Structured matrix methods for computations on Bernstein basis polynomials

by

Ning Yang

A thesis submitted to the

Computer Science

in conformity with the requirements for

the degree of PhD

Sheffield University

England

January 2013

Abstract

This thesis considers structure preserving matrix methods for computations on Bernstein polynomials whose coefficients are corrupted by noise. The ill-posed operations of greatest common divisor computations and polynomial division are considered, and it is shown that structure preserving matrix methods yield excellent results.

With respect to greatest common divisor computations, the most difficult part is the computation of its degree, and several methods for its determination are presented. These are based on the Sylvester resultant matrix, and it is shown that a new form of the Sylvester resultant matrix in the modified Bernstein basis yields the best results. The Bézout resultant matrix in the modified Bernstein basis is also considered, and it is shown that the results from it are inferior to those from the Sylvester resultant matrix in the modified Bernstein basis.

Acknowledgements

Firstly, I sincerely appreciate that my supervisor, Joab Winkler, offers me a great opportunity to study in the University of Sheffield and work with him. During three years, his patient guidance, attentive care and consistent encouragement help me finish this tough and challenging work. Furthermore, his precise attitude to academic research and easy-going and cautiously optimistic personality are what I learnt from him, which is a precious treasure for my future career.

Secondly, I heartily acknowledge my parents. Your support in spirit gives me the courage and makes me strong when I am disappointed and depressed. You are my teacher and inspiration.

The last but not the least, I am thankful to my friends and relatives for their generous help.

Abbreviations and notation

GCD	...	greatest common divisor
AGCD	...	approximate greatest common divisor
LSE	...	least squares with equality
STLN	...	structured total least norm
SNTLN	...	structured nonlinear total least norm
$B(f,g)$	...	Bézout resultant matrix for the Bernstein polynomials $f(x)$ and $g(x)$
$S(f,g)$	...	The conventional form of Sylvester resultant matrix for the Bernstein polynomials $f(x)$ and $g(x)$
$S_k(f,g)$	...	The conventional form of Sylvester subresultant matrix of order k for the Bernstein polynomials $f(x)$ and $g(x)$
$S(f,g)Q$	...	The modified form of Sylvester resultant matrix for the Bernstein polynomials $f(x)$ and $g(x)$
$S_k(f,g)Q_k$	...	The modified form of Sylvester subresultant matrix of order k for the Bernstein polynomials $f(x)$ and $g(x)$
$\log x$	...	$\log_{10} x$
$\|\cdot\|$	...	$\|\cdot\|_2$

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Computer aided geometric design

Computer aided geometric design (CAGD) is a discipline that is concerned with constructing, representing and modeling free-form shapes of curves and surfaces. Before the establishment of CAGD, all design work was done manually, but the introduction of computers enabled this work to be automated. The techniques of CAGD enable the shapes of curves and surfaces to be designed to any precision. In addition, these techniques allow curves and surfaces to be manipulated in an intuitive way. This means that curves and surfaces can be manipulated easily and in a predictable manner, and knowledge of the underlying mathematics is not required. For example, the derivative of a Bézier curve can be calculated easily, and the geometry of curves and surfaces can be stored and reused. The techniques of CAGD are extensively applied in industry, and this is now considered.

The initial use of CAGD was in ship building and automobile design. The automobile design industry became interested in CAGD because the increasing commercial and

public demand urged automobile companies to accelerate the production process in which the prototype design had to be modified frequently in response to the feedback from the manufacturing process.

The techniques of CAGD occur in many industries and arise in all stages of the manufacturing cycle. Furthermore, they are also applied in more recent applications such as computer graphics, computer animation, geographic information system and robot path planning. More applications of the techniques of CAGD can be found in [17, 18].

1.2 The representation of curves and surfaces

This section considers the representation of curves and surfaces in CAGD. Three types of representation, explicit, implicit and parametric, will be discussed. In addition, the conversion between the implicit form and parametric form is often required because both representations are used in CAGD. The conversion between these two forms is also explained. The explicit form is a particular class of the implicit form.

1.2.1 Three types of representation of curves and surfaces

The first type of representation of curves and surfaces is the explicit representation. A curve is represented explicitly as $y = f(x)$. For example, $y = 3x + 1$ represents a straight line and $y = x^2 + 1$ represents a parabola. A surface is represented explicitly as $z = f(x, y)$, for example, $z = 4x + 2y - 6$ represents a plane. For the explicit representation, it is easy to find a point on the curve or surface and check whether a point lies on the curve or surface. However, some curves and surfaces can not

be represented using an explicit equation, for example, a unit circle centered at the origin $x^2 + y^2 = 1$. Solving y in terms of x, we obtain $y = \pm\sqrt{1 - x^2}$. Two explicit equations are required to represent this unit circle. As the example shows, the explicit representation is not suited to represent a closed curve and surface because for the closed curve and surface, one value of x corresponds to several different values of y, and multiple explicit equations are needed to represent it.

The second type of representation is the implicit representation. The curve and surface are represented in the form of $f(x, y) = 0$ and $f(x, y, z) = 0$ respectively. For example, $x^2 + y^2 - 1 = 0$ represents the unit circle and $x^2 + y^2 + z^2 - 1 = 0$ represents the unit sphere. We can easily check whether a point lies on the curve and surface represented in the implicit equation. In addition, an implicit representation can define a closed curve and surface. We can also determine if a point lies inside or outside the closed curve and surface by checking the sign of the implicit equation. Given the unit circle $x^2 + y^2 - 1 = 0$, the point (u, v) lies outside the circle if $u^2 + v^2 - 1 > 0$, and the point lies inside the circle if $u^2 + v^2 - 1 < 0$. Nevertheless, for the implicit representation, it is not easy to find a point on the curve and surface.

The third type of representation is the parametric representation. A plane curve is represented parametrically as $x = x(t)$ and $y = y(t)$. For example, the unit circle centered at the origin is expressed by two parametric equations

$$x(t) = \frac{2t}{1 + t^2} \qquad \text{and} \qquad y(t) = \frac{1 - t^2}{1 + t^2}.$$

A surface is represented in the form of $x = x(s, t)$, $y = y(s, t)$ and $z = z(s, t)$. For instance, the parametric equations

$$x(s, t) = \frac{2s}{1 + s^2 + t^2}, \qquad y(s, t) = \frac{2t}{1 + s^2 + t^2}, \qquad z(s, t) = \frac{1 - s^2 - t^2}{1 + s^2 + t^2},$$

represent the unit sphere. For the parametric representation, we can obtain a point on the curve or surface by evaluating coordinate functions at various values of parameters. Furthermore, the parametric representation can express a closed curve and surface. In addition, the parametric representation is easy to extend to higher dimension. If we want to express a space curve, we can simply add a coordinate function $z = z(t)$ and then $\big(x = x(t), y = y(t), z = z(t)\big)$ represents a space curve. However, it is difficult to check whether a point lies on the curve and surface expressed in the parametric equation.

The implicit and parametric representations are most commonly used in CAGD. From the above discussion, it is obvious that the parametric representation is convenient for obtaining points on a curve and surface, but the implicit representation is easy for determining whether a point lies on a curve and surface. Therefore, the conversion from one representation to the other is desired. In addition, the conversion is also motivated by the intersection problem in surface and solid modeling. Given that one surface is expressed parametrically by $\big(x = x(s,t), y = y(s,t), z = z(s,t)\big)$ and the other surface is represented implicitly by $f(x, y, z) = 0$, the intersection problem of these two surfaces can be simplified by substituting $x = x(s,t)$, $y = y(s,t)$ and $z = z(s,t)$ into $f(x, y, z) = 0$ to yield a single equation $f\big(x = x(s,t), y = y(s,t), z = z(s,t)\big) = 0$, which is the curve of intersection expressed implicitly using the parameters s and t. The conversion from the parametric to the implicit representation is called implicitization, and the conversion from the implicit to the parametric representation is called parameterization. These two conversions will be discussed in the next section.

1.2.2 Implicitization

Implicitization is the process of the conversion from the parametric form of a curve or surface to its implicit form. Two implicitization approaches, direct substitution and resultant, are introduced here.

Direct substitution: Direct substitution can be used to convert some curves and surfaces expressed parametrically to their implicit forms [2]. For example, given a curve represented by two parametric equations

$$x = t + 1 \qquad \text{and} \qquad y = t^2 + 3t + 1,$$

we can solve t in terms of x to obtain $t = x - 1$, and substitute it into $y = t^2 + 3t + 1$ to yield its implicit equation $x^2 + x - y - 1 = 0$. This method is suitable for the implicit forms of linear and quadratic curves. However, it can not be applied to curves of higher degree. A more general approach is to use the resultant of two polynomials.

Resultant: A resultant of a set of polynomials is an expression involving the coefficients of the polynomials such that the vanishing of the resultant is a necessary and sufficient condition for the set of polynomials to have a nontrivial common root [47]. Consider two polynomials

$$f(t) = a_m t^m + a_{m-1} t^{m-1} + \cdots + a_1 t + a_0,$$

and

$$g(t) = b_n t^n + b_{n-1} t^{n-1} + \cdots + b_1 t + b_0,$$

where $a_m \neq 0$ and $b_n \neq 0$. The Bézout resultant matrix requires $\deg f = \deg g$, and if we assume $m \geq n$, then $g(t)$ is padded with $m - n$ leading zero coefficients, that is

$$g(t) = b_m t^m + b_{m-1} t^{m-1} + \cdots + b_1 t + b_0,$$

where $b_m = 0, b_{m-1} = 0, \ldots, b_{n+1} = 0$. Let $C_k = (a_k, b_k), k = 0, 1, \cdots, m$, be the scalar cross product $C_i \times C_j = (a_i b_j - a_j b_i)$. The algorithm to construct the Bézout resultant matrix of $f(t)$ and $g(t)$ is derived in [29]:

$$B(f,g) = \begin{bmatrix} b_{0,0} & \cdots & b_{0,m-1} \\ \vdots & & \vdots \\ b_{m-1,0} & \cdots & b_{m-1,m-1} \end{bmatrix},$$

where the element of $B(f,g)$, $b_{i,j}$ is computed using the equation:

$$b_{i,j} = \sum_{\substack{p \geq \max(m-i, m-j) \\ p+q=2m-i-j-1}} C_p \times C_q. \tag{1.1}$$

The following theorem concerning the Bézout resultant of two polynomials is established in [29]:

Theorem 1.1. *The polynomials $f(t)$ and $g(t)$ have a common root if and only if* $\det B(f,g) = 0$*, where* $\det B(f,g)$ *is the determinant of $B(f,g)$.*

This theorem enables the Bézout resultant of two polynomials to be used for the implicitization process. Consider a curve defined by two parametric equations

$$x = a_m t^m + a_{m-1} t^{m-1} + \cdots + a_1 t + a_0,$$

and

$$y = b_m t^m + b_{m-1} t^{m-1} + \cdots + b_1 t + b_0.$$

To implicitize this curve, two auxiliary polynomials need to be created:

$$f_x(t) = a_m t^m + a_{m-1} t^{m-1} + \cdots + a_1 t + (a_0 - x),$$

and

$$g_y(t) = b_m t^m + b_{m-1} t^{m-1} + \cdots + b_1 t + (b_0 - y).$$

If the point (x, y) lies on the curve, the polynomials $f_x(t)$ and $g_y(t)$ have at least one

common root. Therefore, in terms of Theorem 1.1, $\det B(f_x, g_y) = 0$, and $\det B(f_x, g_y)$ is the resultant of $f_x(t)$ and $g_y(t)$. In particular, $\det B(f_x, g_y)$ is a function of x and y, and thus it is the implicit equation of the curve. We give an example to illustrate the implicitization of curve.

Example 1.1. Consider a curve defined by two parametric equations

$$x = 2t^2 + t + 3,$$
$$y = t^2 + 3t + 1.$$

Create two auxiliary polynomials

$$f_x(t) = 2t^2 + t + (3 - x),$$
$$g_y(t) = t^2 + 3t + (1 - y),$$

and thus $C_2 = (2, 1)$, $C_1 = (1, 3)$ and $C_0 = (3 - x, 1 - y)$. It follows from (1.1) that the Bézout resultant matrix of $f_x(t)$ and $g_y(t)$

$$\begin{aligned} B(f_x, g_y) &= \begin{bmatrix} C_2 \times C_1 & C_2 \times C_0 \\ C_2 \times C_0 & C_1 \times C_0 \end{bmatrix} \\ &= \begin{bmatrix} 5 & x - 2y - 1 \\ x - 2y - 1 & 3x - y - 8 \end{bmatrix}. \end{aligned}$$

The resultant of $f_x(t)$ and $g_y(t)$ is

$$\begin{aligned} \det B(f_x, g_y) &= 5(3x - y - 8) - (x - 2y - 1)^2 \\ &= -x^2 - 4y^2 + 4xy + 17x - 9y - 41. \end{aligned}$$

Therefore, the implicit form of the curve is $x^2 + 4y^2 - 4xy - 17x + 9y + 41 = 0$. □

In addition, the implicitization problem can be solved by Sylvester's dialytic expansion [47]. This approach considers all the individual monomials of a polynomial as independent variables. Therefore, t^3, t^2 and t are considered three independent variables, even though they are dependent. Multiplying the initial polynomials with well-chosen independent variables yields auxiliary equations such that the total number of equations is equal to the total number of independent variables.

For example, given two equations $a_2t^2+a_1t+a_0=0$ and $b_3t^3+b_2t^2+b_1t+b_0=0$, we initially have two equations with 4 independent variables: t^3, t^2, t and 1. Multiplying $a_2t^2+a_1t+a_0$ by t^2 and t, and multiplying $b_3t^3+b_2t^2+b_1t+b_0$ by t, we obtain 5 equations with 5 independent variables: t^4, t^3, t^2, t and 1. These 5 equations can be written as

$$\begin{bmatrix} a_2 & a_1 & a_0 & 0 & 0 \\ 0 & a_2 & a_1 & a_0 & 0 \\ 0 & 0 & a_2 & a_1 & a_0 \\ b_3 & b_2 & b_1 & b_0 & 0 \\ 0 & b_3 & b_2 & b_1 & b_0 \end{bmatrix} \begin{bmatrix} t^4 \\ t^3 \\ t^2 \\ t \\ 1 \end{bmatrix} = 0 \qquad \text{or} \qquad Ax = 0.$$

It follows that if two equations $a_2t^2+a_1t+a_0=0$ and $b_3t^3+b_2t^2+b_1t+b_0=0$ have a common root, $Ax=0$ must have a nontrivial solution. This means that the coefficient matrix A must be rank deficient and thus $\det A = 0$. Therefore, the resultant of these two equations is $\det A$. Example 1.2 illustrates the implicitization of curve using Sylvester's dialytic expansion.

Example 1.2. Consider the curve in Example 1.1, which is defined by two parametric

equations

$$x = 2t^2 + t + 3,$$

$$y = t^2 + 3t + 1.$$

Create two auxiliary polynomials

$$f_x(t) = 2t^2 + t + (3 - x),$$

$$g_y(t) = t^2 + 3t + (1 - y).$$

We have two equations $f_x(t) = 0$ and $g_y(t) = 0$ with 3 independent variables: t^2, t and 1. Multiplying $f_x(t)$ and $g_y(t)$ with t, we obtain 4 equations with 4 independent variables: t^3, t^2, t and 1. These 4 equations can be written as

$$\begin{bmatrix} 2 & 1 & 3-x & 0 \\ 0 & 2 & 1 & 3-x \\ 1 & 3 & 1-y & 0 \\ 0 & 1 & 3 & 1-y \end{bmatrix} \begin{bmatrix} t^3 \\ t^2 \\ t \\ 1 \end{bmatrix} = 0 \qquad \text{or} \qquad Ax = 0.$$

Similarly, if the point (x, y) lies on the curve, the polynomials $f_x(t)$ and $g_y(t)$ have at least one common root. Hence, there exists a nontrivial solution to satisfy $Ax = 0$ and thus $\det A = 0$. Since $\det A$ is a function of x and y, $\det A$ is the implicit form of the curve,

$$\begin{aligned} \det A &= 2\begin{vmatrix} 2 & 1 & 3-x \\ 3 & 1-y & 0 \\ 1 & 3 & 1-y \end{vmatrix} - \begin{vmatrix} 0 & 1 & 3-x \\ 1 & 1-y & 0 \\ 0 & 3 & 1-y \end{vmatrix} + (3-x)\begin{vmatrix} 0 & 2 & 3-x \\ 1 & 3 & 0 \\ 0 & 1 & 1-y \end{vmatrix} \\ &= x^2 + 4y^2 - 4xy - 17x + 9y + 41, \end{aligned}$$

and thus the implicit form of the curve is $x^2 + 4y^2 - 4xy - 17x + 9y + 41 = 0$, which is the same as Example 1.1. □

From Examples 1.1 and 1.2, the Bézout resultant and Sylvester's dialytic expansion yield the same implicit form of the curve, but the only difference is that the determinant of a 2×2 matrix is calculated using the Bézout resultant and the determinant of a 4×4 matrix is computed for Sylvester's dialytic expansion.

If a curve is defined by rational parametric equations, we can still adopt the Bézout resultant and Sylvester's dialytic expansion to implicitize the curve, but we have to rewrite the rational parametric equations. For example, given a curve expressed by two rational parametric equations

$$x = \frac{a_2t^2 + a_1t + a_0}{c_2t^2 + c_1t + c_0} \qquad \text{and} \qquad y = \frac{b_2t^2 + b_1t + b_0}{c_2t^2 + c_1t + c_0},$$

we rewrite these expressions as

$$f_x(t) = (c_2x - a_2)t^2 + (c_1x - a_1)t + (c_0x - a_0),$$

and

$$g_y(t) = (c_2y - b_2)t^2 + (c_1y - b_1)t + (c_0y - b_0),$$

and then implicitize the curve as mentioned above.

For the implicitization of surface, the following observation is stated in [14]:

Given a surface expressed by three parametric equations $x = x(s,t)$, $y = y(s,t)$ and $z = z(s,t)$, create three auxiliary equations

$$P_x(s,t) = x(s,t) - x,$$

$$P_y(s,t) = y(s,t) - y,$$

$$P_z(s,t) = z(s,t) - z.$$

Then we construct the matrix

$$P = \begin{bmatrix} P_x(s,t) & P_y(s,t) & P_z(s,t) \\ P_x(\alpha,t) & P_y(\alpha,t) & P_z(\alpha,t) \\ P_x(\alpha,\beta) & P_y(\alpha,\beta) & P_z(\alpha,\beta) \end{bmatrix},$$

where $\alpha, \beta \in \mathbb{R}$. If there exist $s = s'$ and $t = t'$, which will simultaneously satisfy $P_x(s',t') = P_y(s',t') = P_z(s',t') = 0$, the first row vanishes and thus the equation $\det P = 0$ is independent of the values of α and β. Also, when $s = \alpha$, the first two rows are identical and when $t = \beta$, the last two rows are identical. Therefore, if either $s = \alpha$ or $t = \beta$, $\det P = 0$. This means that $(s - \alpha)$ and $(t - \beta)$ are factors of $\det P$. Define

$$\delta = \frac{\det P}{(s-\alpha)(t-\beta)},$$

and $\delta = 0$ for any value of α and β if and only if $s = s'$ and $t = t'$. In addition, if the surface is of degree n in s and degree m in t, δ is of degree $n-1$ in s, $2m-1$ in t, $2n-1$ in α and $m-1$ in β. Hence, δ can be considered as a polynomial in α and β whose coefficients are polynomials in s and t:

$$\delta = \sum_{i=0}^{2n-1} \sum_{j=0}^{m-1} f_{i,j}(s,t)\alpha^i\beta^j.$$

δ is the sum of $2mn$ polynomials and $f_{i,j}(s,t)$ has $2mn$ terms. Since δ must vanish for any value of α and β if and only if $s = s'$ and $t = t'$, all of the $f_{i,j}(s',t')$ must

vanish. In particular, these $2mn$ polynomials can be written as:

$$\begin{matrix} \alpha^0\beta^0 \\ \vdots \\ \alpha^i\beta^j \\ \vdots \\ \alpha^{2n-1}\beta^{m-1} \end{matrix} \begin{bmatrix} A(0,0,0,0) & \cdots & A(0,0,k,l) & \cdots & A(\tbinom{0,0,n-1,}{2m-1}) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ A(i,j,0,0) & \cdots & A(i,j,k,l) & \cdots & A(\tbinom{i,j,n-1,}{2m-1}) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ A(\tbinom{2n-1,}{m-1,0,0}) & \cdots & A(\tbinom{2n-1,}{m-1,k,l}) & \cdots & A(\tbinom{2n-1,m-1,}{n-1,2m-1}) \end{bmatrix} \begin{bmatrix} s^0t^0 \\ \vdots \\ s^kt^l \\ \vdots \\ s^{n-1}t^{2m-1} \end{bmatrix} = 0,$$

where $A(i,j,k,l)$ is the coefficient of the term s^kt^l in polynomial $f_{i,j}(s,t)$, which is the coefficient of $\alpha^i\beta^j$. The formula for computing $A(i,j,k,l)$ is found in [14]. Since there exists a solution $s = s'$ and $t = t'$, the determinant of the matrix must vanish. Therefore, the resultant is the determinant of the matrix, and calculating the determinant of the matrix yields the implicit form of the surface.

If a surface is defined by three rational parametric equations

$$\begin{aligned} D &= f(d|s^m,t^n), \\ x &= f(a|s^m,t^n)/D, \\ y &= f(b|s^m,t^n)/D, \\ z &= f(c|s^m,t^n)/D, \end{aligned}$$

where $f(k|s^m,t^n)$ is a polynomial with degree m in variable s, degree n in variable t and the coefficients $k_0, k_1, \cdots$. We rewrite

$$\begin{aligned} xD - f(a|s^m,t^n) &= 0, \\ yD - f(b|s^m,t^n) &= 0, \\ zD - f(c|s^m,t^n) &= 0, \end{aligned}$$

then we implicitize the surface as before.

1.2.3 Parameterization

After solving the implicitization problem, we now consider the conversion of the implicit form of a curve and surface to its parametric form, parameterization. Every parametric curve and surface has an implicit form. However, the converse is not true, and some curves and surfaces expressed implicitly by polynomials and rational functions can not be represented in the parametric form. Therefore, the parameterization includes two distinct parts:

1. Determine if a curve or surface has a parametric representation;

2. If it is representable in the parametric form, find its parametric representation.

For the first part, the following theorem is used to determine if a curve has a parametric representation [10]:

Theorem 1.2. *An algebraic curve has a parametric rational polynomial representation if and only if the curve has genus zero.*

The genus is calculated using the following formula:

$$\text{genus} = \frac{(n-1)(n-2)}{2} - \sum_i \frac{r_i(r_i-1)}{2},$$

where n is the degree of the algebraic curve and r_i is the multiplicity of the ith multiple point. A multiple point is the point on a curve through which two or more branches of the curve pass and its multiplicity is the number of branches involved. More details can be found in [55].

Since every quadric curve and surface has a parametric representation, the first part is satisfied, and we only consider finding the parametric forms of a quadric curve and surface. Let us consider a circle defined by a implicit equation $x^2 + y^2 - R^2 = 0$. We

first factor it into

$$x \cdot x = (R + y)(R - y). \tag{1.2}$$

Rearranging these terms and introducing the parameter t, we obtain

$$t = \frac{x}{R + y} = \frac{R - y}{x}. \tag{1.3}$$

Solving this equation for x and y in terms of t yields $x - ty = Rt$ and $tx + y = R$. From these two equations, we get the parametric form of the circle

$$x = \frac{2Rt}{1 + t^2} \qquad \text{and} \qquad y = \frac{R(1 - t^2)}{1 + t^2}. \tag{1.4}$$

The parameterization problem can be solved using another approach. We first select a fixed point $(R, 0)$, and a line passing through this point is

$$y - xt + Rt = 0, \tag{1.5}$$

where the parameter t is the slope of the line. Substituting $t(x - R)$ for y in $x^2 + y^2 - R^2 = 0$, we obtain

$$x^2 + t^2(x - R)^2 - R^2 = 0,$$

and solving for x gives

$$x = R \qquad \text{and} \qquad x = \frac{R(1 - t^2)}{1 + t^2}.$$

Substituting $x = R$ into (1.5) yields

$$y = 0.$$

These two equations $x = R$ and $y = 0$ are not the parametric form of the circle because they only represent one point on the circle. Substituting $x = \frac{R(1-t^2)}{1+t^2}$ into (1.5) yields

$$y = \frac{2Rt}{1 + t^2}.$$

Equations $x = \frac{R(1-t^2)}{1+t^2}$ and $y = \frac{2Rt}{1+t^2}$, which are functions of t, trace out the circle, and therefore they are the parametric form of the circle. These are the same as (1.4) with x and y reversed.

This approach can be applied to certain higher degree curves but the selection of the fixed point must be considered. In particular, the selected fixed point must be singular of the right multiplicity such that except at the fixed point, the line intersects the curve at only one other point.

For the parameterization of surface, consider a sphere: $x^2 + y^2 + z^2 - R^2 = 0$. An auxiliary variable w is introduced:

$$x^2 + y^2 = w^2 \qquad \text{and} \qquad w^2 + z^2 = R^2.$$

Following (1.2), (1.3) and (1.4), we solve these two equations and obtain

$$x = \frac{2ws}{1+s^2}, \qquad y = \frac{w(1-s^2)}{1+s^2}, \qquad w = \frac{2Rt}{1+t^2}, \qquad z = \frac{R(1-t^2)}{1+t^2}.$$

Substituting $w = \frac{2Rt}{1+t^2}$ into x and y yields the parametric representation of the sphere

$$x = \frac{4Rst}{(1+s^2)(1+t^2)}, \qquad y = \frac{2R(1-s^2)t}{(1+s^2)(1+t^2)}, \qquad z = \frac{R(1-t^2)}{1+t^2}.$$

1.3 Summary

This chapter introduced basic ideas about the techniques of CAGD, and its influence and applications have been emphasized. Furthermore, this chapter considered three types of representation of curves and surfaces in CAGD system. The conversion between two most widely used forms, the implicit and parametric forms, was also discussed. The next chapter will introduce one important technique of CAGD, the Bézier curve, which is represented parametrically.

Chapter 2

Bézier curves

In 1959, Paul de Faget de Casteljau began to develop a new method for the design of curves with the aim of making their design intuitive, in order to facilitate interactive design. Meanwhile, another mathematician, Pierre Bézier also realized the importance of the computer representation of curves and developed a system in which a curve is represented as the intersection of two elliptic cylinders. Although his idea is different from that involved in the de Casteljau algorithm, the result is identical to the curve constructed using the de Casteljau algorithm. Pierre Bézier's work was extensively published and the curve was then named the Bézier curve.

The new concept in the Bézier curve is the use of its control polygon. Since the curve follows the control polygon in an intuitive way, we can define and modify the control polygon instead of constructing and changing the curve directly. The de Casteljau algorithm is introduced in the next section. Then, the parametric form of the Bézier curve and its important properties are considered.

2.1 The de Casteljau algorithm

The de Casteljau algorithm can be illustrated by a simple construction of a parabola. Suppose we have b_0, b_1, $b_2 \in \mathbb{R}^3$ and $t \in \mathbb{R}$. Construct

$$\begin{aligned} b_0^1(t) &= (1-t)b_0 + tb_1, \\ b_1^1(t) &= (1-t)b_1 + tb_2, \\ b_0^2(t) &= (1-t)b_0^1(t) + tb_1^1(t). \end{aligned}$$

Inserting the first two equations into the third one, we obtain

$$b_0^2(t) = (1-t)^2 b_0 + 2t(1-t)b_1 + t^2 b_2.$$

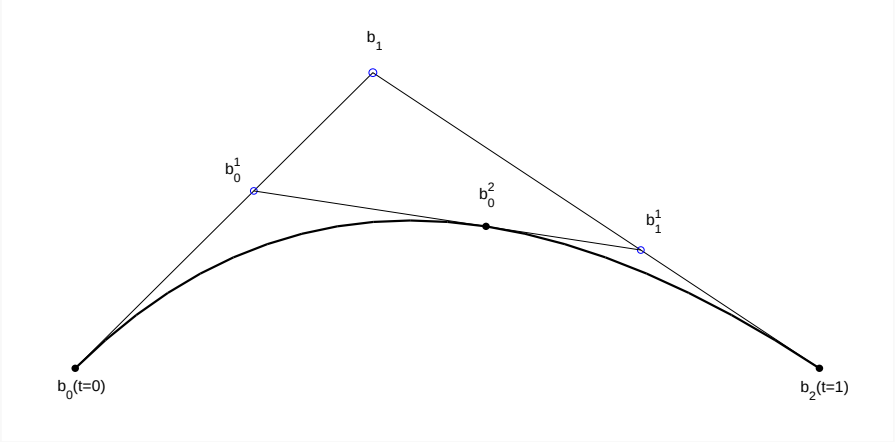


Figure 2.1: The parabola generated by repeated linear interpolation for $t \in [0, 1]$.

As t varies from $-\infty$ to $+\infty$, $b_0^2(t)$ traces a parabola. For t between 0 and 1, $b_0^2(t)$ is inside the triangle formed by b_0, b_1, b_2. This is illustrated in Figure 2.1.

The generation of the parabola involves repeated linear interpolations. This process

can be generalized to construct a polynomial curve with arbitrary degree n. The de Casteljau algorithm is as follows [19]:

de Casteljau algorithm:

Given b_0, b_1, b_2, $\cdots$, $b_n \in \mathbb{R}^3$ and $t \in \mathbb{R}$,

set

$$b_i^r(t) = (1-t)b_i^{r-1}(t) + tb_{i+1}^{r-1}(t), \tag{2.1}$$

where $r = 1, \cdots, n$ and $i = 0, \cdots, n-r$. Set $b_i^0(t) = b_i$ and $b^n(t) = b_0^n(t)$, then $b_0^n(t)$ is the point with parameter t on the Bézier curve b^n.

The vertices b_0, b_1, $\cdots$, b_n are called the control points and the polygon formed by b_0, b_1, $\cdots$, b_n is called the control polygon.

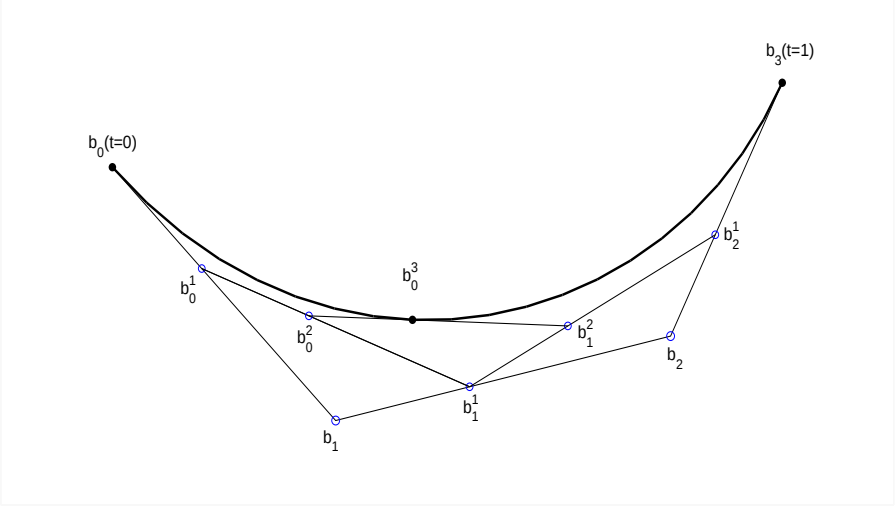


Figure 2.2: The cubic Bézier curve b^3 generated by de Casteljau algorithm for $t \in [0, 1]$.

Figure 2.2 illustrates a cubic Bézier curve b^3 as t varies from 0 to 1. For t between 0 and 1, b^3 lies inside the control polygon formed by b_0, b_1, b_2, b_3.

The Bézier curve with arbitrary degree n can be generated by the de Casteljau algorithm, but it is desirable to represent the Bézier curve in polynomial form to facilitate more detailed theoretical research on it. This requires the Bernstein basis, which is described in the next section.

2.2 Bernstein basis functions

A Bézier curve can be expressed in terms of the Bernstein basis functions. The Bernstein basis functions of degree n are defined explicitly by

$$B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}, \qquad i = 0, \cdots, n, \tag{2.2}$$

where $\binom{n}{i}$ is binomial coefficient. In particular,

$$B_0^0(t) \equiv 1,$$

and

$$B_i^n(t) \equiv 0, \qquad \text{for} \qquad i \notin \{0, \cdots, n\}.$$

Some properties of the Bernstein basis functions will be examined here because they are important for the development of properties of a Bézier curve.

Partition of unity: For any value of t,

$$\sum_{i=0}^{n} B_i^n(t) = 1. \tag{2.3}$$

The proof is

$$1 = [t + (1-t)]^n = \sum_{i=0}^{n} \binom{n}{i} t^i (1-t)^{n-i} = \sum_{i=0}^{n} B_i^n(t).$$

Symmetry: It is easy to verify $B_i^n(t) = B_{n-i}^n(1-t)$ from (2.2).

Nonnegativity: For $t \in [0, 1]$, each Bernstein basis function is nonnegative. Since

t and $(1-t)$ are non-negative for $t \in [0,1]$, and the binomial coefficient $\binom{n}{i}$ is non-negative, the non-negative property follows.

Recursion: The Bernstein basis function with degree n is equal to the sum of two Bernstein basis functions with degree $n-1$,

$$B_i^n(t) = (1-t)B_i^{n-1}(t) + tB_{i-1}^{n-1}(t). \tag{2.4}$$

The proof of (2.4) is

$$\begin{aligned} B_i^n(t) &= \binom{n}{i} t^i (1-t)^{n-i} \\ &= \binom{n-1}{i} t^i (1-t)^{n-i} + \binom{n-1}{i-1} t^i (1-t)^{n-i} \\ &= (1-t)\binom{n-1}{i} t^i (1-t)^{n-i-1} + t\binom{n-1}{i-1} t^{i-1} (1-t)^{n-i} \\ &= (1-t)B_i^{n-1}(t) + tB_{i-1}^{n-1}(t). \end{aligned}$$

One important application of the Bernstein basis functions is the definition of a Bézier curve. The point with position vector $b_0^n(t)$ on a Bézier curve with degree n is a parametric function of the following form:

$$b_0^n(t) = \sum_{i=0}^{n} b_i B_i^n(t), \tag{2.5}$$

where b_i is the vector of the control point and $B_i^n(t)$ is the ith Bernstein basis function. The properties of a Bézier curve can be derived in terms of the de Casteljau algorithm and Bernstein basis functions. This will be addressed in the next section.

2.3 The properties of a Bézier curve

In this section, some properties of a Bézier curve are examined using the de Casteljau algorithm and properties of the Bernstein basis functions [19, 25, 28].

Affine invariance: The Bézier curve is invariant under an affine map, that is, let Φ

be an affine map, then

$$\Phi\left(\sum_{i=0}^{n} b_i B_i^n(t)\right) = \sum_{i=0}^{n} \Phi(b_i) B_i^n(t).$$

The de Casteljau algorithm involves a sequence of repeated linear interpolations and the linear interpolation is invariant under an affine map. Since the Bézier curve is generated by de Casteljau algorithm, the Bézier curve is invariant under affine map. This property can also be verified in terms of Bernstein basis functions. The barycentric combination

$$b = \sum_{i=0}^{n} \alpha_i b_i,$$

where $b_i \in \mathbb{R}^3$ and $\alpha_0 + \cdots + \alpha_n = 1$, is invariant under affine map. From (2.3) and (2.5), the Bézier curve is the barycentric combination of the control points, and it is therefore invariant under an affine map.

This property means the following two processes yield the identical Bézier curve. Let Φ be affine map:

1. Compute the Bézier curve from the control points $\{b_0, b_1, \ldots, b_n\}$ and then apply the affine map to the Bézier curve;

2. Apply the affine map to the control points $\{b_0, b_1, \ldots, b_n\}$ to obtain new control points $\{\Phi(b_0), \Phi(b_1), \ldots, \Phi(b_n)\}$ and then compute the Bézier curve from the new control points.

A practical example can illustrate the function at this property. Suppose we want to generate a cubic Bézier curve by evaluating 100 points and rotate it using an affine map. Two processes can be implemented:

1. Evaluate 100 points to generate the Bézier curve and then rotate each of the 100 points.

2. Rotate the control points and then evaluate the resulting function at 100 points to generate the Bézier curve.

Due to the affine invariance property, these two processes yield the identical Bézier curve but the first process needs 100 rotations, and the second only needs 4 rotations.

Invariance under affine parameter transformations: In most cases, the Bézier curve is defined over interval $[0, 1]$. However, the Bézier curve can be defined over any arbitrary interval $[a, b]$. If $a \leq u \leq b$, the generalized de Casteljau algorithm is given by

$$b_i^r(u) = \frac{b-u}{b-a} b_i^{r-1}(u) + \frac{u-a}{b-a} b_{i+1}^{r-1}(u),$$

and the generalized Bernstein form of Bézier curve is

$$b_0^n(t) = \sum_{i=0}^{n} b_i B_i^n(t) = \sum_{i=0}^{n} b_i B_i^n \left(\frac{u-a}{b-a} \right).$$

Endpoint interpolation: The Bézier curve passes through b_0 and b_n. In terms of the de Casteljau algorithm, when $t = 0$, $b_i^r = b_i^{r-1}$, thus $b_0^n = b_0^{n-1} = \cdots = b_0^1 = b_0^0 = b_0$ and when $t = 1$, $b_i^r = b_{i+1}^{r-1}$, thus $b_0^n = b_1^{n-1} = \cdots = b_{n-1}^1 = b_n^0 = b_n$. From the Bernstein basis functions, $b_0^n(0) = b_0$ and $b_0^n(1) = b_n$. The endpoints of the Bézier curve are two important points. For example, for the design of an escalator using the Bézier curve, it is essential to create a Bézier curve that connects entrance and exit points of the escalator accurately. This property enables us to have direct control on them.

Symmetry: The control points $b_0, b_1, \cdots, b_n$ and $b_n, b_{n-1}, \cdots, b_0$ yield the same Bézier curve. The only difference is that the direction of the Bézier curve is revered. Since $B_i^n(t) = B_{n-i}^n(1-t)$,

$$\sum_{i=0}^{n} b_i B_i^n(t) = \sum_{i=0}^{n} b_{n-i} B_i^n(1-t).$$

This property means that if we want to reverse the direction of the Bézier curve, we first reverse the order of the control points and then generate the Bézier curve.

Invariance under barycentric combinations: For $\alpha + \beta = 1$, we obtain

$$\sum_{i=0}^{n}(\alpha b_i + \beta c_i)B_i^n(t) = \alpha\sum_{i=0}^{n} b_i B_i^n(t) + \beta\sum_{i=0}^{n} c_i B_i^n(t).$$

This property allows us to generate the weighted average of two Bézier curves in two ways:

1. Compute the weighted average of corresponding points on the Bézier curves;
2. Compute the weighted average of corresponding control points and then generate the Bézier curve.

Linear precision: If the control points $b_1, \cdots, b_{n-1}$ are uniformly distributed on the straight line joining control points b_0 and b_n, the Bézier curve generated using these control points is a straight line from b_0 to b_n. The proof of this property needs the relation

$$\sum_{i=0}^{n}\frac{i}{n}B_i^n(t) = t. \tag{2.6}$$

This relation is verified as following:

$$\begin{aligned}
t &= t\times[t+(1-t)]^{n-1}\\
&= t\left[\binom{n-1}{0}t^0(1-t)^{n-1} + \binom{n-1}{1}t^1(1-t)^{n-2} + \cdots + \binom{n-1}{n-1}t^{n-1}(1-t)^0\right]\\
&= \binom{n-1}{0}t^1(1-t)^{n-1} + \binom{n-1}{1}t^2(1-t)^{n-2} + \cdots + \binom{n-1}{n-1}t^n(1-t)^0\\
&= \frac{1}{n}\binom{n}{1}t^1(1-t)^{n-1} + \frac{2}{n}\binom{n}{2}t^2(1-t)^{n-2} + \cdots + \frac{n}{n}\binom{n}{n}t^n(1-t)^0\\
&= \sum_{i=0}^{n}\frac{i}{n}B_i^n(t).
\end{aligned}$$

Thus, the linear precision property can be proved using (2.6). Suppose the control points $b_1, \cdots, b_{n-1}$ are uniformly distributed on the straight line joining control points b_0 and b_n:

$$b_i = \left(1 - \frac{i}{n}\right) b_0 + \frac{i}{n} b_n, \qquad i = 0, \cdots, n,$$

then the Bézier curve b^n generated using this set of control points is

$$\begin{aligned} b^n &= \sum_{i=0}^{n} b_i B_i^n(t) \\ &= \sum_{i=0}^{n} \left((1 - \frac{i}{n}) b_0 + \frac{i}{n} b_n \right) B_i^n(t) \\ &= b_0 \sum_{i=0}^{n} B_i^n(t) - b_0 \sum_{i=0}^{n} \frac{i}{n} B_i^n(t) + b_n \sum_{i=0}^{n} \frac{i}{n} B_i^n(t) \\ &= b_0 + (b_n - b_0)t. \end{aligned}$$

Since $b^n = b_0 + (b_n - b_0)t$, the Bézier curve b^n is a straight line joining the two endpoints b_0 and b_n. Figure 2.3 illustrates this property.

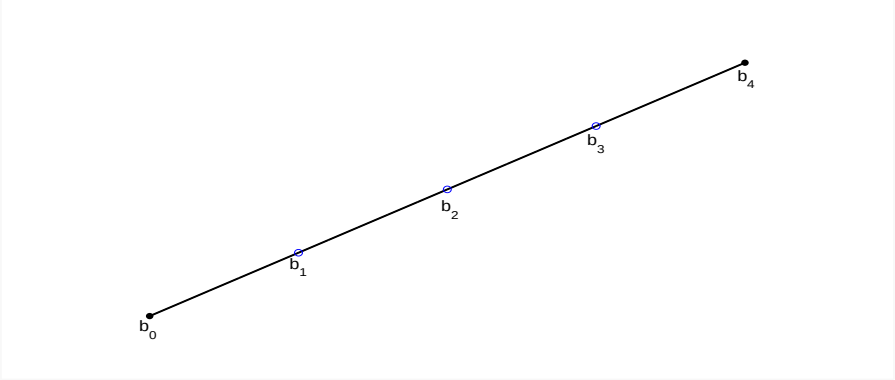


Figure 2.3: Linear precision property: The Bézier curve b^4 generated using uniformly distributed control points for $t \in [0, 1]$.

Convex hull: For $t \in [0,1]$, the Bézier curve lies in the convex hull of the control polygon. We give the definition of the convex hull as follows.

First, the convex set for a set of points is a set that contains the line segment between any two points in the set. Then, the convex hull is the smallest convex set. In particular, the convex hull for a set of points $x_0, x_1, \cdots, x_n$ is the set of all convex combinations of points $x_0, x_1, \cdots, x_n$. The convex combination of points $x_0, x_1, \cdots, x_n$ is

$$\alpha_0 x_0 + \alpha_1 x_1 + \cdots + \alpha_n x_n,$$

where $\alpha_i \geq 0$ and $\alpha_0 + \alpha_1 + \cdots + \alpha_n = 1$.

The proof of this property is straightforward. Remember the Bézier curve $b^n = \sum_{i=0}^{n} b_i B_i^n(t)$. For $t \in [0,1]$, the Bernstein basis polynomial $B_i^n(t)$ is nonnegative and from equation (2.3), $\sum_{i=0}^{n} B_i^n(t) = 1$. Therefore, for $t \in [0,1]$, the point on the Bézier curve is the convex combination of control points contained in the convex hull. Figure 2.4 illustrates the convex hull property.

The convex hull property guarantees that the planar control polygon always generates the planar Bézier curve.

Pseudolocal control: The shape change of the Bézier curve follows the movement of the control points.

This is the most important property of the Bézier curve. First of all, we can change the shape of the Bézier curve by moving the control points instead of changing every point on the Bézier curve. Furthermore, we can change the shape of the Bézier curve in a predictable and intuitive way because the Bézier curve follows the control points. In particular, moving one control point changes the shape of the whole Bézier curve, which is called global control. This is in contrast with the local control of the

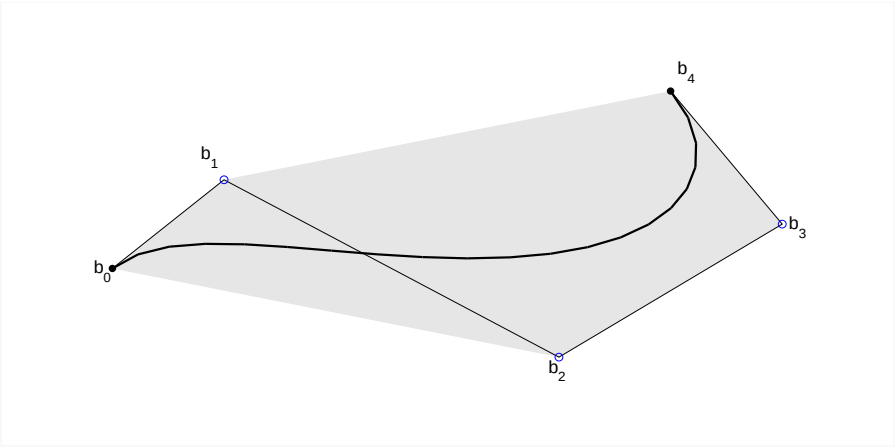
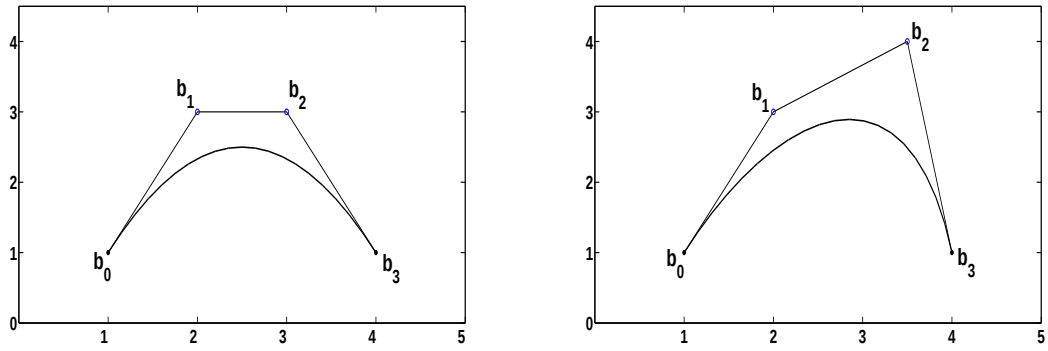


Figure 2.4: Convex hull property: The convex hull of the control polygon is shaded. For $t \in [0, 1]$, the Bézier curve b^4 lies in the convex hull of the control polygon.

B-splines, for which moving one control point alters only part of the curve. As Figure 2.5 shows, if we move one control point b_2 from $(3, 3)$ to $(3.5, 4)$, the whole Bézier curve follows the movement of the control point b_2.

Variation diminishing: If a straight line intersects the Bézier curve n times, then the line intersects its control polygon at least n times. In other words, the Bézier curve can intersect a straight line no more times than its control polygon does. Figure 2.6 shows this property.

The properties of Bézier curves are useful for solving computation problems on Bézier curves. This is demonstrated in the next section, in which one practical computation problem associated with Bézier curves, the intersection problem of Bézier curves, is considered.



(a) (b)

Figure 2.5: Pseudolocal control property: (a) $t \in [0, 1]$, the Bézier curve b^3 employs control points $b_0(1, 1)$, $b_1(2, 3)$, $b_2(3, 3)$ and $b_3(4, 1)$; and (b) $t \in [0, 1]$, the Bézier curve b^3 employs control points $b_0(1, 1)$, $b_1(2, 3)$, $b_2(3.5, 4)$ and $b_3(4, 1)$.

2.4 Intersection problem of Bézier curves

The intersection problem of Bézier curves is a fundamental computation problem in CAGD. Three major approaches for computing intersections of Bézier curves are Bézier subdivision [35, 66], interval subdivision [34, 45] and implicitization [48]. Bézier subdivision and interval subdivision use the geometric property of curves, and implicitization is an algebraic approach. The following sections consider these approaches.

2.4.1 Bézier subdivision

Bézier subdivision relies on the de Casteljau algorithm for subdividing a Bézier curve and uses the convex hull property of a Bézier curve to determine the intersection points.

Subdivision for a Bézier curve was introduced by de Casteljau [13], and proved by E. Staerk [50]. Subdivision of Bézier curve is the process of splitting a Bézier curve

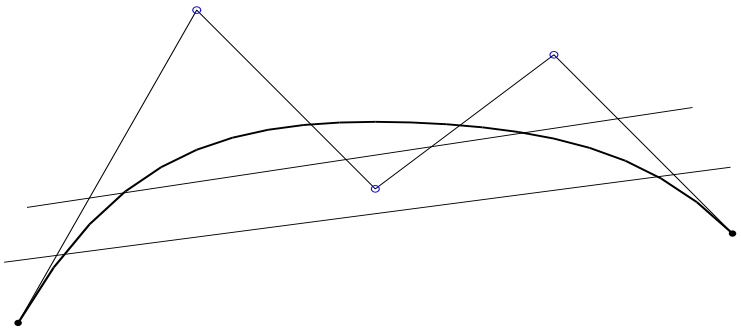


Figure 2.6: Variation diminishing property: Two straight lines intersect one Bézier curve and its control polygon.

into two segments, and each segment forms its own control polygon. It is seen from Figure 2.7 that a cubic Bézier curve is split to two segments, and for each segment, the resulting set of control points forms the control polygon of the segment.

In terms of the convex hull property, a Bézier curve lies entirely within the convex hull defined by its control points. Hence, if the convex hulls of two curves do not overlap, two curves do not intersect. Therefore, whether two Bézier curves intersect can be determined by checking if their convex hulls overlap.

