \frac{0!}{(k-j)!} & & \\ \vdots & & \vdots & & \ddots & \\ 0 & \cdots & 0 & & & \frac{(m+n-k)!}{(m+n-j)!} \end{array}\right],$$

where the zero matrix is of order $(m+n-k+1)\times(k-j)$ and the diagonal matrix is square and of order $(m+n-k+1)$. The product $B_j\cdots B_{k-1}$ has a similar form,

$$B_{j,k}(\hat{g}) = \prod_{i=j}^{k-1} B_i(\hat{g}) = \left[\begin{array}{ccc} 0 & \cdots & 0 \\ \vdots & & \vdots \\ 0 & \cdots & 0 \\ \hline \frac{(k-j)!}{0!} & & \\ & \ddots & \\ & & \frac{(n-j)!}{(n-k)!} \end{array}\right] \in \mathbb{R}^{(n-j+1)\times(n-k+1)},$$

where the zero matrix is of order $(k-j)\times(n-k+1)$ and the diagonal matrix is square and of order $(n-k+1)$.

Proof The equivalence of the left and right hand sides of (21) follows from the definitions of $A_j(\hat{f},\hat{g}), B_j(\hat{g})$ and $\tilde{C}_j(\hat{f})$. ■

5. Preprocessing operations

It has been shown it is advantageous to compute the degree of the GCD of $\hat{f}(y)$ and $\hat{g}(y)$ from $S_k(\hat{f}, \hat{g}) = D_k^{-1} T_k(\hat{f}, \hat{g}) Q_k$ because it requires the evaluation of fewer binomial terms and, as shown in Example 4.1, the effect of the binomial terms on its condition number is smaller than on the condition numbers of $D_k^{-1} T_k(\hat{f}, \hat{g})$ and $T_k(\hat{f}, \hat{g})$. The ratio of the entry of maximum magnitude to the entry of minimum magnitude of $D_k^{-1} T_k(\hat{f}, \hat{g}) Q_k$ may still, however, be large, which may cause numerical problems, and numerical problems may also arise because of the partitioned form of $S_k(\hat{f}, \hat{g})$. This section considers three preprocessing operations that must be performed on $S_k(\hat{f}, \hat{g})$ in order to minimise the effects of these numerical problems. It is shown in [15, 16, 17, 19] that the inclusion of these preprocessing operations on $\hat{f}(y)$ and $\hat{g}(y)$ yields improved results for AGCD computations. These preprocessing operations are:

1. The normalisation of the entries in the first $n-k+1$ columns and the last $m-k+1$ columns of $S_k(\hat{f}, \hat{g})$ by their geometric means.
2. The replacement of $\hat{g}(y)$ by $\alpha \hat{g}(y)$ where α is a non-zero constant whose optimal value is computed.
3. The transformation of the independent variable y to a new independent variable w by the substitution,

$$y = \theta w, \tag{22}$$

   where θ is a parameter whose optimal value is computed.

The first preprocessing operation requires the normalisation of $C_k(\hat{f})$ and $C_k(\hat{g})$ by the geometric means of their non-zero entries. These means are functions of k, and expressions for them, using the explicit computation (4), are derived in [19]. This derivation is reviewed and the calculation of the geometric means when the subresultant matrices are computed from (12) and (13) is then considered.

Consider initially the geometric means computed in [19], which uses the form of $S_k(\hat{f}, \hat{g})$ in (8). In particular, it is shown that the geometric mean of the non-zero entries of $C_k(\hat{f})$ is

$$\prod_{i=0}^{m}\left(\frac{\left|\hat{a}_i\binom{m}{i}\right|^{n-k+1}\prod_{j=0}^{n-k}\binom{n-k}{j}}{\prod_{j=i}^{n-k+i}\binom{m+n-k}{j}}\right)^{\frac{1}{(n-k+1)(m+1)}}, \tag{23}$$

and that the geometric mean of the non-zero entries of $C_k(\hat{g})$ is

$$\prod_{i=0}^{n}\left(\frac{\left|\hat{b}_i\binom{n}{i}\right|^{m-k+1}\prod_{j=0}^{m-k}\binom{m-k}{j}}{\prod_{j=i}^{m-k+i}\binom{m+n-k}{j}}\right)^{\frac{1}{(m-k+1)(n+1)}}. \tag{24}$$

Consider now the computation of the geometric means of the non-zero entries of the modified Sylvester matrix and its subresultant matrices when they are expressed in the forms (12) and (13). The change of index $\hat{i} = i - j$ allows these entries to be expressed as

$$C_k(\hat{f})_{(\hat{i}+j+1,j+1)} = \begin{cases} \frac{\hat{a}_{\hat{i}}\binom{m+n-k-\hat{i}-j}{n-k-j}\binom{\hat{i}+j}{j}}{\binom{m+n-k}{n-k}}, & \begin{array}{l}\hat{i}=0,\ldots,m,\\ j=0,\ldots,n-k,\end{array} \\ 0, & \text{otherwise,} \end{cases} \tag{25}$$

and

$$C_k(\hat{g})_{(\hat{i}+j+1,j+1)} = \begin{cases} \frac{\hat{b}_{\hat{i}}\binom{m+n-k-\hat{i}-j}{m-k-j}\binom{\hat{i}+j}{j}}{\binom{m+n-k}{m-k}}, & \begin{array}{l}\hat{i}=0,\ldots,n,\\ j=0,\ldots,m-k,\end{array} \\ 0, & \text{otherwise,} \end{cases} \tag{26}$$

from which it is seen that the denominator in each entry of $C_k(\hat{f})$ is constant for each value of k, and likewise, the denominator in each entry of $C_k(\hat{g})$ is constant for each value of k. These constant values simplify the calculation of the geometric means of the non-zero entries of $C_k(\hat{f})$ and $C_k(\hat{g})$.

The binomial terms in the numerators of (25) and (26) satisfy a simple relationship that can be exploited for the efficient computation of the geometric means of the non-zero entries in $C_k(\hat{f})$ and $C_k(\hat{g})$. In particular, it follows from the substitutions

$$p = n - k - j \qquad \text{and} \qquad q = m - i,$$

that the evaluation of the geometric means of the numerators of the non-zero terms in (25) is simplified because

$$\prod_{j=0}^{n-k}\prod_{i=0}^{m}\binom{m+n-k-i-j}{n-k-j}=\prod_{j=0}^{n-k}\prod_{i=0}^{m}\binom{i+j}{j}.$$

This equation shows that the product of the first set of binomial terms in the numerators of the entries of $C_k(\hat{f})$ in (25) is equal to the product of the second set of binomial terms, and thus the geometric mean of its non-zero entries is

$$\lambda_k=\frac{\left(\prod_{i=0}^{m}|\hat{a}_i|\right)^{\frac{1}{m+1}}\left(\prod_{j=0}^{n-k}\prod_{i=0}^{m}\binom{i+j}{j}\right)^{\frac{2}{(n-k+1)(m+1)}}}{\binom{m+n-k}{n-k}}. \tag{27}$$

The repetition of this analysis for $C_k(\hat{g})$ in (26) shows that the geometric mean of its non-zero entries is

$$\mu_k=\frac{\left(\prod_{i=0}^{n}\left|\hat{b}_i\right|\right)^{\frac{1}{n+1}}\left(\prod_{j=0}^{m-k}\prod_{i=0}^{n}\binom{i+j}{j}\right)^{\frac{2}{(m-k+1)(n+1)}}}{\binom{m+n-k}{m-k}}. \tag{28}$$

The expressions (27) and (28) enable the normalised forms of $\hat{f}(y)$ and $\hat{g}(y)$ to be calculated, and it is clear they are computationally more efficient than the expressions (23) and (24), respectively, which are obtained from (4).

The second preprocessing operation requires that the normalised form of $\hat{g}(y)$ be scaled by α, and the third preprocessing operation requires the substitution (22). It therefore follows that the three preprocessing operations yield the polynomials

$$\dot{f}(w)=\sum_{i=0}^{m}\left(\bar{a}_i\theta^i\right)\binom{m}{i}(1-\theta w)^{m-i}w^i, \qquad \bar{a}_i=\frac{\hat{a}_i}{\lambda_k}, \tag{29}$$

and

$$\alpha\dot{g}(w)=\alpha\sum_{i=0}^{n}\left(\bar{b}_i\theta^i\right)\binom{n}{i}(1-\theta w)^{n-i}w^i, \qquad \bar{b}_i=\frac{\hat{b}_i}{\mu_k}. \tag{30}$$

The next section considers the computation of the optimal values of α and θ, and it is shown they are functions of k, that is, the computation of their optimal values must be performed for each value of k, $k=1,\ldots,\min(m,n)$.

5.1. The optimal values of α and θ

Numerical problems may occur when computations are performed on polynomials whose coefficients vary widely in magnitude. The optimal values of α and θ are therefore chosen such that the ratio of the entry of maximum magnitude, to the entry of minimum magnitude, of $S_k(\dot{f}, \alpha\dot{g}) = D_k^{-1} T_k(\dot{f}, \alpha\dot{g}) Q_k$, where $\dot{f} = \dot{f}(w)$ and $\dot{g} = \dot{g}(w)$ are defined in (29) and (30) respectively, is minimised.

The general expression for a non-zero entry in the first $n-k+1$ columns of $S_k(\dot{f}, \alpha\dot{g})$ is, from (25) and (29),

$$\frac{\bar{a}_i \binom{m+n-k-i-j}{n-k-j} \binom{i+j}{j} \theta^i}{\binom{m+n-k}{n-k}}, \qquad i = 0, \ldots, m, \qquad j = 0, \ldots, n-k,$$

and similarly, the general expression for a non-zero entry in the last $m-k+1$ columns of $S_k(\dot{f}, \alpha\dot{g})$ is, from (26) and (30),

$$\frac{\alpha \bar{b}_i \binom{m+n-k-i-j}{m-k-j} \binom{i+j}{j} \theta^i}{\binom{m+n-k}{m-k}}, \qquad i = 0, \ldots, n, \qquad j = 0, \ldots, m-k.$$

It is convenient to define the sets $\mathcal{P}_k(\theta)$ and $\mathcal{Q}_k(\alpha, \theta)$ as

$$\mathcal{P}_k(\theta) = \left\{ \frac{\left| \bar{a}_i \binom{m+n-k-i-j}{n-k-j} \binom{i+j}{j} \theta^i \right|}{\binom{m+n-k}{n-k}}; \quad i = 0, \ldots, m, \quad j = 0, \ldots, n-k \right\},$$

and

$$\mathcal{Q}_k(\alpha, \theta) = \left\{ \frac{\left| \alpha \bar{b}_i \binom{m+n-k-i-j}{m-k-j} \binom{i+j}{j} \theta^i \right|}{\binom{m+n-k}{m-k}}; \quad i = 0, \ldots, n, \quad j = 0, \ldots, m-k \right\},$$

respectively, and the optimal values $\alpha_0(k)$ and $\theta_0(k)$ of α and θ minimise the ratio of the entry of maximum magnitude to the entry of minimum magnitude of $S_k(\dot{f}, \alpha\dot{g})$. They are therefore given by

$$\alpha_0(k), \theta_0(k) = \arg\min_{\alpha, \theta} \left\{ \frac{\max\{\max\{\mathcal{P}_k(\theta)\}, \max\{\mathcal{Q}_k(\alpha, \theta)\}\}}{\min\{\min\{\mathcal{P}_k(\theta)\}, \min\{\mathcal{Q}_k(\alpha, \theta)\}\}} \right\},$$

for $k = 1, \ldots, \min(m, n)$, and it is shown in [19] that this minimisation leads to a linear programming problem. The substitution of the solutions $\alpha_0(k)$ and

$\theta_0(k)$ in (29) and (30) shows that the polynomials whose AGCD is computed are $\bar{f}(w)$ and $\alpha_0\bar{g}(w)$,

$$\bar{f}(w) = \sum_{i=0}^{m} \left(\bar{a}_i\theta_0^i\right)\binom{m}{i}(1-\theta_0 w)^{m-i}w^i, \tag{31}$$

and

$$\alpha_0\bar{g}(w) = \alpha_0 \sum_{i=0}^{n} \left(\bar{b}_i\theta_0^i\right)\binom{n}{i}(1-\theta_0 w)^{n-i}w^i, \tag{32}$$

where $\alpha_0 = \alpha_0(k)$ and $\theta_0 = \theta_0(k)$.