Bézier subdivision involves repeated subdivisions and uses the convex hull property to compute intersection points of Bézier curves. In particular, given two Bézier curves, Bézier subdivision begins by comparing the convex hulls of two curves. If they do not overlap, two curves do not intersect. Otherwise, a subdivision algorithm splits each curve into two segments, and each segment forms its own control polygon. Then, the convex hulls of segments are checked for overlap, and segments that do not overlap are rejected. The overlapped segments are then split into new segments by subdivision

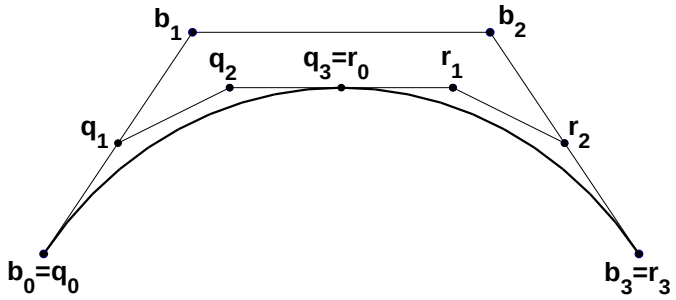


Figure 2.7: Subdivision for a cubic Bézier curve with control points b_0, b_1, b_2 and b_3, for $t \in [0, 1]$. The points q_0, q_1, q_2 and q_3 are the control points of the segment of the original curve from $t = 0$ to $t = \frac{1}{2}$, and the points r_0, r_1, r_2 and r_3 are the control points of the segment of the original curve from $t = \frac{1}{2}$ to $t = 1$.

algorithm, and the convex hulls of new segments are checked for overlap. This process continues until the new curve segment is approximately linear under certain tolerance. If two approximately linear segments overlap, their point of intersection is accepted as an intersection of two curves.

2.4.2 Interval subdivision

Interval subdivision is similar to Bézier subdivision. In particular, given two Bézier curves, each curve is preprocessed to determine its characteristic points such as vertical and horizontal tangents. The curve is then split at characteristic points into intervals, and every interval has characteristic points at the endpoints. For each interval, a rectangle whose diagonal is defined by two endpoints of the interval is computed such that the rectangle bounds the interval completely. This preprocess is

illustrated in Figure 2.8.

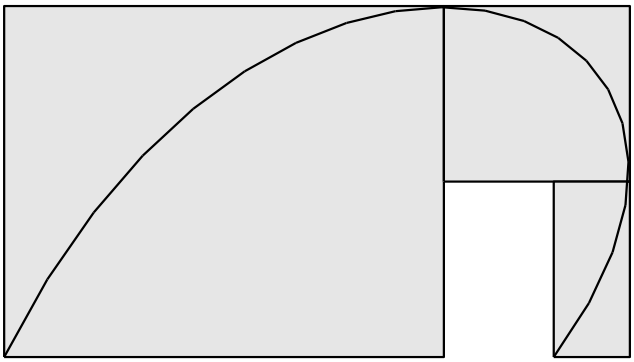


Figure 2.8: Interval preprocess

Because each interval lies entirely within its bounding rectangle, whether two intervals intersect can be determined by comparing their bounding rectangles. If their bounding rectangles do not overlap, the two intervals do not intersect. If their bounding rectangles overlap, the intervals are subdivided at the middle value of interval, and the bounding rectangle of each subinterval is computed for overlap checking. As this procedure proceeds, each iteration rejects intervals which do not contain intersection points. The algorithm terminates when the new interval is approximated by a straight line within a specified tolerance. If two approximately linear intervals overlap, their intersection point is considered an intersection point of two curves. More details and examples about solving the intersection problem of Bézier curves using interval subdivision are shown in [34, 45].

2.4.3 Implicitization

As stated earlier, a curve represented parametrically has a corresponding implicit form, and the Bézier curve is the parametric curve. In order to solve the intersection problem of two Bézier curves, one Bézier curve is implicitized to obtain its implicit form using resultant matrices [46], and then the parametric form of the other Bézier curve is substituted to its implicit form to yield a single equation. The intersection problem is solved by computing the roots of this equation. In this case, the intersection problem of Bézier curves is reduced to finding solutions of a univariate polynomial equation. Some approaches to compute solutions of polynomial equations are Gröbner bases algorithm [7], homotopy method [27], interval arithmetic [49] and iterative methods.

2.5 Summary

In this chapter, one important curve representation in CAGD, the Bézier curve, was introduced. In particular, the de Casteljau algorithm is a process to construct the Bézier curve with a specified set of control points, and the Bézier curve constructed in this way can be represented parametrically by the Bernstein basis functions. Some important properties of the Bézier curve allow it to be easily manipulated in an intuitive way and make the computations associated with it simplified.

Since the Bernstein basis functions are the parametric expressions of a Bézier curve, computation problems involving the Bézier curve are equivalent to manipulating polynomials defined in the Bernstein basis. In addition, it is demonstrated in [21] that

the conversion between a Bernstein basis polynomial and its power basis form is ill-conditioned. Furthermore, in the interval $[0, 1]$, the Bernstein basis is computationally more stable than the power basis [20], and thus numerical computations performed in the Bernstein basis should be considered.

It is noted that resultant matrices are widely applied in CAGD. As stated earlier, they are used for implicitization and intersection problem. Another important problem in CAGD is to compte the greatest common divisor of two polynomials, which can also be solved using resultant matrices. This issue is discussed in the next chapter.

Chapter 3

Greatest common divisor computation

The greatest common divisor (GCD) of two polynomials is a polynomial with the highest degree that divides both polynomials. The calculation of the GCD of polynomials defined in the power basis is usually considered, and its applications include image processing [37, 38], control theory [51], computing theory [1] and the computation of the roots of a polynomial [69]. However, because the Bernstein basis function is the natural choice for the Bézier curve, and the computation performed in the Bernstein basis has computational advantages, it is desirable to consider the computation of the GCD of polynomials defined in the Bernstein basis. The calculation of the GCD of Bernstein polynomials is essential and arises in many applications, including robotics motion planning [8], computer aided geometric design (CAGD) [31, 44] and computer vision [23, 40].

The major algorithms to compute the GCD of polynomials are Euclid's algorithm and resultant matrices. When resultant matrices are used to compute the GCD of

Bernstein polynomials, the form of resultant matrices expressed in the Bernstein basis must be developed. The rest of the chapter will introduce these algorithms.

3.1 Euclid's algorithm

Euclid's algorithm is known as an efficient method for computing the GCD of two polynomials symbolically and it involves a repeated sequence of polynomial divisions [54]. Given two polynomials $\hat{f}(x)$ with degree m and $\hat{g}(x)$ with degree n, where $m \geq n$, we assign $\o_0(x) = \hat{f}(x)$ and $\o_1(x) = \hat{g}(x)$, and then compute polynomials $\o_2(x), \cdots, \o_m(x)$ through the sequence

$$\begin{aligned}
\o_0(x) &= \o_1(x)q_1(x) + \o_2(x), \\
\o_1(x) &= \o_2(x)q_2(x) + \o_3(x), \\
&\cdots \\
\o_{r-1}(x) &= \o_r(x)q_r(x) + \o_{r+1}(x), \\
&\cdots \\
\o_{m-1}(x) &= \o_m(x)q_m(x), \qquad (3.1)
\end{aligned}$$

where $q_r(x)$ and $\o_{r+1}(x)$ are the quotient and remainder of dividing $\o_{r-1}(x)$ by $\o_r(x)$. The division sequence continues until a remainder $\o_{m+1}(x)$ vanishes and the GCD of $\hat{f}(x)$ and $\hat{g}(x)$ is $\o_m(x)$.

Example 3.1. Consider two Bernstein polynomials

$$\begin{aligned}
\hat{f}(x) &= \binom{3}{0}(1-x)^3 + \frac{3}{2}\binom{3}{1}(1-x)^2x + 2\binom{3}{2}(1-x)x^2 + 2\binom{3}{3}x^3 \\
&= (x-2)(x+1)^2,
\end{aligned}$$

and

$$\begin{aligned}\hat{g}(x) &= \binom{2}{0}(1-x)^2 + \frac{4}{3}\binom{2}{1}(1-x)x + \frac{4}{3}\binom{2}{2}x^2 \\ &= (x-3)(x+1),\end{aligned}$$

whose GCD is $\binom{1}{0}(1-x) + 2\binom{1}{1}x$.

Applying Euclid's algorithm to $\hat{f}(x)$ and $\hat{g}(x)$ yields the division sequence

$$\begin{aligned}\hat{f}(x) &= \hat{g}(x) \times \overbrace{\left(3\binom{1}{0}(1-x) + \frac{9}{2}\binom{1}{1}x\right)}^{q_1(x)} + \overbrace{\left(-2\binom{1}{0}(1-x) - 4\binom{1}{1}x\right)}^{\phi_2(x)}, \\ \hat{g}(x) &= \overbrace{\left(-2\binom{1}{0}(1-x) - 4\binom{1}{1}x\right)}^{\phi_2(x)} \times \overbrace{\left(-\frac{1}{2}\binom{1}{0}(1-x) - \frac{1}{3}\binom{1}{1}x\right)}^{q_2(x)}.\end{aligned}$$

Since the remainder of the second equation is equal to zero, its divisor is the GCD of $\hat{f}(x)$ and $\hat{g}(x)$, which is correct because $-2\binom{1}{0}(1-x) - 4\binom{1}{1}x$ is proportional to $\binom{1}{0}(1-x) + 2\binom{1}{1}x$. □

The next section considers an algorithm using resultant matrices for computing the GCD of Bernstein polynomials.

3.2 Bézout resultant matrix

As mentioned before, two resultant matrices, the Bézout and Sylvester resultant matrices, are used to solve the implicitization and intersection problems. Furthermore, the GCD of $\hat{f}(x)$ and $\hat{g}(x)$ can also be computed using the resultant matrices [3]. In this section and Section 3.3, the construction of the Bézout and Sylvester resultant matrices for Bernstein polynomials is developed and the algorithm that uses them to compute the GCD of Bernstein polynomials is explained. The formulae that unite the Bézout and Sylvester resultant matrices for Bernstein polynomials are established

in [58].

This section considers the Bézout resultant matrix of two polynomials expressed in the Bernstein basis and the properties that enable it to be used for computing the GCD of two Bernstein polynomials. The Sylvester resultant matrix defined in the Bernstein basis is discussed in Section 3.3.

The following construction of the Bézout resultant matrix of Bernstein polynomials is presented in [5]:

Consider one Bernstein polynomial $\hat{f}(x)$ with degree m and another Bernstein polynomial $\hat{g}(x)$ with degree n. It is assumed $m \geq n$, and thus the polynomial $\hat{g}(x)$ is degree elevated $(m-n)$ times [22]. Then we obtain

$$\hat{f}(x) = \sum_{i=0}^{m} \hat{a}_i B_i^m(x) \qquad \text{and} \qquad \hat{g}(x) = \sum_{i=0}^{m} \hat{b}_i B_i^m(x),$$

where $B_i^m(x)$ is the ith Bernstein basis function. The Bézout resultant matrix $B(\hat{f}, \hat{g}) = (b_{i,j}) \in \mathbb{R}^{m \times m}$ of $\hat{f}(x)$ and $\hat{g}(x)$ is defined by

$$\frac{\hat{f}(x)\hat{g}(l) - \hat{f}(l)\hat{g}(x)}{x - l} = \sum_{i,j=1}^{m} b_{i,j} B_{i-1}^{m-1}(x) B_{j-1}^{m-1}(l),$$

which can be rewritten as

$$\sum_{i,j=0}^{m} (\hat{a}_i \hat{b}_j - \hat{a}_j \hat{b}_i) B_i^m(x) B_j^m(l) = (x - l) \sum_{i,j=1}^{m} b_{i,j} B_{i-1}^{m-1}(x) B_{j-1}^{m-1}(l).$$

It is shown in [5] that

$$\begin{aligned} & x \sum_{i,j=1}^{m} b_{i,j} B_{i-1}^{m-1}(x) B_{j-1}^{m-1}(l) \\ &= \big(l + (1-l)\big) x \sum_{i,j=1}^{m} b_{i,j} B_{i-1}^{m-1}(x) B_{j-1}^{m-1}(l) \\ &= xl \sum_{i,j=1}^{m} b_{i,j} B_{i-1}^{m-1}(x) B_{j-1}^{m-1}(l) + \sum_{i,j=1}^{m} b_{i,j} x B_{i-1}^{m-1}(x)(1-l) B_{j-1}^{m-1}(l), \end{aligned}$$

and

$$\begin{aligned} & l \sum_{i,j=1}^{m} b_{i,j} B_{i-1}^{m-1}(x) B_{j-1}^{m-1}(l) \\ & \quad = \big(x + (1-x)\big) l \sum_{i,j=1}^{m} b_{i,j} B_{i-1}^{m-1}(x) B_{j-1}^{m-1}(l) \\ & \quad = xl \sum_{i,j=1}^{m} b_{i,j} B_{i-1}^{m-1}(x) B_{j-1}^{m-1}(l) + \sum_{i,j=1}^{m} b_{i,j} (1-x) B_{i-1}^{m-1}(x) l B_{j-1}^{m-1}(l), \end{aligned}$$

hence

$$\begin{aligned} & \sum_{i,j=0}^{m} (\hat{a}_i \hat{b}_j - \hat{a}_j \hat{b}_i) B_i^m(x) B_j^m(l) \\ & \quad = \sum_{i,j=1}^{m} b_{i,j} x B_{i-1}^{m-1}(x) (1-l) B_{j-1}^{m-1}(l) - \sum_{i,j=1}^{m} b_{i,j} (1-x) B_{i-1}^{m-1}(x) l B_{j-1}^{m-1}(l). \end{aligned} \tag{3.2}$$

Since

$$\begin{aligned} x B_{i-1}^{m-1}(x) &= \frac{i}{m} B_i^m(x), \\ (1-l) B_{j-1}^{m-1}(l) &= \frac{m-j+1}{m} B_{j-1}^m(l), \end{aligned}$$

and

$$\begin{aligned} (1-x) B_{i-1}^{m-1}(x) &= \frac{m-i+1}{m} B_{i-1}^m(x), \\ l B_{j-1}^{m-1}(l) &= \frac{j}{m} B_j^m(l), \end{aligned}$$

(3.2) can be rewritten as

$$\begin{aligned} & \sum_{i,j=0}^{m} (\hat{a}_i \hat{b}_j - \hat{a}_j \hat{b}_i) B_i^m(x) B_j^m(l) \\ & \quad = \sum_{i,j=1}^{m} b_{i,j} \frac{i}{m} B_i^m(x) \frac{m-j+1}{m} B_{j-1}^m(l) - \sum_{i,j=1}^{m} b_{i,j} \frac{m-i+1}{m} B_{i-1}^m(x) \frac{j}{m} B_j^m(l). \end{aligned}$$

Equalizing the coefficients of $B_j^m(l)$ on both sides of the previous relation, we obtain

$$\sum_{i=0}^{m} (\hat{a}_i \hat{b}_0 - \hat{a}_0 \hat{b}_i) B_i^m(x) = \sum_{i=1}^{m} b_{i,1} \frac{i}{m} B_i^m(x), \qquad \text{for} \qquad j = 0,$$

and

$$\sum_{i=0}^{m}(\hat{a}_i\hat{b}_j - \hat{a}_j\hat{b}_i)B_i^m(x) = \frac{m-j}{m}\sum_{i=1}^{m} b_{i,j+1}\frac{i}{m}B_i^m(x) - \frac{j}{m}\sum_{i=1}^{m} b_{i,j}\frac{m-i+1}{m}B_{i-1}^m(x),$$

for $j = 1, \cdots, m-1$. Therefore, from the above relation, the formulae for the entries of the Bézout resultant matrix of two Bernstein polynomials are

$$\begin{aligned} b_{i,1} &= \frac{m}{i}(\hat{a}_i\hat{b}_0 - \hat{a}_0\hat{b}_i), \qquad 1 \le i \le m, \\ b_{i,j+1} &= \frac{m^2}{i(m-j)}(\hat{a}_i\hat{b}_j - \hat{a}_j\hat{b}_i) + \frac{j(m-i)}{i(m-j)}b_{i+1,j}, \qquad 1 \le i, j \le m-1, \\ b_{m,j+1} &= \frac{m}{m-j}(\hat{a}_m\hat{b}_j - \hat{a}_j\hat{b}_m), \qquad 1 \le j \le m-1. \end{aligned} \tag{3.3}$$

The Bézout resultant matrix $B(\hat{f}, \hat{g})$ of $\hat{f}(x)$ and $\hat{g}(x)$ satisfies the following properties [5]:

1. The rank loss of $B(\hat{f}, \hat{g})$ is equal to the degree of the GCD of $\hat{f}(x)$ and $\hat{g}(x)$.

2. The coefficients of the GCD of $\hat{f}(x)$ and $\hat{g}(x)$ can be obtained by reducing $B(\hat{f}, \hat{g})$ to upper triangular form, using the QR or LU decompositions.

These two important properties enable us to compute the GCD of $\hat{f}(x)$ and $\hat{g}(x)$ using the Bézout resultant matrix.

Example 3.2. Consider two Bernstein polynomials

$$\hat{f}(x) = \binom{3}{0}(1-x)^3 - \frac{1}{2}\binom{3}{1}(1-x)^2x + \frac{1}{2}\binom{3}{3}x^3,$$

and

$$\begin{aligned} \hat{g}(x) = &\binom{4}{0}(1-x)^4 - \frac{1}{4}\binom{4}{1}(1-x)^3x - \frac{1}{8}\binom{4}{2}(1-x)^2x^2 \\ &+\frac{1}{8}\binom{4}{3}(1-x)x^3 + \frac{1}{4}\binom{4}{4}x^4, \end{aligned}$$

whose GCD is $\hat{f}(x)$ because

$$\hat{g}(x) = \hat{f}(x)\left(\binom{1}{0}(1-x) + \frac{1}{2}\binom{1}{1}x\right).$$

The Bézout resultant matrix of $\hat{f}(x)$ and $\hat{g}(x)$ is

$$B(\hat{f},\hat{g}) = \begin{bmatrix} \frac{1}{2} & -\frac{1}{4} & 0 & \frac{1}{4} \\ -\frac{1}{4} & \frac{1}{8} & 0 & -\frac{1}{8} \\ 0 & 0 & 0 & 0 \\ \frac{1}{4} & -\frac{1}{8} & 0 & \frac{1}{8} \end{bmatrix}.$$

The reduction of $B(\hat{f},\hat{g})$ to row echelon (upper triangular) form yields

$$\begin{bmatrix} \frac{1}{2} & -\frac{1}{4} & 0 & \frac{1}{4} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

and thus the degree of the GCD is 3. The coefficients in the last non-zero row of this matrix yield the GCD

$$\begin{aligned} &\frac{1}{2}\binom{3}{0}(1-x)^3 - \frac{1}{4}\binom{3}{1}(1-x)^2x + \frac{1}{4}\binom{3}{3}x^3 \\ &= \frac{1}{2}\left(\binom{3}{0}(1-x)^3 - \frac{1}{2}\binom{3}{1}(1-x)^2x + \frac{1}{2}\binom{3}{3}x^3\right), \end{aligned}$$

which is proportional to $\hat{f}(x)$. □

The next section considers the computation of the GCD using another resultant matrix, the Sylvester resultant matrix.

3.3 Sylvester resultant matrix

The Sylvester resultant matrix of polynomials defined in Bernstein form is developed in [61]. Consider two Bernstein polynomials

$$\hat{f}(x) = \sum_{i=0}^{m} \hat{a}_i \binom{m}{i} (1-x)^{m-i} x^i \qquad \text{and} \qquad \hat{g}(x) = \sum_{j=0}^{n} \hat{b}_j \binom{n}{j} (1-x)^{n-j} x^j, \quad (3.4)$$

whose GCD is of degree $\hat{d} > 0$, in which case $\hat{f}(x)$ and $\hat{g}(x)$ have one or more common divisors $\hat{t}(x)$ of degree one. Let $\hat{u}(x)$ and $\hat{v}(x)$ be quotient polynomials associated with $\hat{t}(x)$ and therefore

$$\hat{f}(x) = \hat{u}(x)\hat{t}(x) \qquad \text{and} \qquad \hat{g}(x) = \hat{v}(x)\hat{t}(x), \quad (3.5)$$

where $\hat{u}(x)$ and $\hat{v}(x)$ are given by

$$\hat{u}(x) = \sum_{i=0}^{m-1} \hat{u}_i \binom{m-1}{i} (1-x)^{m-1-i} x^i \quad \text{and} \quad \hat{v}(x) = \sum_{i=0}^{n-1} \hat{v}_i \binom{n-1}{i} (1-x)^{n-1-i} x^i.$$

It follows from (3.5) that $\hat{f}(x)\hat{v}(x) = \hat{g}(x)\hat{u}(x)$, which can be written in matrix form as

$$D^{-1} \begin{bmatrix} \hat{a}_0\binom{m}{0} & & & \hat{b}_0\binom{n}{0} & & \\ \hat{a}_1\binom{m}{1} & \ddots & & \hat{b}_1\binom{n}{1} & \ddots & \\ \vdots & \ddots & \hat{a}_0\binom{m}{0} & \vdots & \ddots & \hat{b}_0\binom{n}{0} \\ \vdots & \ddots & \hat{a}_1\binom{m}{1} & \vdots & \ddots & \hat{b}_1\binom{n}{1} \\ \hat{a}_m\binom{m}{m} & \ddots & \vdots & \hat{b}_n\binom{n}{n} & \ddots & \vdots \\ & \ddots & \vdots & & \ddots & \vdots \\ & & \hat{a}_m\binom{m}{m} & & & \hat{b}_n\binom{n}{n} \end{bmatrix} \begin{bmatrix} \hat{v}_0\binom{n-1}{0} \\ \hat{v}_1\binom{n-1}{1} \\ \vdots \\ \hat{v}_{n-1}\binom{n-1}{n-1} \\ -\hat{u}_0\binom{m-1}{0} \\ \vdots \\ -\hat{u}_{m-1}\binom{m-1}{m-1} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad (3.6)$$

where D^{-1} is

$$D^{-1} = \operatorname{diag}\left[\frac{1}{\binom{m+n-1}{0}} \quad \frac{1}{\binom{m+n-1}{1}} \quad \cdots \quad \frac{1}{\binom{m+n-1}{m+n-2}} \quad \frac{1}{\binom{m+n-1}{m+n-1}} \right]. \quad (3.7)$$

Let $T(\hat{f},\hat{g})$ and $p(\hat{u},\hat{v})$ be given by

$$T(\hat{f},\hat{g}) = \begin{bmatrix} \hat{a}_0\binom{m}{0} & & & \hat{b}_0\binom{n}{0} & & \\ \hat{a}_1\binom{m}{1} & \ddots & & \hat{b}_1\binom{n}{1} & \ddots & \\ \vdots & \ddots & \hat{a}_0\binom{m}{0} & \vdots & \ddots & \hat{b}_0\binom{n}{0} \\ \vdots & \ddots & \hat{a}_1\binom{m}{1} & \vdots & \ddots & \hat{b}_1\binom{n}{1} \\ \hat{a}_m\binom{m}{m} & \ddots & \vdots & \hat{b}_n\binom{n}{n} & \ddots & \vdots \\ & \ddots & \vdots & & \ddots & \vdots \\ & & \hat{a}_m\binom{m}{m} & & & \hat{b}_n\binom{n}{n} \end{bmatrix}, \tag{3.8}$$

and

$$p(\hat{u},\hat{v}) = \begin{bmatrix} \hat{v}_0\binom{n-1}{0} \\ \hat{v}_1\binom{n-1}{1} \\ \vdots \\ \hat{v}_{n-1}\binom{n-1}{n-1} \\ -\hat{u}_0\binom{m-1}{0} \\ \vdots \\ -\hat{u}_{m-1}\binom{m-1}{m-1} \end{bmatrix}, \tag{3.9}$$

in which case, (3.6) can be expressed as

$$D^{-1}T(\hat{f},\hat{g})p(\hat{u},\hat{v}) = 0, \tag{3.10}$$

where D^{-1} is defined in (3.7), and $T(\hat{f},\hat{g})$ is the Sylvester resultant matrix when $\hat{f}(x)$ and $\hat{g}(x)$ are expressed in the scaled Bernstein basis [57]. If $\hat{f}(x)$ and $\hat{g}(x)$ have a non-constant common divisor, there exists a solution of $p(\hat{u},\hat{v})$ satisfying (3.10) and thus the determinant of $D^{-1}T(\hat{f},\hat{g})$ vanishes. Therefore, the determinant of $D^{-1}T(\hat{f},\hat{g})$ is a resultant of $\hat{f}(x)$ and $\hat{g}(x)$, and the Sylvester resultant matrix is

$$S(\hat{f},\hat{g}) = D^{-1}T(\hat{f},\hat{g}). \tag{3.11}$$

Similarly, it is shown in [61] that the Sylvester resultant matrix $S(\hat{f}, \hat{g})$ of $\hat{f}(x)$ and $\hat{g}(x)$ satisfies the following properties:

1. The rank loss of $S(\hat{f}, \hat{g})$ is equal to the degree of the GCD of $\hat{f}(x)$ and $\hat{g}(x)$.

2. The coefficients of the GCD of $\hat{f}(x)$ and $\hat{g}(x)$ can be obtained by reducing $S(\hat{f}, \hat{g})^T$ to upper triangular form, using the QR or LU decompositions.

These important properties establish the relation between the Sylvester resultant matrix and the computation of the GCD of two polynomials.

Example 3.3. Consider two Bernstein polynomials

$$\hat{f}(x) = 6\binom{2}{0}(1-x)^2 + \frac{7}{2}\binom{2}{1}(1-x)x + 2\binom{2}{2}x^2,$$

and

$$\hat{g}(x) = 6\binom{3}{0}(1-x)^3 + \frac{19}{3}\binom{3}{1}(1-x)^2x + \frac{16}{3}\binom{3}{2}(1-x)x^2 + 4\binom{3}{3}x^3,$$

whose GCD is $\hat{f}(x)$ because

$$\hat{g}(x) = \hat{f}(x)\left(\binom{1}{0}(1-x) + 2\binom{1}{1}x\right).$$

The transpose of the Sylvester resultant matrix $S(\hat{f},\hat{g})$ of $\hat{f}(x)$ and $\hat{g}(x)$ is

$$\begin{aligned} S(\hat{f},\hat{g})^T &= \begin{bmatrix} 6 & 7 & 2 & 0 & 0 \\ 0 & 6 & 7 & 2 & 0 \\ 0 & 0 & 6 & 7 & 2 \\ 6 & 19 & 16 & 4 & 0 \\ 0 & 6 & 19 & 16 & 4 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{4} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{6} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{4} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 6 & \frac{7}{4} & \frac{1}{3} & 0 & 0 \\ 0 & \frac{3}{2} & \frac{7}{6} & \frac{1}{2} & 0 \\ 0 & 0 & 1 & \frac{7}{4} & 2 \\ 6 & \frac{19}{4} & \frac{8}{3} & 1 & 0 \\ 0 & \frac{3}{2} & \frac{19}{6} & 4 & 4 \end{bmatrix}. \end{aligned}$$

The reduction of $S(\hat{f},\hat{g})^T$ to row echelon (upper triangular) form yields

$$\begin{bmatrix} 1 & 0 & 0 & \frac{1}{5} & \frac{1}{3} \\ 0 & 1 & 0 & -\frac{37}{36} & -\frac{14}{9} \\ 0 & 0 & 1 & \frac{7}{4} & 2 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

and the coefficients in the last non-zero row of this matrix yield the GCD,

$$\begin{aligned} &\binom{4}{2}(1-x)^2x^2 + \frac{7}{4}\binom{4}{3}(1-x)x^3 + 2\binom{4}{4}x^4 \\ &= x^2\left(6\binom{2}{0}(1-x)^2 + \frac{7}{2}\binom{2}{1}(1-x)x + 2\binom{2}{2}x^2\right). \end{aligned}$$

Deletion of the extraneous factor x^2 yields the GCD, $\hat{f}(x)$. □

The next section considers the Sylvester subresultant matrices because the degree of the GCD of polynomials can be determined by calculating the ranks of the

Sylvester subresultant matrices.

3.3.1 Sylvester subresultant matrices

In this section, we consider the Sylvester subresultant matrices defined in the Bernstein basis. A subresultant matrix of the Sylvester resultant matrix is similar to the Sylvester matrix but it has fewer rows and columns. The Sylvester subresultant matrices expressed in the power basis are considered in [64].

It is assumed that two Bernstein polynomials $\hat{f}(x)$ and $\hat{g}(x)$ defined in (3.4) have a GCD of degree $\hat{d} > 0$, and thus they possess a common divisor of degree k, where $1 \leq k \leq \hat{d}$. Therefore, there exists a polynomial $\hat{d}_k(x)$ of degree k such that

$$\hat{f}(x) = \hat{u}_k(x)\hat{d}_k(x) \qquad \text{and} \qquad \hat{g}(x) = \hat{v}_k(x)\hat{d}_k(x), \tag{3.12}$$

where the quotient polynomials $\hat{u}_k(x)$ and $\hat{v}_k(x)$ are

$$\begin{aligned} \hat{u}_k(x) &= \sum_{i=0}^{m-k} \hat{u}_{k,i} \binom{m-k}{i} (1-x)^{m-k-i} x^i, \\ \hat{v}_k(x) &= \sum_{j=0}^{n-k} \hat{v}_{k,j} \binom{n-k}{j} (1-x)^{n-k-j} x^j, \end{aligned}$$

respectively, and the common divisor polynomial $\hat{d}_k(x)$ is

$$\hat{d}_k(x) = \sum_{i=0}^{k} \hat{d}_{k,i} \binom{k}{i} (1-x)^{k-i} x^i.$$

It follows from (3.12) that $\hat{f}\hat{v}_k = \hat{g}\hat{u}_k$, that is,

$$\begin{aligned} &\sum_{i=0}^{m} \hat{a}_i \binom{m}{i} (1-x)^{m-i} x^i \sum_{j=0}^{n-k} \hat{v}_{k,j} \binom{n-k}{j} (1-x)^{n-k-j} x^j \\ &= \sum_{j=0}^{n} \hat{b}_j \binom{n}{j} (1-x)^{n-j} x^j \sum_{i=0}^{m-k} \hat{u}_{k,i} \binom{m-k}{i} (1-x)^{m-k-i} x^i, \end{aligned}$$

and the expression for the coefficients of the product of two Bernstein polynomials yields an expression for each coefficient of the product [22],

$$\sum_{j=\max(0,r-(n-k))}^{\min(m,r)} \left(\frac{\hat{a}_j \binom{m}{j}}{\binom{m+n-k}{r}} \right) \hat{v}_{k,r-j} \binom{n-k}{r-j}$$
$$= \sum_{j=\max(0,r-n)}^{\min(m-k,r)} \left(\frac{\hat{b}_{r-j} \binom{n}{r-j}}{\binom{m+n-k}{r}} \right) \hat{u}_{k,j} \binom{m-k}{j}, \qquad r = 0, \ldots, m+n-k.$$

It follows that the homogeneous equation

$$D_k^{-1} \begin{bmatrix} \hat{a}_0\binom{m}{0} & & & \hat{b}_0\binom{n}{0} & & \\ \hat{a}_1\binom{m}{1} & \ddots & & \hat{b}_1\binom{n}{1} & \ddots & \\ \vdots & \ddots & \hat{a}_0\binom{m}{0} & \vdots & \ddots & \hat{b}_0\binom{n}{0} \\ \hat{a}_{m-1}\binom{m}{m-1} & \ddots & \hat{a}_1\binom{m}{1} & \hat{b}_{n-1}\binom{n}{n-1} & \ddots & \hat{b}_1\binom{n}{1} \\ \hat{a}_m\binom{m}{m} & \ddots & \vdots & \hat{b}_n\binom{n}{n} & \ddots & \vdots \\ & \ddots & \hat{a}_{m-1}\binom{m}{m-1} & & \ddots & \hat{b}_{n-1}\binom{n}{n-1} \\ & & \hat{a}_m\binom{m}{m} & & & \hat{b}_n\binom{n}{n} \end{bmatrix} \times$$
$$\begin{bmatrix} \hat{v}_{k,0}\binom{n-k}{0} \\ \vdots \\ \hat{v}_{k,n-k}\binom{n-k}{n-k} \\ -\hat{u}_{k,0}\binom{m-k}{0} \\ \vdots \\ -\hat{u}_{k,m-k}\binom{m-k}{m-k} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \tag{3.13}$$

where

$$D_k^{-1} = \operatorname{diag}\left[\frac{1}{\binom{m+n-k}{0}} \quad \frac{1}{\binom{m+n-k}{1}} \quad \cdots \quad \frac{1}{\binom{m+n-k}{m+n-k-1}} \quad \frac{1}{\binom{m+n-k}{m+n-k}} \right], \tag{3.14}$$

and $D_1^{-1} = D^{-1}$ defined in (3.7) is attained. Let $T_k(\hat{f}, \hat{g})$ and $p_k(\hat{u}_k, \hat{v}_k)$ be given by

$$T_k(\hat{f},\hat{g}) = \begin{bmatrix} \hat{a}_0\binom{m}{0} & & & \hat{b}_0\binom{n}{0} & & \\ \hat{a}_1\binom{m}{1} & \ddots & & \hat{b}_1\binom{n}{1} & \ddots & \\ \vdots & \ddots & \hat{a}_0\binom{m}{0} & \vdots & \ddots & \hat{b}_0\binom{n}{0} \\ \hat{a}_{m-1}\binom{m}{m-1} & \ddots & \hat{a}_1\binom{m}{1} & \hat{b}_{n-1}\binom{n}{n-1} & \ddots & \hat{b}_1\binom{n}{1} \\ \hat{a}_m\binom{m}{m} & \ddots & \vdots & \hat{b}_n\binom{n}{n} & \ddots & \vdots \\ & \ddots & \hat{a}_{m-1}\binom{m}{m-1} & & \ddots & \hat{b}_{n-1}\binom{n}{n-1} \\ & & \hat{a}_m\binom{m}{m} & & & \hat{b}_n\binom{n}{n} \end{bmatrix}, \quad (3.15)$$

and

$$p_k(\hat{u}_k,\hat{v}_k) = \begin{bmatrix} \hat{v}_{k,0}\binom{n-k}{0} \\ \vdots \\ \hat{v}_{k,n-k}\binom{n-k}{n-k} \\ -\hat{u}_{k,0}\binom{m-k}{0} \\ \vdots \\ -\hat{u}_{k,m-k}\binom{m-k}{m-k} \end{bmatrix}, \quad (3.16)$$

where $T_k(\hat{f}, \hat{g}) \in \mathbb{R}^{(m+n-k+1)\times(m+n-2k+2)}$ and $p_k(\hat{u}_k, \hat{v}_k) \in \mathbb{R}^{m+n-2k+2}$ respectively. Equation (3.13) is rewritten as

$$\left(D_k^{-1} T_k(\hat{f},\hat{g})\right) p_k(\hat{u}_k,\hat{v}_k) = 0, \qquad k = 1,\ldots,\min(m,n), \quad (3.17)$$

and the kth Sylvester subresultant matrix is

$$S_k(\hat{f},\hat{g}) = D_k^{-1} T_k(\hat{f},\hat{g}) \in \mathbb{R}^{(m+n-k+1)\times(m+n-2k+2)}, \quad (3.18)$$

where D_k^{-1} is defined in (3.14) and $T_k(\hat{f}, \hat{g})$ is defined in (3.15). The coefficients of $\hat{f}(x)$ occupy the first $(n-k+1)$ columns, and the coefficients of $\hat{g}(x)$ occupy the last $(m-k+1)$ columns, of $S_k(\hat{f}, \hat{g})$, and $S_k(\hat{f}, \hat{g})$ is square and reduces to the Sylvester matrix if $k = 1$, $S_1(\hat{f}, \hat{g}) = S(\hat{f}, \hat{g})$. If $k > 1$, the number of rows of $S_k(\hat{f}, \hat{g})$ is greater

than its number of columns.

If $\hat{f}(x)$ and $\hat{g}(x)$ have a common divisor of degree $k \geq 1$, then (3.17) possesses a solution, and $S_k(\hat{f}, \hat{g})$ must be rank deficient. Therefore, if $\hat{f}(x)$ and $\hat{g}(x)$ have a common divisor of degree $k \geq 1$, the rank of $S_k(\hat{f}, \hat{g})$ is less than $(m + n - 2k + 2)$. Now assume that the rank of $S_k(\hat{f}, \hat{g})$ is less than $(m + n - 2k + 2)$, from which it follows that one or more of its columns are linearly dependent on the other columns. Therefore, there exist constants $h_{k,0}, \ldots, h_{k,n-k}, q_{k,0}, \ldots, q_{k,m-k}$, not all zero, such that

$$\sum_{i=0}^{n-k} h_{k,i} c_{k,i} - \sum_{j=0}^{m-k} q_{k,j} d_{k,j} = 0, \tag{3.19}$$

where $c_{k,i}, i = 0, \ldots, n-k$, and $d_{k,j}, j = 0, \ldots, m-k$, are the vectors of the first $(n-k+1)$ and last $(m-k+1)$ columns of $S_k(\hat{f}, \hat{g})$, respectively. If the polynomials $h_k(x)$ and $q_k(x)$ are defined as

$$h_k(x) = \sum_{i=0}^{n-k} h_{k,i} \binom{n-k}{i} (1-x)^{n-k-i} x^i,$$

and

$$q_k(x) = \sum_{j=0}^{m-k} q_{k,j} \binom{m-k}{j} (1-x)^{m-k-j} x^j,$$

respectively, then (3.19) states that

$$h_k(x)\hat{f}(x) = q_k(x)\hat{g}(x). \tag{3.20}$$

One important theorem associated with (3.20) must be introduced here.

Theorem 3.1. *Let $\hat{f}(x)$ and $\hat{g}(x)$ be polynomials of degrees m and n respectively, and let $\hat{d}_k(x)$ be a polynomial of degree k. There exist polynomials $h_k(x)$ and $q_k(x)$, of degrees $n-k$ and $m-k$, respectively, that satisfy (3.20), if and only if $\hat{d}_k(x)$ is a*

common divisor of $\hat{f}(x)$ *and* $\hat{g}(x)$.

Proof. If $\hat{d}_k(x)$ is a common divisor of $\hat{f}(x)$ and $\hat{g}(x)$, there exist polynomials $h_k(x)$ and $q_k(x)$ such that

$$\frac{\hat{f}(x)}{\hat{d}_k(x)} = q_k(x) \qquad \text{and} \qquad \frac{\hat{g}(x)}{\hat{d}_k(x)} = h_k(x),$$

and (3.20) follows.

Conversely, assume (3.20) holds such that, without loss of generality, $h_k(x)$ and $q_k(x)$ are coprime. (If these polynomials are not coprime, any common divisors can be removed.) It follows that since $h_k(x)$ is of degree $n-k$ and $\hat{g}(x)$ is of degree n, every divisor of $h_k(x)$ is also a divisor of $\hat{g}(x)$. There therefore exists a polynomial $\hat{d}_{k,1}(x)$ of degree k such that

$$\hat{g}(x) = h_k(x)\hat{d}_{k,1}(x), \tag{3.21}$$

and similarly, consideration of the polynomials $q_k(x)$ and $\hat{f}(x)$ leads to

$$\hat{f}(x) = q_k(x)\hat{d}_{k,2}(x), \tag{3.22}$$

where $\hat{d}_{k,2}(x)$ is of degree k. The substitution of (3.21) and (3.22) into (3.20) shows that $\hat{d}_{k,1}(x) = \hat{d}_{k,2}(x)$, and thus the result is established. □

It follows from Theorem 3.1 that (3.20) shows that $\hat{f}(x)$ and $\hat{g}(x)$ have a common divisor of degree k. Therefore, if the rank of $S_k(\hat{f}, \hat{g})$ is less than $(m+n-2k+2)$, then $\hat{f}(x)$ and $\hat{g}(x)$ have a common divisor of degree k.

From the above discussion, the main theorem is now established.

Theorem 3.2. *A necessary and sufficient condition for the polynomials* $\hat{f}(x)$ *and* $\hat{g}(x)$ *to have a common divisor of degree* $k \geq 1$ *is that the rank of* $S_k(\hat{f}, \hat{g})$ *is less than* $(m+n-2k+2)$*, where* $S_k(\hat{f}, \hat{g})$ *is defined in (3.18).*

Since the degree of the GCD of $\hat{f}(x)$ and $\hat{g}(x)$ is $\hat{d} \geq 1$, these polynomials possess common divisors of degree $1, 2, \ldots, \hat{d}$, but they do not have a common divisor of degree $\hat{d}+1$. Therefore, from Theorem 3.2, the rank of $S_k(\hat{f}, \hat{g})$ can be used to calculate the degree of the GCD of $\hat{f}(x)$ and $\hat{g}(x)$:

$$\begin{aligned} &\operatorname{rank} S_k(\hat{f}, \hat{g}) < m + n - 2k + 2, \qquad k = 1, \ldots, \hat{d}, \\ &\operatorname{rank} S_k(\hat{f}, \hat{g}) = m + n - 2k + 2, \qquad k = \hat{d}+1, \ldots, \min(m, n). \end{aligned} \tag{3.23}$$

Thus, the determination of the degree of the GCD of $\hat{f}(x)$ and $\hat{g}(x)$ reduces to the calculation of the ranks of the subresultant matrices $S_k(\hat{f}, \hat{g}), k = 1, \ldots, \min(m, n)$.

Example 3.4. Consider two Bernstein polynomials

$$\begin{aligned} \hat{f}(x) &= 4\binom{3}{0}(1-x)^3 + 4\binom{3}{1}(1-x)^2 x + 3\binom{3}{2}(1-x)x^2 + 2\binom{3}{3}x^3 \\ &= (x-2)^2(x+1), \end{aligned}$$

and

$$\begin{aligned} \hat{g}(x) &= \binom{2}{0}(1-x)^2 - \frac{1}{4}\binom{2}{1}(1-x)x - \frac{1}{2}\binom{2}{2}x^2 \\ &= (x-2)(x-\frac{1}{2}), \end{aligned}$$

whose GCD is of degree 1. The subresultant matrices $S_k(\hat{f}, \hat{g})$, $k = 1, 2$, of $\hat{f}(x)$ and

$\hat{g}(x)$ are

$$S_1(\hat{f},\hat{g}) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{4} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{6} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{4} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 & 0 & 1 & 0 & 0 \\ 12 & 4 & -\frac{1}{2} & 1 & 0 \\ 9 & 12 & -\frac{1}{2} & -\frac{1}{2} & 1 \\ 2 & 9 & 0 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & 2 & 0 & 0 & -\frac{1}{2} \end{bmatrix}$$

$$= \begin{bmatrix} 4 & 0 & 1 & 0 & 0 \\ 3 & 1 & -\frac{1}{8} & \frac{1}{4} & 0 \\ \frac{3}{2} & 2 & -\frac{1}{12} & -\frac{1}{12} & \frac{1}{6} \\ \frac{1}{2} & \frac{9}{4} & 0 & -\frac{1}{8} & -\frac{1}{8} \\ 0 & 2 & 0 & 0 & -\frac{1}{2} \end{bmatrix} \in \mathbb{R}^{5\times 5},$$

and

$$S_2(\hat{f},\hat{g}) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{3} & 0 & 0 \\ 0 & 0 & \frac{1}{3} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 & 1 & 0 \\ 12 & -\frac{1}{2} & 1 \\ 9 & -\frac{1}{2} & -\frac{1}{2} \\ 2 & 0 & -\frac{1}{2} \end{bmatrix} = \begin{bmatrix} 4 & 1 & 0 \\ 4 & -\frac{1}{6} & \frac{1}{3} \\ 3 & -\frac{1}{6} & -\frac{1}{6} \\ 2 & 0 & -\frac{1}{2} \end{bmatrix} \in \mathbb{R}^{4\times 3},$$

where $S_1(\hat{f},\hat{g}) = S(\hat{f},\hat{g})$. Reducing $S_1(\hat{f},\hat{g})$ and $S_2(\hat{f},\hat{g})$ to their row echelon forms, we obtain $\operatorname{rank} S_1(\hat{f},\hat{g}) = 4$ and $\operatorname{rank} S_2(\hat{f},\hat{g}) = 3$, and thus $S_1(\hat{f},\hat{g})$ is rank deficient and $S_2(\hat{f},\hat{g})$ is of full rank, which implies that the degree of the GCD is 1. □

This section introduced the conventional forms of the Sylvester resultant matrix and its subresultant matrices. However, new forms of the Sylvester resultant matrix and subresultant matrices can be developed with the inclusion of a diagonal matrix, which are considered in the next section. It will be shown in the following chapters that the new forms of the Sylvester resultant matrix and subresultant matrices yield

significantly better results than their conventional forms with respect to the determination of the degree of an approximate GCD, which will be explained in Sections 4.3 and 4.4, and a structured low rank approximation of the Sylvester resultant matrix. The explanation for the superiority of the new forms of the Sylvester resultant matrix and subresultant matrices is considered in Section 6.3.

3.4 A new form of the Sylvester resultant matrix

This section considers another form of the Sylvester resultant matrix. In particular, this new form is obtained with the inclusion of a diagonal matrix, which is discussed in the following.

The vector $p(\hat{u}, \hat{v})$ defined in (3.9) can be written as

$$p(\hat{u}, \hat{v}) = Qr(\hat{u}, \hat{v}), \tag{3.24}$$

where

$$Q = \operatorname{diag}\left[\begin{array}{cccccc} \binom{n-1}{0} & \cdots & \binom{n-1}{n-1} & \binom{m-1}{0} & \cdots & \binom{m-1}{m-1} \end{array} \right] \in \mathbb{R}^{(m+n)\times(m+n)}, \tag{3.25}$$

and

$$r(\hat{u}, \hat{v}) = \left[\begin{array}{cccccc} \hat{v}_0 & \cdots & \hat{v}_{n-1} & -\hat{u}_0 & \cdots & -\hat{u}_{m-1} \end{array} \right]^T \in \mathbb{R}^{m+n}, \tag{3.26}$$

and thus it follows from (3.10) that

$$S(\hat{f}, \hat{g})p(\hat{u}, \hat{v}) = \big(D^{-1}T(\hat{f}, \hat{g})\big)p(\hat{u}, \hat{v}) = \big(D^{-1}T(\hat{f}, \hat{g})Q\big)r(\hat{u}, \hat{v}) = 0,$$

where D^{-1} is defined in (3.7) and $T(\hat{f}, \hat{g})$ is defined in (3.8).

Since Q is non-singular, it follows that

$$\begin{aligned} \deg \mathrm{GCD}(\hat{f}, \hat{g}) &= m + n - \operatorname{rank} S(\hat{f}, \hat{g}) \\ &= m + n - \operatorname{rank} D^{-1} T(\hat{f}, \hat{g}) \\ &= m + n - \operatorname{rank} D^{-1} T(\hat{f}, \hat{g}) Q, \end{aligned} \tag{3.27}$$

and thus

$$S(\hat{f}, \hat{g})Q = D^{-1} T(\hat{f}, \hat{g})Q, \tag{3.28}$$

satisfies the rank loss property of the Sylvester resultant matrix. The second property - the computation of the GCD coefficients from the QR or LU decomposition of $\big(S(\hat{f}, \hat{g})Q\big)^T = QS(\hat{f}, \hat{g})^T$ - follows because Q is a diagonal matrix that scales the rows of $S(\hat{f}, \hat{g})^T$, and thus $S(\hat{f}, \hat{g})Q$ is also a resultant matrix. These two properties allow $S(\hat{f}, \hat{g})Q$ to be used to compute the GCD of $\hat{f}(x)$ and $\hat{g}(x)$.

Example 3.5. Consider two Bernstein polynomials

$$\hat{f}(x) = 2\binom{3}{0}(1-x)^3 + \frac{4}{3}\binom{3}{1}(1-x)^2 x - \frac{1}{2}\binom{3}{2}(1-x)x^2 - \frac{9}{2}\binom{3}{3}x^3,$$

and

$$\hat{g}(x) = \binom{2}{0}(1-x)^2 + \frac{1}{4}\binom{2}{1}(1-x)x - \frac{3}{2}\binom{2}{2}x^2,$$

whose GCD is $\hat{g}(x)$ because

$$\hat{f}(x) = \hat{g}(x)\left(2\binom{1}{0}(1-x) + 3\binom{1}{1}x\right).$$

The transpose of the Sylvester resultant matrix $S(\hat{f},\hat{g})Q$ is

$$QS(\hat{f},\hat{g})^T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 & 4 & -\frac{3}{2} & -\frac{9}{2} & 0 \\ 0 & 2 & 4 & -\frac{3}{2} & -\frac{9}{2} \\ 1 & \frac{1}{2} & -\frac{3}{2} & 0 & 0 \\ 0 & 1 & \frac{1}{2} & -\frac{3}{2} & 0 \\ 0 & 0 & 1 & \frac{1}{2} & -\frac{3}{2} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{4} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{6} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{4} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 2 & 1 & -\frac{1}{4} & -\frac{9}{8} & 0 \\ 0 & \frac{1}{2} & \frac{2}{3} & -\frac{3}{8} & -\frac{9}{2} \\ 1 & \frac{1}{8} & -\frac{1}{4} & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{6} & -\frac{3}{4} & 0 \\ 0 & 0 & \frac{1}{6} & \frac{1}{8} & -\frac{3}{2} \end{bmatrix}.$$

The reduction of $QS(\hat{f},\hat{g})^T$ to row echelon (upper triangular) form yields

$$\begin{bmatrix} 2 & 1 & -\frac{1}{4} & -\frac{9}{8} & 0 \\ 0 & \frac{1}{2} & \frac{2}{3} & -\frac{3}{8} & -\frac{9}{2} \\ 0 & 0 & -\frac{1}{2} & -\frac{3}{8} & \frac{9}{2} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

and the coefficients in the last non-zero row of this matrix yield the GCD,

$$\begin{aligned} &-\frac{1}{2}\binom{4}{2}(1-x)^2x^2 - \frac{3}{8}\binom{4}{3}(1-x)x^3 + \frac{9}{2}\binom{4}{4}x^4 \\ &= -3x^2\left(\binom{2}{0}(1-x)^2 + \frac{1}{4}\binom{2}{1}(1-x)x - \frac{3}{2}\binom{2}{2}x^2\right). \end{aligned}$$

Deletion of the extraneous factor $-3x^2$ yields the GCD, $\hat{g}(x)$. □

Example 3.5 shows that the degree of the GCD of $\hat{f}(x)$ and $\hat{g}(x)$ is equal to the rank loss of their Sylvester matrix $S(\hat{f},\hat{g})Q$ and the last non-zero row of an upper

triangular form of $S(\hat{f},\hat{g})Q$ yields the coefficients of the GCD.

In this section, we introduced the matrix $S(\hat{f},\hat{g})Q$ and explained that it satisfies the properties of the Sylvester resultant matrix. Therefore, $S(\hat{f},\hat{g})Q$ can be considered another form of the Sylvester resultant matrix. The subresultant matrices of this modified form of the Sylvester matrix are discussed in the next section.