6. The degree of an AGCD

This section considers three methods for the computation of the degree of an AGCD of the polynomials $\bar{f}(w)$ and $\alpha_0\bar{g}(w)$, which are defined in (31) and (32) respectively. These methods are described in Sections 6.1 and 6.2, and they use the residuals of a set of approximate linear algebraic equations and the QR decomposition of each matrix $S_k(\bar{f}, \alpha_0\bar{g})$, $k = 1, \ldots, \min(m, n)$, respectively.

6.1. The method of residuals

A method for the calculation of the degree t of an AGCD of $\bar{f}(w)$ and $\alpha_0\bar{g}(w)$ that is based on (5), using the residuals of a set of approximate linear algebraic equations, is considered in [17]. These approximate equations are

$$A_{k,q}x_q \approx c_{k,q}, \quad k = 1, \ldots, \min(m, n), \quad q = 1, \ldots, m+n-2k+2, \tag{33}$$

where $A_{k,q}$ and $c_{k,q}$ are derived from $S_k(\bar{f}, \alpha_0\bar{g})$. The residual $r_{k,q}$ of the least squares solution of each of these approximate equations is computed, and the minimum residual for each value of k is calculated,

$$r(k) = \min_q \{r_{k,q}\}, \qquad k = 1, \ldots, \min(m, n). \tag{34}$$

It is shown in [17] that if $r(k)$ is large, $S_k(\bar{f}, \alpha_0\bar{g})$ has full column rank and thus $\bar{f}(w)$ and $\alpha_0\bar{g}(w)$ do not have an AGCD of degree k. If, however, $r(k)$ is small, then there exists at least one column of $S_k(\bar{f}, \alpha_0\bar{g})$ that is almost linearly dependent on its other columns, and thus $S_k(\bar{f}, \alpha_0 \bar{g})$ is numerically singular. It therefore follows that $\bar{f}(w)$ and $\alpha_0 \bar{g}(w)$ have an approximate common divisor of degree k, and the degree of an AGCD is equal to the largest value of k such that $S_k(\bar{f}, \alpha_0 \bar{g})$ is numerically singular. The degree t of an AGCD of $\bar{f}(w)$ and $\alpha_0 \bar{g}(w)$ is therefore given by

$$t = \arg \max_{k=1,\ldots,\min(m,n)-1} \frac{r(k+1)}{r(k)},$$

that is, t is equal to the value of k for which the ratio between two successive values of $r(k)$ is a maximum because this marks the change from a numerically singular matrix $S_k(\bar{f}, \alpha_0 \bar{g})$ ($r(k)$ is small) to a matrix $S_{k+1}(\bar{f}, \alpha_0 \bar{g})$ that is far from singularity ($r(k+1)$ is large).

The residuals $r_{k,q}$ are usually calculated by the SVD, and this paper also considers their computation by the QR decomposition. This decomposition has been used by other researchers [7, 20], but the methods used to calculate the degree of an AGCD in this paper differ from the methods in these references.

6.2. The application of the QR decomposition

The value of t can also be calculated from the square upper triangular matrix R_k of the QR decomposition of $S_k(\bar{f}, \alpha_0 \bar{g})$,

$$S_k(\bar{f}, \alpha_0 \bar{g}) = Q_k \begin{bmatrix} R_k \\ 0 \end{bmatrix}, \tag{35}$$

where Q_k is an orthogonal matrix.[3] Two tests can be performed on R_k in order to determine the numerical rank of $S_k(\bar{f}, \alpha_0 \bar{g})$, and therefore the degree of an AGCD of $\bar{f}(w)$ and $\alpha_0 \bar{g}(w)$.

1. The ratio $\rho_1(k)$ of the maximum diagonal entry of R_k to the minimum diagonal entry of R_k,

$$\rho_1(k) = \frac{\max_i \{|R_{k,i,i}|\}}{\min_i \{|R_{k,i,i}|\}}, \qquad k = 1, \ldots, \min(m,n), \tag{36}$$

[3]It is clear that this matrix is not related to the diagonal matrix Q_k of binomial factors defined in (3).

is computed, where $R_{k,i,j}$ is entry (i,j) of R_k. Since $\rho_1(k)$ is finite if R_k is non-singular and infinite if R_k is singular, it follows from (5) that t is given by the maximum change between successive values of k,

$$t = \arg \max_{k=1,\ldots,\min(m,n)-1} \frac{\rho_1(k)}{\rho_1(k+1)}.$$

2. If $s_{k,i}$ is the ith row of R_k, then the ratio $\rho_2(k)$ is defined as the ratio of the maximum 2-norm to the minimum 2-norm of the rows of R_k,

$$\rho_2(k) = \frac{\max_i \left\{ \|s_{k,i}\|_2 \right\}}{\min_i \left\{ \|s_{k,i}\|_2 \right\}}, \qquad k = 1, \ldots, \min(m,n). \tag{37}$$

A matrix is near singularity if $\rho_2(k)$ is large, and it therefore follows that t can also be computed from

$$t = \arg \max_{k=1,\ldots,\min(m,n)-1} \frac{\rho_2(k)}{\rho_2(k+1)}.$$

Many AGCD computations, for the power and Bernstein bases, have shown that $\rho_1(k)$ and $\rho_2(k)$ are useful for the (heuristic) computation of the rank of a matrix, and therefore the degree of an AGCD of two polynomials.