3.4.1 The subresultant matrices of the modified Sylvester matrix

This section considers the subresultant matrices of the Sylvester resultant matrix $S(\hat{f},\hat{g})Q$.

The vector $p_k(\hat{u}_k,\hat{v}_k)$ defined in (3.16) can be written as

$$p_k(\hat{u}_k,\hat{v}_k) = Q_k r_k(\hat{u}_k,\hat{v}_k), \tag{3.29}$$

where $Q_k \in \mathbb{R}^{(m+n-2k+2)\times(m+n-2k+2)}$,

$$Q_k = \operatorname{diag}\left[\begin{array}{cccccc} \binom{n-k}{0} & \cdots & \binom{n-k}{n-k} & \binom{m-k}{0} & \cdots & \binom{m-k}{m-k} \end{array} \right], \tag{3.30}$$

and

$$r_k(\hat{u}_k,\hat{v}_k) = \left[\begin{array}{cccccc} \hat{v}_{k,0} & \cdots & \hat{v}_{k,n-k} & -\hat{u}_{k,0} & \cdots & -\hat{u}_{k,m-k} \end{array} \right]^T \in \mathbb{R}^{m+n-2k+2}, \tag{3.31}$$

and thus it follows from (3.17) that

$$\begin{aligned} S_k(\hat{f},\hat{g})p_k(\hat{u}_k,\hat{v}_k) &= \left(D_k^{-1}T_k(\hat{f},\hat{g})\right) p_k(\hat{u}_k,\hat{v}_k) \\ &= \left(D_k^{-1}T_k(\hat{f},\hat{g})Q_k\right) r_k(\hat{u}_k,\hat{v}_k) \\ &= \left(S_k(\hat{f},\hat{g})Q_k\right) r_k(\hat{u}_k,\hat{v}_k) \\ &= 0. \end{aligned} \tag{3.32}$$

Since Q_k is non-singular, the rank of $S_k(\hat{f},\hat{g})$ equals to the rank of $S_k(\hat{f},\hat{g})Q_k$. Therefore, it follows from (3.23) that

$$\begin{aligned} &\operatorname{rank} S_k(\hat{f},\hat{g})Q_k < m+n-2k+2, \qquad k=1,\ldots,\hat{d}, \\ &\operatorname{rank} S_k(\hat{f},\hat{g})Q_k = m+n-2k+2, \qquad k=\hat{d}+1,\ldots,\min(m,n). \end{aligned} \tag{3.33}$$

Therefore,

$$S_k(\hat{f},\hat{g})Q_k = D_k^{-1}T_k(\hat{f},\hat{g})Q_k, \tag{3.34}$$

satisfies the property of the Sylvester subresultant matrices, and the degree of the GCD of $\hat{f}(x)$ and $\hat{g}(x)$ can also be determined by calculating the ranks of the subresultant matrices $S_k(\hat{f},\hat{g})Q_k, k=1,\ldots,\min(m,n)$.

The coefficients of $\hat{f}(x)$ occupy the first $(n-k+1)$ columns, and the coefficients of $\hat{g}(x)$ occupy the last $(m-k+1)$ columns, of $S_k(\hat{f},\hat{g})Q_k$, and when $k=1$, $S_k(\hat{f},\hat{g})Q_k$ is square and equals to the Sylvester resultant matrix $S(\hat{f},\hat{g})Q$, that is, $S_1(\hat{f},\hat{g})Q_1 = S(\hat{f},\hat{g})Q$. If $k>1$, the number of rows of $S_k(\hat{f},\hat{g})Q_k$ is greater than its number of columns.

Example 3.6. Consider two Bernstein polynomials

$$\begin{aligned} \hat{f}(x) &= 2\binom{2}{0}(1-x)^2 + \frac{1}{2}\binom{2}{1}(1-x)x \\ &= (x-2)(x-1), \end{aligned}$$

and

$$\begin{aligned} \hat{g}(x) &= 2\binom{3}{0}(1-x)^3 + 3\binom{3}{1}(1-x)^2x + 4\binom{3}{2}(1-x)x^2 + 4\binom{3}{3}x^3 \\ &= (x-2)(x+1)^2, \end{aligned}$$

whose GCD is of degree 1. The subresultant matrices $S_k(\hat{f},\hat{g})Q_k$, $k=1,2$, of $\hat{f}(x)$

and $\hat{g}(x)$ are

$$S_1(\hat{f},\hat{g})Q_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{4} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{6} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{4} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 & 0 & 0 & 2 & 0 \\ 1 & 2 & 0 & 9 & 2 \\ 0 & 1 & 2 & 12 & 9 \\ 0 & 0 & 1 & 4 & 12 \\ 0 & 0 & 0 & 0 & 4 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 2 & 0 & 0 & 2 & 0 \\ \frac{1}{4} & 1 & 0 & \frac{9}{4} & \frac{1}{2} \\ 0 & \frac{1}{3} & \frac{1}{3} & 2 & \frac{3}{2} \\ 0 & 0 & \frac{1}{4} & 1 & 3 \\ 0 & 0 & 0 & 0 & 4 \end{bmatrix} \in \mathbb{R}^{5\times 5},$$

and

$$S_2(\hat{f},\hat{g})Q_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{3} & 0 & 0 \\ 0 & 0 & \frac{1}{3} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 & 0 & 2 \\ 1 & 2 & 9 \\ 0 & 1 & 12 \\ 0 & 0 & 4 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 2 & 0 & 2 \\ \frac{1}{3} & \frac{2}{3} & 3 \\ 0 & \frac{1}{3} & 4 \\ 0 & 0 & 4 \end{bmatrix} \in \mathbb{R}^{4\times 3},$$

where $S_1(\hat{f},\hat{g})Q_1 = S(\hat{f},\hat{g})Q$. Reducing $S_1(\hat{f},\hat{g})Q_1$ and $S_2(\hat{f},\hat{g})Q_2$ to their row echelon forms yields $\operatorname{rank} S_1(\hat{f},\hat{g})Q_1 = 4$ and $\operatorname{rank} S_2(\hat{f},\hat{g})Q_2 = 3$, and therefore $S_1(\hat{f},\hat{g})Q_1$ is rank deficient and $S_2(\hat{f},\hat{g})Q_2$ has full rank, which implies that the degree of the GCD of $\hat{f}(x)$ and $\hat{g}(x)$ is 1. □

3.5 Summary

This chapter has introduced two classical methods, Euclid's algorithm and resultant matrices, to calculate the GCD of two Bernstein polynomials. It was also shown in this chapter that the degree of the GCD of two Bernstein polynomials can be determined by computing the ranks of the Sylvester subresultant matrices. Examples have shown that they provide an unambiguous and correct result in a symbolic computing environment when Bernstein polynomials are specified exactly.

When these methods are implemented in a floating point environment, however, roundoff error may suggest that a resultant matrix is non-singular, even if it is theoretically singular, and an example of this phenomenon is shown in [64]. This computational problem is more apparent when data errors, which are usually much larger than roundoff error, are present. The problem caused by data errors for GCD computations will be shown in the next chapter.

Chapter 4

GCD computation in the presence of noise

Chapter 3 introduced Euclid's algorithm, and the Bézout and Sylvester resultant matrices, to compute the GCD of two Bernstein polynomials symbolically. However, in practical applications, the GCD computation is performed in a floating point environment, and polynomials are not often specified exactly due to data errors generated from previous computation. Because polynomials are often perturbed by data errors such that inexact forms are usually specified, it is necessary to consider the effect of data errors on GCD computations. In particular, minor noise applied to the coefficients of polynomials makes their inexact forms coprime such that computing the GCD of polynomials from their inexact forms is an ill-posed problem, and this phenomena will be shown in Section 4.2.

This chapter first introduces the addition of noise to the coefficients of a Bernstein polynomial to obtain an inexact form, and then the computation of the GCD of two exact polynomials from their inexact forms, using the three algorithms described in

Chapter 3, in a floating point environment, is discussed.

4.1 Addition of noise

The Bernstein polynomials $\hat{f}(x)$ and $\hat{g}(x)$ are defined in (3.4), and noise perturbs $\hat{f}(x)$ and $\hat{g}(x)$ to inexact forms $f(x)$ and $g(x)$. Noise $\delta\hat{a}_i$ and $\delta\hat{b}_j$ is added in the componentwise sense to the exact coefficients $\hat{a}_i$ and $\hat{b}_j$ of $\hat{f}(x)$ and $\hat{g}(x)$, and thus the coefficients $\hat{a}_i$ and $\hat{b}_j$ are perturbed to $\hat{a}_i+\delta\hat{a}_i$, $i = 0, \ldots, m$, and $\hat{b}_j+\delta\hat{b}_j$, $j = 0, \ldots, n$, respectively,

$$\hat{a}_i + \delta\hat{a}_i = \hat{a}_i(1 + r_i\varepsilon_c) \qquad \text{and} \qquad \hat{b}_j + \delta\hat{b}_j = \hat{b}_j(1 + r_j\varepsilon_c), \tag{4.1}$$

where r_i and r_j are uniformly distributed random variables in the range $[-1, \ldots, +1]$, and $1/\varepsilon_c$ is the upper bound of the componentwise signal-to-noise ratio. It follows from (4.1) that

$$\frac{1}{\varepsilon_c} \leq \frac{|\hat{a}_i|}{|\delta\hat{a}_i|} \qquad \text{and} \qquad \frac{1}{\varepsilon_c} \leq \frac{|\hat{b}_j|}{|\delta\hat{b}_j|},$$

for $i = 0, \ldots, m$, and $j = 0, \ldots, n$. Therefore, the coefficients $\hat{a}_i$ of $\hat{f}(x)$ and $\hat{b}_j$ of $\hat{g}(x)$ are replaced by the coefficients of their inexact polynomials $f(x)$ and $g(x)$ respectively,

$$\hat{a}_i \to \hat{a}_i(1 + r_i\varepsilon_c), \qquad i = 0, \ldots, m,$$

$$\hat{b}_j \to \hat{b}_j(1 + r_j\varepsilon_c), \qquad j = 0, \ldots, n,$$

and the inexact polynomials $f(x)$ and $g(x)$ are

$$f(x) = \sum_{i=0}^{m} a_i \binom{m}{i} (1-x)^{m-i} x^i \qquad \text{and} \qquad g(x) = \sum_{j=0}^{n} b_j \binom{n}{j} (1-x)^{n-j} x^j. \tag{4.2}$$

In this thesis, we set the componentwise signal-to-noise ratio ε_c^{-1} equal to 10^8 because this level of signal-to-noise ratio is typical in practical examples [63, 64].

The next section considers the computation of the GCD of $\hat{f}(x)$ and $\hat{g}(x)$ from their inexact forms $f(x)$ and $g(x)$, using Euclid's algorithm, and the Bézout and Sylvester resultant matrices, in a floating point environment.

4.2 Computation of GCD of polynomials from their inexact forms

This section considers computing the GCD of the polynomials $\hat{f}(x)$ and $\hat{g}(x)$ from their inexact forms $f(x)$ and $g(x)$ defined in (4.2), using Euclid's algorithm, and the Bézout and Sylvester resultant matrices in a floating point environment.

Euclid′s algorithm

Section 3.1 described Euclid's algorithm in a symbolic environment. However, when Euclid's algorithm is implemented in a floating point environment, the vanishing remainder termination criterion is never satisfied precisely due to round off error generated in each division. Therefore, more concern should be given to the implementation of Euclid's algorithm in a floating point environment.

One reasonable termination criterion of Euclid's algorithm performed in a floating point environment is to test the norm of remainder $\|\varnothing_{r+1}\|$ at each division against a prescribed tolerance ϵ. If $\|\varnothing_{r+1}\|$ is less than the tolerance ϵ, the division sequence stops and $\varnothing_r(x)$ is the GCD of polynomials. However, it is shown in [53] that the remainder norm experiences dramatic and unpredictable changes at each division and

thus its comparison with a specified tolerance ϵ is not a reliable indicator for terminating Euclid's algorithm.

An approach based on Euclid's algorithm to calculate the GCD of two Bernstein polynomials is presented in [53]. Given two Bernstein polynomials $\hat{f}(x)$ and $\hat{g}(x)$, the approach firstly normalizes the coefficients of each polynomial by the L_2 norm of its coefficients, $\|P\|$. The square of the L_2 norm of the polynomial coefficients, $\|P\|^2$, is given by

$$\|P\|^2 = \frac{1}{2n+1} \sum_{i=0}^{n} \sum_{j=0}^{n} \frac{\binom{n}{i}\binom{n}{j}}{\binom{2n}{i+j}} C_i^n C_j^n, \tag{4.3}$$

where n is the degree of polynomial, $\binom{n}{k}$ is the binomial coefficient, and C_k^n is the coefficient of polynomial.

Then, the approach in [53], applies the division sequence (3.1) to $\hat{f}(x)$ and $\hat{g}(x)$. At each division, instead of comparing the remainder norm with the specified tolerance ϵ, the approach divides $\hat{f}(x)$ and $\hat{g}(x)$ by $\emptyset_r(x)$ respectively

$$\begin{aligned} \hat{f}(x) &= q_1(x)\emptyset_r(x) + r_1(x), \\ \hat{g}(x) &= q_2(x)\emptyset_r(x) + r_2(x). \end{aligned}$$

If both remainder norms, $\|r_1\|$ and $\|r_2\|$ are less than a specified tolerance ϵ, the division sequence stops and $\emptyset_r(x)$ is the GCD of the polynomials. The reason is that the divisor $\emptyset_r(x)$ at each division is a candidate GCD of $\hat{f}(x)$ and $\hat{g}(x)$, and therefore the divisor $\emptyset_r(x)$ that divides both polynomials with remainders whose norms are sufficiently small is the GCD of polynomials.

Example 4.1. Consider the Bernstein forms of the exact polynomials

$$\hat{f}(x) = (x-0.17)^4(x-0.56)^4(x-0.72)^2,$$

and

$$\hat{g}(x) \;=\; (x-0.17)^3(x-0.35)^4(x-0.91)^2,$$

whose GCD is of degree 3.

Because of noise, the coefficients of $\hat{f}(x)$ and $\hat{g}(x)$ are not specified exactly. Therefore, adding noise with componentwise signal-to-noise ratio $\varepsilon_c^{-1} = 10^8$ to the coefficients of $\hat{f}(x)$ and $\hat{g}(x)$, we obtain their inexact forms $f(x)$ and $g(x)$. Applying Euclid's algorithm described above with tolerance $\epsilon = 10^{-8}$ to $f(x)$ and $g(x)$ yields the result shown in Table 4.1. It is seen from Table 4.1 that the algorithm stops at stage 10, and yields the GCD of degree 0, which implies that $f(x)$ and $g(x)$ are coprime.

Table 4.1: Remainder norms on dividing $f(x)$ and $g(x)$ by $\emptyset_r(x)$ in Example 4.1.

Stage	Divisor Degree	$\|r_1\|$	$\|r_2\|$
1	9	0.8442	3.7396×10^{-15}
2	8	0.2512	1.9233
3	7	4.1573×10^{3}	2.0070×10^{3}
4	6	0.1906	0.1880
5	5	0.0117	1.2119
6	4	0.0043	0.0246
7	3	3.1435×10^{-4}	0.0011
8	2	0.0027	8.4693×10^{-5}
9	1	1.6985×10^{-5}	6.1524×10^{-5}
10	0	**0**	**0**

□

This example shows that when the coefficients of $\hat{f}(x)$ and $\hat{g}(x)$ are perturbed by noise, Euclid's algorithm fails to compute the GCD of $\hat{f}(x)$ and $\hat{g}(x)$ from their inexact forms $f(x)$ and $g(x)$ because minor noise makes $f(x)$ and $g(x)$ coprime. In addition, even if Euclid's algorithm is applied to polynomials in the absence of noise,

the result obtained is dependent on the selection of the tolerance because of roundoff error.

Next, we consider the computation of the GCD of polynomials in the presence of noise, using the Bézout and Sylvester resultant matrices. As stated earlier, given two polynomials, the rank loss of their resultant matrices is equal to the degree of their GCD. However, for the GCD computation using the resultant matrices performed in a floating point environment, in most cases, a row of a matrix will never vanish identically, because of roundoff error, and thus we adopt a method that observes the variation of normalized singular values of the resultant matrices in order to determine the degree of the GCD [11, 15, 26]. In particular, given a matrix $A \in \mathbb{R}^{m\times n}$, where $m > n$, applying singular value decomposition to the matrix A obtains

$$A = USV,$$

where $U \in \mathbb{R}^{m\times m}$ is an orthogonal matrix, $V \in \mathbb{R}^{n\times n}$ is an orthogonal matrix, and $S \in \mathbb{R}^{m\times n}$ has the form

$$\begin{bmatrix} \sigma_1 & 0 & \dots & 0 \\ 0 & \sigma_2 & \dots & 0 \\ & \dots & \dots & \\ 0 & 0 & \dots & \sigma_n \\ 0 & 0 & \dots & 0 \\ & \dots & \dots & \\ 0 & 0 & \dots & 0 \end{bmatrix}.$$

The singular values of the matrix A are $\sigma_i, i = 1, \dots, n$, which are non-negative elements in descending order:

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n \geq 0.$$

Since the orthogonal matrices U and V are non-singular, $\operatorname{rank} A = \operatorname{rank} S$. Therefore, if $\operatorname{rank} A = r < n$, then

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > \sigma_{r+1} = \sigma_{r+2} = \ldots = \sigma_n = 0.$$

Thus, the rank of the matrix A can be determined by counting the number of its non-zero singular values. However, when we apply singular value decomposition to the matrix A in a floating point environment, all of the singular values of the matrix A are not equal to zero due to roundoff error. But $\sigma_{r+1}, \sigma_{r+2}, \ldots, \sigma_n$ are very small and approximately equal to zero, that is

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > \sigma_{r+1} \approx \sigma_{r+2} \approx \ldots \approx \sigma_n \approx 0.$$

Because $\sigma_r > 0$ and $\sigma_{r+1} \approx 0$, there exists a significantly large change between these two successive singular values. Therefore, the rank of the matrix A is equal to the value of i for which the significantly large change between two successive singular values σ_i and σ_{i+1} occurs.

Bézout resultant matrix

Example 4.2. Consider the Bernstein forms of the exact polynomials

$$\hat{f}(x) = (x - 0.36)^4(x - 0.79)^3(x - 1.46)^3,$$

and

$$\hat{g}(x) = (x - 0.36)^2(x - 0.95)^4(x - 1.46)^5,$$

whose GCD is of degree 5.

Noise with componentwise signal-to-noise ratio 10^8 is added to the coefficients of $\hat{f}(x)$ and $\hat{g}(x)$ to obtain their inexact forms $f(x)$ and $g(x)$, and then the Bézout matrix $B(f, g)$ is computed.

Figure 4.1 shows the normalized singular values of $B(f, g)$, and it is seen that $B(f, g)$

is of full rank, which implies that $f(x)$ and $g(x)$ are coprime.

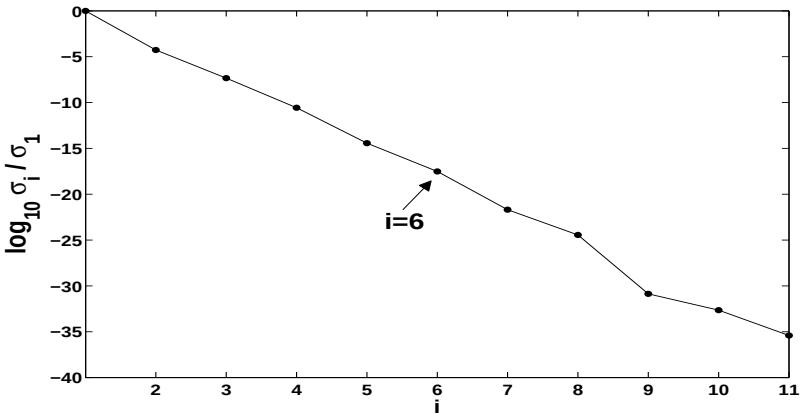


Figure 4.1: The normalized singular values of $B(f, g)$ for Example 4.2.

□

Sylvester resultant matrix

Example 4.3. Consider the Bernstein forms of the exact polynomials

$$\hat{f}(x) = (x - 0.43)^4(x + 0.93)^6(x + 1.47)^5(x - 1.39)^4,$$

and

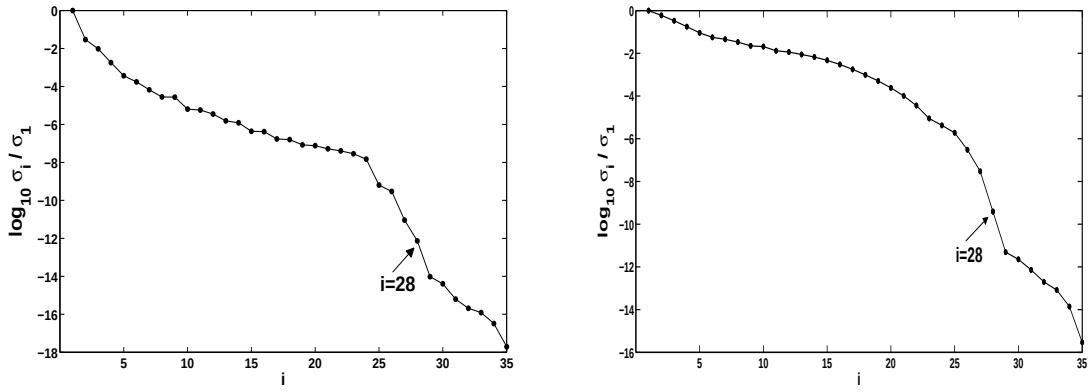
$$\hat{g}(x) = (x - 0.43)^5(x - 0.93)^4(x + 1.47)^3(x - 1.89)^4,$$

whose GCD is of degree 7.

Noise with componentwise signal-to-noise ratio 10^8 is applied to the coefficients of $\hat{f}(x)$ and $\hat{g}(x)$ to obtain their inexact forms $f(x)$ and $g(x)$, and then two forms of the Sylvester matrix $S(f, g)$ and $S(f, g)Q$ are computed.

Figures 4.2(a) and (b) show the normalized singular values of $S(f, g)$ and $S(f, g)Q$

respectively, and it is seen that both $S(f,g)$ and $S(f,g)Q$ are of full rank, which implies that $f(x)$ and $g(x)$ are coprime. □



(a) (b)

Figure 4.2: The normalized singular values of (a) $S(f,g)$ and (b) $S(f,g)Q$ for Example 4.3.

These two examples associated with the Bézout and Sylvester resultant matrices show that when noise is present such that polynomials can not be specified exactly, the Bézout and Sylvester resultant matrices fail to calculate the GCD of polynomials from their inexact forms because minor random noise makes their inexact forms coprime. It is seen from the above examples that if the exact Bernstein polynomials have a non-constant GCD, their inexact forms are coprime with high probability, which makes the computation of the GCD of Bernstein polynomials an ill-posed problem. In this circumstance, the inexact Bernstein polynomials $f(x)$ and $g(x)$ have an approximate greatest common divisor (AGCD) because they are near their theoretically exact forms, $\hat{f}(x)$ and $\hat{g}(x)$ respectively, which are not coprime. Therefore, the GCD of exact Bernstein polynomials is an approximate common divisor of their inexact polynomials. The AGCD is considered in the next section.

4.3 Approximate greatest common divisor

This section discusses an AGCD of inexact polynomials $f(x)$ and $g(x)$, and it will be shown that it differs from the GCD of their exact forms, $\hat{f}(x)$ and $\hat{g}(x)$.

It is assumed that $\hat{f}(x)$ and $\hat{g}(x)$ have a non-constant GCD, and their inexact forms $f(x)$ and $g(x)$ respectively,

$$f(x) = \hat{f}(x) + \delta\hat{f}(x) \qquad \text{and} \qquad g(x) = \hat{g}(x) + \delta\hat{g}(x),$$

are coprime, that is

$$\hat{d} = \deg \mathrm{GCD}(\hat{f}, \hat{g}) > 0 \qquad \text{and} \qquad \deg \mathrm{GCD}(f, g) = 0, \tag{4.4}$$

and

$$d = \deg \mathrm{AGCD}(f, g) > 0. \tag{4.5}$$

It follows from (4.4) and (4.5) that the computation of the GCD of $\hat{f}(x)$ and $\hat{g}(x)$ is an ill-posed problem because random noise imposed on one or both of these polynomials causes the resulting polynomials $f(x)$ and $g(x)$ to be coprime. However, these inexact polynomials are near their theoretically exact forms, which are not coprime, and thus $f(x)$ and $g(x)$ possess an approximate common divisor $h(x)$, that is, $h(x)$ is a polynomial that divides $f(x)$ and $g(x)$ with a small error in each division,

$$f(x) = q_1(x)h(x) + r_1(x) \qquad \text{and} \qquad g(x) = q_2(x)h(x) + r_2(x),$$

where $\|r_1\| \ll \|q_1 h\|$ and $\|r_2\| \ll \|q_2 h\|$.

The GCD of $\hat{f}(x)$ and $\hat{g}(x)$ is a function of the roots of these polynomials, and an AGCD of $f(x)$ and $g(x)$ is a function of the roots of these polynomials, but both these common divisors are independent of arbitrary scalar multipliers that can be

applied to the polynomials. They therefore satisfy

$$\mathrm{GCD}(\hat{f},\hat{g}) = \mathrm{GCD}(\gamma_1\hat{f},\gamma_2\hat{g}) \qquad \text{and} \qquad \mathrm{AGCD}(f,g) = \mathrm{AGCD}(\gamma_3 f,\gamma_4 g),$$

where $\gamma_1,\gamma_2,\gamma_3,\gamma_4 \in \mathbb{R}\backslash 0$.

The GCD of $\hat{f}(x)$ and $\hat{g}(x)$ is unique up to a non-zero scalar multiplier but an AGCD of $f(x)$ and $g(x)$ is not unique. For example, it can be defined as the common divisor polynomial of maximum degree, assumed to be unique, when the magnitude of the perturbations applied to the coefficients of $f(x)$ and $g(x)$ is specified, or the common divisor polynomial, assumed to be unique, obtained when the perturbations of the coefficients of $f(x)$ and $g(x)$ have minimum magnitude, such that the degree of the common divisor polynomial is specified. It follows that there are several definitions of an AGCD in [11, 15, 32, 33, 42]. Each of those definitions formulates the concept with some of the three characteristics as follows [68]:

(a) Nearness: An AGCD is the GCD of another set of polynomials near the given ones.

(b) Max-degree: The AGCD has the highest degree among those polynomials satisfying nearness.

(c) Min-distance: The AGCD minimizes the distance between another set of polynomials and the given ones as mentioned in (a).

Each of these definitions is valid, and the definition used depends on the problem to be solved.

The computation of an AGCD of inexact Bernstein polynomials is rarely considered. However, substantial works have been spent on developing algorithms for calculating

an AGCD of inexact power polynomials. It is shown in [4, 32, 39, 43] that modifications of the Euclidean algorithm are used in order that it can be applied to floating point numbers. In addition, the QR factorization is considered in [12, 67]. For example, the approach in [67] involves the QR decomposition of a matrix derived from the Sylvester matrix of two polynomials, and an incremental condition estimator is used to determine the last non-zero row of the upper triangular matrix R in the QR decomposition. Furthermore, optimization strategies are used in [9, 33]. For example, the approach shown in [9] formulates and solves a nonlinear least squares problem to determine the coefficients of an AGCD, assuming that an estimate for its degree is available. Finally, structured matrix methods involve constructing the resultant matrices of polynomials and computing the structured low rank approximations of the resultant matrices, which yield an AGCD of the polynomials [36, 52, 59, 60]. One algorithm for computing an AGCD of inexact power polynomials used by Zeng is now described [68]:

For any power polynomial

$$\hat{l}(x) = \sum_{i=0}^{m} \hat{l}_i x^i,$$

its kth convolution matrix is

$$C_k(\hat{l}) = \begin{bmatrix} \hat{l}_0 & & \\ \hat{l}_1 & \ddots & \\ \vdots & \ddots & \hat{l}_0 \\ \hat{l}_m & & \hat{l}_1 \\ & \ddots & \vdots \\ & & \hat{l}_m \end{bmatrix} \in \mathbb{R}^{(m+k+1)\times(k+1)}, \qquad k = 0, 1, \cdots.$$

For any polynomials $\hat{p}(x)$ and $\hat{q}(x)$ of degrees j and k respectively, if $\hat{h}(x) = \hat{p}(x)\hat{q}(x)$, then

$$\hat{\mathbf{h}} = C_k(\hat{p})\hat{\mathbf{q}} = C_j(\hat{q})\hat{\mathbf{p}},$$

where $\hat{\mathbf{h}}$, $\hat{\mathbf{p}}$ and $\hat{\mathbf{q}}$ are the vectors containing the coefficients of polynomials $\hat{h}(x)$, $\hat{p}(x)$ and $\hat{q}(x)$.

Consider two exact polynomials $\hat{f}(x)$ and $\hat{g}(x)$

$$\hat{f}(x) = \sum_{i=0}^{m} \hat{a}_i x^i \qquad \text{and} \qquad \hat{g}(x) = \sum_{j=0}^{n} \hat{b}_j x^j,$$

which have a non-constant common divisor $\hat{d}_k(x)$ of degree k. Therefore, there exist quotient polynomials $\hat{u}_k(x)$ and $\hat{v}_k(x)$ such that

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

where

$$\hat{u}_k(x) = \sum_{i=0}^{m-k} \hat{u}_{k,i} x^i \qquad \text{and} \qquad \hat{v}_k(x) = \sum_{i=0}^{n-k} \hat{v}_{k,i} x^i.$$

Then a quadratic system can be established, that is

$$F(\hat{\mathbf{z}}) = \hat{\mathbf{b}}, \tag{4.6}$$

where

$$F(\hat{\mathbf{z}}) = \begin{bmatrix} \mathbf{r}^H \hat{\mathbf{d}}_\mathbf{k} - 1 \\ C_k(\hat{u}_k)\hat{\mathbf{d}}_\mathbf{k} \\ C_k(\hat{v}_k)\hat{\mathbf{d}}_\mathbf{k} \end{bmatrix}, \qquad \hat{\mathbf{z}} = \begin{bmatrix} \hat{\mathbf{d}}_\mathbf{k} \\ \hat{\mathbf{u}}_\mathbf{k} \\ \hat{\mathbf{v}}_\mathbf{k} \end{bmatrix}, \qquad \hat{\mathbf{b}} = \begin{bmatrix} 0 \\ \hat{\mathbf{f}} \\ \hat{\mathbf{g}} \end{bmatrix}. \tag{4.7}$$

The vector $\mathbf{r}$ in (4.7) is a scaling vector and $\mathbf{r}^H$ is the Hermitian adjoint of $\mathbf{r}$. The vectors $\hat{\mathbf{f}}$, $\hat{\mathbf{g}}$, $\hat{\mathbf{u}}_\mathbf{k}$, $\hat{\mathbf{v}}_\mathbf{k}$ and $\hat{\mathbf{d}}_\mathbf{k}$ store the coefficients of the polynomials $\hat{f}(x)$, $\hat{g}(x)$, $\hat{u}_k(x)$, $\hat{v}_k(x)$ and $\hat{d}_k(x)$, and (4.6) is solved for $\hat{\mathbf{u}}_\mathbf{k}$, $\hat{\mathbf{v}}_\mathbf{k}$ and $\hat{\mathbf{d}}_\mathbf{k}$.

However, when their inexact polynomials $f(x)$ and $g(x)$ are specified, (4.6) is replaced

by the approximation

$$F(\mathbf{z}) \approx \mathbf{b}, \tag{4.8}$$

because $f(x)$ and $g(x)$ are coprime. The vectors $F(\mathbf{z})$, $\mathbf{z}$ and $\mathbf{b}$ are given by

$$F(\mathbf{z}) = \begin{bmatrix} \mathbf{r}^H\mathbf{d_k} - 1 \\ C_k(u_k)\mathbf{d_k} \\ C_k(v_k)\mathbf{d_k} \end{bmatrix}, \qquad \mathbf{z} = \begin{bmatrix} \mathbf{d_k} \\ \mathbf{u_k} \\ \mathbf{v_k} \end{bmatrix}, \qquad \mathbf{b} = \begin{bmatrix} 0 \\ \mathbf{f} \\ \mathbf{g} \end{bmatrix}. \tag{4.9}$$

In this algorithm, $d_k(x)$ is an AGCD of the inexact polynomials $f(x)$ and $g(x)$ with tolerance $\varepsilon > 0$, if $d_k(x)$ is of the highest degree k along with quotient polynomials $u_k(x)$ of degree $m-k$ and $v_k(x)$ of degree $n-k$ that form $\mathbf{z}$ in (4.9) satisfying $\| F(\mathbf{z}) - \mathbf{b} \|_2 \leq \varepsilon$.

As stated earlier, if two polynomials have a non-constant common divisor of degree k, their kth subresultant matrix is rank deficient. Since the inexact polynomials $f(x)$ and $g(x)$ are coprime, their subresultant matrices are all of full rank. This algorithm computes the smallest singular value σ_k of the kth subresultant matrix $S_k(f,g)$. When $\sigma_k \leq \varepsilon\sqrt{2k+2}$ occurs, the algorithm assumes $S_k(f,g)$ is close to be rank deficient and thus there is a possibility that $f(x)$ and $g(x)$ have an approximate common divisor of degree k within tolerance ε. The algorithm proceeds as follows:

Consider two inexact polynomials $f(x)$ of degree m and $g(x)$ of degree n, where it is assumed $m \geq n$. As stated earlier, the algorithm looks for an AGCD of the highest degree satisfying the specified tolerance, and the possible highest degree of an AGCD of $f(x)$ and $g(x)$ is the smaller number of m and n. Since it is assumed $m \geq n$, the algorithm first sets $k = n$ and computes the smallest singular value σ_n of $S_n(f,g)$. If $\sigma_n \leq \varepsilon\sqrt{2n+2}$, the approximation (4.8) is established. Then the vector $\mathbf{z}$ is refined

iteratively by Gauss-Newton iteration, that is

$$\mathbf{z}_{j+1} = \mathbf{z}_j - J(\mathbf{z}_j)^+ \left[F(\mathbf{z}_j) - \mathbf{b}\right],$$

where

$$J(\mathbf{z}) = \begin{bmatrix} \mathbf{r}^H & & \\ C_k(u_k) & C_{m-k}(d_k) & \\ C_k(v_k) & & C_{m-k}(d_k) \end{bmatrix},$$

is the Jacobian of $F(\mathbf{z})$ and $J(\mathbf{z})^+ = \left(J(\mathbf{z})^H J(\mathbf{z})\right)^{-1} J(\mathbf{z})^H$ is the pseudo-inverse of $J(\mathbf{z})$.

This iterative refinement terminates when the distance $\varsigma_n \equiv \| F(\mathbf{z}_j) - \mathbf{b} \|_2$ stops decreasing, and then this refinement stage outputs the nearness ς_n and the refined polynomials $d_k(x)$, $u_k(x)$ and $v_k(x)$ embedded in $\mathbf{z} = \mathbf{z}_j$. If $\varsigma_n < \varepsilon$, then $d_k(x)$ is certified as an AGCD of $f(x)$ and $g(x)$, and the algorithm stops. If $\varsigma_n \geq \varepsilon$, which implies that there is no approximate common divisor of degree n satisfying $\| F(\mathbf{z}) - \mathbf{b} \|_2 \leq \varepsilon$, the algorithm then looks for an approximate common divisor of lower degree $n-1$. Therefore, the algorithm sets $k = n-1$ and repeats the above process.

From the above discussion, the computation of an AGCD needs the definition of an AGCD to be specified. This is considered in the next section.

4.4 The definition of an AGCD

In the previous section, the concept of an AGCD has been introduced. It has been mentioned that several definitions of an AGCD exist [11, 15, 32, 33, 42], and they use one or more of the characteristics of nearness, maximum degree and minimum

distance. For example, Zeng's work looks for an AGCD of maximum degree that satisfies an error criterion as described in Section 4.3. However, these definitions are not appropriate for the approach that will be introduced in this thesis because these definitions of an AGCD use an error criterion based on the coefficients of an AGCD. In this thesis, however, the degree d of an AGCD is computed initially, after which the coefficients of an AGCD of degree d are calculated. A test for the correctness of d can not be based on the coefficients of an AGCD, and must be based only on d. The following definition of an AGCD is therefore used in this thesis.

DEFINITION. *The degree d of an AGCD of two inexact polynomials $f(x)$ and $g(x)$ is defined to be correct when it is equal to the degree $\hat{d}$ of the GCD of the exact forms $\hat{f}(x)$ and $\hat{g}(x)$ of $f(x)$ and $g(x)$, respectively.*

This definition of the degree of an AGCD is required because it provides a good measure of the ability of the proposed approach to compute $\hat{d}$, and therefore reproduce in the given inexact polynomials $f(x)$ and $g(x)$ an important property of the exact polynomials $\hat{f}(x)$ and $\hat{g}(x)$.

It is seen from the definition of an AGCD that because the degree d of an AGCD of $f(x)$ and $g(x)$ is defined to be correct when it is equal to the degree $\hat{d}$ of the GCD of their exact forms $\hat{f}(x)$ and $\hat{g}(x)$, the estimate of the degree $\hat{d}$ of the GCD of $\hat{f}(x)$ and $\hat{g}(x)$ is equivalent to the determination of the degree d of an AGCD of $f(x)$ and $g(x)$.

It was stated above that the computation of an AGCD of $f(x)$ and $g(x)$ proceeds in two steps:

1. Calculate the degree d of an AGCD of $f(x)$ and $g(x)$.

2. Given d, calculate the corrections that must be added to the coefficients of $f(x)$ and $g(x)$, such that these corrected polynomials have a GCD of degree d.

For the first step, the degree d of an AGCD of $f(x)$ and $g(x)$ can be determined using the Bézout and Sylvester resultant matrices but the preprocessing operations that will be introduced in the following chapters must be performed on the resultant matrices such that they are more stable computationally. In addition, the advanced methods using the first principal angle and the residual of an approximate linear algebraic equation are also adopted to determine the degree d of an AGCD of $f(x)$ and $g(x)$. After the first step, based on the estimated degree d of an AGCD of $f(x)$ and $g(x)$, the perturbations of minimum magnitude applied to the coefficients of $f(x)$ and $g(x)$ are computed by the method of structured non-linear total least norm (SNTLN) [41]. This approach must be compared with Zeng's method, which computes AGCDs of degrees $\min(m,n)$, $\min(m,n)-1$, ..., until an error criterion on the coefficients of the AGCD is satisfied. It therefore follows that several possible AGCDs are computed, but the approach described in this thesis requires that only one AGCD is computed, which is therefore more efficient.

From the above analysis, we should initially address the determination of the degree d of an AGCD of $f(x)$ and $g(x)$. In particular, this is the most difficult and crucial part of the calculation of an AGCD. Three methods will be discussed in the following chapters respectively.

4.5 Summary

This chapter has shown that when noise is present such that polynomials can not be specified exactly, the classical algorithms, Euclid's algorithm, and the Bézout and

Sylvester resultant matrices, can not be used to compute the GCD of polynomials in a floating point environment because random noise makes their inexact forms coprime. Therefore, the concept of an AGCD was introduced. It follows from the definition of an AGCD specified in Section 4.4 that the computation of an AGCD should firstly determine the degree of an AGCD. The next chapter will consider the determination of the degree of an AGCD of two inexact polynomials using the Bézout resultant matrix, and Chapters 6 and 7 will consider the method based on the Sylvester matrix and advanced methods using the first principal angle and the residual of an approximate linear algebraic equation respectively.

Chapter 5

The degree of an AGCD, Part I

This chapter introduces the method for the computation of the degree of an AGCD of inexact polynomials from their Bézout resultant matrix. It has been shown that minor random noise added to exact polynomials makes them coprime, and thus their Bézout resultant matrix is of full rank. In order to determine the correct degree of an AGCD of inexact polynomials, one preprocessing operation must be performed on the Bézout resultant matrix. Experiments show that this preprocessing operation is essential for the accurate estimate of the degree of an AGCD of inexact polynomials. This preprocessing operation is discussed in the next section.

5.1 Preprocessing operation

It is shown in [24] that computations on a matrix whose entries vary widely in magnitude may be numerically unstable and therefore it is desirable to minimize the ratio of the maximum entry, in magnitude, to the minimum entry, in magnitude. The minimization of the ratio of the maximum and minimum entries of matrix, in magnitude,

can be achieved by one preprocessing operation. In particular, the preprocessing operation introduces a parameter θ which transforms the independent variable x to a new independent variable w. It will be shown that the optimal value of the parameter θ can be easily calculated by solving a standard linear programming problem.

5.1.1 The transformation of the independent variable

Given two inexact Bernstein polynomials $f(x)$ and $g(x)$ defined in (4.2), it was noted in Section 3.2 that the polynomial $g(x)$ must be degree elevated $m-n$ times since it is assumed that $m \geq n$, and it is therefore assumed in this section that both $f(x)$ and $g(x)$ are of degree m,

$$f(x) = \sum_{i=0}^{m} a_i \binom{m}{i} (1-x)^{m-i} x^i \qquad \text{and} \qquad g(x) = \sum_{i=0}^{m} b_i \binom{m}{i} (1-x)^{m-i} x^i.$$

The preprocessing operation is achieved by the transformation

$$x = \theta w, \tag{5.1}$$

where θ is a parameter whose value is to be determined and w is the new independent variable. The polynomials $f(x)$ and $g(x)$ are then transformed to

$$\vec{f}(w,\theta) = \sum_{i=0}^{m} \left(a_i \theta^i\right) \binom{m}{i} (1-\theta w)^{m-i} w^i, \tag{5.2}$$

and

$$\vec{g}(w,\theta) = \sum_{i=0}^{m} \left(b_i \theta^i\right) \binom{m}{i} (1-\theta w)^{m-i} w^i, \tag{5.3}$$

respectively, which are expressed in the modified Bernstein basis, the basis functions of which for a polynomial of degree m are $\phi_i^m(w,\theta)$,

$$\phi_i^m(w,\theta) = \binom{m}{i} (1-\theta w)^{m-i} w^i, \qquad i = 0, \ldots, m. \tag{5.4}$$

The coefficients of $\vec{f}(w,\theta)$ and $\vec{g}(w,\theta)$ are $a_i\theta^i$ and $b_i\theta^i$, $i = 0,\ldots,m$. Therefore, the optimal value of θ must be defined, which allows the coefficients of $\vec{f}(w,\theta)$ and $\vec{g}(w,\theta)$ to be calculated. In particular, the optimal value of θ is determined such that the ratio of maximum and minimum entries of the Bézout matrix of $\vec{f}(w,\theta)$ and $\vec{g}(w,\theta)$, in magnitude, is minimized, which requires that the Bézout matrix of $\vec{f}(w,\theta)$ and $\vec{g}(w,\theta)$ defined in the modified Bernstein basis be developed. This issue is addressed in the next section.

5.1.2 The Bézout resultant matrix for the modified Bernstein basis

It is shown in [5] that the Bézout resultant matrix $B(f,g) \in \mathbb{R}^{m\times m}$ of the Bernstein polynomials $f(x)$ and $g(x)$ satisfies

$$\frac{f(x)g(l) - f(l)g(x)}{x-l} = B^{(m-1)^T}(x)B(f,g)B^{m-1}(l), \tag{5.5}$$

where

$$B^{(m-1)^T}(x) = \begin{bmatrix} B_0^{m-1}(x) & B_1^{m-1}(x) & \cdots & B_{m-1}^{m-1}(x) \end{bmatrix} \in \mathbb{R}^m,$$

and $B_i^{m-1}(x)$ is the ith Bernstein basis function for polynomials of degree $m-1$,

$$B_i^{m-1}(x) = \binom{m-1}{i}(1-x)^{m-1-i}x^i, \qquad i = 0,\ldots,m-1.$$

The Bernstein basis functions and the modified Bernstein basis functions are related by

$$B_i^{m-1}(\theta w) = \binom{m-1}{i}(1-\theta w)^{m-1-i}(\theta w)^i = \theta^i\left(\phi_i^{m-1}(w,\theta)\right),$$

for $i = 0,\ldots,m-1$, and thus (5.1) and the substitution $l = \theta z$ transform (5.5) to

$$\frac{\vec{f}(w,\theta)\vec{g}(z,\theta) - \vec{f}(z,\theta)\vec{g}(w,\theta)}{\theta(w-z)} = \left(\phi^{(m-1)^T}(w,\theta)\right)C(f,g,\theta)\left(\phi^{m-1}(z,\theta)\right), \tag{5.6}$$

where $C(f,g,\theta) \in \mathbb{R}^{m\times m}$ is given by

$$C(f,g,\theta) = H(\theta)B(f,g)H(\theta), \quad H(\theta) = \operatorname{diag}\begin{bmatrix} 1 & \theta & \cdots & \theta^{m-1} \end{bmatrix} \in \mathbb{R}^{m\times m}, \quad (5.7)$$

and

$$\phi^{m-1}(w,\theta) = \begin{bmatrix} \phi_0^{m-1}(w,\theta) & \phi_1^{m-1}(w,\theta) & \cdots & \phi_{m-1}^{m-1}(w,\theta) \end{bmatrix}^T \in \mathbb{R}^m.$$

It follows from (5.7) that $C(f,g,\theta)$ is the Bézout resultant matrix of $\vec{f}(w,\theta)$ and $\vec{g}(w,\theta)$

$$C(f,g,\theta) = H(\theta)B\big(f,g\big)H(\theta), \quad (5.8)$$

and thus the Bézout resultant matrix of two polynomials expressed in the modified Bernstein basis is obtained by pre- and post-multiplying the Bézout resultant matrix of the Bernstein forms of the polynomials by the diagonal matrix $H(\theta)$, which allows the optimal value of θ to be computed.

5.1.3 The optimal value of θ

The calculation of the optimal value of θ requires a general expression for the entries of $C(f,g,\theta)$. In particular, it follows from (5.7) and (5.8) that element (i,j) of $C(f,g,\theta)$ is given by

$$C(f,g,\theta) = b_{i,j}\theta^{i+j-2}, \qquad i,j = 1,\ldots,m,$$

where $b_{i,j}$, which is defined in (3.3), is element (i,j) of the Bernstein Bézout resultant matrix $B(f,g)$. Since θ_0, the optimal value of θ, minimizes the ratio of the maximum element, in magnitude, to the minimum element, in magnitude, of $C(f,g,\theta)$, it follows that

$$\theta_0 = \arg\min_{\theta} \left\{ \frac{\max_{i=1,\ldots,m;j=i,\ldots,m} |b_{i,j}\theta^{i+j-2}|}{\min_{i=1,\ldots,m;j=i,\ldots,m} |b_{i,j}\theta^{i+j-2}|} \right\}. \quad (5.9)$$

This minimization problem can be written as:

Minimize $\frac{u}{v}$

subject to

$$\begin{aligned} u &\geq \left|b_{i,j}\theta^{i+j-2}\right|, \qquad i=1,\ldots,m; j=i,\ldots,m, \\ v &\leq \left|b_{i,j}\theta^{i+j-2}\right|, \qquad i=1,\ldots,m; j=i,\ldots,m, \\ v &> 0, \\ \theta &> 0. \end{aligned} \tag{5.10}$$

The substitutions

$$U=\log u, \qquad V=\log v, \qquad \phi=\log\theta \qquad \text{and} \qquad \beta_{i,j}=\log\left|b_{i,j}\right|, \tag{5.11}$$

where $\log=\log_{10}$, enable the minimization problem (5.10) to be written as

Minimize $U-V$

subject to

$$\begin{aligned} U-(i+j-2)\phi &\geq \beta_{i,j}, \qquad i=1,\ldots,m; j=i,\ldots,m, \\ -V+(i+j-2)\phi &\geq -\beta_{i,j}, \qquad i=1,\ldots,m; j=i,\ldots,m, \end{aligned}$$

which can be expressed as

$$\text{Minimize} \quad \begin{bmatrix} U & V & \phi \end{bmatrix} \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix} \quad \text{subject to} \quad A \begin{bmatrix} U \\ V \\ \phi \end{bmatrix} \geq b, \tag{5.12}$$

where $A \in \mathbb{R}^{r\times 3}$, $b \in \mathbb{R}^r$ and $r=m(m+1)$. Equation (5.12) can be solved using linear programming.

If ϕ_0 is the solution of (5.12), then it follows from (5.9) and (5.11) that the optimal

value of θ is equal to $\theta_0 = 10^{\phi_0}$. Therefore, all computations are performed on the Bézout resultant matrix $\bar{B}(\check{f}, \check{g})$, which is given by

$$\bar{B}\left(\check{f}, \check{g}\right) = H(\theta_0) B\big(f, g\big) H(\theta_0),$$

where

$$\check{f} = \check{f}(w) = \vec{f}(w, \theta_0) = \sum_{i=0}^{m} \left(a_i \theta_0^i\right) \binom{m}{i} (1 - \theta_0 w)^{m-i} w^i, \tag{5.13}$$

and

$$\check{g} = \check{g}(w) = \vec{g}(w, \theta_0) = \sum_{i=0}^{m} \left(b_i \theta_0^i\right) \binom{m}{i} (1 - \theta_0 w)^{m-i} w^i. \tag{5.14}$$

Example 5.1. Consider two Bernstein polynomials

$$f(x) \;=\; \frac{1}{2}\binom{3}{0}(1-x)^3 - \frac{1}{4}\binom{3}{1}(1-x)^2 x + \frac{1}{4}\binom{3}{3}x^3,$$

and

$$g(x) \;=\; \binom{2}{0}(1-x)^2 - \frac{1}{4}\binom{2}{1}(1-x)x - \frac{1}{2}\binom{2}{2}x^2,$$

whose GCD is $g(x)$ because

$$f(x) = g(x)\left(\frac{1}{2}\binom{1}{0}(1-x) - \frac{1}{2}\binom{1}{1}x\right).$$

If the optimal value of θ is $\theta_0 = 2$, it follows that

$$\check{f}(w) = \vec{f}(w, \theta_0) \;=\; \frac{1}{2}\binom{3}{0}(1-2w)^3 - \frac{1}{2}\binom{3}{1}(1-2w)^2 w + 2\binom{3}{3}w^3,$$

and

$$\check{g}(w) = \vec{g}(w, \theta_0) \;=\; \binom{2}{0}(1-2w)^2 - \frac{1}{2}\binom{2}{1}(1-2w)w - 2\binom{2}{2}w^2.$$

The Bézout resultant matrix $\bar{B}\left(\check{f},\check{g}\right)$ of $\check{f}(w)$ and $\check{g}(w)$ is

$$\begin{aligned}\bar{B}\left(\check{f},\check{g}\right) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 4 \end{bmatrix} \begin{bmatrix} -1 & \frac{1}{4} & \frac{1}{2} \\ \frac{1}{4} & -\frac{1}{16} & -\frac{1}{8} \\ \frac{1}{2} & -\frac{1}{8} & -\frac{1}{4} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 4 \end{bmatrix} \\ &= \begin{bmatrix} -1 & \frac{1}{2} & 2 \\ \frac{1}{2} & -\frac{1}{4} & -1 \\ 2 & -1 & -4 \end{bmatrix}.\end{aligned}$$

The reduction of this matrix to row echelon (upper triangular) form yields

$$\begin{bmatrix} 2 & -1 & -4 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

from which it follows that the degree of the GCD of $\check{f}(w)$ and $\check{g}(w)$ is two. The polynomial formed from the last non-zero row of this matrix is

$$2\binom{2}{0}(1-2w)^2 - \binom{2}{1}(1-2w)w - 4\binom{2}{2}w^2,$$

which is proportional to the GCD of $\check{f}(w)$ and $\check{g}(w)$.