7. Examples

This section contains two examples that demonstrate the theory discussed in the previous sections. The coefficients $\hat{a}_i$ of the exact polynomial $\hat{f}(y)$ were corrupted by random noise, such that the upper bound of the componentwise relative error is a uniformly distributed random variable ε_i in the interval $[10^{-p}, 10^{-q}]$, $p > q > 0$, and similarly, for the coefficients $\hat{b}_j$ of $\hat{g}(y)$,

$$a_i = \hat{a}_i + \delta a_i, \qquad \delta a_i = \varepsilon_i \hat{a}_i r_i, \qquad i = 0, \ldots, m, \tag{38}$$

and

$$b_j = \hat{b}_j + \delta b_j, \qquad \delta b_j = \varepsilon_j \hat{b}_j r_j, \qquad j = 0, \ldots, n, \tag{39}$$

where a_i and b_j are the coefficients of the perturbed polynomials $f(y)$ and $g(y)$ respectively, and r_i and r_j are uniformly distributed random variables in the interval $[-1, 1]$. A method for the determination of the degree of an AGCD of $f(y)$ and $g(y)$ that requires a threshold cannot be used because the upper bounds ε_i and ε_j of the relative errors are not constant.

The degree of an AGCD of $f(y)$ and $g(y)$ was also computed from the rank loss of the modified Sylvester matrix $D^{-1}T(\bar{f}, \alpha_0\bar{g})Q = D_1^{-1}T_1(\bar{f}, \alpha_0\bar{g})Q_1$ and the Bézout matrix $B(f, g)$ of $f(y)$ and $g(y)$ [3]. This matrix is square, like the Sylvester matrix, but it differs from it because it is (a) of order $\max(m, n)$, rather than $m + n$, (b) symmetric, and (c) bilinear. It follows from property (c) that the only preprocessing operation that need be applied is the transformation of the independent variable y to the independent variable w, which is defined by (22). The calculation of the optimal value of θ requires the solution of a linear programming problem, which is considered in [19].

Example 7.1. Consider the polynomials $\hat{f}(y)$ and $\hat{g}(y)$,

$$\begin{aligned}
\hat{f}(y) &= \sum_{i=0}^{19} \hat{a}_i \binom{19}{i} (1-y)^{19-i} y^i \\
&= (y-0.10)^4 (y-0.30)^2 (y-0.50)^2 (y-0.70)^3 \times \\
&\quad (y-0.80)^2 (y-2.50)^3 (y+3.40)^3,
\end{aligned}$$

and

$$\begin{aligned}
\hat{g}(y) &= \sum_{i=0}^{16} \hat{b}_i \binom{16}{i} (1-y)^{16-i} y^i \\
&= (y-0.10)^3 (y-0.80)^2 (y-0.85)^4 (y-0.90)^4 (y-1.10)^3,
\end{aligned}$$

whose GCD is of degree five. The polynomials $f(y)$ and $g(y)$ were formed by adding noise, where ε_i and ε_j are uniformly distributed random variables in the range $\left[10^{-10}, 10^{-8}\right]$, as discussed above. The matrices $D_k^{-1}T_k(f, g)Q_k, k = 1, \ldots, 16$, were constructed by the evaluation of the binomial terms in (12) and (13), and the recurrence equation (16) and its equivalent for $g(y)$. The preprocessing operations discussed in Section 5 were applied to these matrices,

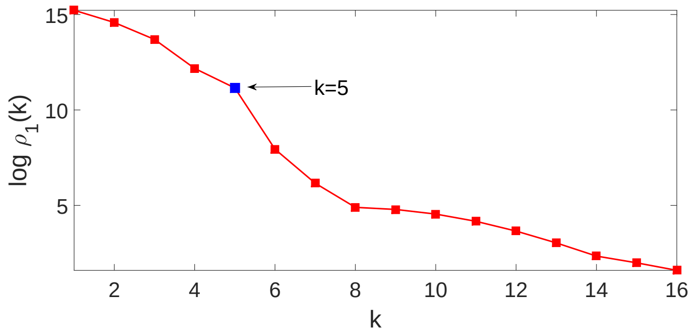

Figure 10: The variation of $\log_{10} \rho_1(k)$ with k, when preprocessing is included, for Example 7.1.

computed by both methods, and thus the polynomials $\bar{f}(w)$ and $\alpha_0 \bar{g}(w)$, which are defined in (31) and (32) respectively, were formed.

The ratios $\rho_1(k)$ and $\rho_2(k)$, which are defined in (36) and (37) respectively, were computed for both forms of construction of the modified Sylvester matrix and its subresultant matrices, and the correct degree of the GCD was obtained for both forms. In particular, Figures 10 and 11 show the variation of $\log_{10} \rho_1(k)$ and $\log_{10} \rho_2(k)$ with k, and it is seen that the greatest change between successive values of $\rho_1(k)$ and $\rho_2(k)$ occurs at $k = 5$, which is correct because this is equal to the degree of the GCD of $\hat{f}(y)$ and $\hat{g}(y)$. Figure 12 shows the variation of the residual $\log_{10} r(k)$, computed by the QR decomposition and SVD, where r_k is defined in (34), against k, and it is seen that the maximum gradient occurs at $k = 5$, which is correct. The greatest difference in the residuals calculated by the two methods occurs for $k < 5$, and the residuals are equal for $k \geq 5$. The residuals computed by the QR decomposition increase monotonically with k, but this property is not shared by the residuals computed by the SVD.

The computations described above were repeated, but the preprocessing operations were not implemented, that is, normalisation by the geometric means is omitted and $\alpha_0 = \theta_0 = 1$. Figures 13 and 14 show, respectively, the variation of $\log_{10} \rho_1(k)$ and $\log_{10} \rho_2(k)$ with k, and Figure 15 shows the variation of $\log_{10} r(k)$ with k, computed by the QR decomposition and the SVD. It is clear

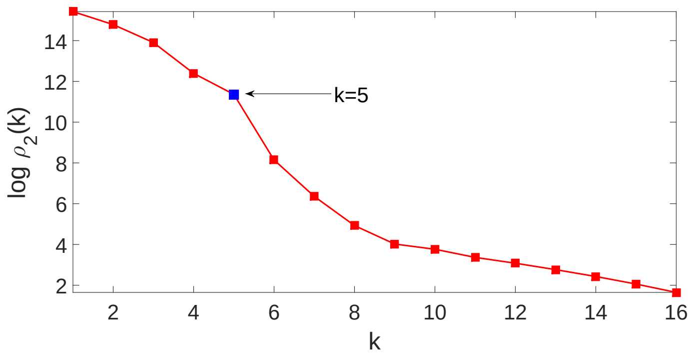

Figure 11: The variation of $\log_{10} \rho_2(k)$ with k, when preprocessing is included, for Example 7.1.

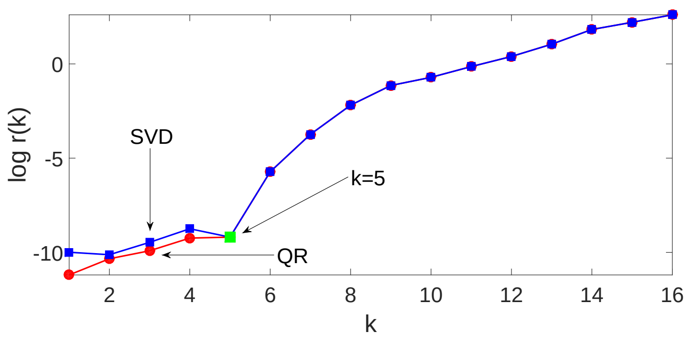

Figure 12: The residual $\log_{10} r(k)$ calculated by the QR decomposition and SVD, against k, when preprocessing is included, for Example 7.1.

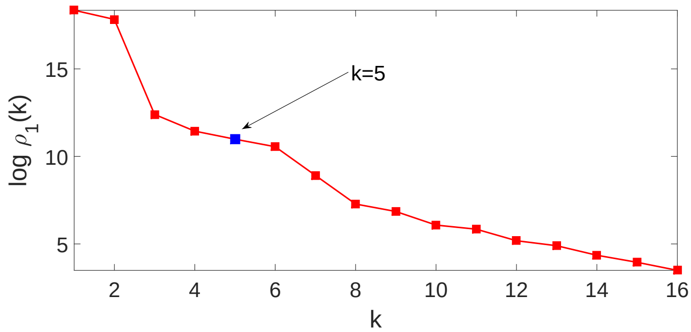

Figure 13: The variation of $\log_{10} \rho_1(k)$ with k, when preprocessing is not included, for Example 7.1.

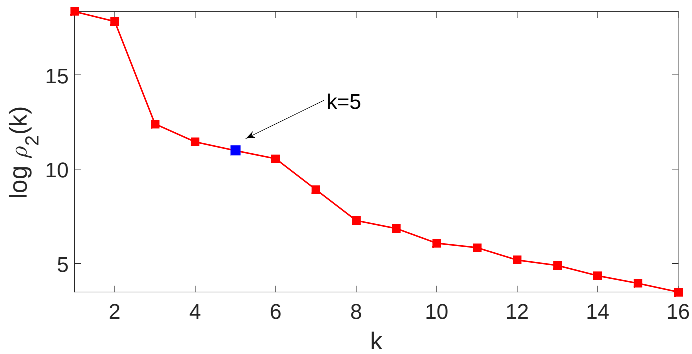

Figure 14: The variation of $\log_{10} \rho_2(k)$ with k, when preprocessing is not included, for Example 7.1.

that Figures 13-15 yield incorrect results, which shows the importance of the preprocessing operations.

The normalised singular values $\log_{10} \sigma_i/\sigma_1$ of $D^{-1}T(\bar{f}, \alpha_0\bar{g})Q$, with and without preprocessing, are shown in Figure 16 . The correct result is obtained when preprocessing is included because the rank loss is five, but an incorrect result ($\deg \text{GCD}(\hat{f}, \hat{g}) = 2$) is obtained when preprocessing is not included, and it is interesting to note that the same incorrect result is obtained in Figures 13-15. Figure 17 shows the normalised singular values of the Bézout matrix $B(f, g)$ when preprocessing is, and is not, included. An incorrect result is obtained when preprocessing is omitted because the rank loss is two, but the correct result is

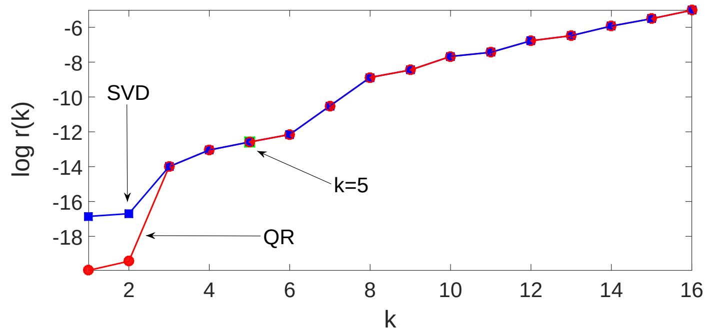

Figure 15: The residual $\log_{10} r(k)$ calculated by the QR decomposition and SVD, against k, when preprocessing is not included, for Example 7.1.

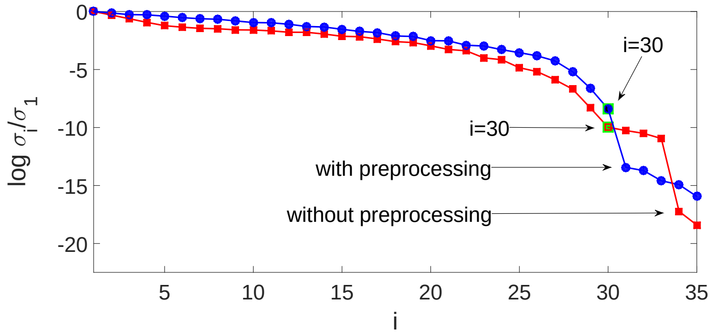

Figure 16: The normalised singular values of $D^{-1}T(\bar{f}, \alpha_0\bar{g})Q$, with and without preprocessing, for Example 7.1.

obtained when preprocessing is included because the rank loss is five, and the figure is therefore consistent with Figures 10-16. ■

Example 7.2. Consider the polynomials $\hat{f}(y)$ and $\hat{g}(y)$,

$$\begin{aligned}\hat{f}(y) &= \sum_{i=0}^{21} \hat{a}_i \binom{21}{i} (1-y)^{21-i} y^i \\ &= (y-0.10)^3 (y-0.56)^4 (y-0.75)^3 (y-0.82)^3 \times \\ &\quad (y-1.37)^3 (y+0.27)^3 (y-1.46)^2,\end{aligned}$$