It is readily verified that the substitution $w = x/\theta = x/2$ yields $g(x)$. □

5.2 Examples

This section includes three examples to illustrate the computation of the degree d of an AGCD of inexact polynomials $f(x)$ and $g(x)$ using their Bézout resultant matrix $\bar{B}\left(\check{f},\check{g}\right)$. The comparison of the results obtained from the Bézout resultant matrix $B(f,g)$ and $\bar{B}\left(\check{f},\check{g}\right)$ is also considered.

Example 5.2. Consider the Bernstein forms of the exact polynomials

$$\hat{f}(x) = (x-0.4)^2(x-0.8)^4(x-0.9)^5(x-1.3)^4(x-2.3)^4,$$

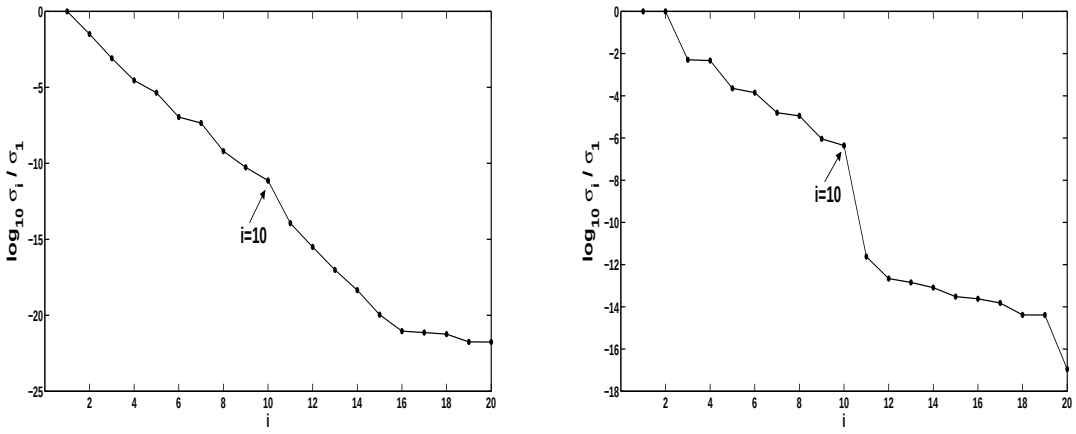
and

$$\hat{g}(x) = (x-0.4)^4(x-0.6)^3(x-0.9)^4(x+1)^4(x-2.3)^5,$$

whose GCD is of degree 10.

Noise with componentwise signal-to-noise ratio 10^8 is added to the coefficients of $\hat{f}(x)$ and $\hat{g}(x)$, and thus we obtain their inexact forms $f(x)$ and $g(x)$. The matrices $B(f,g)$ and $\bar{B}\left(\check{f},\check{g}\right)$ are then computed.

Figures 5.1(a) and (b) show the normalized singular values of $B(f,g)$ and $\bar{B}\left(\check{f},\check{g}\right)$ respectively. It is seen from Figure 5.1(b) that the rank loss of the Bézout matrix $\bar{B}\left(\check{f},\check{g}\right)$ is equal to $\deg \mathrm{GCD}(\hat{f},\hat{g}) = 10$. The result in Figure 5.1(b) was obtained with $\theta_0 = 2.1912$. However, Figure 5.1(a) shows that the Bézout matrix $B(f,g)$ is of full rank, which suggests that $\hat{f}(x)$ and $\hat{g}(x)$ are coprime. □



(a) (b)

Figure 5.1: The normalized singular values of (a) $B(f,g)$ and (b) $\bar{B}\left(\check{f},\check{g}\right)$ for Example 5.2.

Example 5.3. Consider the Bernstein forms of the exact polynomials

$$\hat{f}(x) = (x-0.45)^4(x-0.98)^4(x-1.23)^6(x-2.34)^3,$$

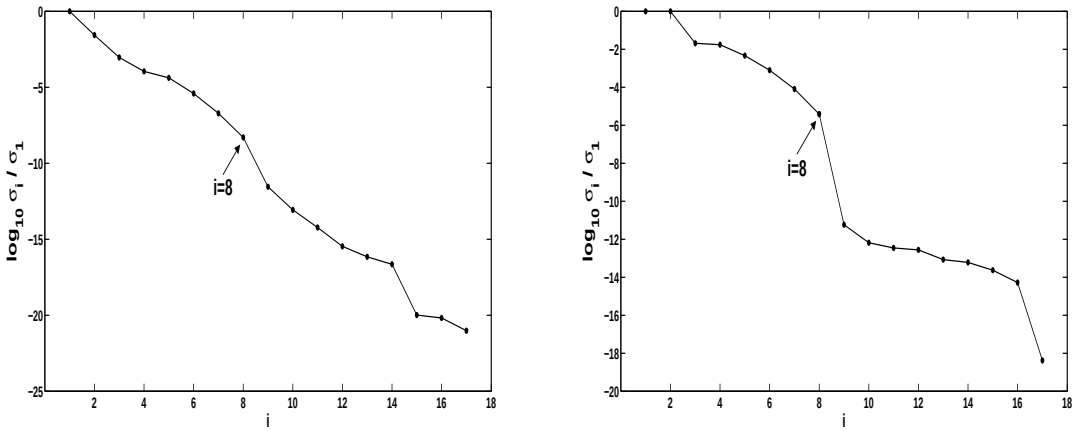
and

$$\hat{g}(x) = (x-0.45)^5(x-0.98)^2(x+1.19)^5(x-2.34)^3,$$

whose GCD is of degree 9.

Each polynomial is corrupted by noise with componentwise signal-to-noise ratio 10^8 to yield their inexact forms $f(x)$ and $g(x)$, and then the matrices $B(f,g)$ and $\bar{B}\left(\check{f},\check{g}\right)$ are computed.

Figure 5.2(b) shows that the rank of the Bézout matrix $\bar{B}\left(\check{f},\check{g}\right)$ is clearly defined and equal to 8, which is correct because $\deg\mathrm{GCD}(\hat{f},\hat{g}) = 9$. The result in Figure 5.2(b) was obtained with $\theta_0 = 2.116$. Figure 5.2(a) shows, however, that the Bézout matrix $B(f,g)$ is not rank deficient, which implies that $\hat{f}(x)$ and $\hat{g}(x)$ are coprime. □



(a) (b)

Figure 5.2: The normalized singular values of (a) $B(f,g)$ and (b) $\bar{B}\left(\check{f},\check{g}\right)$ for Example 5.3.

Example 5.4. Consider the Bernstein forms of the exact polynomials

$$\hat{f}(x) = (x-0.23)^4(x-0.59)^4(x-0.98)^4(x-1.23)^3(x-5.23)^3,$$

and

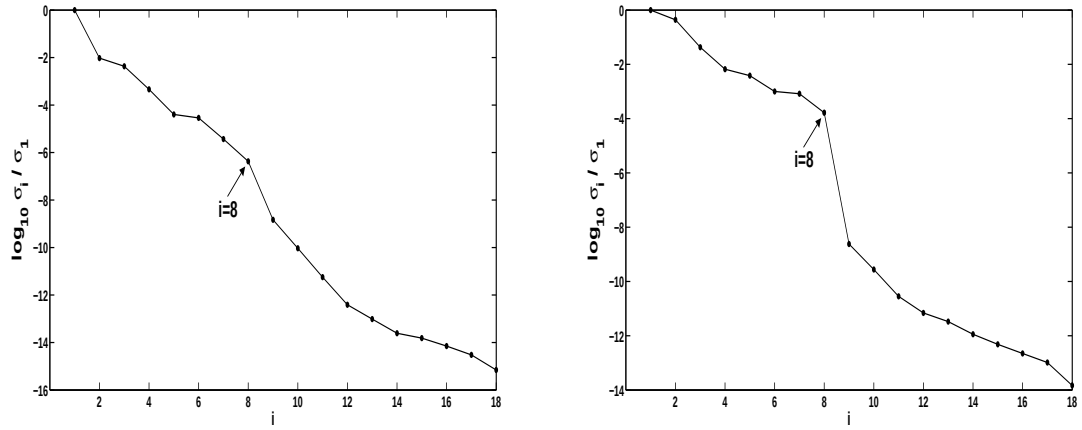
$$\hat{g}(x) = (x-0.23)^3(x-0.59)^5(x-0.73)^4(x+2.36)^2(x-5.23)^4,$$

whose GCD is of degree 10.

Noise with componentwise signal-to-noise ratio 10^8 is added to each polynomial to yield their inexact forms $f(x)$ and $g(x)$, and then we compute the matrices $B(f,g)$ and $\bar{B}\left(\check{f},\check{g}\right)$.

It is seen from Figures 5.3(a) and (b) that the matrices $B(f,g)$ and $\bar{B}\left(\check{f},\check{g}\right)$ yield, respectively, incorrect and correct results because $B(f,g)$ has full rank and the rank of $\bar{B}\left(\check{f},\check{g}\right)$ is equal to 10. The result in Figure 5.3(b) was obtained with $\theta_0 = 1.2494$.

□



(a) (b)

Figure 5.3: The normalized singular values of (a) $B(f,g)$ and (b) $\bar{B}\left(\check{f},\check{g}\right)$ for Example 5.4.

It is seen from these three examples that $\bar{B}\left(\check{f},\check{g}\right)$ yields better results than $B(f,g)$,

which shows that the inclusion of the parameter θ improves the computational results. The results shown in these three examples are consistent with results obtained from many other examples.

5.3 Summary

This chapter introduced an approach to determine the degree of an AGCD of two inexact polynomials from their Bézout resultant matrix. The examples in Section 5.2, which are typical of many other results that were obtained, show that the preprocessing operation allows us to obtain the improved and correct estimate of the degree of an AGCD of inexact polynomials. In particular, the preprocessing operation introduces a new parameter θ that transforms inexact polynomials expressed in the Bernstein basis to their corresponding polynomials defined in the modified Bernstein basis, which requires the modified Bernstein form of their Bézout resultant matrix to be developed. The optimal value of θ minimizes the ratio of maximum and minimum elements in magnitude of their Bézout matrix defined in the modified Bernstein basis.

This chapter considered the determination of the degree of an AGCD of two inexact polynomials from their Bézout resultant matrix. However, the degree of an AGCD of inexact polynomials can also be computed from another resultant matrix, the Sylvester resultant matrix but three preprocessing operations must be performed on the Sylvester resultant matrix because of its partitioned structure. This issue is addressed in the next chapter.

Chapter 6

The degree of an AGCD, Part II

This chapter extends the work of Chapter 5 by considering the Sylvester resultant matrix for the computation of the degree of an AGCD of two inexact Bernstein polynomials. Previous work [62, 63] has shown that two inexact polynomials expressed in the power basis must be preprocessed before their Sylvester matrix is used to compute the degree of an AGCD, and it is shown in these references that the inclusion of these operations improves the result. Therefore, it is desirable to consider the preprocessing operations performed on the Sylvester matrix of two Bernstein polynomials. In particular, three preprocessing operations are required, and our experiments indicate that these preprocessing operations are necessary for significantly better results. These preprocessing operations are considered in the next section.

6.1 Preprocessing operations

Consider two inexact Bernstein polynomials $f(x)$ and $g(x)$ defined in (4.2). The preprocessing operations are:

1. The normalization of $f(x)$ and $g(x)$.

2. The introduction of a parameter α.

3. A transformation of the independent variable x to a new independent variable w.

It will be shown that these preprocessing operations allow the Sylvester matrix of $f(x)$ and $g(x)$ to yield the correct estimate of the degree of an AGCD.

It is shown in Chapter 3 that there exist two forms of the Sylvester matrix, $S(f,g)$ and $S(f,g)Q$, which are defined in (3.11) and (3.28) respectively, and both forms should be considered. In particular, the preprocessing operations associated with these two forms are slightly different. The entries of $S(f,g)Q$ are more complicated than the entries of $S(f,g)$, and therefore it is convenient to consider the preprocessing operations for $S(f,g)Q$ because their simplification allows the preprocessing operations for $S(f,g)$ to be easily obtained.

6.1.1 Normalization of the polynomials

The Sylvester matrix $S(f,g)Q$ of $f(x)$ and $g(x)$ is defined in (3.28),

$$S(f,g)Q = \begin{bmatrix} \frac{a_0\binom{m}{0}\binom{n-1}{0}}{\binom{m+n-1}{0}} & & & \frac{b_0\binom{n}{0}\binom{m-1}{0}}{\binom{m+n-1}{0}} & & \\ \frac{a_1\binom{m}{1}\binom{n-1}{0}}{\binom{m+n-1}{1}} & \ddots & & \frac{b_1\binom{n}{1}\binom{m-1}{0}}{\binom{m+n-1}{1}} & \ddots & \\ \vdots & \ddots & \frac{a_0\binom{m}{0}\binom{n-1}{n-1}}{\binom{m+n-1}{n-1}} & \vdots & \ddots & \frac{b_0\binom{n}{0}\binom{m-1}{m-1}}{\binom{m+n-1}{m-1}} \\ \vdots & \ddots & \frac{a_1\binom{m}{1}\binom{n-1}{n-1}}{\binom{m+n-1}{n}} & \vdots & \ddots & \frac{b_1\binom{n}{1}\binom{m-1}{m-1}}{\binom{m+n-1}{m}} \\ \frac{a_m\binom{m}{m}\binom{n-1}{0}}{\binom{m+n-1}{m}} & \ddots & \vdots & \frac{b_n\binom{n}{n}\binom{m-1}{0}}{\binom{m+n-1}{n}} & \ddots & \vdots \\ & \ddots & \vdots & & \ddots & \vdots \\ & & \frac{a_m\binom{m}{m}\binom{n-1}{n-1}}{\binom{m+n-1}{m+n-1}} & & & \frac{b_n\binom{n}{n}\binom{m-1}{m-1}}{\binom{m+n-1}{m+n-1}} \end{bmatrix}.$$

It was noted that the coefficients of $f(x)$ and $g(x)$ occupy the first n columns and last m columns of $S(f,g)Q$, respectively, and $S(f,g)Q$ satisfies

$$S(\alpha f, \beta g)Q \neq \alpha\beta S(f,g)Q, \qquad \alpha, \beta \in \mathbb{R}\backslash 0. \tag{6.1}$$

Equation (6.1) shows that the Sylvester matrix $S(f,g)Q$ is not scale invariant because of its partitioned structure. If the coefficients of $f(x)$ are much larger or smaller than the coefficients of $g(x)$, this may cause the Sylvester matrix $S(f,g)Q$ to be unbalanced. For example, if $|a_i| \gg |b_j|, i = 0, \ldots, m, j = 0, \ldots, n$, the entries in the first n columns of $S(f,g)Q$ may be much larger than the entries in the last m columns, in magnitude, such that the rank of $S(f,g)Q$ is approximately equal to n, even if $f(x)$ and $g(x)$ are coprime. Similarly, if $|a_i| \ll |b_j|, i = 0, \ldots, m, j = 0, \ldots, n$, the entries in the first n columns of $S(f,g)Q$ may be much smaller than the entries in the last m columns, in magnitude, such that the rank of $S(f,g)Q$ is approximately equal to m. Therefore, it is necessary to normalize the entries of the first n columns and last m columns of $S(f,g)Q$, respectively, to make $S(f,g)Q$ better balanced.

It is advantageous to normalize the entries of the first n columns and last m columns of $S(f,g)Q$ by the geometric mean of the entries of each part, respectively because it provides a better average when the entries vary widely, in magnitude [63]. This can be easily illustrated by the following example.

Example 6.1. Consider a data set $v = \{10^{-3}, 1, 10^{15}\}$, and it is seen that the numbers in data set v vary substantially in magnitude.

The geometric mean of numbers in data set v, GM_v, is equal to 10^4, and the norms of numbers in data set v, $\| v \|_p$ for $p = 1, 2, \infty$, are approximately equal to 10^{15}.

If the first number in data set v, 10^{-3}, is reduced to 10^{-9}, GM_v is then equal to 10^2 but $\| v \|_p$ for $p = 1, 2, \infty$, are still approximately equal to 10^{15}.

This example illustrates that the norms of numbers in data set v are dominated by the extremely large number, however, the geometric mean treats each number equally and any change in small value can affect the value of the geometric mean. □

Consider the coefficients $a_i\binom{m}{i}, i = 0, \ldots, m$, which occupy the first n columns of $S(f,g)Q$. It follows from (3.7), (3.8), (3.25) and (3.28) that the product of the magnitudes of the terms that contain the coefficient $a_0\binom{m}{0}$ in $S(f,g)Q$ is

$$\left|\frac{a_0\binom{m}{0}\binom{n-1}{0}}{\binom{m+n-1}{0}}\right|\left|\frac{a_0\binom{m}{0}\binom{n-1}{1}}{\binom{m+n-1}{1}}\right|\left|\frac{a_0\binom{m}{0}\binom{n-1}{2}}{\binom{m+n-1}{2}}\right|\cdots\left|\frac{a_0\binom{m}{0}\binom{n-1}{n-1}}{\binom{m+n-1}{n-1}}\right| = \frac{\left|a_0\binom{m}{0}\right|^n\prod_{r=0}^{n-1}\binom{n-1}{r}}{\prod_{t=0}^{n-1}\binom{m+n-1}{t}},$$

and the product of the magnitudes of the terms that contain the coefficient $a_1\binom{m}{1}$ in $S(f,g)Q$ is

$$\left|\frac{a_1\binom{m}{1}\binom{n-1}{0}}{\binom{m+n-1}{1}}\right|\left|\frac{a_1\binom{m}{1}\binom{n-1}{1}}{\binom{m+n-1}{2}}\right|\left|\frac{a_1\binom{m}{1}\binom{n-1}{2}}{\binom{m+n-1}{3}}\right|\cdots\left|\frac{a_1\binom{m}{1}\binom{n-1}{n-1}}{\binom{m+n-1}{n}}\right| = \frac{\left|a_1\binom{m}{1}\right|^n\prod_{r=0}^{n-1}\binom{n-1}{r}}{\prod_{t=1}^{n}\binom{m+n-1}{t}}.$$

Therefore, the product of the magnitudes of the terms that contain the coefficient $a_i\binom{m}{i}$ in $S(f,g)Q$ is

$$\left|\frac{a_i\binom{m}{i}\binom{n-1}{0}}{\binom{m+n-1}{i}}\right|\left|\frac{a_i\binom{m}{i}\binom{n-1}{1}}{\binom{m+n-1}{i+1}}\right|\left|\frac{a_i\binom{m}{i}\binom{n-1}{2}}{\binom{m+n-1}{i+2}}\right|\cdots\left|\frac{a_i\binom{m}{i}\binom{n-1}{n-1}}{\binom{m+n-1}{i+n-1}}\right| = \frac{\left|a_i\binom{m}{i}\right|^n\prod_{r=0}^{n-1}\binom{n-1}{r}}{\prod_{t=i}^{n-1+i}\binom{m+n-1}{t}},$$

and thus the product of all the terms in $S(f,g)Q$ that contain the coefficients of $f(x)$ is

$$\prod_{i=0}^{m}\left(\frac{\left|a_i\binom{m}{i}\right|^n\prod_{r=0}^{n-1}\binom{n-1}{r}}{\prod_{t=i}^{n-1+i}\binom{m+n-1}{t}}\right).$$

Since the coefficients of $f(x)$ occur $n(m+1)$ times in $S(f,g)Q$, the geometric mean of these terms is

$$\lambda = \left\{\prod_{i=0}^{m}\left(\frac{\left|a_i\binom{m}{i}\right|^n\prod_{r=0}^{n-1}\binom{n-1}{r}}{\prod_{t=i}^{n-1+i}\binom{m+n-1}{t}}\right)\right\}^{\frac{1}{n(m+1)}}, \tag{6.2}$$

and the numerator of this expression simplifies to

$$\left\{\prod_{i=0}^{m}\left|a_i\binom{m}{i}\right|\right\}^{\frac{1}{m+1}}\left\{\prod_{r=0}^{n-1}\binom{n-1}{r}\right\}^{\frac{1}{n}},$$

where care must be taken in the computation of these terms in order to prevent overflow.

Consider the denominator in (6.2),

$$\left\{\prod_{i=0}^{m}\prod_{t=i}^{n-1+i}\binom{m+n-1}{t}\right\}^{\frac{1}{n(m+1)}},$$

which can be evaluated efficiently by a recurrence equation. In particular, if P_i is defined as

$$P_i = \prod_{t=i}^{n-1+i}\binom{m+n-1}{t}, \qquad i = 0, \ldots, m, \tag{6.3}$$

then

$$P_{i+1} = \prod_{t=i+1}^{n+i}\binom{m+n-1}{t} = \frac{\binom{m+n-1}{n+i}\prod_{t=i}^{n-1+i}\binom{m+n-1}{t}}{\binom{m+n-1}{i}},$$

and thus

$$P_{i+1} = P_i\frac{\binom{m+n-1}{n+i}}{\binom{m+n-1}{i}} = P_i\prod_{t=0}^{n-1}\frac{(m-i+t)}{(i+1+t)}, \qquad i = 0, \ldots, m-1.$$

The starting value of this recurrence relationship is

$$P_0 = \prod_{t=0}^{n-1}\binom{m+n-1}{t},$$

and thus the geometric mean (6.2) of all the terms that contain the coefficients of $f(x)$ is

$$\lambda = \frac{\left\{\prod_{i=0}^{m}\left|a_i\binom{m}{i}\right|\right\}^{\frac{1}{m+1}}\left\{\prod_{r=0}^{n-1}\binom{n-1}{r}\right\}^{\frac{1}{n}}}{\left\{\prod_{i=0}^{m}P_i\right\}^{\frac{1}{n(m+1)}}}. \tag{6.4}$$

It follows that the normalized form of $f(x)$ is

$$\breve{f}(x) = \sum_{i=0}^{m}\bar{a}_i\binom{m}{i}(1-x)^{m-i}x^i, \qquad \bar{a}_i = \frac{a_i}{\lambda}. \tag{6.5}$$

This analysis can be repeated for $g(x)$, and its normalized form is

$$\breve{g}(x) = \sum_{j=0}^{n}\bar{b}_j\binom{n}{j}(1-x)^{n-j}x^j, \qquad \bar{b}_j = \frac{b_j}{\mu}, \tag{6.6}$$

where

$$\mu = \frac{\left\{\prod_{j=0}^{n}\left|b_j\binom{n}{j}\right|\right\}^{\frac{1}{n+1}}\left\{\prod_{r=0}^{m-1}\binom{m-1}{r}\right\}^{\frac{1}{m}}}{\left\{\prod_{j=0}^{n} L_j\right\}^{\frac{1}{m(n+1)}}}, \tag{6.7}$$

and

$$L_j = \prod_{t=j}^{m-1+j}\binom{m+n-1}{t}, \qquad j = 0, \ldots, n. \tag{6.8}$$

The normalized coefficients $\bar{a}_i$ and $\bar{b}_j$ in (6.5) and (6.6) enable the Sylvester matrix $S(\breve{f}, \breve{g})Q = D^{-1}T(\breve{f}, \breve{g})Q$, where the matrices D^{-1}, $T(\breve{f}, \breve{g})$ and Q are defined in (3.7), (3.8) and (3.25), respectively, to be computed. The importance of normalization of polynomials for the correct estimate of their degree of an AGCD is illustrated by the following example.

Example 6.2. Consider the Bernstein forms of the exact polynomials

$$\hat{f}(x) \;=\; (x-0.01)^3(x-0.4)^4(x-0.6)^3(x-0.9)^3,$$

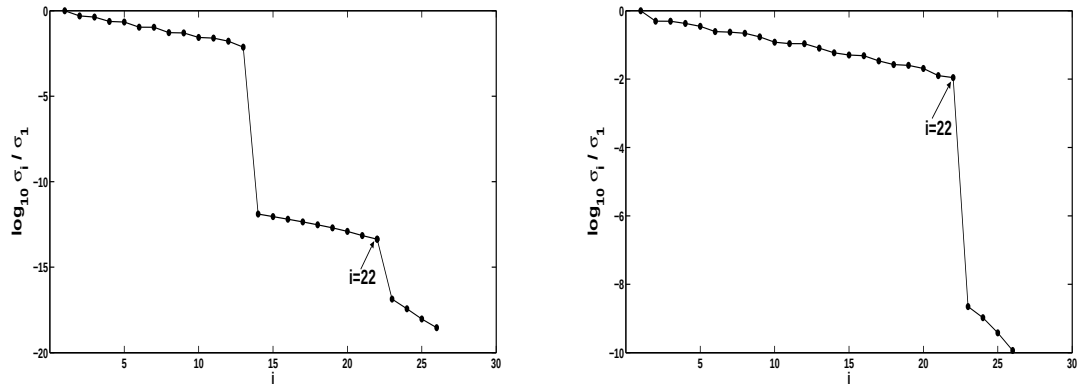
and

$$\hat{g}(x) \;=\; (x-0.4)^2(x-0.6)^2(x-10)^5(x-12)^4,$$

whose GCD is of degree 4. Noise with componentwise signal-to-noise ratio 10^8 is added to the coefficients of $\hat{f}(x)$ and $\hat{g}(x)$, and then the matrices $S(f,g)Q$ and $S(\breve{f}, \breve{g})Q$ are computed.

It is seen from Figure 6.1(b) that the rank of $S(\breve{f}, \breve{g})Q$ is clearly defined and equal to 22, which is correct because $\deg \mathrm{GCD}(\hat{f}, \hat{g}) = 4$. However, Figure 6.1(a) shows that the rank of $S(f,g)Q$ is equal to 13, which is incorrect, and it is interesting to note that this implies that $\hat{f}(x)$ is a constant multiple of $\hat{g}(x)$ because $\deg \hat{f}(x) = \deg \hat{g}(x) = 13$.

□



(a) (b)

Figure 6.1: The normalized singular values of (a) $S(f,g)Q$ and (b) $S(\breve{f},\breve{g})Q$ for Example 6.2.

The above normalization analysis can be repeated for the Sylvester matrix $S(f,g) = D^{-1}T(f,g)$, and it is easy to see that the normalization constants for $f(x)$ and $g(x)$ in this matrix are, respectively,

$$\eta = \frac{\{\prod_{i=0}^{m} \left|a_i \binom{m}{i}\right|\}^{\frac{1}{m+1}}}{\{\prod_{i=0}^{m} P_i\}^{\frac{1}{n(m+1)}}} \qquad \text{and} \qquad \rho = \frac{\left\{\prod_{j=0}^{n} \left|b_j \binom{n}{j}\right|\right\}^{\frac{1}{n+1}}}{\left\{\prod_{j=0}^{n} L_j\right\}^{\frac{1}{m(n+1)}}}, \tag{6.9}$$

where P_i and L_j are defined in (6.3) and (6.8) respectively, and thus the polynomials $\dot{f}(x)$ and $\dot{g}(x)$ in the matrix $S(\dot{f},\dot{g}) = D^{-1}T(\dot{f},\dot{g})$ are

$$\dot{f}(x) = \sum_{i=0}^{m} \ddot{a}_i \binom{m}{i} (1-x)^{m-i} x^i, \qquad \ddot{a}_i = \frac{a_i}{\eta}, \tag{6.10}$$

and

$$\dot{g}(x) = \sum_{j=0}^{n} \ddot{b}_j \binom{n}{j} (1-x)^{n-j} x^j, \qquad \ddot{b}_j = \frac{b_j}{\rho}. \tag{6.11}$$

6.1.2 Scaling a polynomial by an arbitrary constant

The second preprocessing operation arises because the GCD of the exact polynomials $\hat{f}(x)$ and $\hat{g}(x)$ is defined to within an arbitrary scalar multiplier $\alpha \in \mathbb{R}\backslash 0$,

$$\hat{d} = \deg \mathrm{GCD}(\hat{f}, \hat{g}) = \deg \mathrm{GCD}(\hat{f}, \alpha\hat{g}), \tag{6.12}$$

and

$$\text{rank } S(\hat{f}, \hat{g}) = \text{rank } S(\hat{f}, \alpha\hat{g}). \tag{6.13}$$

It cannot be assumed, however, that these equations are satisfied for all real non-zero values of α when inexact polynomials in a floating point environment are considered. For example, the rank of $D^{-1}T(\breve{f}, \alpha\breve{g})Q$ is a function of α, which is most easily seen by noting that

$$\lim_{\alpha\to\delta} \text{rank } D^{-1}T(\breve{f}, \alpha\breve{g})Q = \deg \breve{g}(x) = n, \qquad 0 < |\delta| \ll 1, \tag{6.14}$$

and

$$\lim_{\alpha\to\pm\infty} \text{rank } D^{-1}T(\breve{f}, \alpha\breve{g})Q = \deg \breve{f}(x) = m. \tag{6.15}$$

More general examples of the dependence of d, the degree of an AGCD of $\breve{f}(x)$ and $\breve{g}(x)$, on α for polynomials expressed in the power basis are considered in [63], and it is shown that d is a function of α, such that an incorrect value of α may yield either an incorrect value of d, or the value of d cannot be computed because the numerical rank of the Sylvester matrix is not defined.

It follows from (6.12) and (6.13) that d is given by

$$d = \deg \mathrm{AGCD}(\breve{f}, \alpha\breve{g}) = \text{rank loss } S(\breve{f}, \alpha\breve{g}),$$

where, as shown by (6.14) and (6.15), and the examples in [63], α must be chosen with care. It can therefore be considered a parameter that defines a degree of freedom that

can be used to obtain a superior estimate of $\hat{d}$. The criterion for the determination of the optimal value of α, and its computation, will be considered in the next section, where the third preprocessing operation is described. Furthermore, it will be shown that the optimal value of α for $D^{-1}T(\breve{f}, \alpha\breve{g})Q$ is not equal to the optimal value of α for $D^{-1}T(\dot{f}, \alpha\dot{g})$, but the same criterion is used for the determination of these optimal values.

The first and second preprocessing operations described in this chapter need not be applied to $f(x)$ and $g(x)$ before computations are performed on the Bézout resultant matrix $B(f, g)$, due to the bilinear property of every element of $B(f, g)$ [5].

6.1.3 The transformation of the independent variable

As noted earlier, computations on a matrix whose entries vary widely in magnitude may cause problems, and it is therefore advantageous to preprocess the matrix, such that the ratio of the maximum entry, in magnitude, to the minimum entry, in magnitude, is minimized. This computation, which defines the third preprocessing operation, is implemented by the substitution (5.1).

The polynomials $\breve{f}(x)$ and $\breve{g}(x)$, which are defined in (6.5) and (6.6) respectively, are therefore transformed to

$$\bar{f}(w, \theta) = \sum_{i=0}^{m} \left(\bar{a}_i \theta^i\right) \binom{m}{i} (1 - \theta w)^{m-i} w^i, \tag{6.16}$$

and

$$\bar{g}(w, \theta) = \sum_{j=0}^{n} \left(\bar{b}_j \theta^j\right) \binom{n}{j} (1 - \theta w)^{n-j} w^j, \tag{6.17}$$

respectively, and $\dot{f}(x)$ and $\dot{g}(x)$, which are defined in (6.10) and (6.11), are transformed similarly to

$$\ddot{f}(w,\theta) = \sum_{i=0}^{m} \left(\ddot{a}_i \theta^i\right) \binom{m}{i} (1-\theta w)^{m-i} w^i, \tag{6.18}$$

and

$$\ddot{g}(w,\theta) = \sum_{j=0}^{n} \left(\ddot{b}_j \theta^j\right) \binom{n}{j} (1-\theta w)^{n-j} w^j. \tag{6.19}$$

The analysis in this section and the next section only considers the polynomials $\bar{f}(w,\theta)$ and $\bar{g}(w,\theta)$, but it is also applicable to the polynomials $\ddot{f}(w,\theta)$ and $\ddot{g}(w,\theta)$. As stated earlier, the substitution (5.1) transforms the Bernstein basis to the modified Bernstein basis, whose basis functions for polynomials of degree m are $\phi_i^m(w,\theta), i = 0,\ldots,m$, which are defined in (5.4).

The coefficients of $\bar{f}(w,\theta)$ and $\bar{g}(w,\theta)$ are $\bar{a}_i\theta^i, i = 0,\ldots,m$, and $\bar{b}_j\theta^j, j = 0,\ldots,n$, respectively, and therefore the criterion to select the optimal value of θ must be defined in order that the coefficients of $\bar{f}(w,\theta)$ and $\bar{g}(w,\theta)$ are computed. As mentioned in Section 6.1.2, the rank of the Sylvester matrix is a function of α in a floating point environment, and thus the optimal value of α must be calculated. In particular, the optimal values of α and θ minimize the ratio of the maximum element, in magnitude, to the minimum element, in magnitude, of $\bar{S}(\bar{f}(w,\theta), \alpha\bar{g}(w,\theta))$, which is the Sylvester matrix of the modified Bernstein basis polynomials $\bar{f}(w,\theta)$ and $\alpha\bar{g}(w,\theta)$. The form of $\bar{S}(\bar{f}(w,\theta), \alpha\bar{g}(w,\theta))$ must therefore be developed, which enables the optimal values of α and θ for this matrix to be calculated. This topic is addressed in the next section.

6.1.4 The Sylvester resultant matrix for the modified Bernstein basis

The substitution (5.1) transforms the Bernstein basis to the modified Bernstein basis, and it is therefore necessary to consider computations on this basis, and to develop expressions for the entries of $\bar{S}(\bar{f}(w,\theta), \alpha\bar{g}(w,\theta))$.

The addition and multiplication of two polynomials expressed in the modified Bernstein basis are very similar to their equivalents for polynomials expressed in the Bernstein basis. The development of the Sylvester matrix of the polynomials $\bar{p}(w,\theta)$ and $\bar{q}(w,\theta)$ expressed in the modified Bernstein basis,

$$\bar{p}(w,\theta) = \sum_{i=0}^{m} \left(\bar{c}_i\theta^i\right) \binom{m}{i} (1-\theta w)^{m-i} w^i,$$

and

$$\bar{q}(w,\theta) = \sum_{j=0}^{n} \left(\bar{d}_j\theta^j\right) \binom{n}{j} (1-\theta w)^{n-j} w^j,$$

follows closely the development of (3.6). In particular, if $\bar{p}(w,\theta)$ and $\bar{q}(w,\theta)$ have a non-constant common divisor, and $\bar{u}(w,\theta)$ and $\bar{v}(w,\theta)$ are quotient polynomials of degrees $m-1$ and $n-1$ respectively, then

$$\bar{p}(w,\theta)\bar{v}(w,\theta) = \bar{q}(w,\theta)\bar{u}(w,\theta), \tag{6.20}$$

where

$$\bar{u}(w,\theta) = \sum_{i=0}^{m-1} \left(\bar{u}_i\theta^i\right) \binom{m-1}{i} (1-\theta w)^{m-1-i} w^i,$$

and

$$\bar{v}(w,\theta) = \sum_{j=0}^{n-1} \left(\bar{v}_j\theta^j\right) \binom{n-1}{j} (1-\theta w)^{n-1-j} w^j.$$

Equation (6.20) can be expressed in matrix form as

$$\left(D^{-1}U(\bar{p},\bar{q})\right) s(\bar{u},\bar{v}) = 0, \tag{6.21}$$

where $\bar{p} = \bar{p}(w,\theta), \bar{q} = \bar{q}(w,\theta), \bar{u} = \bar{u}(w,\theta), \bar{v} = \bar{v}(w,\theta)$, the diagonal matrix D^{-1} is defined in (3.7),

$$U(\bar{p},\bar{q}) = \begin{bmatrix} C(\bar{p}) & D(\bar{q}) \end{bmatrix} \in \mathbb{R}^{(m+n)\times(m+n)}, \tag{6.22}$$

the Toeplitz matrices $C(\bar{p})$ and $D(\bar{q})$ are given by, respectively,

$$C(\bar{p}) = \begin{bmatrix} \bar{c}_0\binom{m}{0} & & \\ \bar{c}_1\binom{m}{1}\theta & \ddots & \\ \vdots & \ddots & \bar{c}_0\binom{m}{0} \\ \vdots & \ddots & \bar{c}_1\binom{m}{1}\theta \\ \bar{c}_{m-1}\binom{m}{m-1}\theta^{m-1} & \ddots & \vdots \\ \bar{c}_m\binom{m}{m}\theta^m & \ddots & \vdots \\ & \ddots & \bar{c}_{m-1}\binom{m}{m-1}\theta^{m-1} \\ & & \bar{c}_m\binom{m}{m}\theta^m \end{bmatrix} \in \mathbb{R}^{(m+n)\times n},$$

and

$$D(\bar{q}) = \begin{bmatrix} \bar{d}_0\binom{n}{0} & & \\ \bar{d}_1\binom{n}{1}\theta & \ddots & \\ \vdots & \ddots & \bar{d}_0\binom{n}{0} \\ \vdots & \ddots & \bar{d}_1\binom{n}{1}\theta \\ \bar{d}_{n-1}\binom{n}{n-1}\theta^{n-1} & \ddots & \vdots \\ \bar{d}_n\binom{n}{n}\theta^n & \ddots & \vdots \\ & \ddots & \bar{d}_{n-1}\binom{n}{n-1}\theta^{n-1} \\ & & \bar{d}_n\binom{n}{n}\theta^n \end{bmatrix} \in \mathbb{R}^{(m+n)\times m},$$

and $s(\bar{u},\bar{v}) \in \mathbb{R}^{m+n}$ is equal to

$$\begin{bmatrix} \bar{v}_0\binom{n-1}{0} & \bar{v}_1\binom{n-1}{1}\theta & \cdots & \bar{v}_{n-2}\binom{n-1}{n-2}\theta^{n-2} & \bar{v}_{n-1}\binom{n-1}{n-1}\theta^{n-1} \\ -\bar{u}_0\binom{m-1}{0} & -\bar{u}_1\binom{m-1}{1}\theta & \cdots & -\bar{u}_{m-2}\binom{m-1}{m-2}\theta^{m-2} & -\bar{u}_{m-1}\binom{m-1}{m-1}\theta^{m-1} \end{bmatrix}^T.$$

Following (3.24), the vector $s(\bar{u},\bar{v})$ is written as

$$s(\bar{u},\bar{v}) = Qt(\bar{u},\bar{v}),$$

where Q is defined in (3.25), and

$$t(\bar{u},\bar{v}) = \begin{bmatrix} \bar{v}_0 & \bar{v}_1\theta & \cdots & \bar{v}_{n-1}\theta^{n-1} & -\bar{u}_0 & -\bar{u}_1\theta & \cdots & -\bar{u}_{m-1}\theta^{m-1} \end{bmatrix}^T \in \mathbb{R}^{m+n}.$$

It therefore follows that (6.21) can be written as

$$\left(D^{-1}U(\bar{p},\bar{q})Q\right) t(\bar{u},\bar{v}) = 0,$$

and a slight modification to the proof of the rank loss property (3.27) shows that

$$\deg \mathrm{GCD}(\bar{p},\bar{q}) = m+n-\operatorname{rank} D^{-1}U(\bar{p},\bar{q}) = m+n-\operatorname{rank} D^{-1}U(\bar{p},\bar{q})Q. \quad (6.23)$$

This equation is a generalization of (3.27), which is only applicable to the Bernstein basis and therefore restricted to $\theta = 1$, to arbitrary values of θ, and therefore the modified Bernstein basis, because $U = T$ if $\theta = 1$, where T is defined in (3.8).

Equation (6.23) and Section 6.1.2 show that the degree d of an AGCD of $\bar{f} = \bar{f}(w,\theta)$ and $\bar{g} = \bar{g}(w,\theta)$, which are defined in (6.16) and (6.17) respectively, can be calculated from

$$\bar{S}(\bar{f},\alpha\bar{g}) = D^{-1}U(\bar{f},\alpha\bar{g}) \qquad \text{and} \qquad \bar{S}(\bar{f},\alpha\bar{g})Q = D^{-1}U(\bar{f},\alpha\bar{g})Q,$$

where $U(\bar{p},\bar{q})$ is defined in (6.22), but it must also be shown that the coefficients of the GCD can be computed from these matrices in order that they satisfy the requirements of resultant matrices. This property is easily established for both $D^{-1}U$ and $D^{-1}UQ$ by a small modification to the proof of Theorem 1 in [12], and thus $D^{-1}U$

and $D^{-1}UQ$ are resultant matrices.

This theoretical analysis is valid for arbitrary values of α and θ, but their optimal values for the calculation of d, that is, the degree of an AGCD of two inexact polynomials in a floating point environment, must be considered. This issue is addressed in the next section.

6.1.5 The optimal values of α and θ

The degree d of an AGCD of $\bar{f} = \bar{f}(w, \theta)$ and $\bar{g} = \bar{g}(w, \theta)$, which are defined in (6.16) and (6.17) respectively, can be calculated from $\bar{S}(\bar{f}, \alpha\bar{g})Q$. In particular,

$$d = \text{rank loss } \bar{S}(\bar{f}, \alpha\bar{g})Q = \text{rank loss } D^{-1}U(\bar{f}, \alpha\bar{g})Q,$$

and a criterion for the calculation of the optimal values of α and θ must be established. As stated earlier, computations performed on a matrix whose elements vary widely in magnitude are unreliable. Therefore, it is desirable to choose α_1 and θ_1, the optimal values of α and θ respectively, such that the ratio of the maximum element, in magnitude, of $\bar{S}(\bar{f}, \alpha\bar{g})Q$ to the minimum element, in magnitude, of $\bar{S}(\bar{f}, \alpha\bar{g})Q$ is minimized.

The same criterion is appropriate for $\bar{S}(\ddot{f}, \alpha\ddot{g})$, where $\ddot{f} = \ddot{f}(w, \theta)$ and $\ddot{g} = \ddot{g}(w, \theta)$, which are defined in (6.18) and (6.19) respectively, and the same method can be used for both $\bar{S}(\ddot{f}, \alpha\ddot{g})$ and $\bar{S}(\bar{f}, \alpha\bar{g})Q$, but the optimal values of α and θ are different. It is adequate to consider the computation of the optimal values of α and θ when d is calculated from $\bar{S}(\bar{f}, \alpha\bar{g})Q$ because the computation of their optimal values when $\bar{S}(\ddot{f}, \alpha\ddot{g})$ is used follows easily.

The general expression for a non-zero element in the first n columns of $\bar{S}(\bar{f}, \alpha\bar{g})Q$ is

$$\frac{\bar{a}_j \binom{m}{j} \binom{n-1}{i} \theta^j}{\binom{m+n-1}{i+j}}, \qquad j = 0, \ldots, m; i = 0, \ldots, n-1,$$

and similarly, the general expression for a non-zero element in the last m columns of $\bar{S}(\bar{f}, \alpha\bar{g})Q$ is

$$\frac{\alpha\bar{b}_j \binom{n}{j} \binom{m-1}{i} \theta^j}{\binom{m+n-1}{i+j}}, \qquad j = 0, \ldots, n; i = 0, \ldots, m-1.$$

It is convenient to define the sets $\rho(\theta)$ and $\sigma(\alpha, \theta)$ as

$$\rho(\theta) = \left\{ \frac{\left| \bar{a}_j \binom{m}{j} \binom{n-1}{i} \theta^j \right|}{\binom{m+n-1}{i+j}} : j = 0, \ldots, m; i = 0, \ldots, n-1 \right\},$$

and

$$\sigma(\alpha, \theta) = \left\{ \frac{\left| \alpha\bar{b}_j \binom{n}{j} \binom{m-1}{i} \theta^j \right|}{\binom{m+n-1}{i+j}} : j = 0, \ldots, n; i = 0, \ldots, m-1 \right\},$$

respectively, and the values α_1 and θ_1 of α and θ, respectively, minimize the ratio of the maximum element, in magnitude, to the minimum element, in magnitude, of $\bar{S}(\bar{f}, \alpha\bar{g})Q$,

$$\alpha_1, \theta_1 = \arg\min_{\alpha,\theta} \left\{ \frac{\max\left\{ \max\{\rho(\theta)\}, \max\{\sigma(\alpha, \theta)\} \right\}}{\min\left\{ \min\{\rho(\theta)\}, \min\{\sigma(\alpha, \theta)\} \right\}} \right\}.$$

This minimization problem can be written as:

Minimize $\frac{u}{v}$

subject to

$$\begin{aligned}
u &\geq \frac{\left|\bar{a}_j\binom{m}{j}\binom{n-1}{i}\theta^j\right|}{\binom{m+n-1}{i+j}}, && j=0,\ldots,m; i=0,\ldots,n-1,\\
u &\geq \frac{\left|\alpha\bar{b}_j\binom{n}{j}\binom{m-1}{i}\theta^j\right|}{\binom{m+n-1}{i+j}}, && j=0,\ldots,n; i=0,\ldots,m-1,\\
v &\leq \frac{\left|\bar{a}_j\binom{m}{j}\binom{n-1}{i}\theta^j\right|}{\binom{m+n-1}{i+j}}, && j=0,\ldots,m; i=0,\ldots,n-1,\\
v &\leq \frac{\left|\alpha\bar{b}_j\binom{n}{j}\binom{m-1}{i}\theta^j\right|}{\binom{m+n-1}{i+j}}, && j=0,\ldots,n; i=0,\ldots,m-1,\\
v &> 0,\\
\theta &> 0,\\
\alpha &> 0.
\end{aligned}$$

The transformations

$$U=\log u, \qquad V=\log v, \qquad \phi=\log\theta, \qquad \mu=\log\alpha, \tag{6.24}$$

and

$$\bar{\alpha}_{i,j}=\log\frac{\left|\bar{a}_j\binom{m}{j}\binom{n-1}{i}\right|}{\binom{m+n-1}{i+j}}, \qquad \bar{\beta}_{i,j}=\log\frac{\left|\bar{b}_j\binom{n}{j}\binom{m-1}{i}\right|}{\binom{m+n-1}{i+j}},$$

where $\log=\log_{10}$, enable this constrained minimization problem to be written as:

Minimize $U-V$

subject to

$$\begin{aligned}
U - j\phi &\geq \bar{\alpha}_{i,j}, && j=0,\ldots,m; i=0,\ldots,n-1,\\
U - j\phi - \mu &\geq \bar{\beta}_{i,j}, && j=0,\ldots,n; i=0,\ldots,m-1,\\
-V + j\phi &\geq -\bar{\alpha}_{i,j}, && j=0,\ldots,m; i=0,\ldots,n-1,\\
-V + j\phi + \mu &\geq -\bar{\beta}_{i,j}, && j=0,\ldots,n; i=0,\ldots,m-1.
\end{aligned} \tag{6.25}$$

The counter i appears on the right hand sides only of these inequalities, and thus if $\bar{\lambda}_j$, $\bar{\mu}_j$, $\bar{\rho}_j$ and $\bar{\tau}_j$ are defined as

$$\begin{aligned}
\bar{\lambda}_j &= \max_{i=0,\ldots,n-1}\{\bar{\alpha}_{i,j}\} = \max_{i=0,\ldots,n-1}\left\{\log\frac{\left|\bar{a}_j\binom{m}{j}\binom{n-1}{i}\right|}{\binom{m+n-1}{i+j}}\right\}, \qquad j=0,\ldots,m,\\
\bar{\mu}_j &= \max_{i=0,\ldots,m-1}\{\bar{\beta}_{i,j}\} = \max_{i=0,\ldots,m-1}\left\{\log\frac{\left|\bar{b}_j\binom{n}{j}\binom{m-1}{i}\right|}{\binom{m+n-1}{i+j}}\right\}, \qquad j=0,\ldots,n,\\
\bar{\rho}_j &= \min_{i=0,\ldots,n-1}\{\bar{\alpha}_{i,j}\} = \min_{i=0,\ldots,n-1}\left\{\log\frac{\left|\bar{a}_j\binom{m}{j}\binom{n-1}{i}\right|}{\binom{m+n-1}{i+j}}\right\}, \qquad j=0,\ldots,m,\\
\bar{\tau}_j &= \min_{i=0,\ldots,m-1}\{\bar{\beta}_{i,j}\} = \min_{i=0,\ldots,m-1}\left\{\log\frac{\left|\bar{b}_j\binom{n}{j}\binom{m-1}{i}\right|}{\binom{m+n-1}{i+j}}\right\}, \qquad j=0,\ldots,n,
\end{aligned}$$

then (6.25) can be written as:

Minimize $U - V$

subject to

$$\begin{aligned}
U - j\phi &\geq \bar{\lambda}_j, \qquad j=0,\ldots,m,\\
U - j\phi - \mu &\geq \bar{\mu}_j, \qquad j=0,\ldots,n,\\
-V + j\phi &\geq -\bar{\rho}_j, \qquad j=0,\ldots,m,\\
-V + j\phi + \mu &\geq -\bar{\tau}_j, \qquad j=0,\ldots,n.
\end{aligned}$$

This minimization problem can be written as:

$$\text{Minimize} \quad \begin{bmatrix} 1 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} U \\ V \\ \phi \\ \mu \end{bmatrix} \qquad \text{subject to} \qquad A \begin{bmatrix} U \\ V \\ \phi \\ \mu \end{bmatrix} \geq b, \tag{6.26}$$

where $A \in \mathbb{R}^{(2m+2n+4)\times 4}$ and

$$b = \begin{bmatrix} \bar{\lambda}_0, \cdots, \bar{\lambda}_m, \bar{\mu}_0, \cdots, \bar{\mu}_n, -\bar{\rho}_0, \cdots, -\bar{\rho}_m, -\bar{\tau}_0, \cdots, -\bar{\tau}_n \end{bmatrix}^T \in \mathbb{R}^{2m+2n+4},$$

which is a standard linear programming problem. Since α_1 and θ_1 are computed from the solution of (6.26), using (6.24), it follows from (6.16) and (6.17) that $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ is used to compute the degree of an AGCD of the inexact polynomials (4.2), where

$$\tilde{f} = \tilde{f}(w) = \bar{f}(w, \theta_1) = \sum_{i=0}^{m} \left(\bar{a}_i \theta_1^i\right) \binom{m}{i} (1 - \theta_1 w)^{m-i} w^i, \tag{6.27}$$

and

$$\tilde{g} = \tilde{g}(w) = \bar{g}(w, \theta_1) = \sum_{j=0}^{n} \left(\bar{b}_j \theta_1^j\right) \binom{n}{j} (1 - \theta_1 w)^{n-j} w^j. \tag{6.28}$$

This analysis can be repeated for $\bar{S}(\ddot{f}, \alpha \ddot{g})$, but α_2 and θ_2, the optimal values of α and θ for $\bar{S}(\ddot{f}, \alpha \ddot{g})$ are different because of the absence of Q, and the normalization constants λ and μ, which are defined in (6.4) and (6.7), are replaced by, respectively, η and ρ, which are defined in (6.9). Therefore, the degree of an AGCD of the inexact polynomials (4.2) can also be computed from $\bar{S}(\acute{f}, \alpha_2 \acute{g})$, where

$$\acute{f} = \acute{f}(w) = \ddot{f}(w, \theta_2) = \sum_{i=0}^{m} \left(\ddot{a}_i \theta_2^i\right) \binom{m}{i} (1 - \theta_2 w)^{m-i} w^i, \tag{6.29}$$

and

$$\acute{g} = \acute{g}(w) = \ddot{g}(w, \theta_2) = \sum_{j=0}^{n} \left(\ddot{b}_j \theta_2^j\right) \binom{n}{j} (1 - \theta_2 w)^{n-j} w^j. \tag{6.30}$$

Example 6.3. Consider the Bernstein polynomials $f(x)$ and $g(x)$

$$f(x) = \binom{3}{0}(1-x)^3 + \frac{1}{2}\binom{3}{1}(1-x)^2 x - \frac{1}{2}\binom{3}{2}(1-x)x^2 - \binom{3}{3}x^3,$$

and

$$g(x) = \binom{2}{0}(1-x)^2 - \frac{1}{4}\binom{2}{1}(1-x)x - \frac{1}{2}\binom{2}{2}x^2,$$

whose GCD is $g(x)$ because

$$f(x) = g(x)\left(\binom{1}{0}(1-x) + 2\binom{1}{1}x\right).$$

The forms of $f(x)$ and $g(x)$ in the modified Bernstein basis for the value of $\theta = \theta_1 = 2$,

are

$$\tilde{f}(w) = \bar{f}(w,\theta_1) = \binom{3}{0}(1-2w)^3 + \binom{3}{1}(1-2w)^2 w - 2\binom{3}{2}(1-2w)w^2 - 8\binom{3}{3}w^3,$$

and

$$\tilde{g}(w) = \bar{g}(w,\theta_1) = \binom{2}{0}(1-2w)^2 - \frac{1}{2}\binom{2}{1}(1-2w)w - 2\binom{2}{2}w^2,$$

and thus the transpose of the Sylvester matrix $\bar{S}(\tilde{f},\tilde{g}) = D^{-1}U(\tilde{f},\tilde{g})$ is equal to

$$\begin{aligned}
\bar{S}(\tilde{f},\tilde{g})^T &= \begin{bmatrix} 1 & 3 & -6 & -8 & 0 \\ 0 & 1 & 3 & -6 & -8 \\ 1 & -1 & -2 & 0 & 0 \\ 0 & 1 & -1 & -2 & 0 \\ 0 & 0 & 1 & -1 & -2 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{4} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{6} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{4} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} 1 & \frac{3}{4} & -1 & -2 & 0 \\ 0 & \frac{1}{4} & \frac{1}{2} & -\frac{3}{2} & -8 \\ 1 & -\frac{1}{4} & -\frac{1}{3} & 0 & 0 \\ 0 & \frac{1}{4} & -\frac{1}{6} & -\frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{6} & -\frac{1}{4} & -2 \end{bmatrix}.
\end{aligned}$$

The reduction of this matrix to row echelon (upper triangular) form yields

$$\begin{bmatrix} 1 & \frac{3}{4} & -1 & -2 & 0 \\ 0 & -1 & \frac{2}{3} & 2 & 0 \\ 0 & 0 & \frac{2}{3} & -1 & -8 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

from which it follows that the degree of the GCD of $\tilde{f}(w)$ and $\tilde{g}(w)$ is two. The

polynomial formed from the last non-zero row of this matrix is

$$\frac{2}{3}\binom{4}{2}(1-2w)^2w^2 - \binom{4}{3}(1-2w)w^3 - 8\binom{4}{4}w^4,$$

and the deletion of the extraneous factor w^2 yields the GCD,

$$\bar{d}(w, \theta = 2) = \binom{2}{0}(1-2w)^2 - \frac{1}{2}\binom{2}{1}(1-2w)w - 2\binom{2}{2}w^2.$$

It is readily verified that the substitution $w = x/\theta = x/2$ yields $g(x)$.