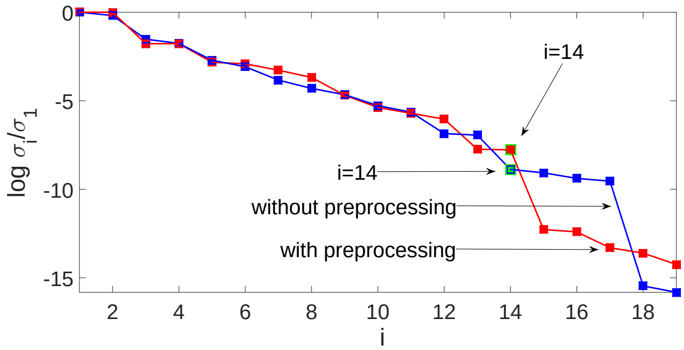

Figure 17: The normalised singular values of the Bézout matrix $B(f, g)$, with and without preprocessing, for Example 7.1.

and

$$
\begin{aligned}
\hat{g}(y) &= \sum_{i=0}^{22} \hat{b}_i \binom{22}{i} (1-y)^{22-i} y^i \\
&= (y-0.10)^2 (y-0.56)^4 (y-0.75)^3 (y-0.99)^4 \times \\
&\quad (y-1.37)^3 (y-2.12)^3 (y-1.20)^3,
\end{aligned}
$$

for which $\deg \mathrm{GCD}\left(\hat{f}, \hat{g}\right) = 12$. Each coefficient of $\hat{f}(y)$ and $\hat{g}(y)$ was perturbed by a uniformly distributed random variable in the interval $\left[10^{-10}, 10^{-8}\right]$, as shown in (38) and (39), thereby yielding the noisy polynomials $f(y)$ and $g(y)$. Both forms of the modified Sylvester matrix (using (12) and (13), and (16)) and its subresultant matrices were computed, and the preprocessing operations were implemented, as described in Example 7.1.

Figures 18 and 19 show the variation of $\log_{10} \rho_1(k)$ and $\log_{10} \rho_2(k)$ with k, and the degree $k = 12$ of the GCD of $\hat{f}(y)$ and $\hat{g}(y)$ is clearly defined in both figures. These figures were obtained using the explicit construction of $D_k^{-1} T_k(\bar{f}, \alpha_0 \bar{g}) Q_k$, and identical results were obtained when the recurrence equation (16) and its equivalent for $g(y)$ were used. Figures 20 and 21 show the variation of $\log_{10} r(k)$, computed by the QR decomposition and the SVD, with k for the two forms of construction of the modified Sylvester matrix and its subresultant matrices. It is seen that both forms yield the correct result, that the graphs are similar for $k \leq 12$, and that they differ for $k > 12$.

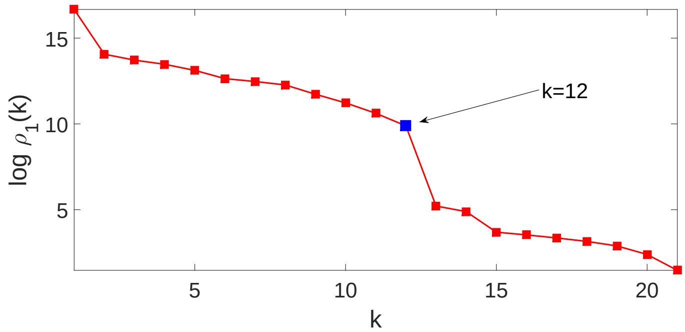

Figure 18: The variation of $\log_{10} \rho_1(k)$ with k, when preprocessing is included, for Example 7.2.

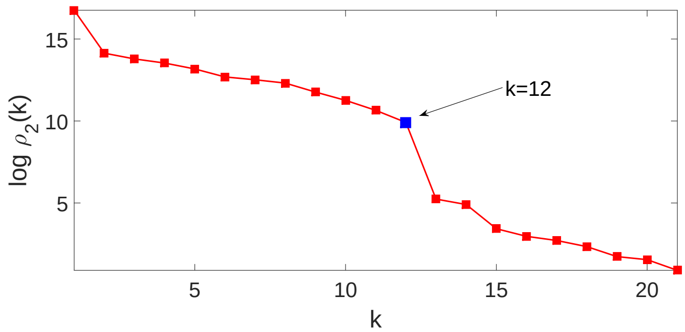

Figure 19: The variation of $\log_{10} \rho_2(k)$ with k, when preprocessing is included, for Example 7.2.

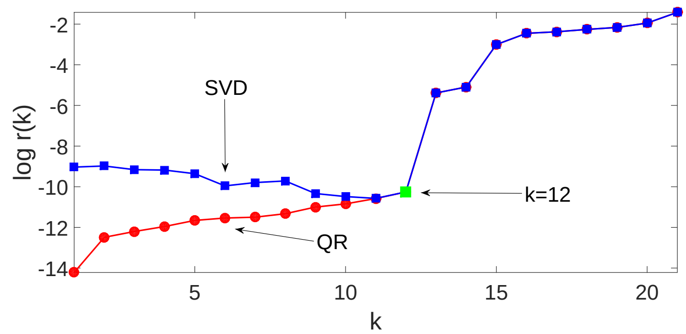

Figure 20: The residual $\log_{10} r(k)$ calculated by the QR decomposition and SVD, against k, when $D_k^{-1} T_k(\bar{f}, \alpha_0 \bar{g}) Q_k$ is constructed explicitly and preprocessing is included, for Example 7.2.