Consider now the transpose of $\bar{S}(\tilde{f}, \tilde{g})Q$,

$$\left(\bar{S}(\tilde{f}, \tilde{g})Q\right)^T = QU(\tilde{f}, \tilde{g})^T D^{-1},$$

where, from (3.25),

$$Q = \text{diag}\,[\ 1\ \ 1\ \ 1\ \ 2\ \ 1\ ],$$

because $m = 3$ and $n = 2$. It follows that the effect of Q is the multiplication of the 4th row of $\bar{S}(\tilde{f}, \tilde{g})^T$ by 2, and thus the reduction of $\left(\bar{S}(\tilde{f}, \tilde{g})Q\right)^T$ to upper triangular form also yields the GCD of $f(x)$ and $g(x)$. □

6.2 Examples

In this section, three examples that illustrate the theory in the previous sections are considered. The results obtained from $S(\dot{f}, \dot{g})$, $\bar{S}(\acute{f}, \alpha_2\acute{g})$ and $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ are shown because they enable the improvement in the computed estimate of $\hat{d}$ obtained by the inclusion of α and θ, and Q, to be observed. The matrices $S(\dot{f}, \dot{g})$, $\bar{S}(\acute{f}, \alpha_2\acute{g})$ and $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ are described as following:

- $S(\dot{f}, \dot{g})$ is the Sylvester matrix of the normalized Bernstein polynomials $\dot{f}(x)$ and $\dot{g}(x)$, which are defined in (6.10) and (6.11) respectively. The second and third preprocessing operations are not implemented, that is, $\alpha = \theta = 1$.

- $\bar{S}(\acute{f}, \alpha_2 \acute{g})$ is the Sylvester matrix of the modified Bernstein polynomials $\acute{f}(w)$ and $\acute{g}(w)$, which are defined in (6.29) and (6.30) respectively, that arise after the three preprocessing operations have been implemented.

- $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ is the Sylvester matrix of the modified Bernstein polynomials $\tilde{f}(w)$ and $\tilde{g}(w)$, which are defined in (6.27) and (6.28) respectively, that arise after the three preprocessing operations have been implemented.

Example 6.4. Consider the exact Bernstein polynomials $\hat{f}(x)$ and $\hat{g}(x)$, whose roots and multiplicities are specified in Table 6.1. It is seen that the degree of their GCD is $\hat{d} = 6$.

Root of $\hat{f}(x)$	Multiplicity
0.1300e+000	3
0.4300e+000	2
0.7800e+000	4
-0.8800e+000	3
0.9300e+000	4
1.3400e+000	6
3.2000e+000	1

Root of $\hat{g}(x)$	Multiplicity
0.1300e+000	4
0.2300e+000	4
-0.3600e+000	2
0.5300e+000	4
0.9300e+000	3
-1.4700e+000	2
2.4700e+000	4

Table 6.1: The roots and multiplicities of $\hat{f}(x)$ and $\hat{g}(x)$ for Example 6.4.

Noise with componentwise signal-to-noise ratio 10^8 is added to the coefficients of $\hat{f}(x)$ and $\hat{g}(x)$, and then the matrices $S(\dot{f}, \dot{g})$, $\bar{S}(\acute{f}, \alpha_2 \acute{g})$ and $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ are computed. The normalized singular values of $S(\dot{f}, \dot{g})$, $\bar{S}(\acute{f}, \alpha_2 \acute{g})$ and $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ are shown in Figures 6.2(a), (b) and (c) respectively. It is seen from Figures 6.2(b) and (c) that the numerical ranks of $\bar{S}(\acute{f}, \alpha_2 \acute{g})$ and $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ are clearly defined and equal to 40, which is correct because $\deg\mathrm{GCD}(\hat{f}, \hat{g}) = 6$. The results in Figures 6.2(b) and (c) were obtained with $\alpha_2 = 1.2401$ and $\theta_2 = 1.2335$, and $\alpha_1 = 1.2401$ and $\theta_1 = 1.2335$,

respectively. Furthermore, it is noted that the numerical rank of $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ is more clearly defined because a significantly larger gap in the singular values of $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ can be observed. However, Figure 6.2(a) shows that the Sylvester matrix $S(\dot{f}, \dot{g})$ is of full rank, which implies that $\hat{f}(x)$ and $\hat{g}(x)$ are coprime. □

Example 6.5. Consider the exact Bernstein polynomials $\hat{f}(x)$ and $\hat{g}(x)$, whose roots and multiplicities are specified in Table 6.2. It is seen that the degree of their GCD is $\hat{d} = 18$.

Root of $\hat{f}(x)$	Multiplicity
-0.3285e+000	5
0.3791e+000	6
-0.7113e+000	6
0.9214e+000	6
2.3125e+000	5
9.1474e+000	6

Root of $\hat{g}(x)$	Multiplicity
-0.3285e+000	3
0.3791e+000	7
0.5217e+000	3
0.9214e+000	7
1.4397e+000	3
9.1474e+000	3

Table 6.2: The roots and multiplicities of $\hat{f}(x)$ and $\hat{g}(x)$ for Example 6.5.

Noise with componentwise signal-to-noise ratio 10^8 is added to the coefficients of $\hat{f}(x)$ and $\hat{g}(x)$, and the matrices $S(\dot{f}, \dot{g})$, $\bar{S}(\acute{f}, \alpha_2\acute{g})$ and $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ are then computed.

Figures 6.3(a), (b) and (c) show the normalized singular values of $S(\dot{f}, \dot{g})$, $\bar{S}(\acute{f}, \alpha_2\acute{g})$ and $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ respectively. It is seen from Figure 6.3(c) that the rank loss of $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ is equal to $\deg\mathrm{GCD}(\hat{f}, \hat{g}) = 18$. The result in Figure 6.3(c) was obtained with $\alpha_1 = 4.7326$ and $\theta_1 = 1.3298$. However, it is seen from Figures 6.3(a) and (b) that $\hat{f}(x)$ and $\hat{g}(x)$ are coprime because the matrices $S(\dot{f}, \dot{g})$ and $\bar{S}(\acute{f}, \alpha_2\acute{g})$ are not rank deficient. The result in Figure 6.3(b) was obtained with $\alpha_2 = 0.1253$ and $\theta_2 = 1.3387$. □

Example 6.6. Consider the exact Bernstein polynomials $\hat{f}(x)$ and $\hat{g}(x)$, whose roots and multiplicities are specified in Table 6.3. It is seen that the degree of their GCD is $\hat{d} = 22$.

Root of $\hat{f}(x)$	Multiplicity
0.1000e+000	7
-0.2700e+000	3
0.5600e+000	5
0.7500e+000	6
0.8200e+000	3
1.3700e+000	5
1.4600e+000	4

Root of $\hat{g}(x)$	Multiplicity
0.1000e+000	8
0.5600e+000	6
0.7500e+000	6
0.9900e+000	5
-1.2000e+000	4
1.3700e+000	4
2.1200e+000	3

Table 6.3: The roots and multiplicities of $\hat{f}(x)$ and $\hat{g}(x)$ for Example 6.6.

Noise with componentwise signal-to-noise ratio 10^8 is applied to the coefficients of $\hat{f}(x)$ and $\hat{g}(x)$, and the matrices $S(\dot{f}, \dot{g})$, $\bar{S}(\acute{f}, \alpha_2 \acute{g})$ and $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ are then computed.

The results shown in Figure 6.4 are the same as those obtained in Figure 6.3 for Example 6.5. In particular, the matrices $S(\dot{f}, \dot{g})$ and $\bar{S}(\acute{f}, \alpha_2 \acute{g})$ yield the incorrect results respectively, but $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ yields the correct result because $S(\dot{f}, \dot{g})$ and $\bar{S}(\acute{f}, \alpha_2 \acute{g})$ have full rank, and the numerical rank of $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ is equal to 47. The result in Figure 6.4(b) was obtained with $\alpha_2 = 4.6176e + 002$ and $\theta_2 = 1.3771$, and the result in Figure 6.4(c) was obtained with $\alpha_1 = 2.8062e + 001$ and $\theta_1 = 1.4308$. □

6.3 Discussion

The results of the examples in Section 6.2 are consistent because the matrices $S(\dot{f}, \dot{g})$ and $\bar{S}(\acute{f}, \alpha_2 \acute{g})$ yield incorrect results but $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ yields the correct result. Therefore, it is instructive to consider the superiority of $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ with respect to the matrices $S(\dot{f}, \dot{g})$ and $\bar{S}(\acute{f}, \alpha_2 \acute{g})$. We will firstly consider $S(\dot{f}, \dot{g})$ and $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$.

If $A \in \mathbb{R}^{p \times q}, p \geq q$, is an arbitrary matrix of rank r that is perturbed to $A + \delta A$, then it is shown in [30] that

$$|\sigma_i(A + \delta A) - \sigma_i(A)| \leq \|\delta A\|_2, \qquad i = 1, \ldots, q,$$

where $\sigma_i(A), i = 1, \ldots, q$, are the singular values of A. It follows that

$$|\sigma_i(A + \delta A) - \sigma_i(A)| \leq \left(\frac{\|\delta A\|_2}{\|A\|_2} \right) \sigma_1(A), \qquad i = 1, \ldots, q,$$

and thus if the errors in $\sigma_i(A)$ and $\|A\|_2$ are

$$|\delta\sigma_i(A)| = |\sigma_i(A + \delta A) - \sigma_i(A)| \qquad \text{and} \qquad \Delta A = \frac{\|\delta A\|_2}{\|A\|_2},$$

respectively, then

$$\frac{|\delta\sigma_i(A)|}{\Delta A} \leq \sigma_1(A) = \|A\|_2 \leq \sqrt{q}\|A\|_1, \qquad i = 1, \ldots, q, \tag{6.31}$$

and it follows that the upper bound on the ratio of the absolute error in the singular values of A to the relative error in $\|A\|_2$ is decreased by reducing $\|A\|_1$. The upper bound in (6.31) is expressed in terms of the 1-norm of A, rather than its 2-norm, because the diagonal matrix Q postmultiplies $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})$, and its effect is therefore most clearly quantified by examining the column sums of $S(\dot{f}, \dot{g})$ and $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$.

The application of (6.31) to $S(\dot{f}, \dot{g})$ and $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ yields

$$\frac{|\delta\sigma_i(S)|}{\Delta S} \leq \sqrt{m+n}\|S\|_1, \qquad i = 1, \ldots, m+n, \qquad S = S(\dot{f}, \dot{g}),$$

and

$$\frac{|\delta\sigma_i(\bar{S}Q)|}{\Delta(\bar{S}Q)} \leq \sqrt{m+n}\|\bar{S}Q\|_1, \qquad i = 1, \ldots, m+n, \qquad \bar{S}Q = \bar{S}(\tilde{f}, \alpha_1\tilde{g})Q,$$

and since $\Delta S = \Delta\Big(\bar{S}Q\Big) \approx \varepsilon$, where ε is the componentwise signal-to-noise ratio, it follows that if

$$\|\bar{S}Q\|_1 < \|S\|_1, \tag{6.32}$$

the singular nature of $S(\hat{f}, \hat{g})$ is more faithfully preserved when computations are performed on $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ than when computations are performed on $S(\dot{f}, \dot{g})$.

Figures 6.5, 6.6 and 6.7 show the column sums of $S(\dot{f}, \dot{g})$ and $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$,

$$\sigma_j = \log \sum_{i=1}^{m+n} |S(\dot{f}, \dot{g})|_{i,j}, \qquad \tau_j = \log \sum_{i=1}^{m+n} |\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q|_{i,j}, \qquad j = 1, \ldots, m+n,$$

for Examples 6.4, 6.5 and 6.6 respectively, where $|P|_{i,j}$ denotes the absolute value of element (i, j) of P and $\log = \log_{10}$. The figures show that the maximum absolute column sum of $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ is significantly smaller than the maximum absolute column sum of $S(\dot{f}, \dot{g})$, and thus the inequality (6.32) is satisfied, which implies that the inclusion of α_1 and θ_1, and Q, yields greatly improved computational results. Furthermore, the column sums of $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ span a smaller range than the column sums of $S(\dot{f}, \dot{g})$ by several orders of magnitude. For example, Figure 6.6 shows that the column sums τ_j span approximately 3 orders of magnitude, but the column sums σ_j span about 10 orders of magnitude.

Consider now the superiority of $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ with respect to $\bar{S}(\acute{f}, \alpha_2\acute{g})$. The above analysis suggests that it is desirable to examine the absolute column sums of $\bar{S}(\acute{f}, \alpha_2\acute{g})$ and $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$.

It follows from (3.7), (3.8), (3.25), (3.28), (6.4) and (6.7) that the sums of the absolute values of the entries in each of the first n columns, and each of the last m columns,

of $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ are

$$\Gamma_j(a) = \frac{1}{\lambda}\sum_{i=0}^{m} \frac{|a_i|\binom{m}{i}\binom{n-1}{j-1}\theta_1^i}{\binom{m+n-1}{i+j-1}}, \qquad j = 1, \ldots, n, \tag{6.33}$$

and

$$\Gamma_j(b) = \frac{\alpha_1}{\mu}\sum_{i=0}^{n} \frac{|b_i|\binom{n}{i}\binom{m-1}{j-1}\theta_1^i}{\binom{m+n-1}{i+j-1}}, \qquad j = 1, \ldots, m, \tag{6.34}$$

where

$$\|\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q\|_1 = \max_{j=1,\ldots,n;k=1,\ldots,m}\{\Gamma_j(a), \Gamma_k(b)\}. \tag{6.35}$$

Similarly, it follows from (3.7), (3.8), (3.11) and (6.9) that the sums of the absolute values of the entries in each of the first n columns, and each of the last m columns, of $\bar{S}(\acute{f}, \alpha_2\acute{g})$ are

$$\Sigma_j(a) = \frac{1}{\eta}\sum_{i=0}^{m} \frac{|a_i|\binom{m}{i}\theta_2^i}{\binom{m+n-1}{i+j-1}}, \qquad j = 1, \ldots, n, \tag{6.36}$$

and

$$\Sigma_j(b) = \frac{\alpha_2}{\rho}\sum_{i=0}^{n} \frac{|b_i|\binom{n}{i}\theta_2^i}{\binom{m+n-1}{i+j-1}}, \qquad j = 1, \ldots, m, \tag{6.37}$$

where

$$\|\bar{S}(\acute{f}, \alpha_2\acute{g})\|_1 = \max_{j=1,\ldots,n;k=1,\ldots,m}\{\Sigma_j(a), \Sigma_k(b)\}. \tag{6.38}$$

The above analysis based on $S(\dot{f}, \dot{g})$ and $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ can be repeated for $\bar{S}(\acute{f}, \alpha_2\acute{g})$ and $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$, and therefore if

$$\|\bar{S}Q\|_1 < \|\bar{S}\|_1, \tag{6.39}$$

where $\bar{S}Q = \bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ and $\bar{S} = \bar{S}(\acute{f}, \alpha_2\acute{g})$, the singular nature of $S(\hat{f}, \hat{g})$ is more faithfully preserved when computations are performed on $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ than when computations are performed on $\bar{S}(\acute{f}, \alpha_2\acute{g})$.

It is seen from Examples 6.4, 6.5 and 6.6 that since α_1 for $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ and α_2 for

$\bar{S}(\acute{f}, \alpha_2 \acute{g})$ are $O(1)$, they are much smaller than terms of the form $\binom{m}{i}$, $\binom{n}{j}$ and $\binom{m+n-1}{k}$, they can be set equal to one. In addition, it is seen from these three examples that θ_1 for $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ is approximately equal to θ_2 for $\bar{S}(\acute{f}, \alpha_2 \acute{g})$, that is, $\theta_1 \approx \theta_2$. Therefore, the effect of α_1 and θ_1 on $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ is similar to the effect of α_2 and θ_2 on $\bar{S}(\acute{f}, \alpha_2 \acute{g})$, and thus the difference between $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ and $\bar{S}(\acute{f}, \alpha_2 \acute{g})$ is mainly caused by the matrix Q. In particular, since $\alpha_1 = O(1)$ and $\alpha_2 = O(1)$, and $\theta_1 \approx \theta_2$ for Examples 6.4, 6.5 and 6.6, and λ, μ, η and ρ are normalization constants, it follows that the differences between (6.33) and (6.34), and (6.36) and (6.37), arise from the combinatorial factors $\binom{n-1}{j-1}$, $j = 1, \ldots, n$, and $\binom{m-1}{j-1}$, $j = 1, \ldots, m$, in (6.33) and (6.34) respectively.

Figures 6.8, 6.9 and 6.10 show the column sums of $\bar{S}(\acute{f}, \alpha_2 \acute{g})$ and $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ for Examples 6.4, 6.5 and 6.6 respectively. It is seen from these figures that the variation in magnitude of $\{\Gamma_j(a), \Gamma_k(b)\}$ is much smaller than the variation in magnitude of $\{\Sigma_j(a), \Sigma_k(b)\}$ for $j = 1, \ldots, n$, and $k = 1, \ldots, m$. In particular, the maximum value of $\{\Gamma_j(a), \Gamma_k(b)\}$ is less than the maximum value of $\{\Sigma_j(a), \Sigma_k(b)\}$ for $j = 1, \ldots, n$, and $k = 1, \ldots, m$, and it therefore follows from (6.35) and (6.38) that (6.39) is satisfied. This confirms that the inclusion of Q is important for improved computational results because it reduces the column sums of $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$.

The parameters α_1 and θ_1 are included in the computation of d in order to minimize the ratio of the entry of maximum absolute value, to the entry of minimum absolute value, of $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$, where $\tilde{f} = \tilde{f}(w, \theta)$ and $\tilde{g} = \tilde{g}(w, \theta)$ are defined in (6.27) and (6.28) respectively. This objective is consistent with the effect of Q, but there is an important difference between α_1 and θ_1, and Q:

- The parameters α_1 and θ_1 are functions of the Bernstein basis coefficients $\bar{a}_i$ and

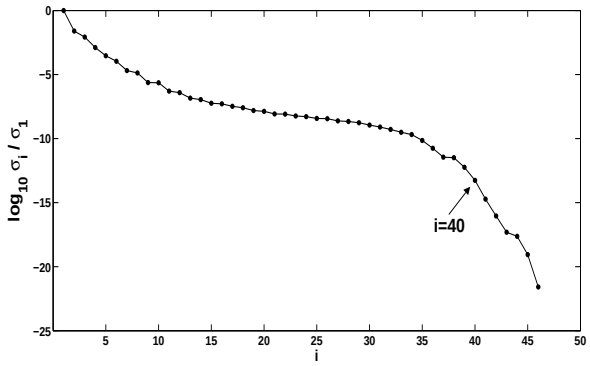
$\bar{b}_j$, combinatorial factors of the forms $\binom{m}{i}$, $\binom{n}{j}$ and $\binom{m+n-1}{k}$, and the entries of Q. The importance of their inclusion therefore increases as the degrees of $f(x)$ and $g(x)$, and the variation in magnitude of the coefficients $\bar{a}_i$ and $\bar{b}_j$, increase.

- The entries of Q are functions of m and n, but they are independent of the coefficients $\bar{a}_i$ and $\bar{b}_j$. They therefore mitigate the effects of the combinatorial factors $\binom{m}{i}$, $\binom{n}{j}$ and $\binom{m+n-1}{k}$, for large values of m and n, in a Sylvester matrix.

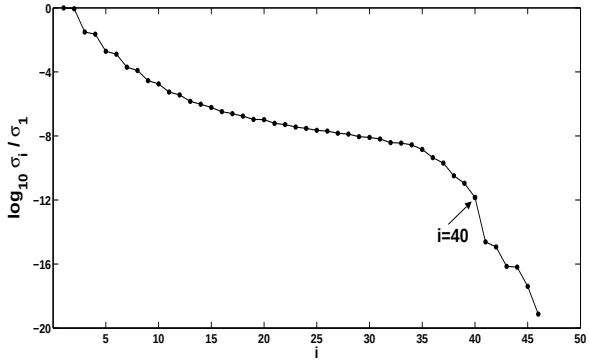
6.4 Summary

This chapter has introduced the computation of the degree of an AGCD of two inexact Bernstein polynomials $f(x)$ and $g(x)$ using their Sylvester matrix. It was shown that in order to obtain the best results, it is necessary to include the diagonal matrix Q and preprocess the Sylvester matrix $S(f, g)Q$ by three operations, such that all the computations are performed on the Sylvester matrix $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$, where α is a scaling parameter. The third preprocessing operation introduces the parameter θ, which transforms the polynomials from the Bernstein basis to the modified Bernstein basis. The optimal values of α and θ are obtained from the solution of a linear programming problem.

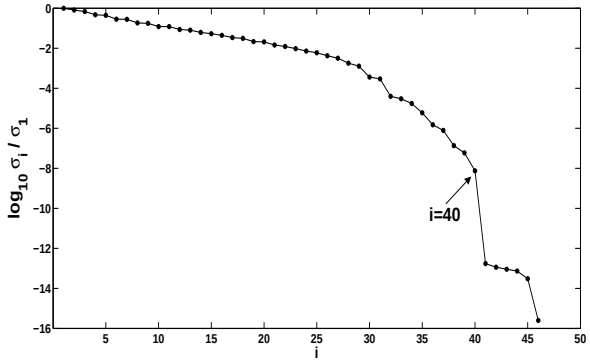
In Chapters 5 and 6, the degree of an AGCD of inexact polynomials is determined by observing the variation of the singular values of their Bézout and Sylvester resultant matrices. Furthermore, some advanced techniques using the first principal angle and the residual of an approximate linear algebraic equation can also be used to estimate the degree of an AGCD, which involves the Sylvester subresultant matrices. These issues are discussed in the next chapter.



(a)

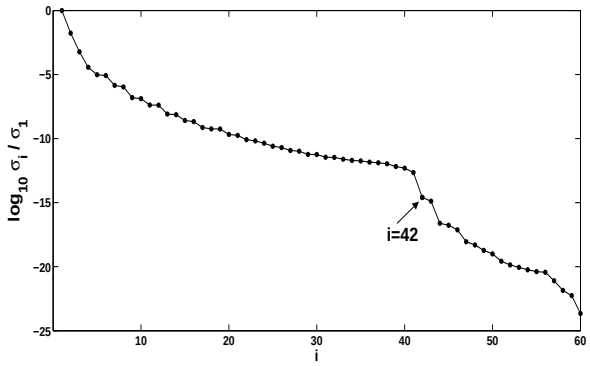


(b)

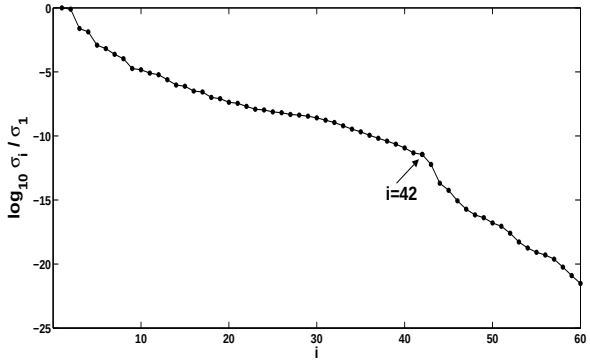


(c)

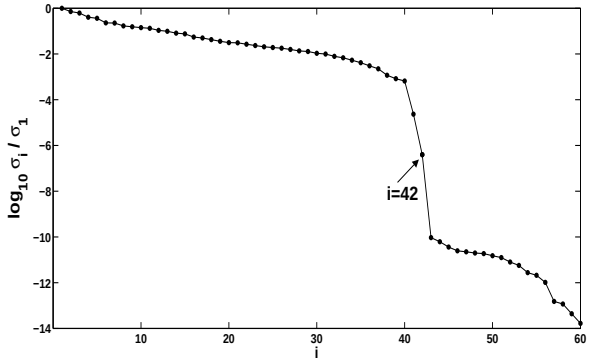
Figure 6.2: The normalized singular values of (a) $S(\dot{f}, \dot{g})$, (b) $\bar{S}(\acute{f}, \alpha_2 \acute{g})$ and (c) $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ for Example 6.4.



(a)

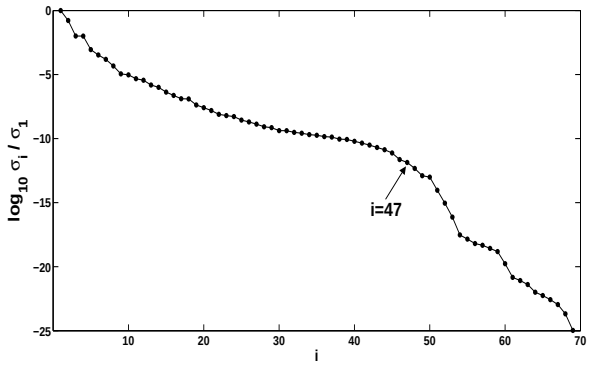


(b)

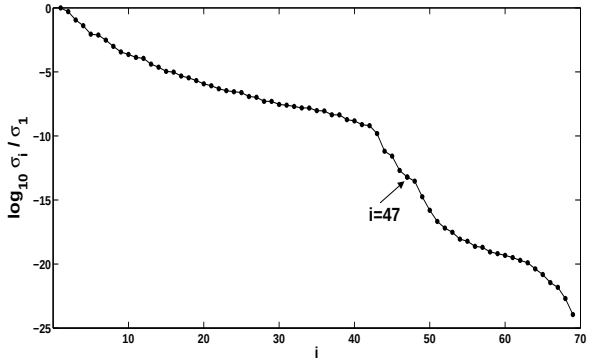


(c)

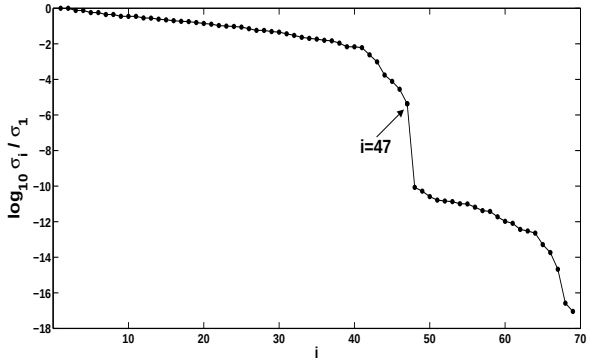
Figure 6.3: The normalized singular values of (a) $S(\dot{f}, \dot{g})$, (b) $\bar{S}(\acute{f}, \alpha_2 \acute{g})$ and (c) $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ for Example 6.5.



(a)

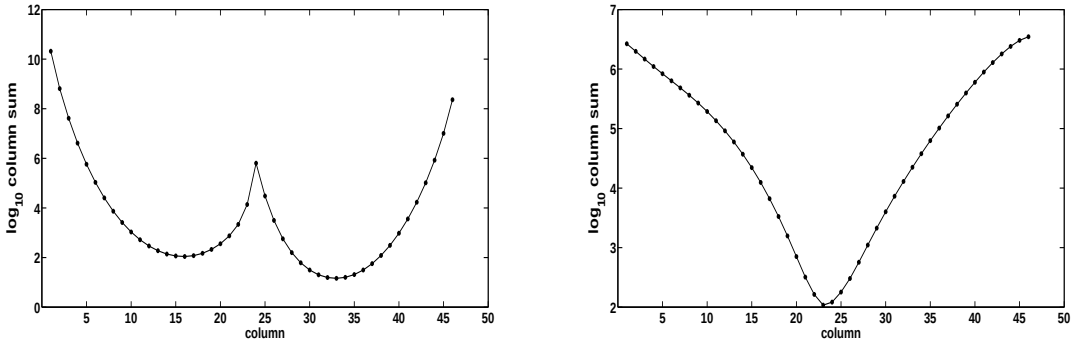


(b)



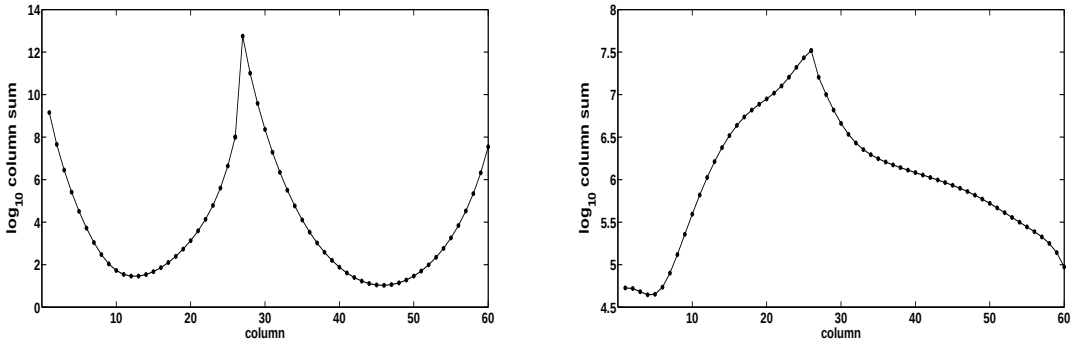
(c)

Figure 6.4: The normalized singular values of (a) $S(\dot{f}, \dot{g})$, (b) $\bar{S}(\acute{f}, \alpha_2 \acute{g})$ and (c) $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ for Example 6.6.



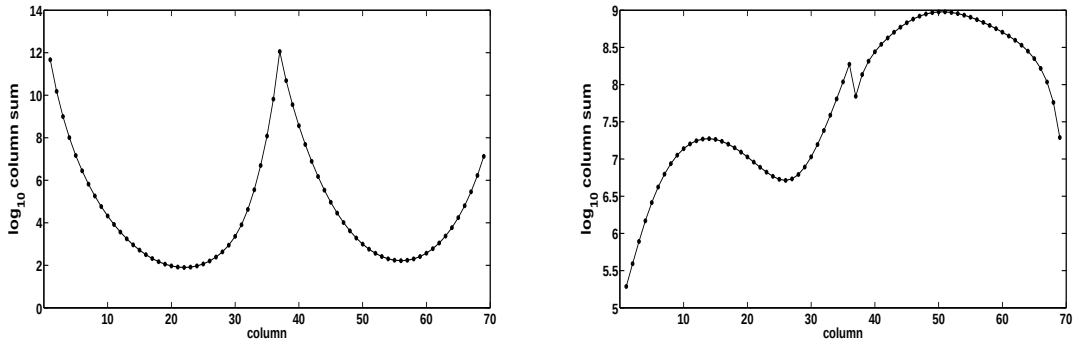
(a) (b)

Figure 6.5: The column sums of (a) $S(\dot{f}, \dot{g})$ and (b) $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$, for Example 6.4.



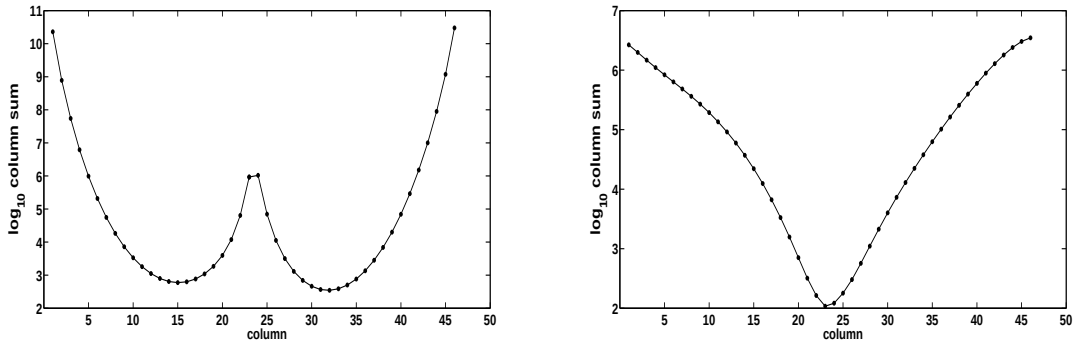
(a) (b)

Figure 6.6: The column sums of (a) $S(\dot{f}, \dot{g})$ and (b) $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$, for Example 6.5.



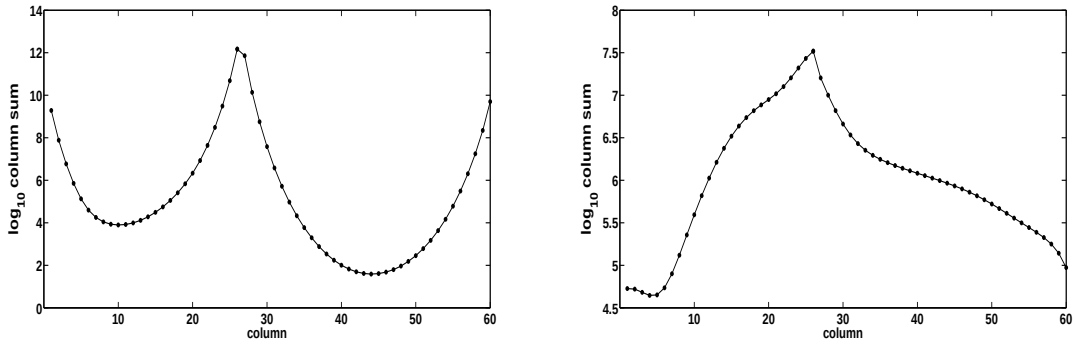
(a) (b)

Figure 6.7: The column sums of (a) $S(\dot{f}, \dot{g})$ and (b) $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$, for Example 6.6.



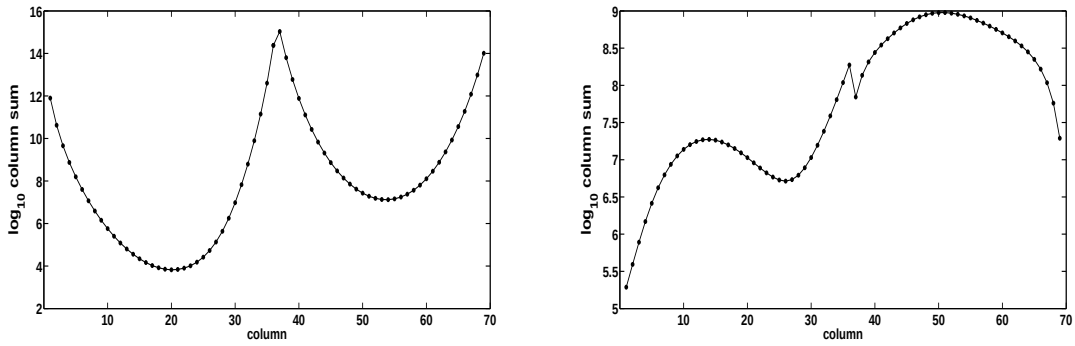
(a) (b)

Figure 6.8: The column sums of (a) $\bar{S}(\acute{f}, \alpha_2 \acute{g})$ and (b) $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$, for Example 6.4.



(a) (b)

Figure 6.9: The column sums of (a) $\bar{S}(\acute{f}, \alpha_2 \acute{g})$ and (b) $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$, for Example 6.5.



(a) (b)

Figure 6.10: The column sums of (a) $\bar{S}(\acute{f}, \alpha_2 \acute{g})$ and (b) $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$, for Example 6.6.

Chapter 7

The degree of an AGCD, Part III

Chapter 6 introduced the determination of the degree of an AGCD from two forms of the Sylvester matrix, $S(f, g)$ and $S(f, g)Q$. This chapter extends the work in Chapter 6 and involves the Sylvester subresultant matrices of $S(f, g)$ and $S(f, g)Q$, $S_k(f, g)$ and $S_k(f, g)Q_k$, which are introduced in Chapter 3. In particular, this chapter considers two methods to calculate the degree of an AGCD of the inexact polynomials $f(x)$ and $g(x)$. One method uses the first principal angle between a line and a hyperplane, the equations of which are calculated from the Sylvester subresultant matrices, and the other method uses the residual of a linear algebraic equation whose coefficient matrix and right hand side vector are also derived from the Sylvester subresultant matrices. Before these two methods are introduced, the criteria that measure the error in a linear algebraic equation must be considered. This issue is discussed in the next section.

7.1 Measures of the error in a linear algebraic equation

This section considers two criteria that measure the error in a linear algebraic equation. One criterion is based on the first principal angle between a line and a hyperplane, and the other criterion is based on the residual of a linear algebraic equation. The concepts of the first principal angle and residual are introduced in this section, and the computations of the first principal angle and residual will be described in Sections 7.4.1 and 7.4.2 respectively.

Consider a linear algebraic equation

$$Ax = b, \tag{7.1}$$

where $A \in \mathbb{R}^{m \times n}$ is a matrix, $b \in \mathbb{R}^m$ is a vector and $x \in \mathbb{R}^n$ is the solution vector for this equation. The first principal angle between the vector b and the matrix A is the smallest angle between b and an arbitrary vector in the space spanned by the columns of A, and the residual of (7.1) is equal to $\|b - Ax\|$, where $\|\cdot\| = \|\cdot\|_2$. Equation (7.1) possesses a non-zero solution, and thus b lies in the column space of A, which implies that the first principal angle between b and the column space of A and the residual of (7.1) are equal to zero.

However, when

$$Ax \neq b, \tag{7.2}$$

is specified, b does not lie in the column space of A because (7.2) does not possess a non-zero solution, which implies that the first principal angle between b and the column space of A and the residual of (7.2) are not equal to zero.

The above analysis suggests that the error in a linear algebraic equation can be

measured by the criteria based on the first principal angle and residual. In addition, the error measures using the criteria based on the first principal angle and residual can be used to determine if a linear algebraic equation has a non-zero solution. This analysis is extended for the Sylvester subresultant matrices, which will be addressed in the next section.

7.2 The error measures for the Sylvester subresultant matrices

This section considers the error measures using the criteria based on the first principal angle and residual for the Sylvester subresultant matrices. Two forms of the Sylvester subresultant matrices, $S_k(f,g)$ and $S_k(f,g)Q_k$ should be considered. The following analysis is developed for $S_k(f,g)$, and the same analysis can be repeated for $S_k(f,g)Q_k$.

If the exact Bernstein polynomials $\hat{f}(x)$ and $\hat{g}(x)$ defined in (3.4) have a GCD of degree $\hat{d} > 0$, they possess a common divisor of degree k, where $1 \leq k \leq \hat{d}$. Therefore, there exists a polynomial $\hat{d}_k(x)$ of degree k such that

$$\hat{f}(x) = \hat{u}_k(x)\hat{d}_k(x) \qquad \text{and} \qquad \hat{g}(x) = \hat{v}_k(x)\hat{d}_k(x), \tag{7.3}$$

where the quotient polynomials $\hat{u}_k(x)$ and $\hat{v}_k(x)$ are

$$\begin{aligned} \hat{u}_k(x) &= \sum_{i=0}^{m-k} \hat{u}_{k,i} \binom{m-k}{i} (1-x)^{m-k-i} x^i, \\ \hat{v}_k(x) &= \sum_{j=0}^{n-k} \hat{v}_{k,j} \binom{n-k}{j} (1-x)^{n-k-j} x^j, \end{aligned}$$

respectively, and the common divisor polynomial $\hat{d}_k(x)$ is

$$\hat{d}_k(x) = \sum_{i=0}^{k} \hat{d}_{k,i} \binom{k}{i} (1-x)^{k-i} x^i.$$

It has been shown in Section 3.3.1 that it follows from (7.3) that $\hat{f}\hat{v} - \hat{g}\hat{u} = 0$, which can be written in matrix form as

$$S_k(\hat{f}, \hat{g}) p_k(\hat{u}_k, \hat{v}_k) = 0, \qquad k = 1, \ldots, \min(m, n), \tag{7.4}$$

where $S_k(\hat{f}, \hat{g})$ is the kth Sylvester subresultant matrix defined in (3.18) and

$$p_k(\hat{u}_k, \hat{v}_k) = \begin{bmatrix} \hat{v}_{k,0}\binom{n-k}{0} \\ \vdots \\ \hat{v}_{k,n-k}\binom{n-k}{n-k} \\ -\hat{u}_{k,0}\binom{m-k}{0} \\ \vdots \\ -\hat{u}_{k,m-k}\binom{m-k}{m-k} \end{bmatrix}.$$

Since the degree of the GCD of $\hat{f}(x)$ and $\hat{g}(x)$ is $\hat{d} \geq 1$, these polynomials possess common divisors of degree $1, 2, \ldots, \hat{d}$, but they do not have a common divisor of degree $\hat{d}+1$. It has been shown in Section 3.3.1 that (7.4) possesses a non-zero solution for $k = 1, \ldots, \hat{d}$, but it does not possess a non-zero solution for $k = \hat{d}+1, \ldots, \min(m, n)$. The methods based on the first principal angle and residual of an approximate linear algebraic equation for the computation of the degree of an AGCD of two polynomials require that the homogeneous equation (7.4) is converted to a linear algebraic equation, which can be achieved by moving one column of $S_k(\hat{f}, \hat{g})$ to the right hand side. For $k = 1, \ldots, \hat{d}$, (7.4) possesses a non-zero solution and therefore $S_k(\hat{f}, \hat{g})$ must be rank deficient, which implies that at least one column of $S_k(\hat{f}, \hat{g})$ is linearly dependent

on its other columns. This analysis is expressed as

$$L_{k,i}x_{k,i} = l_{k,i} \qquad \text{for} \qquad k = 1, \ldots, \hat{d}, \tag{7.5}$$

for at least one column of $S_k(\hat{f}, \hat{g})$, where $l_{k,i} \in \mathbb{R}^{m+n-k+1}$ is the ith column of $S_k(\hat{f}, \hat{g})$, $L_{k,i} \in \mathbb{R}^{(m+n-k+1)\times(m+n-2k+1)}$ is the remaining matrix of $S_k(\hat{f}, \hat{g})$ after the removal of the ith column and

$$x_{k,i} = \begin{bmatrix} x_1 & \cdots & x_{i-1} & x_{i+1} & \cdots & x_{m+n-2k+2} \end{bmatrix}^T \in \mathbb{R}^{m+n-2k+1},$$

and

$$p_k(\hat{u}_k, \hat{v}_k) = \begin{bmatrix} \hat{v}_{k,0}\binom{n-k}{0} \\ \vdots \\ \hat{v}_{k,n-k}\binom{n-k}{n-k} \\ -\hat{u}_{k,0}\binom{m-k}{0} \\ \vdots \\ -\hat{u}_{k,m-k}\binom{m-k}{m-k} \end{bmatrix} = \begin{bmatrix} x_1 \\ \vdots \\ x_{i-1} \\ -1 \\ x_{i+1} \\ \vdots \\ x_{m+n-2k+2} \end{bmatrix} \in \mathbb{R}^{m+n-2k+2}.$$

However, there does not exist a column of $S_k(\hat{f}, \hat{g})$, such that

$$L_{k,i}x_{k,i} = l_{k,i} \qquad \text{for} \qquad k = \hat{d}+1, \ldots, \min(m, n). \tag{7.6}$$

The operation of removing the ith column from $S_k(\hat{f}, \hat{g})$ is achieved by postmultiplying $S_k(\hat{f}, \hat{g})$ by $M_{k,i} \in \mathbb{R}^{(m+n-2k+2)\times(m+n-2k+1)}$, which is equal to the identity matrix after the removal of the ith column,

$$M_{k,i} = \begin{bmatrix} e_{k,1} & e_{k,2} & \cdots & e_{k,i-1} & e_{k,i+1} & \cdots & e_{k,m+n-2k+1} & e_{k,m+n-2k+2} \end{bmatrix},$$

where $i = 1, \ldots, m+n-2k+2$, and $e_{k,i} \in \mathbb{R}^{m+n-2k+2}$ is the ith unit basis vector. The ith column of $S_k(\hat{f}, \hat{g})$ is equal to $S_k(\hat{f}, \hat{g})e_{k,i}$.

Example 7.1. Let $m = 3$, $n = 2$ and $k = 2$. Thus $S_2 = S_2(\hat{f}, \hat{g}) \in \mathbb{R}^{4\times 3}$ is

$$S_2 = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \\ j & k & l \end{bmatrix},$$

$$S_2 M_{2,1} = S_2 \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} b & c \\ e & f \\ h & i \\ k & l \end{bmatrix}, \qquad S_2 e_{2,1} = S_2 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} a \\ d \\ g \\ j \end{bmatrix},$$

$$S_2 M_{2,2} = S_2 \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} a & c \\ d & f \\ g & i \\ j & l \end{bmatrix}, \qquad S_2 e_{2,2} = S_2 \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} b \\ e \\ h \\ k \end{bmatrix},$$

$$S_2 M_{2,3} = S_2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} a & b \\ d & e \\ g & h \\ j & k \end{bmatrix}, \qquad S_2 e_{2,3} = S_2 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} c \\ f \\ i \\ l \end{bmatrix}.$$

□

This analysis must be compared with the previous work in [63], which considers the determination of the degree of an AGCD of two power polynomials using the methods based on the first principal angle and residual. The work in [63] moves the first column of $S_k(\hat{f}, \hat{g})$ to the right hand side of (7.4). The reason is that for

$k = 1, \ldots, \hat{d}$, (7.4) possesses a non-zero solution, which stores the coefficients of the quotient polynomials $\hat{v}_k(x)$ and $\hat{u}_k(x)$. Since the leading coefficient of power polynomial is non-zero, the leading coefficient of $\hat{v}_k(x)$ is non-zero. This implies that the first element in the solution vector, which is the leading coefficient of $\hat{v}_k(x)$, is non-zero, and therefore the first column of $S_k(\hat{f}, \hat{g})$ is linearly dependent on the other columns of $S_k(\hat{f}, \hat{g})$. When the first column is moved to the right hand side of (7.4), the equation is still satisfied. However, since a Bernstein polynomial is a combination of Bernstein basis functions and the degree of each Bernstein basis function is equal to the degree of Bernstein polynomial, the leading coefficient of Bernstein polynomial is not always non-zero. When the methods are performed in the Bernstein basis, for $k = 1, \ldots, \hat{d}$, (7.4) has a non-zero solution but the first element in the solution vector, which is the leading scaled coefficient of the quotient polynomial $\hat{v}_k(x)$, is not always non-zero, which implies that the first column of $S_k(\hat{f}, \hat{g})$ is not always linearly dependent, and therefore we can not guarantee that the equation is satisfied when the first column of $S_k(\hat{f}, \hat{g})$ is moved to the right hand side of (7.4).