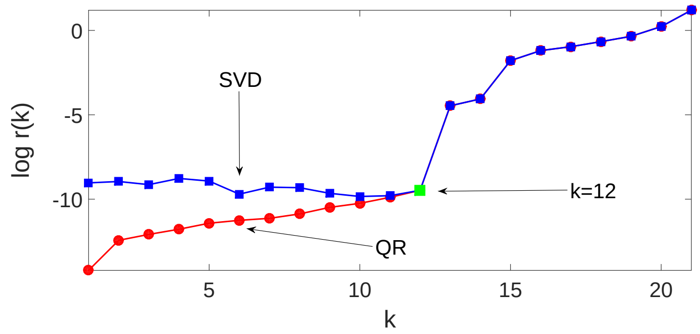

Figure 21: The residual $\log_{10} r(k)$ calculated by the QR decomposition and SVD, against k, when $D_k^{-1}T_k(\bar{f}, \alpha_0\bar{g})Q_k$ is constructed using the recurrence equation (16) and its equivalent for $g(y)$, and preprocessing is included, for Example 7.2.

Good results were also obtained when the denominator was not included, that is, the form of the modified Sylvester matrix and its subresultant matrices (20) were used. Incorrect results were, however, obtained when the preprocessing operations were not included, which are therefore consistent with the results of Example 7.1. Figure 22 shows the normalised singular values $\log_{10} \sigma_i/\sigma_1$ of $D^{-1}T(\bar{f}, \alpha_0\bar{g})Q$, with and without preprocessing. The correct result is obtained when preprocessing is included because the rank loss is 12, but an incorrect result is obtained when preprocessing is not included because the numerical rank is not defined, that is, a clear gap between two successive singular values does not exist. Figures 23 shows the normalised singular values of the Bézout matrix $B(f, g)$ when preprocessing is, and is not, included. It is seen that the correct result is obtained when preprocessing is included, but the numerical rank of $B(f, g)$ is not defined when $f(y)$ and $g(y)$ are not preprocessed. The results in Figures 22 and 23 are therefore consistent with the results in Figures 16 and 17 for Example 7.1. ■

The computation of the degree of an AGCD of $f(y)$ and $g(y)$ was repeated (noise was added to the coefficients of the exact polynomials, which were then preprocessed), but the matrices $T_k(f, \alpha_0 g)$ and $D_k^{-1}T_k(f, \alpha_0 g)$, where $T_k(\hat{f}, \hat{g})$

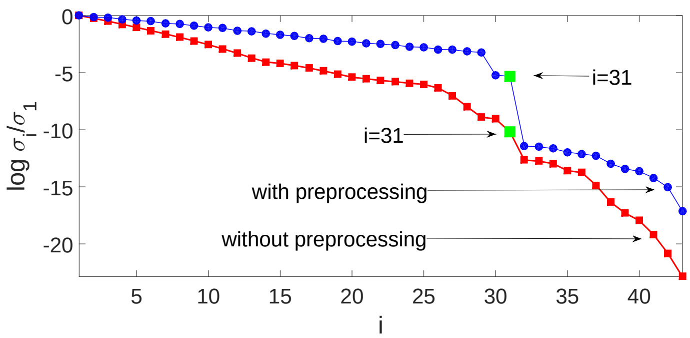

Figure 22: The normalised singular values of $D^{-1}T(\bar{f}, \alpha_0\bar{g})Q$, with and without preprocessing, for Example 7.2.

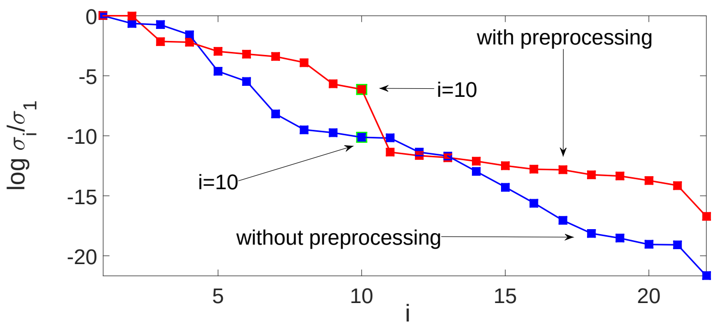

Figure 23: The normalised singular values of the Bézout matrix $B(f, g)$, with and without preprocessing, for Example 7.2.

is defined in (2), were used. The results were not good because the relative errors in the computed coprime polynomials and AGCD were much larger than their corresponding values obtained from $D_k^{-1}T_k(f, \alpha_0 g)Q_k$. Also, the rank loss of $T_k(f, \alpha_0 g)$ and the rank loss of $D_k^{-1}T_k(f, \alpha_0 g)$ were not clearly defined, which shows it is better to compute a structured low rank approximation of the Sylvester matrix of $f(y)$ and $g(y)$ from $D_k^{-1}T_k(f, \alpha_0 g)Q_k$ than from $T_k(f, \alpha_0 g)$ and $D_k^{-1}T_k(f, \alpha_0 g)$.

Figures 12, 15, 20 and 21 show that the residual computed by the QR decomposition is smaller than the residual computed by the SVD for $k < t$. The simple use of the SVD of a matrix A that is near singularity may lead to a bad result in a least squares problem because the reciprocal of the small singular values of A must be computed. This solution is essentially the same as the solution obtained with the function `pinv` in MATLAB, and it is associated with a large error in the least squares solution if a tolerance for the small singular values of A is not specified. A better solution is obtained when these small singular values are set equal to zero, which can be implemented by the addition of a parameter `tol` to `pinv`. In this circumstance, the QR decomposition and SVD yield residuals that may be considered equal, and thus the larger residuals obtained by the SVD follow from the absence of the specification of `tol` in the arguments of `pinv`. It is noted that if A is ill-conditioned and its numerical rank is not defined, that is, its singular values cannot be divided into two groups that are separated by a large and well-defined gap, then the results from `pinv` may be dependent upon the value of `tol`.

The results in Examples 7.1 and 7.2 are typical of the results obtained with other polynomials. For example, the inclusion of the preprocessing operations yielded a significant improvement in the results, and the best results were obtained when the terms in the denominator were included in the computations, that is, the form $S_k(\bar{f}, \alpha_0 \bar{g})$, rather than the form $\tilde{S}_k(\bar{f}, \alpha_0 \bar{g})$, was used, where

$$S_k(\bar{f}, \alpha_0 \bar{g}) = \begin{bmatrix} C_k(\bar{f}) & \alpha_0 C_k(\bar{g}) \end{bmatrix}, \qquad \tilde{S}_k(\bar{f}, \alpha_0 \bar{g}) = \begin{bmatrix} \tilde{C}_k(\bar{f}) & \alpha_0 \tilde{C}_k(\bar{g}) \end{bmatrix},$$

and $\tilde{C}(\hat{f})$ and $\tilde{C}(\hat{g})$ are defined in (15). The values of $\rho_1(k)$ and $\rho_2(k)$ were good measures of the change from singularity to non-singularity for the calculation of the rank of R_k, which is defined in (35).