For $k = 1, \ldots, \hat{d}$, (7.5) possesses a non-zero solution for at least one column of $S_k(\hat{f}, \hat{g})$, which implies that one column of $S_k(\hat{f}, \hat{g})$, $l_{k,i}$, lies in the space spanned by the remaining $m+n-2k+1$ columns of $S_k(\hat{f}, \hat{g})$, $L_{k,i}$, for these values of k. However, for $k = \hat{d}+1, \ldots, \min(m, n)$, (7.6) does not possess a non-zero solution for any column of $S_k(\hat{f}, \hat{g})$, and therefore no column of $S_k(\hat{f}, \hat{g})$, $l_{k,i}$, lies in the column space of $L_{k,i}$ for these values of k. As stated in Section 7.1, the first principal angle and residual can be used to determine if $l_{k,i}$ lies in the column space of $L_{k,i}$. Since at least one column of $S_k(\hat{f}, \hat{g})$, $l_{k,i}$, lies in the column space of $L_{k,i}$ for $k = 1, \ldots, \hat{d}$, the first principal angle and residual between $l_{k,i}$ and $L_{k,i}$ are equal to zero for these values of

k. However, for $k = \hat{d}+1, \ldots, \min(m,n)$, there does not exist one column of $S_k(\hat{f},\hat{g})$, $l_{k,i}$, such that the first principal angle and residual between $l_{k,i}$ and $L_{k,i}$ are equal to zero.

However, when noise is added to the exact polynomials $\hat{f}(x)$ and $\hat{g}(x)$, such that the inexact polynomials $f(x)$ and $g(x)$ are specified, $L_{k,i}x_{k,i} = l_{k,i}$ does not possess a non-zero solution for all $k = 1, \ldots, \min(m,n)$, because $f(x)$ and $g(x)$ are coprime. As explained above, in the exact case, there may be several columns that are linearly dependent upon the other columns in $S_k(\hat{f},\hat{g})$ for $k = 1, \ldots, \hat{d}$. Therefore, when the inexact polynomials $f(x)$ and $g(x)$ are specified, there exist several columns that are almost linearly dependent upon the other columns in $S_k(f,g)$ for $k = 1, \ldots, \hat{d}$, because the noise level is small. It is therefore necessary to choose one column of $S_k(f,g)$ as the optimal column for $k = 1, \ldots, \hat{d}$, in terms of a specified criterion. In particular, the optimal column of $S_k(f,g)$, l_{k,i^*}, is defined as the column that is closest to be linearly dependent upon the other columns in $S_k(f,g)$, such that $L_{k,i^*}x_{k,i^*} \approx l_{k,i^*}$ has an approximate solution with smaller error. The optimal column of $S_k(f,g)$ is selected for each value of $k = 1, \ldots, \hat{d}$, and therefore $L_{k,i^*}x_{k,i^*} \approx l_{k,i^*}$ has an approximate solution for $k = 1, \ldots, \hat{d}$, which implies that the first principal angle and residual between l_{k,i^*} and L_{k,i^*} are not equal to zero but relatively small for these values of k. However, it was shown in the exact case that there exists no column that is linearly dependent upon the other columns in $S_k(\hat{f},\hat{g})$ for $k = \hat{d}+1, \ldots, \min(m,n)$. Therefore, there exists no column that is almost linearly dependent upon the other columns in $S_k(f,g)$ for $k = \hat{d}+1, \ldots, \min(m,n)$, when the inexact polynomials $f(x)$ and $g(x)$ are specified, and thus there does not exist a column of $S_k(f,g)$, such that $L_{k,i}x_{k,i} \approx l_{k,i}$ for $k = \hat{d}+1, \ldots, \min(m,n)$. The approximation $L_{k,i}x_{k,i} \approx l_{k,i}$ does not

possess an approximate solution for $k = \hat{d}+1, \ldots, \min(m,n)$, which implies that the first principal angle and residual between $l_{k,i}$ and $L_{k,i}$ are relatively large for these values of k, and therefore $\hat{d}$ is equal to the value of k for which the maximum change in the first principal angle and residual occurs. This analysis suggests that the degree of an AGCD can be determined by observing the maximum change in the first principal angle and residual. The analysis is written as

$$\begin{aligned} &L_{k,i^*}x_{k,i^*} \approx l_{k,i^*} \qquad \text{for} \qquad k = 1, \ldots, \hat{d}, \\ &L_{k,i}x_{k,i} \neq l_{k,i} \qquad \text{for} \qquad k = \hat{d}+1, \ldots, \min(m,n); i = 1, \ldots, m+n-2k+2, \end{aligned} \tag{7.7}$$

where i^* denotes the index of optimal column of $S_k(f,g)$.

Since the optimal column of $S_k(f,g)$ is unknown for $k = 1, \ldots, \hat{d}$, it is necessary to span each column in $S_k(f,g)$ in order to determine the optimal column yielding the smaller error in (7.7) for these values of k. The approximation (7.7) is then rewritten as

$$\begin{aligned} &L_{k,i}x_{k,i} \approx l_{k,i} \qquad \text{for} \qquad k = 1, \ldots, \hat{d}; i = 1, \ldots, m+n-2k+2, \\ &L_{k,i}x_{k,i} \neq l_{k,i} \qquad \text{for} \qquad k = \hat{d}+1, \ldots, \min(m,n); i = 1, \ldots, m+n-2k+2. \end{aligned} \tag{7.8}$$

In addition, the degree $\hat{d}$ of the GCD is unknown and to be determined, and therefore it is necessary to choose the optimal column for $k = 1, \ldots, \min(m,n)$. The approximation (7.8) is then replaced by

$$L_{k,i}x_{k,i} \approx l_{k,i} \qquad \text{for} \qquad k = 1, \ldots, \min(m,n); i = 1, \ldots, m+n-2k+2, \tag{7.9}$$

and the largest value of k for which (7.9) has an approximate solution with the smaller error is equal to $\hat{d}$. As stated earlier, the error in (7.9) can be measured by the criteria based on the first principal angle and residual.

The above analysis can be repeated for another form of the Sylvester subresultant

matrices, $S_k(f,g)Q_k$, which is defined in (3.34), and therefore given the inexact polynomials $f(x)$ and $g(x)$, the approximation

$$H_{k,i}x_{k,i} \approx h_{k,i} \qquad \text{for} \qquad k = 1, \ldots, \min(m,n); i = 1, \ldots, m+n-2k+2, \quad (7.10)$$

is established. The vector $h_{k,i} \in \mathbb{R}^{m+n-k+1}$ is the ith column of $S_k(f,g)Q_k$ and $H_{k,i} \in \mathbb{R}^{(m+n-k+1)\times(m+n-2k+1)}$ is the remaining matrix of $S_k(f,g)Q_k$ after the removal of the ith column. The largest value of k for which (7.10) has an approximate solution with the smaller error is equal to the degree of an AGCD.

Computational experiments have shown that two forms of the Sylvester subresultant matrices, $S_k(f,g)$ and $S_k(f,g)Q_k$, must be preprocessed before computations on them are performed. In particular, since each of the Sylvester subresultant matrices has the same partitioned structure as the Sylvester resultant matrix, the preprocessing operations for the Sylvester resultant matrix described in Chapter 6 are required for each of the Sylvester subresultant matrices for the same reason. The preprocessing operations are addressed in the next section.

7.3 Preprocessing operations

Given two inexact Bernstein polynomials $f(x)$ and $g(x)$ defined in (4.2), the preprocessing operations performed on two forms of the Sylvester subresultant matrices, $S_k(f,g)$ and $S_k(f,g)Q_k$ are considered. The three preprocessing operations shown in Chapter 6, normalization of the coefficients of $f(x)$ and $g(x)$, the introduction of a parameter α and a transformation of the independent variable x to a new independent variable w, are required to be implemented for $S_k(f,g)$ and $S_k(f,g)Q_k$. It should be noted that because the entries of the subresultant matrices

$S_k(f,g)Q_k$, $k = 1, \ldots, \min(m,n)$, are different, the three preprocessing operations must be implemented for each value of k. Similarly, these three preprocessing operations are necessary to be performed for each of the subresultant matrices $S_k(f,g)$, $k = 1, \ldots, \min(m,n)$.

Because the entries of $S_k(f,g)Q_k$ are more complex than the entries of $S_k(f,g)$, it is better to consider the preprocessing operations associated with $S_k(f,g)Q_k$, and then the preprocessing operations for $S_k(f,g)$ can be simply obtained from it.

7.3.1 Normalization of the polynomials

It follows from (3.14), (3.15), (3.30) and (3.34) that the coefficients of $f(x)$ occupy the first $(n-k+1)$ columns, and the coefficients of $g(x)$ occupy the last $(m-k+1)$ columns, of $S_k(f,g)Q_k$. As stated earlier, the partitioned nature of $S_k(f,g)Q_k$ may cause computational problems when the coefficients of $f(x)$ are significantly larger or smaller than the coefficients of $g(x)$. Therefore, it is necessary to normalize the entries of the first $(n-k+1)$ columns and last $(m-k+1)$ columns of $S_k(f,g)Q_k$, respectively, to obtain a more balanced matrix $S_k(f,g)Q_k$. The geometric mean normalization was used in Section 6.1.1 and this form of normalization is also used in this section.

This section develops the general forms of normalization constants for $S_k(f,g)Q_k$, $k = 1, \ldots, \min(m,n)$, and the general forms are functions of k. The following analysis is a generalization of the analysis in Section 6.1.1, which only considers the calculation of normalization constants for the Sylvester matrix $S(f,g)Q$, that is for $k = 1$ because $S_1(f,g)Q_1 = S(f,g)Q$.

Consider the coefficients $a_i\binom{m}{i}, i = 0, \ldots, m$, which occupy the first $(n-k+1)$ columns of $S_k(f,g)Q_k$. It follows from (3.14), (3.15), (3.30) and (3.34) that the

general expression for the product of the magnitudes of the terms that contain the coefficient $a_i\binom{m}{i}$ in $S_k(f,g)Q_k$ is

$$\left|\frac{a_i\binom{m}{i}\binom{n-k}{0}}{\binom{m+n-k}{i}}\right|\left|\frac{a_i\binom{m}{i}\binom{n-k}{1}}{\binom{m+n-k}{i+1}}\right|\left|\frac{a_i\binom{m}{i}\binom{n-k}{2}}{\binom{m+n-k}{i+2}}\right|\cdots\left|\frac{a_i\binom{m}{i}\binom{n-k}{n-k}}{\binom{m+n-k}{i+n-k}}\right| = \frac{\left|a_i\binom{m}{i}\right|^{n-k+1}\prod_{r=0}^{n-k}\binom{n-k}{r}}{\prod_{t=i}^{n-k+i}\binom{m+n-k}{t}}.$$

Therefore, the product of all the terms that contain the coefficients of $f(x)$ in $S_k(f,g)Q_k$, $k=1,\ldots,\min(m,n)$, is

$$\prod_{i=0}^{m}\left(\frac{\left|a_i\binom{m}{i}\right|^{n-k+1}\prod_{r=0}^{n-k}\binom{n-k}{r}}{\prod_{t=i}^{n-k+i}\binom{m+n-k}{t}}\right),$$

and since the coefficients of $f(x)$ occur a total of $(n-k+1)(m+1)$ times in $S_k(f,g)Q_k$, the geometric mean of these terms in $S_k(f,g)Q_k$ is

$$\lambda_k = \left\{\prod_{i=0}^{m}\left(\frac{\left|a_i\binom{m}{i}\right|^{n-k+1}\prod_{r=0}^{n-k}\binom{n-k}{r}}{\prod_{t=i}^{n-k+i}\binom{m+n-k}{t}}\right)\right\}^{\frac{1}{(n-k+1)(m+1)}}, \quad k=1,\ldots,\min(m,n). \tag{7.11}$$

The numerator of this expression simplifies to

$$\left\{\prod_{i=0}^{m}\left|a_i\binom{m}{i}\right|\right\}^{\frac{1}{m+1}}\left\{\prod_{r=0}^{n-k}\binom{n-k}{r}\right\}^{\frac{1}{n-k+1}},$$

where care must be taken in the computation of these terms in order to prevent overflow.

Consider now the denominator in (7.11),

$$\left\{\prod_{i=0}^{m}\prod_{t=i}^{n-k+i}\binom{m+n-k}{t}\right\}^{\frac{1}{(n-k+1)(m+1)}},$$

which can be evaluated efficiently by a recurrence equation. In particular, if $P_{i,k}$ is defined as

$$P_{i,k} = \prod_{t=i}^{n-k+i}\binom{m+n-k}{t}, \qquad i=0,\ldots,m; k=1,\ldots,\min(m,n), \tag{7.12}$$

then

$$P_{i+1,k} = \prod_{t=i+1}^{n-k+i+1}\binom{m+n-k}{t} = \frac{\binom{m+n-k}{n-k+i+1}\prod_{t=i}^{n-k+i}\binom{m+n-k}{t}}{\binom{m+n-k}{i}},$$

and thus

$$P_{i+1,k} = P_{i,k}\frac{\binom{m+n-k}{n-k+i+1}}{\binom{m+n-k}{i}} = P_{i,k}\prod_{t=0}^{n-k}\frac{(m-i+t)}{(i+1+t)}, \qquad i = 0,\ldots,m-1.$$

The starting value of this recurrence relationship, for each value of $k = 1,\ldots,\min(m,n)$, is

$$P_{0,k} = \prod_{t=0}^{n-k}\binom{m+n-k}{t}.$$

Therefore, the geometric mean (7.11) of all the terms that contain the coefficients $a_i\binom{m}{i}$ is

$$\lambda_k = \frac{\left(\prod_{i=0}^{m}\left|a_i\binom{m}{i}\right|\right)^{\frac{1}{m+1}}\left(\prod_{r=0}^{n-k}\binom{n-k}{r}\right)^{\frac{1}{n-k+1}}}{\{\prod_{i=0}^{m}P_{i,k}\}^{\frac{1}{(n-k+1)(m+1)}}}, \qquad k = 1,\ldots,\min(m,n).$$

It follows that the normalized form of $f(x)$ for $S_k(f,g)Q_k$, $k = 1,\ldots,\min(m,n)$ is

$$\breve{f}_k(x) = \sum_{i=0}^{m}\bar{a}_{k,i}\binom{m}{i}(1-x)^{m-i}x^i, \qquad \bar{a}_{k,i} = \frac{a_i}{\lambda_k}, \tag{7.13}$$

where $\breve{f}_1(x) = \breve{f}(x)$ which is defined in (6.5), because $S_1(f,g)Q_1 = S(f,g)Q$.

This analysis can be repeated for $g(x)$, and its normalized form for $S_k(f,g)Q_k$ is

$$\breve{g}_k(x) = \sum_{j=0}^{n}\bar{b}_{k,j}\binom{n}{j}(1-x)^{n-j}x^j, \qquad \bar{b}_{k,j} = \frac{b_j}{\mu_k}, \tag{7.14}$$

where

$$\mu_k = \frac{\left(\prod_{j=0}^{n}\left|b_j\binom{n}{j}\right|\right)^{\frac{1}{n+1}}\left(\prod_{r=0}^{m-k}\binom{m-k}{r}\right)^{\frac{1}{m-k+1}}}{\left\{\prod_{j=0}^{n}L_{j,k}\right\}^{\frac{1}{(m-k+1)(n+1)}}},$$

and

$$L_{j,k} = \prod_{t=j}^{m-k+j}\binom{m+n-k}{t}, \qquad j = 0,\ldots,n; k = 1,\ldots,\min(m,n), \tag{7.15}$$

and $\breve{g}_1(x) = \breve{g}(x)$ which is defined in (6.6), because $S_1(f,g)Q_1 = S(f,g)Q$.

It is noted that the normalization constants λ_k and μ_k are functions of k. The normalized coefficients $\bar{a}_{k,i}$ and $\bar{b}_{k,j}$ in (7.13) and (7.14) enable the subresultant matrices $S_k(\breve{f}_k, \breve{g}_k)Q_k = D_k^{-1}T_k(\breve{f}_k, \breve{g}_k)Q_k, k = 1, \ldots, \min(m,n)$, where the matrices D_k^{-1}, $T_k(\breve{f}_k, \breve{g}_k)$ and Q_k are defined in (3.14), (3.15) and (3.30), respectively, to be computed. This analysis can be repeated for the subresultant matrices $S_k(f,g) = D_k^{-1}T_k(f,g), k = 1, \ldots, \min(m,n)$, and the normalization constants for $f(x)$ and $g(x)$ in $S_k(f,g)$ are, respectively,

$$\eta_k = \frac{\left(\prod_{i=0}^{m} \left|a_i \binom{m}{i}\right|\right)^{\frac{1}{m+1}}}{\left(\prod_{i=0}^{m} P_{i,k}\right)^{\frac{1}{(n-k+1)(m+1)}}}, \qquad k = 1, \ldots, \min(m,n),$$

and

$$\rho_k = \frac{\left(\prod_{j=0}^{n} \left|b_j \binom{n}{j}\right|\right)^{\frac{1}{n+1}}}{\left(\prod_{j=0}^{n} L_{j,k}\right)^{\frac{1}{(m-k+1)(n+1)}}}, \qquad k = 1, \ldots, \min(m,n),$$

where $P_{i,k}$ and $L_{j,k}$ are defined in (7.12) and (7.15) respectively, and thus the normalized forms of $f(x)$ and $g(x)$ for $S_k(f,g)$ are

$$\dot{f}_k(x) = \sum_{i=0}^{m} \ddot{a}_{k,i} \binom{m}{i} (1-x)^{m-i} x^i, \qquad \ddot{a}_{k,i} = \frac{a_i}{\eta_k}, \tag{7.16}$$

and

$$\dot{g}_k(x) = \sum_{j=0}^{n} \ddot{b}_{k,j} \binom{n}{j} (1-x)^{n-j} x^j, \qquad \ddot{b}_{k,j} = \frac{b_j}{\rho_k}, \tag{7.17}$$

where $\dot{f}_1(x) = \dot{f}(x)$ which is defined in (6.10), and $\dot{g}_1(x) = \dot{g}(x)$ which is defined in (6.11), because $S_1(f,g) = S(f,g)$.

The subresultant matrices $S_k(\dot{f}_k, \dot{g}_k) = D_k^{-1}T_k(\dot{f}_k, \dot{g}_k), k = 1, \ldots, \min(m,n)$, are computed from the normalized coefficients $\ddot{a}_{k,i}$ and $\ddot{b}_{k,j}$ in (7.16) and (7.17).

7.3.2 Scaling a polynomial by an arbitrary constant

It is shown in Section 6.1.2 that when two inexact polynomials in a floating point environment are considered, the numerical rank of their Sylvester matrix is a function of α, due to its partitioned structure. Because the subresultant matrices have the same structure, it is necessary to choose the value of α with care in order to obtain the best results. Therefore, the subresultant matrices,

$$S_k(\breve{f}_k, \alpha\breve{g}_k)Q_k, \qquad k = 1, \ldots, \min(m, n),$$

and

$$S_k(\dot{f}_k, \alpha\dot{g}_k), \qquad k = 1, \ldots, \min(m, n),$$

should be considered. The computation of the optimal value of α is considered in Section 7.3.5, after the third preprocessing operation has been introduced.

7.3.3 A transformation of the independent variable

As stated in Chapters 5 and 6, numerical problems may occur when computations are performed on a matrix whose entries vary widely in magnitude. Therefore, it is necessary to minimize the ratio of the maximum entry, in magnitude, to the minimum entry, in magnitude, of each subresultant matrix. As shown in Chapters 5 and 6, this can be achieved by the introduction of a new parameter θ. In particular, for the subresultant matrices, $S_k(\breve{f}_k, \alpha\breve{g}_k)Q_k,\ k = 1, \ldots, \min(m, n)$, this preprocessing operation is implemented by the substitution (5.1), which transforms the polynomials $\breve{f}_k(x)$ and $\breve{g}_k(x)$ defined in (7.13) and (7.14) respectively to

$$\bar{f}_k(w, \theta) = \sum_{i=0}^{m} \left(\bar{a}_{k,i}\theta^i\right) \binom{m}{i} (1 - \theta w)^{m-i} w^i, \tag{7.18}$$

and

$$\bar{g}_k(w,\theta) = \sum_{j=0}^{n} \left(\bar{b}_{k,j}\theta^j\right) \binom{n}{j} (1-\theta w)^{n-j} w^j, \tag{7.19}$$

respectively. It follows from (7.18) and (7.19) that the substitution (5.1) transforms the Bernstein basis to the modified Bernstein basis, whose basis functions for polynomials of degree m, $\phi_i^m(w,\theta)$, are defined in (5.4). Therefore, the subresultant matrices, $S_k(\breve{f}_k, \alpha\breve{g}_k)Q_k$, $k = 1,\ldots,\min(m,n)$, which are expressed in the Bernstein basis, are transformed to $\bar{S}_k(\bar{f}_k, \alpha\bar{g}_k)Q_k$, $k = 1,\ldots,\min(m,n)$, which are the subresultant matrices of the modified Bernstein polynomials $\bar{f}_k(w,\theta)$ and $\alpha\bar{g}_k(w,\theta)$.

For the subresultant matrices, $S_k(\dot{f}_k, \alpha\dot{g}_k)$, $k = 1,\ldots,\min(m,n)$, $\dot{f}_k(x)$ and $\dot{g}_k(x)$, which are defined in (7.16) and (7.17) respectively, are transformed similarly to

$$\ddot{\bar{f}}_k(w,\theta) = \sum_{i=0}^{m} \left(\ddot{a}_{k,i}\theta^i\right) \binom{m}{i} (1-\theta w)^{m-i} w^i, \tag{7.20}$$

and

$$\ddot{\bar{g}}_k(w,\theta) = \sum_{j=0}^{n} \left(\ddot{b}_{k,j}\theta^j\right) \binom{n}{j} (1-\theta w)^{n-j} w^j, \tag{7.21}$$

and thus the subresultant matrices, $S_k(\dot{f}_k, \alpha\dot{g}_k)$, $k = 1,\ldots,\min(m,n)$, defined in the Bernstein basis, are transformed to $\bar{S}_k(\ddot{f}_k, \alpha\ddot{g}_k)$, $k = 1,\ldots,\min(m,n)$, which are the subresultant matrices of the modified Bernstein polynomials $\ddot{\bar{f}}_k(w,\theta)$ and $\alpha\ddot{\bar{g}}_k(w,\theta)$.

The coefficients of $\bar{f}_k(w,\theta)$ and $\bar{g}_k(w,\theta)$ are $\bar{a}_{k,i}\theta^i, i = 0,\ldots,m$, and $\bar{b}_{k,j}\theta^j, j = 0,\ldots,n$, respectively, and it will be shown that the optimal values of the parameters α and θ minimize the ratio of the maximum element, in magnitude, to the minimum element, in magnitude, of $\bar{S}_k(\bar{f}_k, \alpha\bar{g}_k)Q_k$. Therefore, the form of $\bar{S}_k(\bar{f}_k, \alpha\bar{g}_k)Q_k$ must be developed, and this is considered in the next section.

7.3.4 The Sylvester subresultant matrices for the modified Bernstein basis

The substitution (5.1) transforms $S_k(\breve{f}_k, \alpha\breve{g}_k)Q_k$, which is defined in the Bernstein basis, to $\bar{S}_k(\bar{f}_k, \alpha\bar{g}_k)Q_k$, which is defined in the modified Bernstein basis, and therefore it is necessary to develop expressions for the entries of $\bar{S}_k(\bar{f}_k, \alpha\bar{g}_k)Q_k$. Section 6.1.4 develops the form of the Sylvester matrix in the modified Bernstein basis, $\bar{S}(\bar{f}, \alpha\bar{g})Q$, which is a particular case for $k = 1$ because $\bar{S}_1(\bar{f}_1, \alpha\bar{g}_1)Q_1 = \bar{S}(\bar{f}, \alpha\bar{g})Q$.

Consider the polynomials $\hat{p}(x)$ and $\hat{q}(x)$ expressed in the Bernstein basis,

$$\hat{p}(x) = \sum_{i=0}^{m} \hat{c}_i \binom{m}{i} (1-x)^{m-i} x^i,$$

and

$$\hat{q}(x) = \sum_{j=0}^{n} \hat{d}_j \binom{n}{j} (1-x)^{n-j} x^j,$$

whose GCD is of degree $\hat{d}$. If $\hat{u}_k(x)$ and $\hat{v}_k(x)$ are quotient polynomials expressed in the Bernstein basis, and $\hat{d}_k(x)$ is a common divisor polynomial of degree k, also expressed in the Bernstein basis, $k = 1, \ldots, \min(m, n)$, then

$$\hat{d}_k(x) = \frac{\hat{p}(x)}{\hat{u}_k(x)} = \frac{\hat{q}(x)}{\hat{v}_k(x)}, \quad \deg \hat{u}_k < \deg \hat{p} = m, \quad \deg \hat{v}_k < \deg \hat{q} = n, \tag{7.22}$$

where $\hat{u}_k(x)$ and $\hat{v}_k(x)$ are of degrees $m-k$ and $n-k$ respectively. The normalization described in Section 7.3.1 and parameter substitution (5.1) transform (7.22) to

$$\bar{d}_k(w, \theta) = \frac{\bar{p}_k(w, \theta)}{\bar{u}_k(w, \theta)} = \frac{\bar{q}_k(w, \theta)}{\bar{v}_k(w, \theta)}, \ \deg \bar{u}_k < \deg \bar{p}_k = m, \ \deg \bar{v}_k < \deg \bar{q}_k = n, \tag{7.23}$$

where $k = 1, \ldots, \hat{d}$,

$$\bar{p}_k(w, \theta) = \sum_{i=0}^{m} \left(\bar{c}_{k,i} \theta^i\right) \binom{m}{i} (1 - \theta w)^{m-i} w^i,$$

$$\bar{q}_k(w,\theta) = \sum_{j=0}^{n} \left(\bar{d}_{k,j}\theta^j\right) \binom{n}{j} (1-\theta w)^{n-j} w^j,$$

$$\bar{u}_k(w,\theta) = \sum_{i=0}^{m-k} \left(\bar{u}_{k,i}\theta^i\right) \binom{m-k}{i} (1-\theta w)^{m-k-i} w^i,$$

$$\bar{v}_k(w,\theta) = \sum_{i=0}^{n-k} \left(\bar{v}_{k,i}\theta^i\right) \binom{n-k}{i} (1-\theta w)^{n-k-i} w^i,$$

and

$$\bar{d}_k(w,\theta) = \sum_{i=0}^{k} \left(\bar{d}_{k,i}\theta^i\right) \binom{k}{i} (1-\theta w)^{k-i} w^i.$$

It follows from (7.23) that

$$\bar{p}_k(w,\theta)\bar{v}_k(w,\theta) = \bar{q}_k(w,\theta)\bar{u}_k(w,\theta), \qquad k = 1,\ldots,\hat{d}, \tag{7.24}$$

and then (7.24) can be expressed in matrix form as

$$\left(D_k^{-1} U_k(\bar{p}_k,\bar{q}_k)\right) s_k(\bar{u}_k,\bar{v}_k) = 0, \tag{7.25}$$

where $\bar{p}_k = \bar{p}_k(w,\theta)$, $\bar{q}_k = \bar{q}_k(w,\theta)$, $\bar{u}_k = \bar{u}_k(w,\theta)$, $\bar{v}_k = \bar{v}_k(w,\theta)$, the diagonal matrix D_k^{-1} is defined in (3.14),

$$U_k(\bar{p}_k,\bar{q}_k) = \begin{bmatrix} C_k(\bar{p}_k) & D_k(\bar{q}_k) \end{bmatrix} \in \mathbb{R}^{(m+n-k+1)\times(m+n-2k+2)}, \tag{7.26}$$

the Toeplitz matrices $C_k(\bar{p}_k)$ and $D_k(\bar{q}_k)$ are given by, respectively,

$$C_k(\bar{p}_k) = \begin{bmatrix} \bar{c}_{k,0}\binom{m}{0} & & \\ \bar{c}_{k,1}\binom{m}{1}\theta & \ddots & \\ \vdots & \ddots & \bar{c}_{k,0}\binom{m}{0} \\ \vdots & \ddots & \bar{c}_{k,1}\binom{m}{1}\theta \\ \bar{c}_{k,m-1}\binom{m}{m-1}\theta^{m-1} & \ddots & \vdots \\ \bar{c}_{k,m}\binom{m}{m}\theta^m & \ddots & \vdots \\ & \ddots & \bar{c}_{k,m-1}\binom{m}{m-1}\theta^{m-1} \\ & & \bar{c}_{k,m}\binom{m}{m}\theta^m \end{bmatrix} \in \mathbb{R}^{(m+n-k+1)\times(n-k+1)},$$

and

$$D_k(\bar{q}_k) = \begin{bmatrix} \bar{d}_{k,0}\binom{n}{0} & & \\ \bar{d}_{k,1}\binom{n}{1}\theta & \ddots & \\ \vdots & \ddots & \bar{d}_{k,0}\binom{n}{0} \\ \vdots & \ddots & \bar{d}_{k,1}\binom{n}{1}\theta \\ \bar{d}_{k,n-1}\binom{n}{n-1}\theta^{n-1} & \ddots & \vdots \\ \bar{d}_{k,n}\binom{n}{n}\theta^n & \ddots & \vdots \\ & \ddots & \bar{d}_{k,n-1}\binom{n}{n-1}\theta^{n-1} \\ & & \bar{d}_{k,n}\binom{n}{n}\theta^n \end{bmatrix} \in \mathbb{R}^{(m+n-k+1)\times(m-k+1)},$$

and $s_k(\bar{u}_k, \bar{v}_k) \in \mathbb{R}^{m+n-2k+2}$ is equal to

$$\begin{bmatrix} \bar{v}_{k,0}\binom{n-k}{0} & \bar{v}_{k,1}\binom{n-k}{1}\theta & \dots & \dots & \bar{v}_{k,n-k}\binom{n-k}{n-k}\theta^{n-k} \\ -\bar{u}_{k,0}\binom{m-k}{0} & -\bar{u}_{k,1}\binom{m-k}{1}\theta & \dots & \dots & -\bar{u}_{k,m-k}\binom{m-k}{m-k}\theta^{m-k} \end{bmatrix}^T.$$

It follows from (3.29) that the vector $s_k(\bar{u}_k, \bar{v}_k)$ is written as

$$s_k(\bar{u}_k, \bar{v}_k) = Q_k t_k(\bar{u}_k, \bar{v}_k),$$

where Q_k is defined in (3.30), and $t_k(\bar{u}_k, \bar{v}_k) \in \mathbb{R}^{m+n-2k+2}$ is equal to

$$\begin{bmatrix} \bar{v}_{k,0} & \bar{v}_{k,1}\theta & \cdots & \bar{v}_{k,n-k}\theta^{n-k} & -\bar{u}_{k,0} & -\bar{u}_{k,1}\theta & \cdots & -\bar{u}_{k,m-k}\theta^{m-k} \end{bmatrix}^T, \quad (7.27)$$

and therefore (7.25) can be written as

$$\left(D_k^{-1}U_k(\bar{p}_k, \bar{q}_k)Q_k\right) t_k(\bar{u}_k, \bar{v}_k) = 0. \quad (7.28)$$

Since the degree of the GCD of $\bar{p}_k(w, \theta)$ and $\bar{q}_k(w, \theta)$ is $\hat{d} \geq 1$, these polynomials possess common divisors of degree $1, 2, \ldots, \hat{d}$, but they do not have a common divisor of degree $\hat{d} + 1$:

$$\begin{aligned} &\text{rank } D_k^{-1}U_k(\bar{p}_k, \bar{q}_k) < m + n - 2k + 2, \qquad k = 1, \ldots, \hat{d}, \\ &\text{rank } D_k^{-1}U_k(\bar{p}_k, \bar{q}_k) = m + n - 2k + 2, \qquad k = \hat{d} + 1, \ldots, \min(m, n), \end{aligned}$$

and

$$\begin{aligned} &\text{rank } D_k^{-1}U_k(\bar{p}_k, \bar{q}_k)Q_k < m + n - 2k + 2, \qquad k = 1, \ldots, \hat{d}, \\ &\text{rank } D_k^{-1}U_k(\bar{p}_k, \bar{q}_k)Q_k = m + n - 2k + 2, \qquad k = \hat{d} + 1, \ldots, \min(m, n). \end{aligned}$$

It follows from (7.28) that the kth subresultant matrix, $\bar{S}_k(\bar{f}_k, \alpha\bar{g}_k)Q_k$, of $\bar{f}_k(w, \theta)$ and $\alpha\bar{g}_k(w, \theta)$, which are defined in (7.18) and (7.19) respectively, is

$$\bar{S}_k(\bar{f}_k, \alpha\bar{g}_k)Q_k = D_k^{-1}U_k(\bar{f}_k, \alpha\bar{g}_k)Q_k, \quad (7.29)$$

and similarly, it follows from (7.25) that the kth subresultant matrix, $\bar{S}_k(\ddot{f}_k, \alpha\ddot{g}_k)$, of $\ddot{f}_k(w, \theta)$ and $\alpha\ddot{g}_k(w, \theta)$, which are defined in (7.20) and (7.21) respectively, is

$$\bar{S}_k(\ddot{f}_k, \alpha\ddot{g}_k) = D_k^{-1}U_k(\ddot{f}_k, \alpha\ddot{g}_k). \quad (7.30)$$

The form of $\bar{S}_k(\bar{f}_k, \alpha\bar{g}_k)Q_k$ is established in (7.29), and a criterion for the calculation of its optimal values of α and θ is considered in the next section.

7.3.5 The optimal values of α and θ

This section considers the calculation of the optimal values of α and θ. The criterion described in Section 6.1.5 is appropriate for both $\bar{S}_k(\bar{f}_k, \alpha\bar{g}_k)Q_k$ and $\bar{S}_k(\ddot{f}_k, \alpha\ddot{g}_k)$, but the optimal values of α and θ are different because of the diagonal matrix Q_k. It is adequate to consider the calculation of the optimal values of α and θ for $\bar{S}_k(\bar{f}_k, \alpha\bar{g}_k)Q_k$ because the computation of the optimal values of α and θ for $\bar{S}_k(\ddot{f}_k, \alpha\ddot{g}_k)$ follows easily. As stated before, computations performed on a matrix whose elements vary widely in magnitude may cause computational problems. Therefore, as shown in Section 6.1.5, it is desirable to choose the optimal values of α and θ respectively, such that the ratio of the maximum element to the minimum element, in magnitude, of $\bar{S}_k(\bar{f}_k, \alpha\bar{g}_k)Q_k$ is minimized.

For the subresultant matrices $\bar{S}_k(\bar{f}_k, \alpha\bar{g}_k)Q_k$, $k = 1, \ldots, \min(m, n)$, the optimal values of α and θ must be computed for each value of k. In particular, the calculation of the optimal values of α and θ for $k = 1$ is identical to the computation of the optimal values of α and θ for $\bar{S}(\bar{f}, \alpha\bar{g})Q$ shown in Section 6.1.5 because $\bar{S}_1(\bar{f}_1, \alpha\bar{g}_1)Q_1 = \bar{S}(\bar{f}, \alpha\bar{g})Q$.

It follows from (3.14), (3.30), (7.26) and (7.29) that the general expression for a non-zero element in the first $n - k + 1$ columns of $\bar{S}_k(\bar{f}_k, \alpha\bar{g}_k)Q_k$, $k = 1, \ldots, \min(m, n)$, is

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

and similarly, the general expression for a non-zero element in the last $m - k + 1$ columns of $\bar{S}_k(\bar{f}_k, \alpha\bar{g}_k)Q_k$, $k = 1, \ldots, \min(m, n)$, is

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

It is convenient to define the sets $\rho_k(\theta)$ and $\sigma_k(\alpha,\theta)$ as

$$\rho_k(\theta) = \left\{ \frac{\left|\bar{a}_{k,j}\binom{m}{j}\binom{n-k}{i}\theta^j\right|}{\binom{m+n-k}{i+j}} : j = 0,\ldots,m; i = 0,\ldots,n-k \right\},$$

and

$$\sigma_k(\alpha,\theta) = \left\{ \frac{\left|\alpha\bar{b}_{k,j}\binom{n}{j}\binom{m-k}{i}\theta^j\right|}{\binom{m+n-k}{i+j}} : j = 0,\ldots,n; i = 0,\ldots,m-k \right\},$$

respectively, and the optimal values of α and θ, $\alpha_1(k)$ and $\theta_1(k)$, minimize the ratio of the maximum element, in magnitude, to the minimum element, in magnitude, of $\bar{S}_k(\bar{f}_k, \alpha\bar{g}_k)Q_k$,

$$\alpha_1(k), \theta_1(k) = \arg\min_{\alpha,\theta} \left\{ \frac{\max\left\{ \max\{\rho_k(\theta)\}, \max\{\sigma_k(\alpha,\theta)\} \right\}}{\min\left\{ \min\{\rho_k(\theta)\}, \min\{\sigma_k(\alpha,\theta)\} \right\}} \right\}, k = 1,\ldots,\min(m,n).$$

This minimization problem is a function of k and Section 6.1.5 considers the same minimization problem for $k = 1$. Therefore, the following analysis is similar to the analysis shown in Section 6.1.5. In addition, it is important to note that the optimal values of α and θ are functions of k, which must be compared with the situation that prevails for the power basis because the optimal values of α and θ are independent of k for this basis.

This minimization problem can be written as:

Minimize $\frac{u}{v}$

subject to

$$\begin{aligned}
u &\geq \frac{\left|\bar{a}_{k,j}\binom{m}{j}\binom{n-k}{i}\theta^j\right|}{\binom{m+n-k}{i+j}}, && j=0,\ldots,m; i=0,\ldots,n-k,\\
u &\geq \frac{\left|\alpha\bar{b}_{k,j}\binom{n}{j}\binom{m-k}{i}\theta^j\right|}{\binom{m+n-k}{i+j}}, && j=0,\ldots,n; i=0,\ldots,m-k,\\
v &\leq \frac{\left|\bar{a}_{k,j}\binom{m}{j}\binom{n-k}{i}\theta^j\right|}{\binom{m+n-k}{i+j}}, && j=0,\ldots,m; i=0,\ldots,n-k,\\
v &\leq \frac{\left|\alpha\bar{b}_{k,j}\binom{n}{j}\binom{m-k}{i}\theta^j\right|}{\binom{m+n-k}{i+j}}, && j=0,\ldots,n; i=0,\ldots,m-k,\\
v &> 0,\\
\theta &> 0,\\
\alpha &> 0.
\end{aligned}$$

The transformations

$$U=\log u, \qquad V=\log v, \qquad \phi=\log\theta, \qquad \mu=\log\alpha, \tag{7.31}$$

and

$$\bar{\alpha}_{i,j}=\log\frac{\left|\bar{a}_{k,j}\binom{m}{j}\binom{n-k}{i}\right|}{\binom{m+n-k}{i+j}}, \qquad \bar{\beta}_{i,j}=\log\frac{\left|\bar{b}_{k,j}\binom{n}{j}\binom{m-k}{i}\right|}{\binom{m+n-k}{i+j}},$$

where $\log=\log_{10}$, enable this constrained minimization problem to be written as:

Minimize $U-V$

subject to

$$\begin{aligned}
U - j\phi &\geq \bar{\alpha}_{i,j}, && j=0,\ldots,m; i=0,\ldots,n-k,\\
U - j\phi - \mu &\geq \bar{\beta}_{i,j}, && j=0,\ldots,n; i=0,\ldots,m-k,\\
-V + j\phi &\geq -\bar{\alpha}_{i,j}, && j=0,\ldots,m; i=0,\ldots,n-k,\\
-V + j\phi + \mu &\geq -\bar{\beta}_{i,j}, && j=0,\ldots,n; i=0,\ldots,m-k.
\end{aligned} \tag{7.32}$$

The counter i appears only on the right hand side of these inequalities, and thus if $\bar{\lambda}_j$, $\bar{\mu}_j$, $\bar{\rho}_j$ and $\bar{\tau}_j$ are defined as

$$\bar{\lambda}_j = \max_{i=0,\ldots,n-k}\{\bar{\alpha}_{i,j}\} = \max_{i=0,\ldots,n-k}\left\{\log\frac{\left|\bar{a}_{k,j}\binom{m}{j}\binom{n-k}{i}\right|}{\binom{m+n-k}{i+j}}\right\}, \qquad j=0,\ldots,m,$$

$$\bar{\mu}_j = \max_{i=0,\ldots,m-k}\{\bar{\beta}_{i,j}\} = \max_{i=0,\ldots,m-k}\left\{\log\frac{\left|\bar{b}_{k,j}\binom{n}{j}\binom{m-k}{i}\right|}{\binom{m+n-k}{i+j}}\right\}, \qquad j=0,\ldots,n,$$

$$\bar{\rho}_j = \min_{i=0,\ldots,n-k}\{\bar{\alpha}_{i,j}\} = \min_{i=0,\ldots,n-k}\left\{\log\frac{\left|\bar{a}_{k,j}\binom{m}{j}\binom{n-k}{i}\right|}{\binom{m+n-k}{i+j}}\right\}, \qquad j=0,\ldots,m,$$

$$\bar{\tau}_j = \min_{i=0,\ldots,m-k}\{\bar{\beta}_{i,j}\} = \min_{i=0,\ldots,m-k}\left\{\log\frac{\left|\bar{b}_{k,j}\binom{n}{j}\binom{m-k}{i}\right|}{\binom{m+n-k}{i+j}}\right\}, \qquad j=0,\ldots,n,$$

then (7.32) can be written as

Minimize $U - V$

subject to

$$\begin{aligned}
U &- j\phi &&&&\geq \bar{\lambda}_j, \quad j=0,\ldots,m,\\
U &- j\phi &- \mu &&\geq \bar{\mu}_j, \quad j=0,\ldots,n,\\
-V &+ j\phi &&&\geq -\bar{\rho}_j, \quad j=0,\ldots,m,\\
-V &+ j\phi &+ \mu &&\geq -\bar{\tau}_j, \quad j=0,\ldots,n.
\end{aligned}$$

This minimization problem can be written as:

$$\text{Minimize} \quad \begin{bmatrix} 1 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} U \\ V \\ \phi \\ \mu \end{bmatrix} \qquad \text{subject to} \qquad A \begin{bmatrix} U \\ V \\ \phi \\ \mu \end{bmatrix} \geq b, \tag{7.33}$$

where $A \in \mathbb{R}^{(2m+2n+4)\times 4}$ and

$$b = \begin{bmatrix} \bar{\lambda}_0, \cdots, \bar{\lambda}_m, \bar{\mu}_0, \cdots, \bar{\mu}_n, -\bar{\rho}_0, \cdots, -\bar{\rho}_m, -\bar{\tau}_0, \cdots, -\bar{\tau}_n \end{bmatrix}^T \in \mathbb{R}^{2m+2n+4},$$

which is a standard linear programming problem. If $\alpha_1(k)$ and $\theta_1(k)$ are the solutions of the linear programming problem (7.33), then the polynomials (7.18) and (7.19) become

$$\tilde{f}_k = \tilde{f}_k(w) = \bar{f}_k(w, \theta_1) = \sum_{i=0}^{m} \left(\bar{a}_{k,i}\theta_1^i\right) \binom{m}{i} (1 - \theta_1 w)^{m-i} w^i, \tag{7.34}$$

and

$$\tilde{g}_k = \tilde{g}_k(w) = \bar{g}_k(w, \theta_1) = \sum_{j=0}^{n} \left(\bar{b}_{k,j}\theta_1^j\right) \binom{n}{j} (1 - \theta_1 w)^{n-j} w^j, \tag{7.35}$$

respectively, where $\alpha_1 = \alpha_1(k)$ and $\theta_1 = \theta_1(k)$, and the coefficients of these polynomials form the entries of the subresultant matrices $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$, $k = 1, \ldots, \min(m, n)$. Because the entries of the subresultant matrices $S_k(f, g)Q_k$, $k = 1, \ldots, \min(m, n)$, are functions of k, the three preprocessing operations must be implemented for each value of k.

A slight modification to the linear programming problem (7.33) allows the optimal values of α and θ, $\alpha_2(k)$ and $\theta_2(k)$ for $\bar{S}_k(\ddot{f}_k, \alpha\ddot{g}_k)$ to be calculated, and therefore the polynomials (7.20) and (7.21) become

$$\acute{f}_k = \acute{f}_k(w) = \ddot{\bar{f}}_k(w, \theta_2) = \sum_{i=0}^{m} \left(\ddot{a}_{k,i}\theta_2^i\right) \binom{m}{i} (1 - \theta_2 w)^{m-i} w^i, \tag{7.36}$$

and

$$\acute{g}_k = \acute{g}_k(w) = \ddot{\bar{g}}_k(w, \theta_2) = \sum_{j=0}^{n} \left(\ddot{\bar{b}}_{k,j}\theta_2^j\right) \binom{n}{j} (1 - \theta_2 w)^{n-j} w^j, \tag{7.37}$$

respectively, where $\alpha_2 = \alpha_2(k)$ and $\theta_2 = \theta_2(k)$, and the entries of $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$, $k = 1, \ldots, \min(m, n)$, are calculated from the coefficients of these polynomials. Similarly, since the entries of the subresultant matrices $S_k(f, g)$, $k = 1, \ldots, \min(m, n)$, are also functions of k, each of the subresultant matrices, $S_k(f, g)$, $k = 1, \ldots, \min(m, n)$, must be processed by the three preprocessing operations to obtain the subresultant

matrices defined in the modified Bernstein basis, $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$.

The degree of an AGCD of the inexact polynomials $f(x)$ and $g(x)$ can be determined from the subresultant matrices $\bar{S}_k(\tilde{f}_k, \alpha_1 \tilde{g}_k)Q_k$ and $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$, $k = 1, \ldots, \min(m, n)$, which is considered in the next section.

7.4 The determination of the degree of an AGCD

The preprocessing operations described in Section 7.3 transform the given inexact polynomials $f(x)$ and $g(x)$ to $\tilde{f}_k(w)$ and $\tilde{g}_k(w)$ defined in (7.34) and (7.35) respectively for $k = 1, \ldots, \min(m, n)$, and the kth subresultant matrix $\bar{S}_k(\tilde{f}_k, \alpha_1 \tilde{g}_k)Q_k$ is computed from the polynomials $\tilde{f}_k(w)$ and $\tilde{g}_k(w)$.

As shown in Section 7.2, when inexact polynomials are specified, noise that is added to the polynomials makes them coprime, and thus $\bar{S}_k(\tilde{f}_k, \alpha_1 \tilde{g}_k)Q_k$ has full column rank for all $k = 1, \ldots, \min(m, n)$. In order to determine the degree of an AGCD using the methods based on the first principal angle and residual, the approximation (7.10) is established.

It was discussed in Section 7.2 that it is necessary to choose the optimal column of $\bar{S}_k(\tilde{f}_k, \alpha_1 \tilde{g}_k)Q_k$ for $k = 1, \ldots, \min(m, n)$, such that the smallest error in the approximation (7.10) is achieved. In particular, the smallest error in the approximation (7.10) for each value of $k = 1, \ldots, \min(m, n)$, can be achieved by choosing the column of $\bar{S}_k(\tilde{f}_k, \alpha_1 \tilde{g}_k)Q_k$ as optimal column, such that the angle between this column and the space spanned by the remaining $m+n-2k+1$ columns of $\bar{S}_k(\tilde{f}_k, \alpha_1 \tilde{g}_k)Q_k$ is minimum, which implies that the smaller the angle, the smaller the error in the approximation (7.10). An alternative method considers the residual of the approximation (7.10) to calculate the optimal column for each value of k.

The discussion suggests that two issues must be addressed:

(1) The calculation of the index $i = q$ of the column of $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$ that defines the optimal column $h_{k,i}$ in (7.10) for $k = 1, \ldots, \min(m, n)$.

(2) The calculation of the degree $k = d$ of an AGCD of $\tilde{f}_k(w)$ and $\tilde{g}_k(w)$.

Two methods, one based on the first principal angle and the other based on the residual of (7.10) are used to solve this problem.

7.4.1 The method of the first principal angle

The first principal angle between the vector $h_{k,i}$ and the matrix $H_{k,i}$, which are defined in (7.10), is the smallest angle between the space $\mathcal{L}_{k,i}$ spanned by $h_{k,i}$, and the space $\mathcal{H}_{k,i}$ spanned by the columns of $H_{k,i}$,

$$\psi_{k,i} = \angle(\mathcal{L}_{k,i}, \mathcal{H}_{k,i}), \qquad k = 1, \ldots, \min(m, n); i = 1, \ldots, m + n - 2k + 2, \quad (7.38)$$

where

$$\begin{aligned} \mathcal{L}_{k,i} &= \text{span}\{\ h_{k,i}\ \}, \\ \mathcal{H}_{k,i} &= \text{span}\{\ h_{k,1}\ \ \cdots\ \ h_{k,i-1}\ \ h_{k,i+1}\ \ \cdots\ \ h_{k,m+n-2k+2}\ \}. \end{aligned}$$

The calculation of the degree of an AGCD using the criterion of the first principal angle firstly chooses the optimal column for each value $k = 1, \ldots, \min(m, n)$. Thus, the minimum value ϕ_k of $\psi_{k,i}$ for each value of k is calculated,

$$\phi_k = \min\left\{\psi_{k,i} : i = 1, \ldots, m + n - 2k + 2\right\}, \quad k = 1, \ldots, \min(m, n), \qquad (7.39)$$

and the column $i = q_k^\phi$ for which each of the $\min(m, n)$ minima occurs is recorded, thereby yielding the vector

$$q^\phi = \left[\ q_1^\phi\ \ q_2^\phi\ \ \cdots\ \ q_{\min(m,n)-1}^\phi\ \ q_{\min(m,n)}^\phi\right] \in \mathbb{R}^{\min(m,n)},$$

where the superscript ϕ denotes that these optimal column indices are calculated using the criterion based on the first principal angle.

It was stated in Section 7.2 that the degree d^ϕ of an AGCD is equal to the index k for which the change in ϕ_k between two successive values of k is maximum,

$$d^\phi = \left\{ k : \max\left(\frac{\phi_{k+1}}{\phi_k}\right); k = 1, \ldots, \min(m,n) - 1 \right\}. \tag{7.40}$$

Equation (7.40) is stated in terms of the maximum ratio of successive first principal angles, rather than the minimum value of the first principal angles. The reason for this criterion is easily seen by considering an example. In particular, let $\min(m,n) = 7$ and let $\phi \in \mathbb{R}^7$ be the vector of first principal angles ϕ_k, $k = 1, \ldots, 7$,

$$\begin{aligned} \phi &:= \begin{bmatrix} \phi_1 & \phi_2 & \phi_3 & \phi_4 & \phi_5 & \phi_6 & \phi_7 \end{bmatrix} \\ &= \begin{bmatrix} 2 \times 10^{-12} & 5 \times 10^{-13} & 4 \times 10^{-11} & 7 \times 10^{-12} & 3 \times 10^{-1} & 10^{-3} & 10^{-2} \end{bmatrix}, \end{aligned}$$

and thus

$$\log \phi = \begin{bmatrix} -11.7 & -12.3 & -10.4 & -11.2 & -0.5 & -3 & -2 \end{bmatrix}.$$

The variation of the first principal angles $\log \phi_1, \ldots, \log \phi_4$, is relatively minor, such that these four first principal angles are sufficiently small, which implies that the vector h_{k,q_k^ϕ} almost lies in the column space H_{k,q_k^ϕ}, and therefore the associated approximate solutions of (7.10) are acceptable. In particular, these small values show that the polynomials have approximate common divisors of degrees $1, 2, 3$ and 4. The maximum ratio $\phi_{k+1}/\phi_k, k = 1, \ldots, 6$, occurs for $k = 4$, which implies that the first principal angle between the vector h_{k,q_k^ϕ} and the column space H_{k,q_k^ϕ} is unacceptably large for $k = 5$, and thus the degree d^ϕ of an AGCD is equal to four.

Equation (7.40) defines the criterion to calculate the degree of an AGCD using the first principal angle, but the method to compute $\psi_{k,i}$, which is defined in (7.38), must

be obtained. The following considers the calculation of $\psi_{k,i}$ [56]:

The unit vector $u_{k,i}$ that spans $\mathcal{L}_{k,i}$ is

$$u_{k,i} = \frac{h_{k,i}}{\|h_{k,i}\|} \in \mathcal{L}_{k,i}, \qquad \dim \mathcal{L}_{k,i} = 1.$$

The calculation of $\psi_{k,i}$ requires an orthonormal basis for $\mathcal{H}_{k,i}$, which can be obtained by applying the QR decomposition to $H_{k,i}$,

$$H_{k,i} = O_{k,i} R_{k,i}, \qquad O_{k,i}^T O_{k,i} = I_{m+n-2k+1},$$

where $O_{k,i} \in \mathbb{R}^{(m+n-k+1)\times(m+n-2k+1)}$, $R_{k,i} \in \mathbb{R}^{(m+n-2k+1)\times(m+n-2k+1)}$ is an upper triangular matrix, and the columns of $O_{k,i}$ define an orthonormal basis for $\mathcal{H}_{k,i}$. Therefore, every vector $v_{k,i} \in \mathcal{H}_{k,i}$ can be written as

$$v_{k,i} = O_{k,i} z_{k,i},$$

where $z_{k,i} \in \mathbb{R}^{m+n-2k+1}$, and the cosine of the angle θ between $u_{k,i}$ and $v_{k,i}$ is

$$\cos\theta = u_{k,i}^T v_{k,i}, \qquad \|u_{k,i}\| = \|v_{k,i}\| = 1.$$

The first principal angle $\psi_{k,i}$ between $\mathcal{L}_{k,i}$ and $\mathcal{H}_{k,i}$ is defined to be the smallest angle between $u_{k,i} \in \mathcal{L}_{k,i}$ and an arbitrary vector $v_{k,i} \in \mathcal{H}_{k,i}$, and thus

$$\cos\psi_{k,i} = \max_{\|v_{k,i}\|=1} u_{k,i}^T v_{k,i} = \max_{\|z_{k,i}\|=1} (u_{k,i}^T O_{k,i}) z_{k,i}. \tag{7.41}$$

If the SVD of $u_{k,i}^T O_{k,i}$ is

$$u_{k,i}^T O_{k,i} = \Sigma_{k,i} W_{k,i}^T,$$

where $\Sigma_{k,i} \in \mathbb{R}^{1\times(m+n-2k+1)}$ and $W_{k,i} \in \mathbb{R}^{(m+n-2k+1)\times(m+n-2k+1)}$, then (7.41) yields

$$\cos\psi_{k,i} = \max_{\|v_{k,i}\|=1} u_{k,i}^T v_{k,i} = \max_{\|z_{k,i}\|=1} (\Sigma_{k,i} W_{k,i}^T) z_{k,i},$$

which implies that $\cos\psi_{k,i}$ is equal to the non-zero singular value of $u_{k,i}^T O_{k,i}$,

$$\cos\psi_{k,i} = \sigma_{k,i,1}. \tag{7.42}$$

It follows from (7.42) that the first principal angle, $\psi_{k,i}$, between $\mathcal{L}_{k,i}$ and $\mathcal{H}_{k,i}$ is given by

$$\psi_{k,i} = \cos^{-1} \sigma_{k,i,1}.$$

However, computational problems arise when $\psi_{k,i} \approx 0$ because

$$\delta\psi_{k,i} = -\frac{\delta\sigma_{k,i,1}}{\sin \psi_{k,i}}, \tag{7.43}$$

and thus $|\delta\psi_{k,i}| \gg |\delta\sigma_{k,i,1}|$ if $\psi_{k,i} \approx 0$. Therefore, a stable method for the first principal angle computation must be developed, which requires the following theorem [56].

Theorem 7.1. *Let the columns of $W \in \mathbb{R}^{r\times p}$ be orthonormal, and let W be partitioned as*

$$W = \begin{bmatrix} W_1 \\ W_2 \end{bmatrix}, \qquad W_1 \in \mathbb{R}^{r_1\times p}, \qquad W_2 \in \mathbb{R}^{r_2\times p}, \qquad r_1 + r_2 = r.$$

Let $\gamma_1 \geq \gamma_2 \geq \cdots \geq \gamma_p$ be the singular values of W_1, and let $\sigma_1 \leq \sigma_2 \leq \cdots \leq \sigma_p$ be the singular values of W_2, then

$$\gamma_j^2 + \sigma_j^2 = 1, \qquad j = 1, \ldots, p. \tag{7.44}$$

Proof. Since the columns of W are orthonormal, it follows that

$$W_1^T W_1 + W_2^T W_2 = I_p.$$

If (λ, v) is an eigenpair of $W_1^T W_1$, then

$$(W_1^T W_1)v = \lambda v,$$

and thus

$$(I_p - W_1^T W_1)v = (1 - \lambda)v,$$

that is

$$(W_2^T W_2)v = \mu v,$$

from which it follows that (μ, v) is an eigenpair of $W_2^T W_2$, where

$$\lambda + \mu = 1. \tag{7.45}$$

The jth eigenvalue of $W_1^T W_1$ is $\gamma_j^2, j = 1, \ldots, p$, and thus it follows from (7.45) that the jth eigenvalue of $W_2^T W_2$ is $1 - \gamma_j^2$. Since the jth eigenvalue of $W_2^T W_2$ is equal to σ_j^2, it follows that the sum of the jth eigenvalues of $W_1^T W_1$ and $W_2^T W_2$ is equal to one, and thus (7.44) is established. □

It will be shown that the instability that arises when $\psi_{k,i} \approx 0$ can be overcome by computing the orthogonal complements $\mathcal{L}_{k,i}^\perp$ and $\mathcal{H}_{k,i}^\perp$, where

$$\mathcal{L}_{k,i} \cup \mathcal{L}_{k,i}^\perp = \mathbb{R}^{m+n-k+1} \qquad \text{and} \qquad \mathcal{H}_{k,i} \cup \mathcal{H}_{k,i}^\perp = \mathbb{R}^{m+n-k+1},$$

and

$$\dim \mathcal{L}_{k,i}^\perp = m + n - k \qquad \text{and} \qquad \dim \mathcal{H}_{k,i}^\perp = k.$$

It will be required to calculate orthonormal bases for $\mathcal{L}_{k,i}^\perp$ and $\mathcal{H}_{k,i}^\perp$, and these bases will define the columns of matrices $\overline{U}_{k,i} \in \mathbb{R}^{(m+n-k+1)\times(m+n-k)}$ and $\overline{O}_{k,i} \in \mathbb{R}^{(m+n-k+1)\times k}$, respectively. It follows that the columns of $U_{k,i}$ and $N_{k,i}$ are given by

$$U_{k,i} = \begin{bmatrix} u_{k,i} & \overline{U}_{k,i} \end{bmatrix} \in \mathbb{R}^{(m+n-k+1)\times(m+n-k+1)}, \qquad U_{k,i}^T U_{k,i} = U_{k,i} U_{k,i}^T = I_{m+n-k+1}, \tag{7.46}$$

and

$$N_{k,i} = \begin{bmatrix} O_{k,i} & \overline{O}_{k,i} \end{bmatrix} \in \mathbb{R}^{(m+n-k+1)\times(m+n-k+1)}, \qquad N_{k,i}^T N_{k,i} = N_{k,i} N_{k,i}^T = I_{m+n-k+1}, \tag{7.47}$$

respectively, which define orthonormal bases for $\mathbb{R}^{m+n-k+1}$. The following theorem is established in [56].

Theorem 7.2. *Let $\mathcal{L}_{k,i}$ and $\mathcal{H}_{k,i}$ be subspaces of $\mathbb{R}^{m+n-k+1}$, and let θ_j be the jth*

principal angle between them. The unit vector $u_{k,i} \in \mathbb{R}^{m+n-k+1}$ *spans the line* $\mathcal{L}_{k,i}$*, and the columns of* $O_{k,i} \in \mathbb{R}^{(m+n-k+1)\times(m+n-2k+1)}$ *define an orthonormal basis for* $\mathcal{H}_{k,i}$*. Also, let the columns of* $\overline{U}_{k,i} \in \mathbb{R}^{(m+n-k+1)\times(m+n-k)}$ *and* $\overline{O}_{k,i} \in \mathbb{R}^{(m+n-k+1)\times k}$ *define orthonormal bases for* $\mathcal{L}_{k,i}^{\perp}$ *and* $\mathcal{H}_{k,i}^{\perp}$ *respectively, where (7.46) and (7.47) are satisfied. Then the singular values of* $\overline{U}_{k,i}^T O_{k,i} \in \mathbb{R}^{(m+n-k)\times(m+n-2k+1)}$ *and* $u_{k,i}^T \overline{O}_{k,i} \in \mathbb{R}^k$ *are*

$$\sin\theta_1 \leq \sin\theta_2 \leq \cdots \leq \sin\theta_{m+n-2k+1}.$$

Proof. Since $U_{k,i}$ is an orthogonal matrix and $O_{k,i}$ has orthonormal columns, the columns of $W_{k,i,1} \in \mathbb{R}^{(m+n-k+1)\times(m+n-2k+1)}$,

$$W_{k,i,1} = U_{k,i}^T O_{k,i} = \begin{bmatrix} u_{k,i}^T O_{k,i} \\ \overline{U}_{k,i}^T O_{k,i} \end{bmatrix}, \quad u_{k,i}^T O_{k,i} \in \mathbb{R}^{m+n-2k+1}, \quad \overline{U}_{k,i}^T O_{k,i} \in \mathbb{R}^{(m+n-k)\times(m+n-2k+1)},$$

are also orthonormal. Also, the singular values of $u_{k,i}^T O_{k,i}$ are $\gamma_{k,i,j} = \cos\theta_{k,i,j}, j = 1, \ldots, m+n-2k+1$, and it follows from Theorem 7.1 that the singular values of $\overline{U}_{k,i}^T O_{k,i}$ are

$$\sigma_{k,i,j} = \sqrt{1-\gamma_{k,i,j}^2} = \sin\theta_{k,i,j}, \qquad j = 1, \ldots, m+n-2k+1.$$

Consider now the vector $W_{k,i,2} \in \mathbb{R}^{m+n-k+1}$,

$$W_{k,i,2} = N_{k,i}^T u_{k,i} = \begin{bmatrix} O_{k,i}^T u_{k,i} \\ \overline{O}_{k,i}^T u_{k,i} \end{bmatrix}, \qquad O_{k,i}^T u_{k,i} \in \mathbb{R}^{m+n-2k+1}, \qquad \overline{O}_{k,i}^T u_{k,i} \in \mathbb{R}^k.$$

The singular values of $O_{k,i}^T u_{k,i}$ are $\cos\theta_{k,i,j}, j = 1, \ldots, m+n-2k+1$, and thus it follows from Theorem 7.1 that the singular values of $\overline{O}_{k,i}^T u_{k,i}$ and $u_{k,i}^T \overline{O}_{k,i}$ are $\sin\theta_{k,i,j}, j = 1, \ldots, m+n-2k+1$. □

Since the singular values of $u_{k,i}^T \overline{O}_{k,i}$ and $\overline{U}_{k,i}^T O_{k,i}$ are $\sigma_{k,i,j} = \sin\theta_{k,i,j}, j = 1, \ldots, m+n-2k+1$, it follows that the principal angles are

$$\theta_{k,i,j} = \sin^{-1}\sigma_{k,i,j}, \qquad j = 1, \ldots, m+n-2k+1,$$

and thus the first principal angle is given by

$$\psi_{k,i} = \sin^{-1} \sigma_{k,i,1}.$$

When $\psi_{k,i} \approx 0$, then,

$$\delta\psi_{k,i} = \frac{\delta\sigma_{k,i,1}}{\cos\psi_{k,i}},$$

from which it follows that if $\psi_{k,i} \approx 0$, then $|\delta\psi_{k,i}| \approx |\delta\sigma_{k,i,1}|$. The first principal angle $\psi_{k,i}$ is therefore stable with respect to changes in $\sigma_{k,i,1}$ when $\psi_{k,i} \approx 0$.

7.4.2 The method of the residual

An alternative method to calculate the optimal column is to consider the residual of (7.10). Let $z_{k,i}$ be the least squares solution of (7.10) and let $r_{k,i} = r_{k,i}(H_{k,i}, h_{k,i})$ be the residual associated with this solution,

$$z_{k,i} = H_{k,i}^{\dagger} h_{k,i}, \qquad r_{k,i} = h_{k,i} - H_{k,i} z_{k,i}, \qquad H_{k,i}^{\dagger} = (H_{k,i}^T H_{k,i})^{-1} H_{k,i}^T, \tag{7.48}$$

for $k = 1, \ldots, \min(m, n)$, and $i = 1, \ldots, m + n - 2k + 2$, where

$$\|h_{k,i}\|^2 = \|r_{k,i}\|^2 + \|H_{k,i} z_{k,i}\|^2, \qquad r_{k,i}^T (H_{k,i} z_{k,i}) = 0.$$

It follows that $\|r_{k,i}\|$ is equal to the perpendicular distance of the point with position vector $h_{k,i}$ to the point with position vector $H_{k,i} z_{k,i}$ on the plane $t = H_{k,i} x_{k,i}$ that defines the column space of $H_{k,i}$, which is shown in Figure 7.1.

The determination of the degree of an AGCD using the method based on the residual also includes two steps, which is similar to the determination of the degree of an AGCD using the method based on the first principal angle. Firstly, the minimum

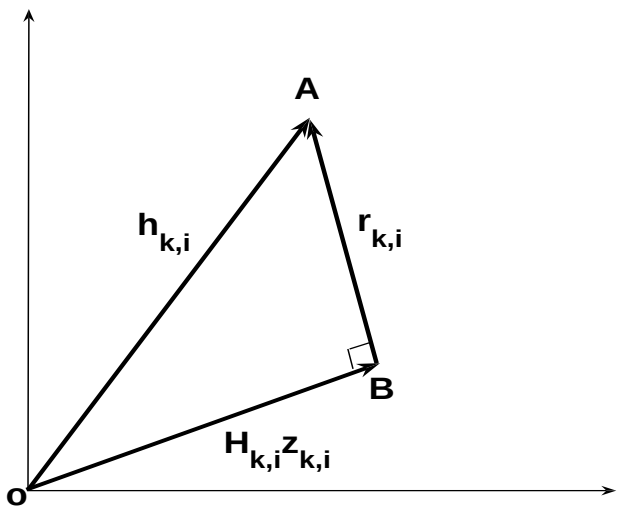


Figure 7.1: The first principal angle $\angle$ AOB and the residual $r_{k,i}$ between the vector $h_{k,i}$ and the column space $H_{k,i}$.

value of $\|r_{k,i}\|$ for each value of $k = 1, \ldots, \min(m, n)$, is calculated using (7.48),

$$\begin{aligned} r_k &= \min \frac{\|r_{k,i}\|}{\|h_{k,i}\|} \\ &= \min \left\{ \frac{\left|\left|\left(I - H_{k,i} H_{k,i}^{\dagger}\right) h_{k,i}\right|\right|}{||h_{k,i}||} : i = 1, \ldots, m + n - 2k + 2 \right\}, \end{aligned} \tag{7.49}$$

for $k = 1, \ldots, \min(m, n)$. The column $i = q_k^r$ for which each of the $\min(m, n)$ minima occurs is recorded, therefore yielding the vector

$$q^r = \left[\begin{array}{cccc} q_1^r & q_2^r & \cdots & q_{\min(m,n)-1}^r \end{array} \; q_{\min(m,n)}^r \right] \in \mathbb{R}^{\min(m,n)},$$

where the superscript r denotes that these optimal column indices are calculated using the criterion based on the residual. The degree d^r of an AGCD equals to the index k for which the change in r_k between two successive values of k is maximum,

$$d^r = \left\{ k : \max \left(\frac{r_{k+1}}{r_k} \right); k = 1, \ldots, \min(m, n) - 1 \right\}. \tag{7.50}$$

It is noted that (7.50) uses the maximum ratio of successive values of the residual, which is of the same form as (7.40) for the calculation of d^{ϕ}.

Algorithm 7.1: The calculation of the degree of an AGCD of two inexact Bernstein polynomials

Input Two inexact Bernstein polynomials $f(x)$ and $g(x)$ defined in (4.2).

Output Two estimates, d^{ϕ} and d^{r}, of the degree of an AGCD of $f(x)$ and $g(x)$, and the column indices q^{ϕ} and q^{r} associated with the first principal angle and residual respectively, for each value of $k = 1, \ldots, \min(m, n)$.

Begin

1. **For** $k = 1, \ldots, \min(m, n)$ % Loop for all the subresultant matrices

   1.1 Preprocess $f(x)$ and $g(x)$ to yield their modified Bernstein polynomials $\tilde{f}_k(w)$ and $\tilde{g}_k(w)$, which are defined in (7.34) and (7.35) respectively, as shown in Section 7.3.

   1.2 Compute the kth subresultant matrix, $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$, of $\tilde{f}_k(w)$ and $\tilde{g}_k(w)$.

   1.3 **For** $i = 1, \ldots, m+n-2k+2$ % Loop for the columns of $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$

   (i) Define the column $h_{k,i}$ from $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$.

   (ii) Define the matrix $H_{k,i}$ from $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$.

   (iii) Calculate the angle $\psi_{k,i}$ and residual $r_{k,i}$.

   End i

   1.4 Calculate ϕ_k and q_k^{ϕ} from (7.39), and r_k and q_k^{r} from (7.49).

End k

2. Calculate two estimates d^{ϕ} and d^{r} of the degree of an AGCD from (7.40) and (7.50).

End

The above analysis about the first principal angle and residual can be repeated for the subresultant matrices $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$, $k = 1, \ldots,\min(m, n)$, which are computed from the modified Bernstein polynomials $\acute{f}_k(w)$ and $\acute{g}_k(w)$ defined in (7.36) and (7.37) respectively. In particular, since the inexact Bernstein polynomials are coprime, $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ has full column rank for all $k = 1, \ldots, \min(m, n)$. Therefore, the approximation (7.9) is established.

Similarly, the optimal column for each index k must be calculated, such that the error in the approximation (7.9) is a minimum. Two criteria, the first principal angle and residual, are used to select the optimal columns for $k = 1, \ldots, \min(m, n)$, and then the degree of an AGCD is determined using the same procedures described in Sections 7.4.1 and 7.4.2 respectively.

7.5 Examples

In this section, three examples are illustrated to demonstrate the computation of the degree of an AGCD from three forms of the subresultant matrices, $S_k(\dot{f}_k, \dot{g}_k)$, $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$, using the methods based on the first principal angle and residual. These three forms of the subresultant matrices are described as follows:

- $S_k(\dot{f}_k, \dot{g}_k)$ is the kth subresultant matrix of the normalized Bernstein polynomials $\dot{f}_k(x)$ and $\dot{g}_k(x)$, which are defined in (7.16) and (7.17) respectively. The second and third preprocessing operations are not implemented, that is, $\alpha = \theta = 1$.

- $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$ is the kth subresultant matrix of the modified Bernstein polynomials $\acute{f}_k(w)$ and $\acute{g}_k(w)$, which are defined in (7.36) and (7.37) respectively, that arise after the three preprocessing operations have been implemented.

- $\bar{S}_k(\tilde{f}_k, \alpha_1 \tilde{g}_k)Q_k$ is the kth subresultant matrix of the modified Bernstein polynomials $\tilde{f}_k(w)$ and $\tilde{g}_k(w)$, which are defined in (7.34) and (7.35) respectively, that arise after the three preprocessing operations have been implemented.

Experiments show that the second and third preprocessing operations, which introduce the parameters α and θ, are important for the correct estimate of the degree of an AGCD. The importance of the second and third preprocessing operations can be easily recognized by observing the differences in the results obtained from these three forms of the subresultant matrices. In addition, it will be shown in the examples that both $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1 \tilde{g}_k)Q_k$ return good results. Furthermore, the examples will also demonstrate that angle and residual yield different optimal columns for some values of k.

Example 7.2. Consider the exact Bernstein polynomials $\hat{f}(x)$ and $\hat{g}(x)$, whose roots and multiplicities are specified in Table 7.1. It is seen that $m = 48$, $n = 47$ and the degree of their GCD is $\hat{d} = 37$.

Root of $\hat{f}(x)$	Multiplicity
0.2792e+000	11
0.3129e+000	4
0.7326e+000	6
0.7912e+000	9
-0.8139e+000	4
1.3741e+000	8
-3.3561e+000	6

Root of $\hat{g}(x)$	Multiplicity
0.2792e+000	10
0.7326e+000	7
0.7912e+000	11
0.9783e+000	6
1.3741e+000	6
-3.3561e+000	7

Table 7.1: The roots and multiplicities of $\hat{f}(x)$ and $\hat{g}(x)$ for Example 7.2.

Noise with componentwise signal-to-noise ratio 10^8 is added to each polynomial.

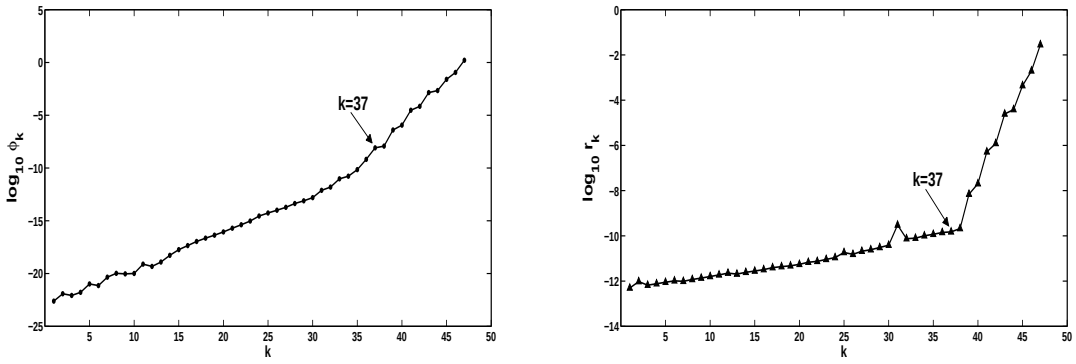


Figure 7.2: The variation of $\log_{10} \phi_k$ and $\log_{10} r_k$ computed from $S_k(\dot{f}_k, \dot{g}_k)$, $k = 1, \ldots, 47$, for Example 7.2.

Figure 7.2 shows the variation of $\log_{10} \phi_k$ and $\log_{10} r_k$ computed from the subresultant matrices, $S_k(\dot{f}_k, \dot{g}_k)$, $k = 1, \ldots, 47$. It is seen from Figure 7.2 that the maximum changes in $\log_{10} \phi_k$ and $\log_{10} r_k$ are not clearly defined.

Figures 7.3 and 7.4 show the variation of $\log_{10}\phi_k$ and $\log_{10} r_k$ computed from $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$, $k = 1, \ldots, 47$, respectively. It is seen from Figures 7.3 and 7.4 that the maximum gradient in each graph occurs when $k = 37$, which is correct because $\deg \mathrm{GCD}(\hat{f}, \hat{g}) = 37$.

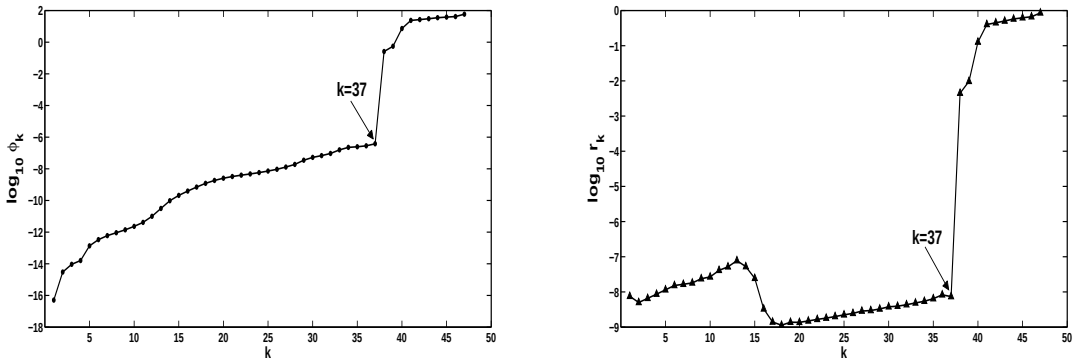


Figure 7.3: The variation of $\log_{10}\phi_k$ and $\log_{10} r_k$ computed from $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$, $k = 1, \ldots, 47$, for Example 7.2.

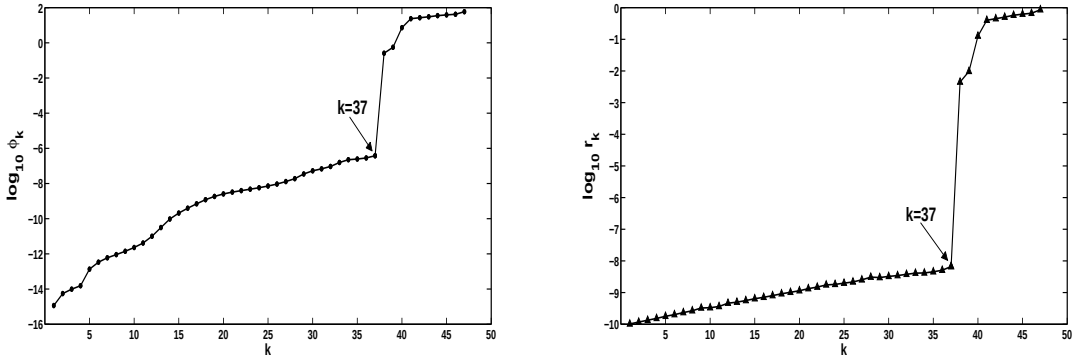


Figure 7.4: The variation of $\log_{10}\phi_k$ and $\log_{10} r_k$ computed from $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$, $k = 1, \ldots, 47$, for Example 7.2.

Comparing the result obtained from $S_k(\dot{f}_k, \dot{g}_k)$ with the results obtained from $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$ indicates that the second and third preprocessing

operations, which introduce the parameters α and θ, are important for yielding the correct estimate of the degree of an AGCD.

Figures 7.5 and 7.6 show the indices of the optimal columns of $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1 \tilde{g}_k)Q_k$, $k = 1, \ldots, 47$, respectively, using the criteria based on the first principal angle and residual. Both figures suggest that the criteria do not yield the same optimal column for all values of k.

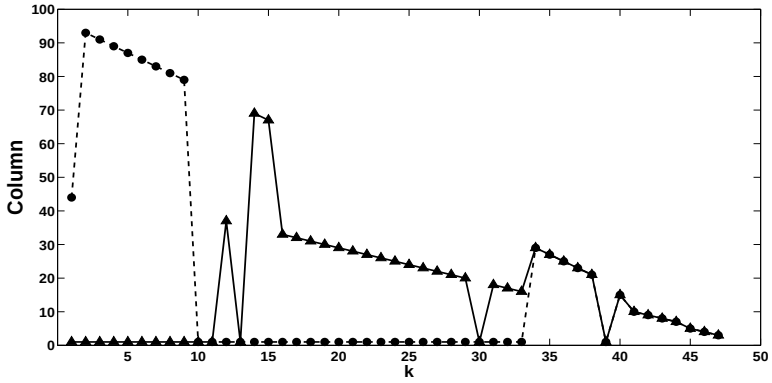


Figure 7.5: The column of $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$ for which the error in (7.9) is a minimum, using the first principal angle •, Method 1, and the residual ▲, Method 2, against k, for Example 7.2.

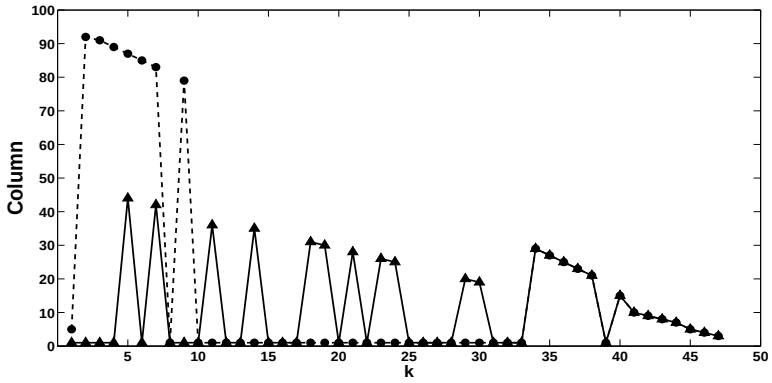


Figure 7.6: The column of $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$ for which the error in (7.10) is a minimum, using the first principal angle •, Method 1, and the residual ▲, Method 2, against k, for Example 7.2.

□

Example 7.3. Consider the exact Bernstein polynomials $\hat{f}(x)$ and $\hat{g}(x)$, whose roots and multiplicities are specified in Table 7.2. It is seen that $m = 29$, $n = 32$ and the degree of their GCD is $\hat{d} = 14$.

Root of $\hat{f}(x)$	Multiplicity
0.3569e+000	7
0.4521e+000	7
1.2383e+000	9
-1.3521e+000	6

Root of $\hat{g}(x)$	Multiplicity
0.8761e+000	9
0.9132e+000	9
1.2383e+000	8
-1.3521e+000	6

Table 7.2: The roots and multiplicities of $\hat{f}(x)$ and $\hat{g}(x)$ for Example 7.3.

Noise with componentwise signal-to-noise ratio 10^8 is added to the polynomials in order to yield their inexact forms.

Figure 7.7 shows the variation of $\log_{10}\phi_k$ and $\log_{10} r_k$ computed from $S_k(\dot{f}_k, \dot{g}_k)$, $k =$

$1, \ldots, 29$. It is seen from Figure 7.7 that the maximum change in $\log_{10} \phi_k$ occurs for $k = 8$, which is incorrect because $\deg \mathrm{GCD}(\hat{f}, \hat{g}) = 14$, and the maximum change in $\log_{10} r_k$ is not clearly defined, such that the degree of an AGCD can not be determined from it.

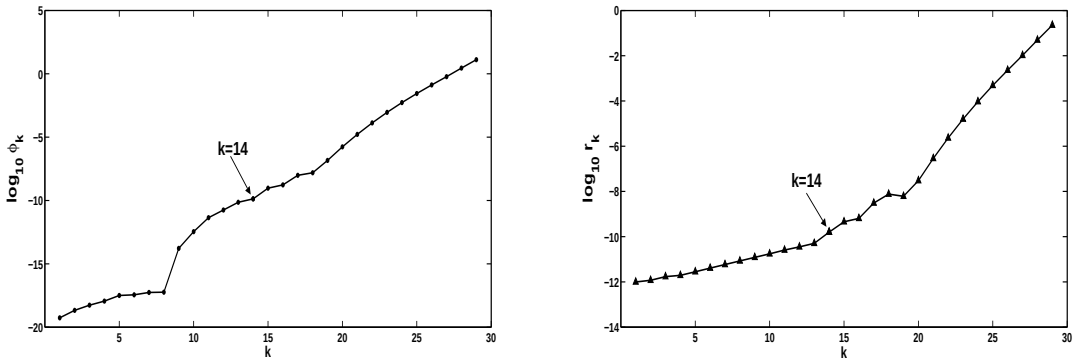


Figure 7.7: The variation of $\log_{10} \phi_k$ and $\log_{10} r_k$ computed from $S_k(\dot{f}_k, \dot{g}_k)$, $k = 1, \ldots, 29$, for Example 7.3.

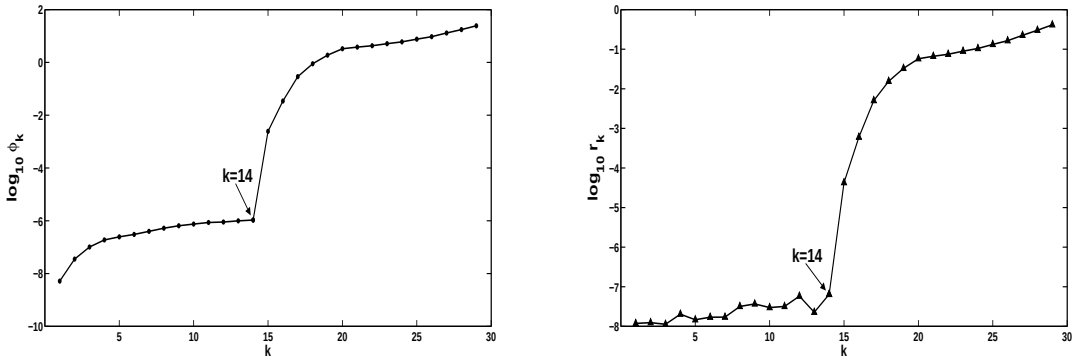


Figure 7.8: The variation of $\log_{10} \phi_k$ and $\log_{10} r_k$ computed from $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$, $k = 1, \ldots, 29$, for Example 7.3.

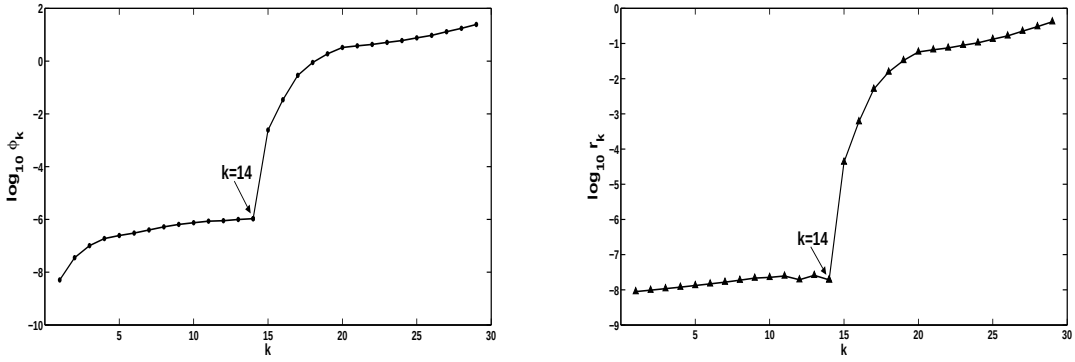


Figure 7.9: The variation of $\log_{10}\phi_k$ and $\log_{10} r_k$ computed from $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$, $k = 1, \ldots, 29$, for Example 7.3.

Figure 7.8 shows the variation of $\log_{10}\phi_k$ and $\log_{10} r_k$ computed from $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$, $k = 1, \ldots, 29$, and Figure 7.9 shows the variation of $\log_{10}\phi_k$ and $\log_{10} r_k$ computed from $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$, $k = 1, \ldots, 29$. It is seen from Figures 7.8 and 7.9 that the maximum gradient in each graph occurs for $k = 14$, which is correct, and that these values of k are clearly defined. The correct results shown in Figures 7.8 and 7.9 must be compared with the incorrect results shown in Figure 7.7. In particular, $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$, which are processed by three preprocessing operations, yield significantly better results than $S_k(\dot{f}_k, \dot{g}_k)$, which is only preprocessed by the normalization operation.

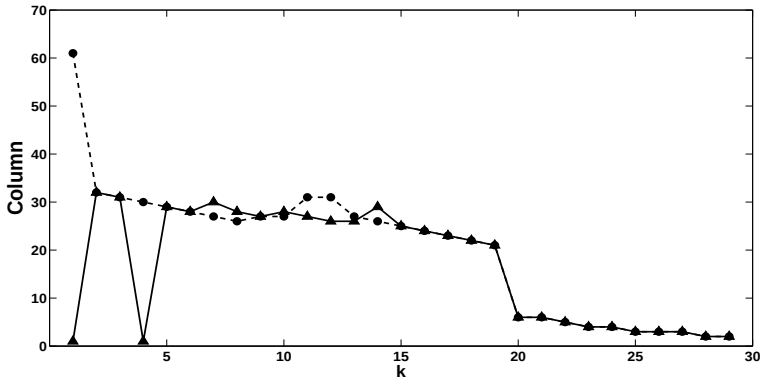


Figure 7.10: The column of $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$ for which the error in (7.9) is a minimum, using the first principal angle •, Method 1, and the residual ▲, Method 2, against k, for Example 7.3.

Figures 7.10 and 7.11 show the column of $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$ for which the error in (7.9) is minimum and the column of $\bar{S}_k(\tilde{f}_k, \alpha_1 \tilde{g}_k)Q_k$ for which the error in (7.10) is minimum, respectively, using the criteria based on the first principal angle and residual. It is seen from Figures 7.10 and 7.11 that the optimal column selected by the criterion based on the first principal angle is the same as the optimal column chosen by the criterion based on the residual for most values of k, and the greatest differences occur only for small values of k for both criteria.

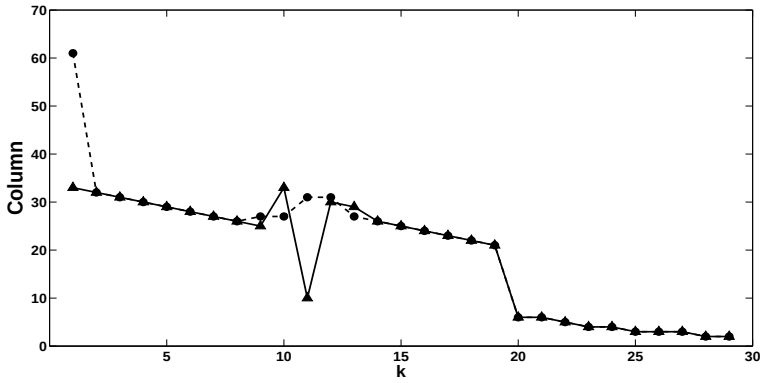


Figure 7.11: The column of $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$ for which the error in (7.10) is a minimum, using the first principal angle •, Method 1, and the residual ▲, Method 2, against k, for Example 7.3.

□

Example 7.4. Consider the exact Bernstein polynomials $\hat{f}(x)$ and $\hat{g}(x)$, whose roots and multiplicities are specified in Table 7.3. It is seen that $m = 27$, $n = 27$ and the degree of their GCD is $\hat{d} = 17$.

Root of $\hat{f}(x)$	Multiplicity
1.3679e-006	6
-2.4583e-005	4
3.6782e-007	5
7.1341e-006	7
-9.4731e-005	5

Root of $\hat{g}(x)$	Multiplicity
1.3679e-006	7
2.3684e-006	4
3.6782e-007	4
-5.7936e-006	5
7.1341e-006	7

Table 7.3: The roots and multiplicities of $\hat{f}(x)$ and $\hat{g}(x)$ for Example 7.4.

The polynomials are perturbed by noise, such that the componentwise signal-to-noise ratio equals to 10^8.

It is seen from Figures 7.13 and 7.14 that $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$ yield the

correct results because the maximum change in each graph occurs at $k = 17$, which is correct because $\deg \mathrm{GCD}(\hat{f}, \hat{g}) = 17$. However, $S_k(\dot{f}_k, \dot{g}_k)$ returns the incorrect results because Figure 7.12 shows that the maximum change in $\log_{10} \phi_k$ is not clearly defined and the maximum change in $\log_{10} r_k$ occurs for $k = 26$. In addition, Figures 7.15 and 7.16 demonstrate that two criteria based on the first principal angle and residual, yield different optimal columns for most values of k.

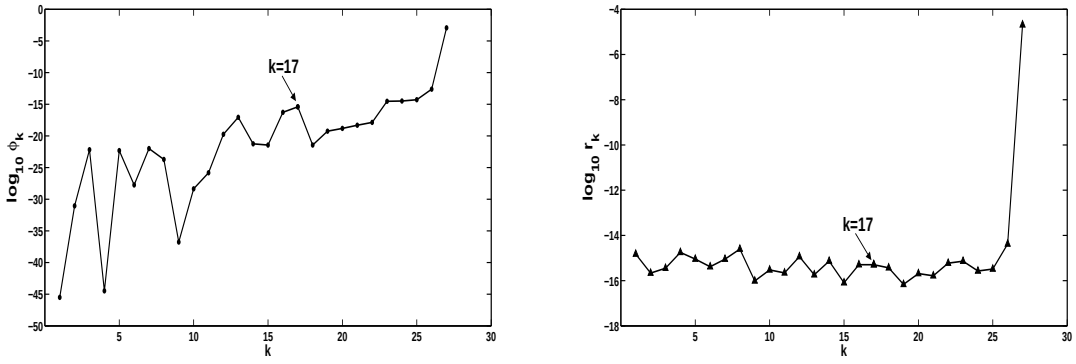


Figure 7.12: The variation of $\log_{10} \phi_k$ and $\log_{10} r_k$ computed from $S_k(\dot{f}_k, \dot{g}_k)$, $k = 1, \ldots, 27$, for Example 7.4.

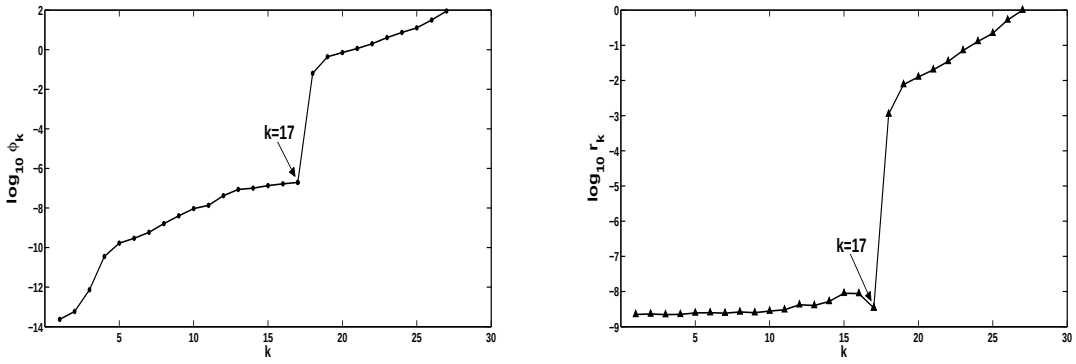


Figure 7.13: The variation of $\log_{10} \phi_k$ and $\log_{10} r_k$ computed from $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$, $k = 1, \ldots, 27$, for Example 7.4.

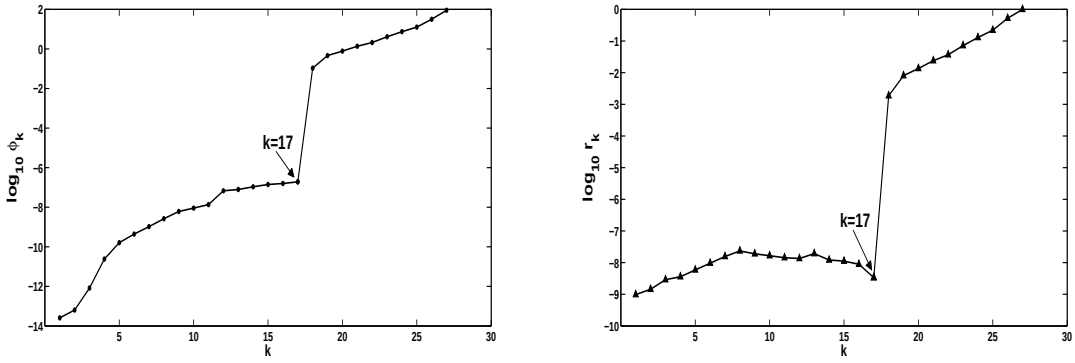


Figure 7.14: The variation of $\log_{10}\phi_k$ and $\log_{10} r_k$ computed from $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$, $k = 1, \ldots, 27$, for Example 7.4.

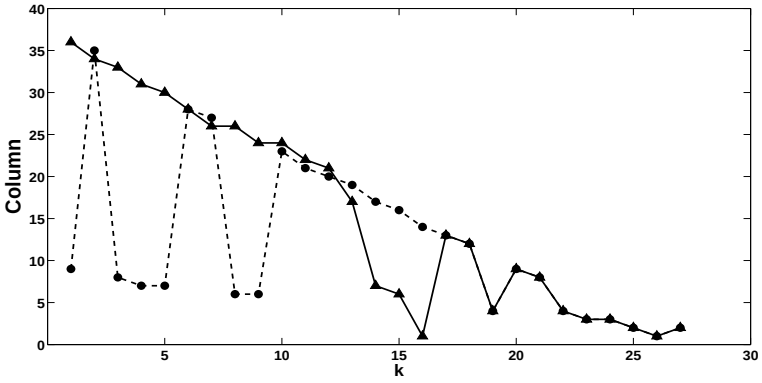


Figure 7.15: The column of $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ for which the error in (7.9) is a minimum, using the first principal angle •, Method 1, and the residual ▲, Method 2, against k, for Example 7.4.

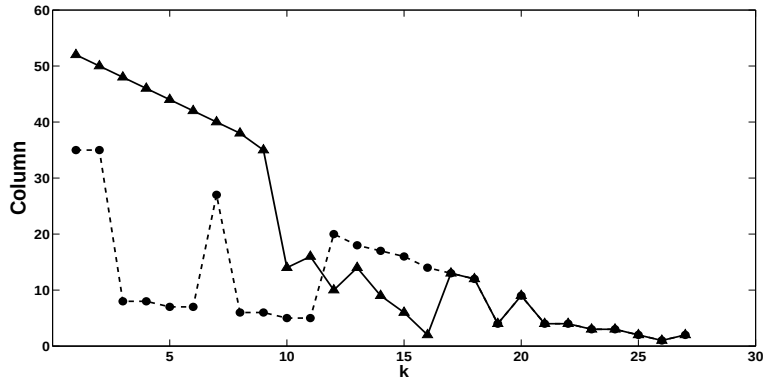


Figure 7.16: The column of $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$ for which the error in (7.10) is a minimum, using the first principal angle •, Method 1, and the residual ▲, Method 2, against k, for Example 7.4.

□

7.6 Discussion

It was shown in Chapter 6 that the inclusion of the diagonal matrix Q is important for the improvement of results because the Sylvester matrix $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ yields significantly better results than the Sylvester matrix $\bar{S}(\acute{f}, \alpha_2\acute{g})$. However, it was seen from the examples in Section 7.5 that when the methods using the first principal angle and residual are applied to the Sylvester subresultant matrices of $\bar{S}(\acute{f}, \alpha_2\acute{g})$ and $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$, $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$, both $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$ return similar correct results, and the improvement of results caused by the inclusion of the diagonal matrix Q_k is not obvious, which is explained as following.

Consider the computation of the degree of an AGCD using the method of the first principal angle applied to two forms of the Sylvester subresultant matrices,

$\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$.

Consider the subresultant matrices $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$, $k = 1, \ldots, \min(m, n)$. It was shown in Section 7.4.1 that the selection of the optimal column of $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ for $k = 1, \ldots, \min(m, n)$, requires the calculation of the first principal angle between $l_{k,i}$ and the column space of $L_{k,i}$ for $i = 1, \ldots, m + n - 2k + 2$, where $l_{k,i}$ is the ith column of $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ and $L_{k,i}$ is the remaining matrix of $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ after the removal of the ith column,

$$L_{k,i} = \begin{bmatrix} l_{k,1} & \cdots & l_{k,i-1} & l_{k,i+1} & \cdots & l_{k,m+n-2k+2} \end{bmatrix} \in \mathbb{R}^{(m+n-k+1)\times(m+n-2k+1)}.$$

The calculation of the first principal angle between $l_{k,i}$ and the column space of $L_{k,i}$ requires the unit vector of $l_{k,i}$ and an orthonormal basis for $L_{k,i}$ to be computed. The unit vector $p_{k,i}$ of $l_{k,i}$ is

$$p_{k,i} = \frac{l_{k,i}}{\|l_{k,i}\|},$$

and the orthonormal basis $O(L_{k,i})$ for $L_{k,i}$ is obtained by applying the QR decomposition to $L_{k,i}$, which involves the Gram-Schmidt process,

$$\begin{aligned}
v_1 &= l_{k,1}, \qquad t_1 = \frac{v_1}{\|v_1\|}, \\
v_2 &= l_{k,2} - \big(l_{k,2} \cdot t_1\big)t_1, \qquad t_2 = \frac{v_2}{\|v_2\|}, \\
&\vdots \\
v_{i-1} &= l_{k,i-1} - \sum_{j=1}^{i-2} \big(l_{k,i-1} \cdot t_j\big)t_j, \qquad t_{i-1} = \frac{v_{i-1}}{\|v_{i-1}\|}, \\
v_{i+1} &= l_{k,i+1} - \sum_{j=1}^{i} \big(l_{k,i+1} \cdot t_j\big)t_j, \qquad t_{i+1} = \frac{v_{i+1}}{\|v_{i+1}\|}, \\
&\vdots \\
v_{m+n-2k+2} &= l_{k,m+n-2k+2} - \sum_{j=1}^{m+n-2k+1} \big(l_{k,m+n-2k+2} \cdot t_j\big)t_j, \quad t_{m+n-2k+2} = \frac{v_{m+n-2k+2}}{\|v_{m+n-2k+2}\|},
\end{aligned}$$

where $r \cdot w$ denotes the inner product of the vectors r and w, and $\| \cdot \|$ denotes the 2-norm.