The rank loss of the Bézout matrix of $f(y)$ and $g(y)$ did not yield good results for the degree of an AGCD when the preprocessing operations were omitted, and better results were obtained when the polynomials were preprocessed. Other examples showed that the modified Sylvester matrix and its subresultant matrices yielded better results than the Bézout matrix in the presence of errors in the coefficients of $f(y)$ and $g(y)$. The inferior results obtained with the Bézout matrix are noted by Bini and Marco [4], and it is believed this arises because each entry of this matrix is of the form $\sum_{i,j}(a_ib_j - a_jb_i)$ and thus small errors in a_ib_j and a_jb_i may lead to a large error in the difference $(a_ib_j - a_jb_i)$ because of numerical cancellation.

All computations were performed in double precision using MATLAB on a standard desktop computer using Windows 7.

8. Summary

This paper has considered the application of the Sylvester and Bézout resultant matrices to the calculation of the degree of an AGCD of two Bernstein polynomials. It was shown that the binomial terms in the Bernstein basis functions may cause numerical problems, the effects of which can be mitigated by preprocessing the polynomials. It was shown that the best results are obtained when a modified form $D_k^{-1}T_k(\bar{f}, \alpha_0\bar{g})Q_k$ of the Sylvester matrix and its subresultant matrices is used.

References

[1] D. A. Bini, P. Boito, Structured matrix-based methods for polynomial ϵ-GCD: Analysis and comparisons, in: Proc. Int. Symp. Symbolic and Algebraic Computation, Waterloo, Canada, 2007, pp. 9–16.

[2] D. A. Bini, P. Boito, A fast algorithm for approximate polynomial GCD based on structured matrix computations, in: D. A. Bini, V. Mehrmann,

V. Olshevsky, E. Tyrtshnikov, M. V. Barel (Eds.), Numerical Methods for Structured Matrices and Applications: The Georg Heinig Memorial Volume, Birkhäuser, 2010, pp. 155–173.

[3] D. A. Bini, L. Gemignani, Bernstein-Bezoutian matrices, Theoretical Computer Science 315 (2004) 319–333.

[4] D. A. Bini, A. Marco, Computing curve intersection by means of simultaneous iterations, Numer. Algor. 43 (2006) 151–175.

[5] P. Chin, R. M. Corless, Optimization strategies for the approximate GCD problem, in: Proc. Int. Symp. Symbolic and Algebraic Computation, Rostock, Germany, 1998, pp. 228–235.

[6] R. M. Corless, P. M. Gianni, B. M. Trager, S. M. Watt, The singular value decomposition for polynomial systems, in: Proc. Int. Symp. Symbolic and Algebraic Computation, ACM Press, New York, 1995, pp. 195–207.

[7] R. M. Corless, S. M. Watt, L. Zhi, QR factoring to compute the GCD of univariate approximate polynomials, IEEE Trans. Signal Processing 52 (12) (2004) 3394–3402.

[8] D. K. Dunaway, A composite algorithm for finding zeros of real polynomials, Ph.D. thesis, Southern Methodist University, Texas (1972).

[9] I. Emiris, A. Galligo, H. Lombardi, Certified approximate univariate GCDs, J. Pure and Applied Algebra 117,118 (1997) 229–251.

[10] S. Ghaderpanah, S. Klasa, Polynomial scaling, SIAM J. Numer. Anal. 27 (1) (1990) 117–135.

[11] E. Kaltofen, Z. Yang, L. Zhi, Structured low rank approximation of a Sylvester matrix, in: D. Wang, L. Zhi (Eds.), Trends in Mathematics, Birkhäuser Verlag, Basel, Switzerland, 2006, pp. 69–83.

[12] B. Li, Z. Yang, L. Zhi, Fast low rank approximation of a Sylvester matrix by structured total least norm, J. Japan Soc. Symbolic and Algebraic Comp. 11 (2005) 165–174.

[13] Y. Tsai, R. T. Farouki, Algorithm 812: BPOLY: An object-orinted library for numerical algorithms for polynomials in Bernstein form, ACM Trans. Mathematical Software 27 (2) (2001) 267–296.

[14] J. R. Winkler, Structured matrix methods for the computation of multiple roots of a polynomial, Journal of Computational and Applied Mathematics 272 (2014) 449–467.

[15] J. R. Winkler, M. Hasan, A non-linear structure preserving matrix method for the low rank approximation of the Sylvester resultant matrix, Journal of Computational and Applied Mathematics 234 (2010) 3226–3242.

[16] J. R. Winkler, M. Hasan, An improved non-linear method for the computation of a structured low rank approximation of the Sylvester resultant matrix, Journal of Computational and Applied Mathematics 237 (1) (2013) 253–268.

[17] J. R. Winkler, M. Hasan, X. Y. Lao, Two methods for the calculation of the degree of an approximate greatest common divisor of two inexact polynomials, Calcolo 49 (2012) 241–267.

[18] J. R. Winkler, X. Y. Lao, M. Hasan, The computation of multiple roots of a polynomial, Journal of Computational and Applied Mathematics 236 (2012) 3478–3497.

[19] J. R. Winkler, N. Yang, Resultant matrices and the computation of the degree of an approximate greatest common divisor of two inexact Bernstein basis polynomials, Computer Aided Geometric Design 30 (4) (2013) 410–429.

[20] C. J. Zarowski, X. Ma, F. W. Fairman, QR-factorization method for computing the greatest common divisor of polynomials with inexact coefficients, IEEE Trans. Signal Processing 48 (11) (2000) 3042–3051.

[21] Z. Zeng, The approximate GCD of inexact polynomials. Part 1: A univariate algorithm, preprint (2004).