The orthonormal basis $O(L_{k,i})$ for $L_{k,i}$ is

$$O(L_{k,i}) = \begin{bmatrix} t_1 & \cdots & t_{i-1} & t_{i+1} & \cdots & t_{m+n-2k+2} \end{bmatrix} \in \mathbb{R}^{(m+n-k+1)\times(m+n-2k+1)}.$$

It was shown in Section 7.4.1 that the unit vector $p_{k,i}$ of $l_{k,i}$ and the orthonormal basis $O(L_{k,i})$ for $L_{k,i}$ are used for the calculation of the first principal angle between $l_{k,i}$ and the column space of $L_{k,i}$.

Consider the subresultant matrices $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$, $k = 1, \ldots, \min(m, n)$, which are equivalent to postmultiplying $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)$ by the diagonal matrix Q_k.

Suppose that the matrix $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k) \in \mathbb{R}^{(m+n-k+1)\times(m+n-2k+2)}$ is

$$\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k) = \begin{bmatrix} c_{k,1} & c_{k,2} & \cdots & \cdots & c_{k,m+n-2k+1} & c_{k,m+n-2k+2} \end{bmatrix}, \tag{7.51}$$

where $c_{k,i}$ is the ith column of $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)$.

It follows from (3.30) that the entries on the diagonal of Q_k are the combinatorial factors, and postmultiplying $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)$ by the diagonal matrix Q_k is equivalent to multiplying the ith column of $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)$, $c_{k,i}$, by the ith entry on the diagonal of Q_k, $q_{k,i}$, that is

$$\begin{aligned} \bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k &= \begin{bmatrix} h_{k,1} & h_{k,2} & \cdots & \cdots & h_{k,m+n-2k+2} \end{bmatrix} \\ &= \begin{bmatrix} q_{k,1}c_{k,1} & q_{k,2}c_{k,2} & \cdots & \cdots & q_{k,m+n-2k+2}c_{k,m+n-2k+2} \end{bmatrix}, \end{aligned} \tag{7.52}$$

where $q_{k,i}$ is a combinatorial factor, $c_{k,i}$ is a vector and $h_{k,i} = q_{k,i}c_{k,i}$.

The selection of the optimal column of $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$ for $k = 1, \ldots, \min(m, n)$, also requires the calculation of the first principal angle between $h_{k,i}$ and the column space of $H_{k,i}$ for $i = 1, \ldots, m + n - 2k + 2$, where $h_{k,i}$ is the ith column of $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$ and $H_{k,i} \in \mathbb{R}^{(m+n-k+1)\times(m+n-2k+1)}$ is the remaining matrix of $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$ after the

removal of the ith column,

$$\begin{aligned} H_{k,i} &= \begin{bmatrix} h_{k,1} & \cdots & h_{k,i-1} & h_{k,i+1} & \cdots & h_{k,m+n-2k+2} \end{bmatrix} \\ &= \begin{bmatrix} q_{k,1}c_{k,1} & \cdots & q_{k,i-1}c_{k,i-1} & q_{k,i+1}c_{k,i+1} & \cdots & q_{k,m+n-2k+2}c_{k,m+n-2k+2} \end{bmatrix}. \end{aligned}$$

The calculation of the first principal angle between $h_{k,i}$ and the column space of $H_{k,i}$ requires the unit vector of $h_{k,i}$ and an orthonormal basis for $H_{k,i}$ to be computed. The unit vector $u_{k,i}$ of $h_{k,i}$ is

$$u_{k,i} = \frac{h_{k,i}}{\|h_{k,i}\|} = \frac{q_{k,i}c_{k,i}}{\|q_{k,i}c_{k,i}\|} = \frac{q_{k,i}c_{k,i}}{q_{k,i}\|c_{k,i}\|} = \frac{c_{k,i}}{\|c_{k,i}\|},$$

and the orthonormal basis $O(H_{k,i})$ for $H_{k,i}$ is computed by applying the QR decomposition to $H_{k,i}$, which involves the Gram-Schmidt process,

$$\begin{aligned} r_1 &= h_{k,1} = q_{k,1}c_{k,1}, \\ e_1 &= \frac{r_1}{\|r_1\|} = \frac{q_{k,1}c_{k,1}}{\|q_{k,1}c_{k,1}\|} = \frac{c_{k,1}}{\|c_{k,1}\|}, \\ r_2 &= h_{k,2} - \big(h_{k,2}\cdot e_1\big)e_1 = q_{k,2}\Big(c_{k,2} - \big(c_{k,2}\cdot e_1\big)e_1\Big) = q_{k,2}w_{k,2}, \\ e_2 &= \frac{r_2}{\|r_2\|} = \frac{q_{k,2}w_{k,2}}{\|q_{k,2}w_{k,2}\|} = \frac{w_{k,2}}{\|w_{k,2}\|}, \\ &\vdots \\ r_{i-1} &= h_{k,i-1} - \sum_{j=1}^{i-2}\big(h_{k,i-1}\cdot e_j\big)e_j = q_{k,i-1}\Big(c_{k,i-1} - \sum_{j=1}^{i-2}\big(c_{k,i-1}\cdot e_j\big)e_j\Big) \\ &= q_{k,i-1}w_{k,i-1}, \\ e_{i-1} &= \frac{r_{i-1}}{\|r_{i-1}\|} = \frac{q_{k,i-1}w_{k,i-1}}{\|q_{k,i-1}w_{k,i-1}\|} = \frac{w_{k,i-1}}{\|w_{k,i-1}\|}, \\ r_{i+1} &= h_{k,i+1} - \sum_{j=1}^{i}\big(h_{k,i+1}\cdot e_j\big)e_j = q_{k,i+1}\Big(c_{k,i+1} - \sum_{j=1}^{i}\big(c_{k,i+1}\cdot e_j\big)e_j\Big) \\ &= q_{k,i+1}w_{k,i+1}, \\ e_{i+1} &= \frac{r_{i+1}}{\|r_{i+1}\|} = \frac{q_{k,i+1}w_{k,i+1}}{\|q_{k,i+1}w_{k,i+1}\|} = \frac{w_{k,i+1}}{\|w_{k,i+1}\|}, \\ &\vdots \end{aligned}$$

$$
\begin{aligned}
r_{m+n-2k+2} &= h_{k,m+n-2k+2} - \sum_{j=1}^{m+n-2k+1} \big(h_{k,m+n-2k+2} \cdot e_j\big) e_j \\
&= q_{k,m+n-2k+2}\Big(c_{k,m+n-2k+2} - \sum_{j=1}^{m+n-2k+1} \big(c_{k,m+n-2k+2} \cdot e_j\big) e_j\Big) \\
&= q_{k,m+n-2k+2} w_{k,m+n-2k+2}, \\
e_{m+n-2k+2} &= \frac{r_{m+n-2k+2}}{\|r_{m+n-2k+2}\|} = \frac{q_{k,m+n-2k+2} w_{k,m+n-2k+2}}{\|q_{k,m+n-2k+2} w_{k,m+n-2k+2}\|} = \frac{w_{k,m+n-2k+2}}{\|w_{k,m+n-2k+2}\|}.
\end{aligned}
$$

The orthonormal basis $O(H_{k,i})$ for $H_{k,i}$ is

$$
O(H_{k,i}) = \begin{bmatrix} e_1 & \cdots & e_{i-1} & e_{i+1} & \cdots & e_{m+n-2k+2} \end{bmatrix} \in \mathbb{R}^{(m+n-k+1)\times(m+n-2k+1)}.
$$

It is seen from the process of calculating the unit vector $u_{k,i}$ of $h_{k,i}$ and the orthonormal basis $O(H_{k,i})$ for $H_{k,i}$ that the effect of the ith entry on the diagonal of Q_k, $q_{k,i}$, is canceled out because the process involves normalizing the vector by its 2 norm. The unit vector $u_{k,i}$ and the orthonormal basis $O(H_{k,i})$ are used for the calculation of the first principal angle between $h_{k,i}$ and the column space of $H_{k,i}$, and therefore the diagonal matrix Q_k has no effect on the computation of the first principal angle.

Consider now the computation of the degree of an AGCD using the method of the residual applied to $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1 \tilde{g}_k)Q_k$.

When the subresultant matrices $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$, $k = 1, \ldots, \min(m, n)$, are used, it was shown in Section 7.4.2 that the selection of the optimal column of $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$ for $k = 1, \ldots, \min(m, n)$, requires the calculation of the residual $r_{k,i}(L_{k,i}, l_{k,i})$ for $i = 1, \ldots, m + n - 2k + 2$, where $l_{k,i}$ is the ith column of $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$ and $L_{k,i}$ is the remaining matrix of $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$ after the removal of the ith column.

It follows from (7.49) that

$$
\frac{\|r_{k,i}(L_{k,i}, l_{k,i})\|}{\|l_{k,i}\|} = \frac{\left|\left|\left(I - L_{k,i} L_{k,i}^{\dagger}\right) l_{k,i}\right|\right|}{||l_{k,i}||}.
$$

Consider the method of the residual applied to $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$. Similarly, the selection of the optimal column of $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$ for $k = 1, \ldots, \min(m, n)$, requires the calculation of the residual $r_{k,i}(H_{k,i}, h_{k,i})$ for $i = 1, \ldots, m + n - 2k + 2$, where $h_{k,i}$ is the ith column of $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$ and $H_{k,i}$ is the remaining matrix of $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$ after the removal of the ith column.

Suppose that $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)$ is defined in (7.51). The matrix $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$ is obtained by postmultiplying $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)$ by the diagonal matrix Q_k, which is equivalent to multiplying the ith column of $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)$, $c_{k,i}$, by the ith entry on the diagonal of Q_k, $q_{k,i}$. It therefore follows from (7.52) that

$$h_{k,i} = q_{k,i}c_{k,i}.$$

Furthermore, if $C_{k,i} \in \mathbb{R}^{(m+n-k+1)\times(m+n-2k+1)}$ is the remaining matrix of $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)$ after the removal of the ith column and $\bar{Q}_{k,i}$ is the remaining matrix of Q_k after removing the ith entry on the diagonal of Q_k,

$$\bar{Q}_{k,i} = \operatorname{diag}\begin{bmatrix} q_{k,1} & \cdots & q_{k,i-1} & q_{k,i+1} & \cdots & q_{k,m+n-2k+2} \end{bmatrix} \in \mathbb{R}^{(m+n-2k+1)\times(m+n-2k+1)},$$

where $q_{k,j}$ is the combinatorial factor, then

$$H_{k,i} = C_{k,i}\bar{Q}_{k,i}.$$

It follows from (7.49) that

$$\begin{aligned}
\frac{\|r_{k,i}(H_{k,i}, h_{k,i})\|}{\|h_{k,i}\|} &= \frac{\left|\left|\left(I - H_{k,i}H_{k,i}^{\dagger}\right)h_{k,i}\right|\right|}{||h_{k,i}||} = \frac{\left|\left|\left(I - C_{k,i}\bar{Q}_{k,i}\left(C_{k,i}\bar{Q}_{k,i}\right)^{\dagger}\right)q_{k,i}c_{k,i}\right|\right|}{||q_{k,i}c_{k,i}||} \\
&= \frac{q_{k,i}\left|\left|\left(I - C_{k,i}\bar{Q}_{k,i}\left(C_{k,i}\bar{Q}_{k,i}\right)^{\dagger}\right)c_{k,i}\right|\right|}{q_{k,i}\,||c_{k,i}||} \\
&= \frac{\left|\left|\left(I - C_{k,i}\bar{Q}_{k,i}\left(C_{k,i}\bar{Q}_{k,i}\right)^{\dagger}\right)c_{k,i}\right|\right|}{||c_{k,i}||}.
\end{aligned}$$

Since

$$\begin{aligned}\left(C_{k,i}\bar{Q}_{k,i}\right)^{\dagger} &= \left(\left(C_{k,i}\bar{Q}_{k,i}\right)^{T}\left(C_{k,i}\bar{Q}_{k,i}\right)\right)^{-1}\left(C_{k,i}\bar{Q}_{k,i}\right)^{T} \\ &= \left(\bar{Q}_{k,i}^{T}\left(C_{k,i}^{T}C_{k,i}\right)\bar{Q}_{k,i}\right)^{-1}\left(C_{k,i}\bar{Q}_{k,i}\right)^{T} \\ &= \left(\bar{Q}_{k,i}^{-1}\left(C_{k,i}^{T}C_{k,i}\right)^{-1}\bar{Q}_{k,i}^{-T}\right)\left(\bar{Q}_{k,i}^{T}C_{k,i}^{T}\right) \\ &= \bar{Q}_{k,i}^{-1}\left(C_{k,i}^{T}C_{k,i}\right)^{-1}C_{k,i}^{T},\end{aligned}$$

then

$$C_{k,i}\bar{Q}_{k,i}\left(C_{k,i}\bar{Q}_{k,i}\right)^{\dagger} = C_{k,i}\bar{Q}_{k,i}\bar{Q}_{k,i}^{-1}\left(C_{k,i}^{T}C_{k,i}\right)^{-1}C_{k,i}^{T} = C_{k,i}\left(C_{k,i}^{T}C_{k,i}\right)^{-1}C_{k,i}^{T} = C_{k,i}C_{k,i}^{\dagger}.$$

Thus

$$\begin{aligned}\frac{\|r_{k,i}(H_{k,i},h_{k,i})\|}{\|h_{k,i}\|} &= \frac{\left|\left|\left(I - C_{k,i}\bar{Q}_{k,i}\left(C_{k,i}\bar{Q}_{k,i}\right)^{\dagger}\right)c_{k,i}\right|\right|}{||c_{k,i}||} = \frac{\left|\left|\left(I - C_{k,i}C_{k,i}^{\dagger}\right)c_{k,i}\right|\right|}{||c_{k,i}||} \\ &= \frac{\|r_{k,i}(C_{k,i},c_{k,i})\|}{\|c_{k,i}\|}.\end{aligned}$$

This analysis shows that the normalized value of the 2-norm of the residual between the remaining matrix of $\bar{S}_k(\tilde{f}_k,\alpha_1\tilde{g}_k)Q_k$ after the removal of the ith column, $H_{k,i}$, and the ith column of $\bar{S}_k(\tilde{f}_k,\alpha_1\tilde{g}_k)Q_k$, $h_{k,i}$, is equal to the normalized value of the 2-norm of the residual between the remaining matrix of $\bar{S}_k(\tilde{f}_k,\alpha_1\tilde{g}_k)$ after the removal of the ith column, $C_{k,i}$, and the ith column of $\bar{S}_k(\tilde{f}_k,\alpha_1\tilde{g}_k)$, $c_{k,i}$, and therefore the diagonal matrix Q_k has no effect on the computation.

7.7 Summary

This chapter has introduced the computation of the degree of an AGCD of inexact polynomials using two methods, one based on the first principal angle and the other based on the residual of a linear algebraic equation. The computation is performed on the subresultant matrices $\bar{S}_k(\acute{f}_k,\alpha_2\acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k,\alpha_1\tilde{g}_k)Q_k$, $k = 1,\ldots,\min(m,n)$,

which are processed by three preprocessing operations. Experiments show that both methods yield correct estimates of the degree of an AGCD, and the preprocessing operations are crucial for the improvement of results. Furthermore, different criteria used to define the error in the approximate linear algebraic equation (7.9) or (7.10) may select different optimal columns.

Chapters 5, 6 and 7 present three methods to determine the degree of an AGCD of inexact polynomials, and it is desirable to compare these three methods to determine the method yielding the best results. This issue is discussed in the next chapter.

Chapter 8

The comparison of three methods

Chapters 5, 6 and 7 present three methods to determine the degree of an AGCD of inexact polynomials. It is shown in Chapter 5 that the Bézout resultant matrix defined in the modified Bernstein basis, $\bar{B}\left(\check{f}, \check{g}\right)$ yields better results than the Bézout resultant matrix defined in the Bernstein basis, $B(f, g)$. Chapter 6 considers two forms of the Sylvester resultant matrix defined in the modified Bernstein basis, $\bar{S}(\acute{f}, \alpha_2 \acute{g})$ and $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$. The comparison of the results obtained from $\bar{S}(\acute{f}, \alpha_2 \acute{g})$ and $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ suggests that $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$ yields better results. In Chapter 7, the methods based on the first principal angle and residual are implemented on two forms of the Sylvester subresultant matrices defined in the modified Bernstein basis, $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1 \tilde{g}_k)Q_k$. Experiments show that both $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1 \tilde{g}_k)Q_k$ return good results.

In the following examples, the results obtained from $\bar{B}\left(\check{f}, \check{g}\right)$, $\bar{S}(\tilde{f}, \alpha_1 \tilde{g})Q$, $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1 \tilde{g}_k)Q_k$ are compared to determine the method yielding the best results.

8.1 Examples

In this section, the results obtained with $\bar{B}\left(\check{f},\check{g}\right)$, $\bar{S}(\tilde{f},\alpha_1\tilde{g})Q$, $\bar{S}_k(\acute{f}_k,\alpha_2\acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k,\alpha_1\tilde{g}_k)Q_k$ are shown and it is therefore instructive to review their definitions:

- $\bar{B}\left(\check{f},\check{g}\right)$ is the Bézout matrix of the modified Bernstein polynomials $\check{f}(w)$ and $\check{g}(w)$, which are defined in (5.13) and (5.14) respectively, that arise after the preprocessing operation shown in Section 5.1 has been implemented.

- $\bar{S}(\tilde{f},\alpha_1\tilde{g})Q$ is the Sylvester matrix of the modified Bernstein polynomials $\tilde{f}(w)$ and $\tilde{g}(w)$, which are defined in (6.27) and (6.28) respectively, that arise after the three preprocessing operations shown in Section 6.1 have been implemented.

- $\bar{S}_k(\acute{f}_k,\alpha_2\acute{g}_k)$ is the kth subresultant matrix of the modified Bernstein polynomials $\acute{f}_k(w)$ and $\acute{g}_k(w)$, which are defined in (7.36) and (7.37) respectively, that arise after the three preprocessing operations shown in Section 7.3 have been implemented.

- $\bar{S}_k(\tilde{f}_k,\alpha_1\tilde{g}_k)Q_k$ is the kth subresultant matrix of the modified Bernstein polynomials $\tilde{f}_k(w)$ and $\tilde{g}_k(w)$, which are defined in (7.34) and (7.35) respectively, that arise after the three preprocessing operations shown in Section 7.3 have been implemented.

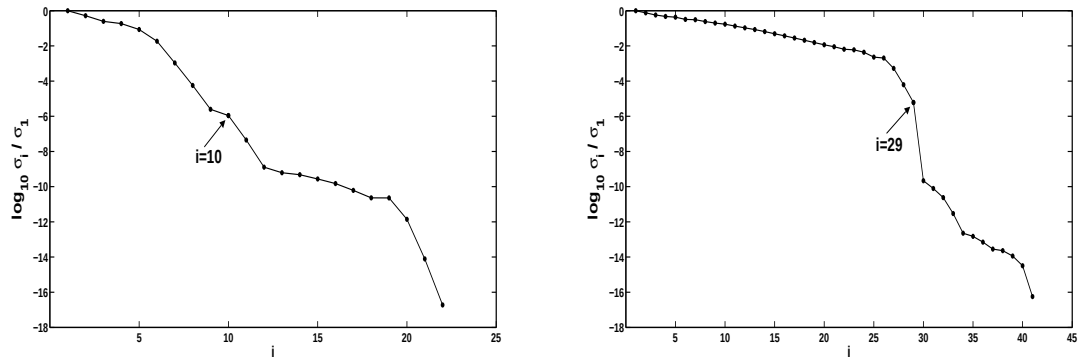
Example 8.1. Consider the exact Bernstein polynomials $\hat{f}(x)$ and $\hat{g}(x)$, whose roots and multiplicities are specified in Table 8.1. It is seen that $m = 22$, $n = 19$ and the degree of their GCD is $\hat{d} = 12$.

Root of $\hat{f}(x)$	Multiplicity
1.3974e-005	5
2.9147e-006	4
7.1963e-006	8
-8.8579e-005	5

Root of $\hat{g}(x)$	Multiplicity
1.3974e-005	4
1.9867e-007	6
2.9147e-006	3
-8.8579e-005	6

Table 8.1: The roots and multiplicities of $\hat{f}(x)$ and $\hat{g}(x)$ for Example 8.1.

Noise with componentwise signal-to-noise ratio 10^8 is added to each polynomial.



(a) (b)

Figure 8.1: The normalized singular values of (a) $\bar{B}\left(\check{f},\check{g}\right)$ and (b) $\bar{S}(\tilde{f},\alpha_1\tilde{g})Q$ for Example 8.1.

The normalized singular values of $\bar{B}\left(\check{f},\check{g}\right)$ and $\bar{S}(\tilde{f},\alpha_1\tilde{g})Q$ are shown in Figures 8.1(a) and (b) respectively. It is seen from Figure 8.1(a) that the Bézout matrix $\bar{B}\left(\check{f},\check{g}\right)$ is of full rank, which implies that $\hat{f}(x)$ and $\hat{g}(x)$ are coprime. The result in Figure 8.1(a) was obtained with $\theta_0 = 1.6029e-005$. The rank of the Sylvester matrix $\bar{S}(\tilde{f},\alpha_1\tilde{g})Q$ is, however, clearly defined and equal to 29, which is correct because $\deg \mathrm{GCD}(\hat{f},\hat{g}) = 12$. The result in Figure 8.1(b) was obtained with $\alpha_1 = 1.9065e-007$ and $\theta_1 = 7.0564e-006$.

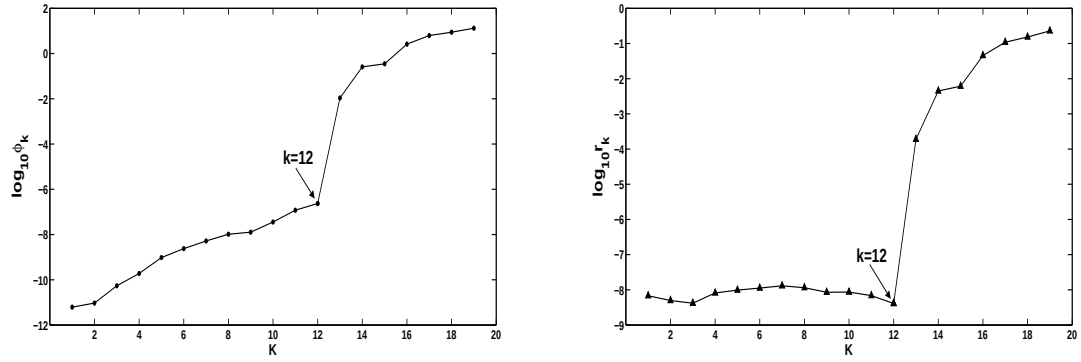


Figure 8.2: The variation of $\log_{10}\phi_k$ and $\log_{10} r_k$ computed from $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$, $k = 1, \ldots, 19$, for Example 8.1.

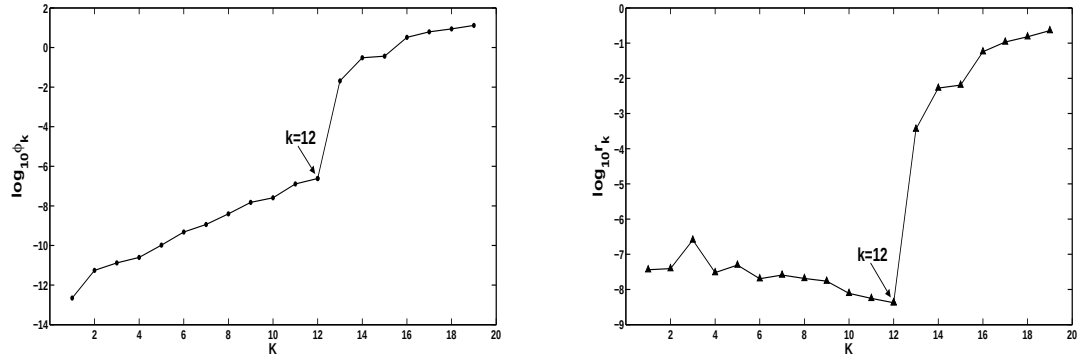


Figure 8.3: The variation of $\log_{10}\phi_k$ and $\log_{10} r_k$ computed from $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$, $k = 1, \ldots, 19$, for Example 8.1.

Figures 8.2 and 8.3 show the variation of $\log_{10}\phi_k$ and $\log_{10} r_k$ computed from $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$, $k = 1, \ldots, 19$, respectively. It is seen from Figures 8.2 and 8.3 that the maximum gradient in each graph occurs when $k = 12$, which is correct because $\deg \mathrm{GCD}(\hat{f}, \hat{g}) = 12$. □

Example 8.2. Consider the exact Bernstein polynomials $\hat{f}(x)$ and $\hat{g}(x)$, whose roots and multiplicities are specified in Table 8.2. It is seen that $m = 38$, $n = 31$ and the

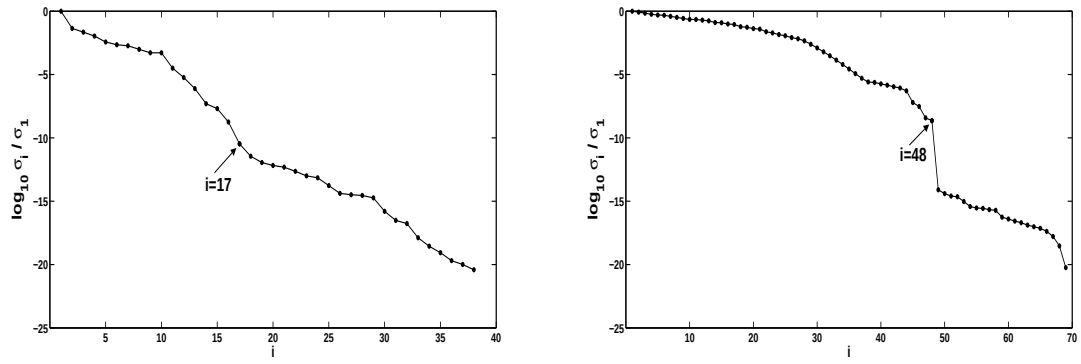
degree of their GCD is $\hat{d} = 21$.

Root of $\hat{f}(x)$	Multiplicity
0.1278e+000	6
0.2374e+000	8
-0.5679e+000	6
0.7937e+000	5
1.7359e+000	9
-2.1455e+000	4

Root of $\hat{g}(x)$	Multiplicity
0.1278e+000	5
0.2374e+000	7
-0.5679e+000	5
0.9949e+000	6
-2.1455e+000	5
-3.4998e+000	3

Table 8.2: The roots and multiplicities of $\hat{f}(x)$ and $\hat{g}(x)$ for Example 8.2.

Noise with componentwise signal-to-noise ratio 10^8 is added to each polynomial.



(a) (b)

Figure 8.4: The normalized singular values of (a) $\bar{B}\left(\check{f}, \check{g}\right)$ and (b) $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ for Example 8.2.

Figure 8.4(a) shows the normalized singular values of $\bar{B}\left(\check{f}, \check{g}\right)$, and it is seen that the Bézout matrix $\bar{B}\left(\check{f}, \check{g}\right)$ has full rank, which suggests that $\hat{f}(x)$ and $\hat{g}(x)$ are coprime. The result in Figure 8.4(a) was obtained with $\theta_0 = 0.7102$. Figure 8.4(b) shows the normalized singular values of $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$, and its rank is equal to 48, which

is correct, and furthermore, it is clearly defined. The result in Figure 8.4(b) was obtained with $\alpha_1 = 29.3094$ and $\theta_1 = 1.07$.

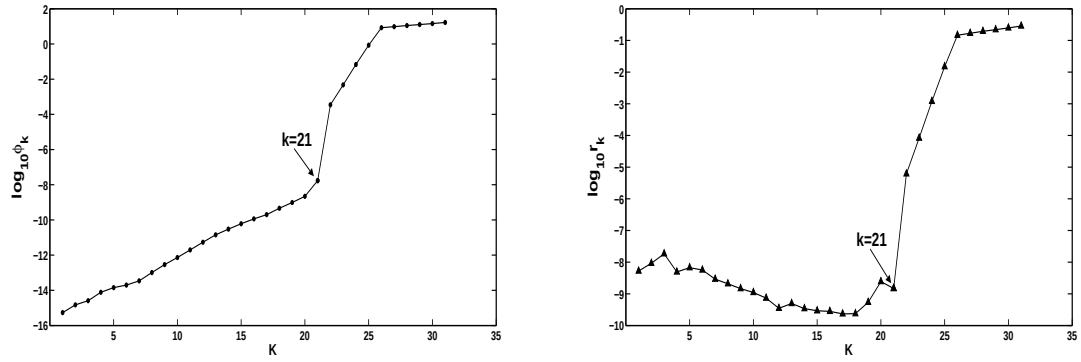


Figure 8.5: The variation of $\log_{10}\phi_k$ and $\log_{10} r_k$ computed from $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$, $k = 1, \ldots, 31$, for Example 8.2.

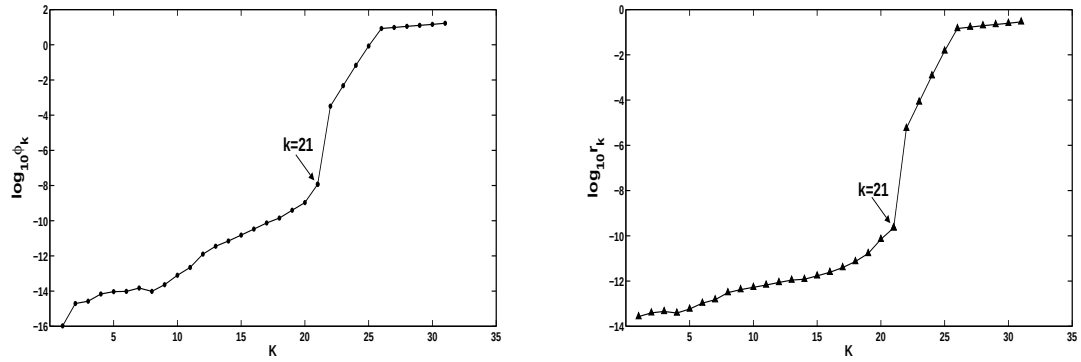


Figure 8.6: The variation of $\log_{10}\phi_k$ and $\log_{10} r_k$ computed from $\bar{S}_k(\tilde{f}_k, \alpha_1 \tilde{g}_k)Q_k$, $k = 1, \ldots, 31$, for Example 8.2.

Figure 8.5 shows the variation of $\log_{10}\phi_k$ and $\log_{10} r_k$ computed from $\bar{S}_k(\acute{f}_k, \alpha_2 \acute{g}_k)$, $k = 1, \ldots, 31$, and Figure 8.6 shows the variation of $\log_{10}\phi_k$ and $\log_{10} r_k$ computed from $\bar{S}_k(\tilde{f}_k, \alpha_1 \tilde{g}_k)Q_k$, $k = 1, \ldots, 31$. It is seen from Figures 8.5 and 8.6 that the

maximum gradient in each graph occurs for $k = 21$, which is correct, and that these values of k are clearly defined. □

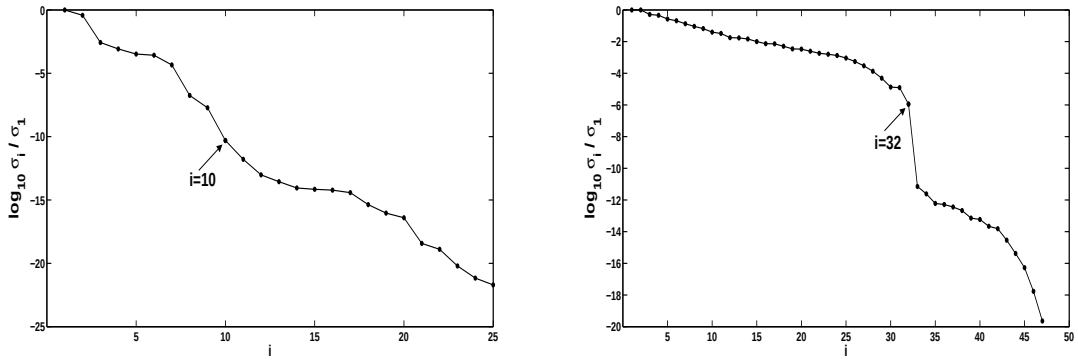
Example 8.3. Consider the exact Bernstein polynomials $\hat{f}(x)$ and $\hat{g}(x)$, whose roots and multiplicities are specified in Table 8.3. It is seen that $m = 22$, $n = 25$ and the degree of their GCD is $\hat{d} = 15$.

Root of $\hat{f}(x)$	Multiplicity
0.3473e+000	4
0.5961e+000	6
1.4793e+000	3
-2.6893e+000	4
3.7913e+000	5

Root of $\hat{g}(x)$	Multiplicity
-0.1124e+000	4
0.5961e+000	7
-1.1794e+000	3
-2.6893e+000	5
3.7913e+000	6

Table 8.3: The roots and multiplicities of $\hat{f}(x)$ and $\hat{g}(x)$ for Example 8.3.

Noise with componentwise signal-to-noise ratio 10^8 is added to each polynomial.



(a) (b)

Figure 8.7: The normalized singular values of (a) $\bar{B}\left(\check{f}, \check{g}\right)$ and (b) $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ for Example 8.3.

The normalized singular values of $\bar{B}\left(\check{f}, \check{g}\right)$ and $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ are shown in Figure

8.7, and similarly the matrices $\bar{B}\left(\check{f},\check{g}\right)$ and $\bar{S}(\tilde{f},\alpha_1\tilde{g})Q$ yield, respectively, incorrect and correct results because $\bar{B}\left(\check{f},\check{g}\right)$ has full rank and the rank of $\bar{S}(\tilde{f},\alpha_1\tilde{g})Q$ is equal to 32. The result in Figure 8.7(a) was obtained with $\theta_0 = 1.0621$, and the result in Figure 8.7(b) was obtained with $\alpha_1 = 8.2488$ and $\theta_1 = 0.9098$.

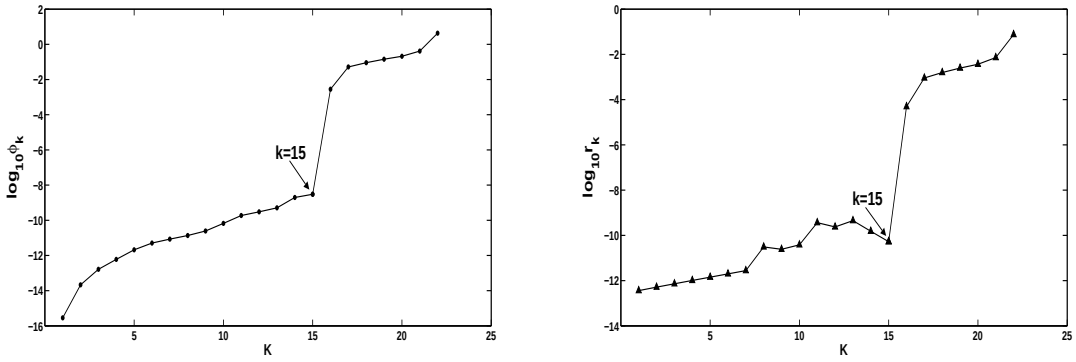


Figure 8.8: The variation of $\log_{10}\phi_k$ and $\log_{10} r_k$ computed from $\bar{S}_k(\acute{f}_k,\alpha_2\acute{g}_k)$, $k = 1,\ldots,22$, for Example 8.3.

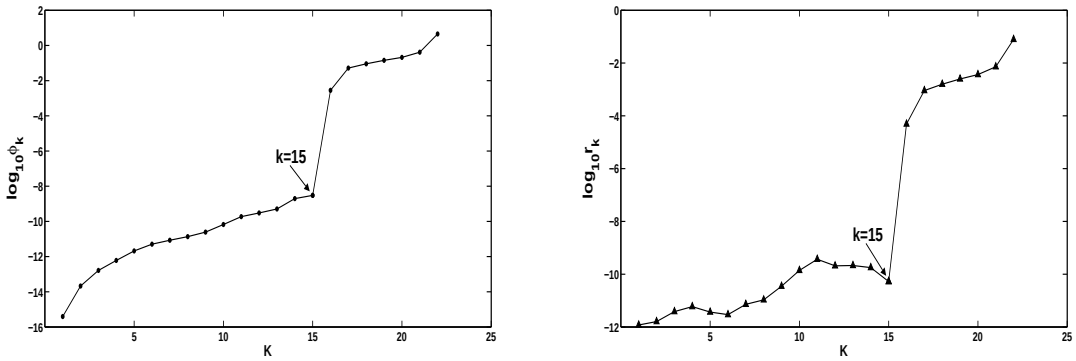


Figure 8.9: The variation of $\log_{10}\phi_k$ and $\log_{10} r_k$ computed from $\bar{S}_k(\tilde{f}_k,\alpha_1\tilde{g}_k)Q_k$, $k = 1,\ldots,22$, for Example 8.3.

Figures 8.8 and 8.9 show the variation of $\log_{10}\phi_k$ and $\log_{10} r_k$ computed from

$\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$, $k = 1, \ldots, 22$, respectively. It is seen from Figures 8.8 and 8.9 that $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$ yield the correct results because the maximum change in each graph occurs at $k = 15$, which is correct because $\deg \mathrm{GCD}(\hat{f}, \hat{g}) = 15$. □

The comparison between the result obtained from $\bar{B}\left(\check{f}, \check{g}\right)$ and the results obtained from $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$, $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$, indicates that the result obtained from $\bar{B}\left(\check{f}, \check{g}\right)$ is inferior to the results obtained from $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$, $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$. This result may therefore confirm the remark by Bini and Marco [6] that the additions required for the computation of the entries of the Bézout matrix may cause numerical cancellation in a floating point environment. This was also investigated by considering the situation that occurs when noise is not added, and the Bézout matrix $\bar{B}\left(\check{f}, \check{g}\right)$ returned the correct numerical rank in most, but not all, examples, but the Sylvester matrix $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ returned the correct numerical rank in all examples. This shows that $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ is numerically superior to $\bar{B}\left(\check{f}, \check{g}\right)$, and it is therefore expected that the result obtained with $\bar{B}\left(\check{f}, \check{g}\right)$ deteriorates when noise is added to the coefficients of the polynomials, as shown in the examples.

It is shown that both the Sylvester resultant matrix $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ and two forms of the Sylvester subresultant matrices, $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$, yield correct estimate of the degree d of an AGCD, and therefore the definition of d used in this thesis, which is stated in Section 4.4, is practical because there exist methods for which this definition of d can be realized. Furthermore, the knowledge of the noise level is not required for these methods.

8.2 Summary

This chapter compared three methods to determine the degree of an AGCD of inexact polynomials, and it was shown that the Sylvester matrix and its subresultant matrices yield better results than the Bézout matrix.

The determination of the degree d of an AGCD of inexact polynomials has been considered, and it is therefore desirable to consider the computation of the coefficients of an AGCD. In particular, the perturbations added to the coefficients of inexact polynomials are calculated, such that the perturbed forms of inexact polynomials possess a non-constant common divisor of degree d. This topic is discussed in the next chapter.

Chapter 9

The coefficients of an AGCD

As stated in Chapter 4, the calculation of an AGCD of inexact polynomials involves two steps: The degree of an AGCD is determined initially, after which the coefficients of an AGCD are computed. The determination of the degree of an AGCD has been covered in Chapters 5, 6 and 7, and the computation of the coefficients of an AGCD is discussed in this chapter.

It is assumed that the degree d of an AGCD $d(x)$ of two inexact polynomials $f(x)$ and $g(x)$, which are defined in (4.2), is determined using the methods described in Chapters 5, 6 and 7. There therefore exist quotient polynomials $u(x)$ and $v(x)$, such that

$$f(x) \approx d(x)u(x) \qquad \text{and} \qquad g(x) \approx d(x)v(x).$$

Since

$$\frac{f(x)}{u(x)} \approx \frac{g(x)}{v(x)},$$

we obtain

$$f(x)v(x) - g(x)u(x) \approx 0,$$

which can be written in matrix form,

$$S_d(f,g)\begin{bmatrix}\mathbf{v}\\ -\mathbf{u}\end{bmatrix}\approx 0, \tag{9.1}$$

where $S_d(f,g)$ is the dth subresultant matrix of $f(x)$ and $g(x)$, which is defined in (3.18), and **v, u** are the scaled coefficients vectors of $v(x)$ and $u(x)$ respectively.

It was shown in Chapter 7 that the approximate homogeneous equation (9.1) can be converted to an approximate linear algebraic equation by moving the optimal column of $S_d(f,g)$, $b_{d,q}$, to the right hand side of (9.1),

$$A_{d,q}x \approx b_{d,q}, \tag{9.2}$$

where $A_{d,q} \in \mathbb{R}^{(m+n-d+1)\times(m+n-2d+1)}$ is the remaining matrix of $S_d(f,g)$ after the removal of its qth column, $b_{d,q} \in \mathbb{R}^{m+n-d+1}$, and

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_{q-1} \\ x_{q+1} \\ \vdots \\ x_{m+n-2d+2} \end{bmatrix} \in \mathbb{R}^{m+n-2d+1}, \qquad \begin{bmatrix}\mathbf{v}\\ -\mathbf{u}\end{bmatrix} = \begin{bmatrix} x_1 \\ \vdots \\ x_{q-1} \\ -1 \\ x_{q+1} \\ \vdots \\ x_{m+n-2d+2} \end{bmatrix} \in \mathbb{R}^{m+n-2d+2}.$$

The approximation (9.2) must be corrected to induce an exact solution. The structure preserving method is used here, which adds a matrix F that has the same structure as $A_{d,q}$ to $A_{d,q}$ and adds a vector c to $b_{d,q}$ respectively, such that

$$\big(A_{d,q}+F\big)x = \big(b_{d,q}+c\big).$$

The perturbation matrix F and vector c are calculated using the method of structured nonlinear total least norm (SNTLN) [41], which will be considered in the next section.

9.1 The method of SNTLN

This section considers the computation of the coefficients of an AGCD using the method of SNTLN. Since the determination of the degree of an AGCD of two inexact polynomials $f(x)$ and $g(x)$ has been introduced in the previous chapters, this section assumes that the degree d of an AGCD is known. Chapter 7 shows that two forms of the subresultant matrices defined in the modified Bernstein basis, $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ and $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$, which are processed by all three preprocessing operations described in Section 7.3 respectively, yield significantly better results than the Sylvester subresultant matrices defined in the Bernstein basis, $S_k(f, g)$. Therefore, it is desirable to consider the method of SNTLN implemented on these two forms of the subresultant matrices. The matrix $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$ has a more complex form than $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$, and therefore this section only describes the method of SNTLN for $\bar{S}_k(\tilde{f}_k, \alpha_1\tilde{g}_k)Q_k$, since the method of SNTLN for $\bar{S}_k(\acute{f}_k, \alpha_2\acute{g}_k)$ can be easily obtained from it.

It is assumed that the inexact Bernstein polynomials $f(x)$ and $g(x)$, which are defined in (4.2), are coprime, and the degree d of an AGCD of $f(x)$ and $g(x)$ is known. The three preprocessing operations described in Section 7.3 transform $f(x)$ and $g(x)$ to the modified Bernstein polynomials $\tilde{f}_d(w)$ and $\alpha_1\tilde{g}_d(w)$, which are defined in (7.34) and (7.35) respectively. Since $\tilde{f}_d(w)$ and $\alpha_1\tilde{g}_d(w)$ have an AGCD $\tilde{d}_d(w)$ of degree d, there exist quotient polynomials $\tilde{u}_d(w)$ and $\tilde{v}_d(w)$, such that

$$\tilde{f}_d(w) \approx \tilde{d}_d(w)\tilde{u}_d(w) \qquad \text{and} \qquad \alpha_1\tilde{g}_d(w) \approx \tilde{d}_d(w)\tilde{v}_d(w),$$

and $\tilde{u}_d(w)$, $\tilde{v}_d(w)$ and $\tilde{d}_d(w)$ are defined as

$$\tilde{u}_d(w) = \sum_{i=0}^{m-d} \left(\bar{u}_{d,i}\theta_1^i\right) \binom{m-d}{i} (1-\theta_1 w)^{m-d-i} w^i,$$

$$\tilde{v}_d(w) = \sum_{i=0}^{n-d} \left(\bar{v}_{d,i}\theta_1^i\right) \binom{n-d}{i} (1-\theta_1 w)^{n-d-i} w^i,$$

and

$$\tilde{d}_d(w) = \sum_{i=0}^{d} \left(\bar{d}_{d,i}\theta_1^i\right) \binom{d}{i} (1-\theta_1 w)^{d-i} w^i,$$

respectively, where $\alpha_1 = \alpha_1(d)$ and $\theta_1 = \theta_1(d)$ are the solutions of the minimization problem (7.33).

Since

$$\frac{\tilde{f}_d(w)}{\tilde{u}_d(w)} \approx \frac{\alpha_1 \tilde{g}_d(w)}{\tilde{v}_d(w)},$$

then

$$\tilde{f}_d(w)\tilde{v}_d(w) - \alpha_1 \tilde{g}_d(w)\tilde{u}_d(w) \approx 0,$$

which can be written in matrix form,

$$\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d) Q_d \begin{bmatrix} \tilde{\mathbf{v}}_d(\theta_1) \\ -\tilde{\mathbf{u}}_d(\theta_1) \end{bmatrix} \approx 0, \tag{9.3}$$

where $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$ is the dth subresultant matrix of $\tilde{f}_d(w)$ and $\alpha_1\tilde{g}_d(w)$, which is defined in (7.29), and $\tilde{\mathbf{v}}_d(\theta_1)$ and $\tilde{\mathbf{u}}_d(\theta_1)$ are the coefficients vectors of $\tilde{v}_d(w)$ and $\tilde{u}_d(w)$ respectively.

Likewise, it is assumed that $h_{d,q}$, the qth column of $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$, is the optimal column, which is chosen by the criterion based on the first principal angle or the residual, and $H_{d,q} \in \mathbb{R}^{(m+n-d+1)\times(m+n-2d+1)}$ is the matrix formed after the removal of

the optimal column $h_{d,q}$ from $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$, that is

$$H_{d,q} = \begin{bmatrix} h_{d,1} & \cdots & h_{d,q-1} & h_{d,q+1} & \cdots & h_{d,m+n-2d+2} \end{bmatrix}.$$

Moving the qth column of $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$ to the right hand side of (9.3) therefore yields the approximation

$$H_{d,q}x \approx h_{d,q}, \tag{9.4}$$

where

$$x = \begin{bmatrix} x_1 & \cdots & x_{q-1} & x_{q+1} & \cdots & x_{m+n-2d+2} \end{bmatrix}^T \in \mathbb{R}^{m+n-2d+1},$$

and

$$\begin{bmatrix} \tilde{\mathbf{v}}_d(\theta_1) \\ -\tilde{\mathbf{u}}_d(\theta_1) \end{bmatrix} = \begin{bmatrix} \bar{v}_{d,0} \\ \vdots \\ \bar{v}_{d,n-d}\theta_1^{n-d} \\ -\bar{u}_{d,0} \\ \vdots \\ -\bar{u}_{d,m-d}\theta_1^{m-d} \end{bmatrix} = \begin{bmatrix} x_1 \\ \vdots \\ x_{q-1} \\ -1 \\ x_{q+1} \\ \vdots \\ x_{m+n-2d+2} \end{bmatrix} \in \mathbb{R}^{m+n-2d+2}.$$

It was shown in Section 7.2 that the operation of removing the qth column from $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$ is achieved by postmultiplying $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$ by $M_{d,q}$, which is equal to the identity matrix after the removal of the qth column,

$$M_{d,q} = \begin{bmatrix} e_{d,1} & e_{d,2} & \cdots & e_{d,q-1} & e_{d,q+1} & \cdots & e_{d,m+n-2d+1} & e_{d,m+n-2d+2} \end{bmatrix},$$

where $M_{d,q} \in \mathbb{R}^{(m+n-2d+2)\times(m+n-2d+1)}$, $q = 1, \ldots, m+n-2d+2$, and $e_{d,q} \in \mathbb{R}^{m+n-2d+2}$ is the qth unit basis vector. Since the qth column of $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$ is equal to $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d e_{d,q}$, it follows that

$$H_{d,q} = \bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d M_{d,q} \qquad \text{and} \qquad h_{d,q} = \bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d e_{d,q},$$

and thus (9.4) is rewritten as

$$\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d M_{d,q}x \approx \bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d e_{d,q}. \tag{9.5}$$

In the method of SNTLN, $\alpha_1 = \alpha_1(d)$ and $\theta_1 = \theta_1(d)$, which are the solutions of the minimization problem (7.33), are the initial values of α and θ respectively. The values of α and θ are then refined in each iteration for the calculation of the corrected forms of $\tilde{f}_d(w)$ and $\tilde{g}_d(w)$ using α_1 and θ_1 as the initial values in the iterative refinement procedure, such that these corrected forms have a non-constant common divisor. Therefore, the constants α_1 and θ_1 in $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$ are replaced by the parameters α and θ. It follows from (7.29) that the Sylvester subresultant matrix $\bar{S}_d(\tilde{f}_d, \alpha\tilde{g}_d)Q_d$ of $\tilde{f}_d(w)$ and $\alpha\tilde{g}_d(w)$ is given by

$$\bar{S}_d(\tilde{f}_d, \alpha\tilde{g}_d)Q_d = D_d^{-1}U_d(\tilde{f}_d, \alpha\tilde{g}_d)Q_d,$$

where $U_d(\tilde{f}_d, \alpha\tilde{g}_d) \in \mathbb{R}^{(m+n-d+1)\times(m+n-2d+2)}$ is equal to

$$\begin{bmatrix}
\bar{a}_{d,0}\binom{m}{0} & & & \alpha\bar{b}_{d,0}\binom{n}{0} & & \\
\bar{a}_{d,1}\binom{m}{1}\theta & \ddots & & \alpha\bar{b}_{d,1}\binom{n}{1}\theta & \ddots & \\
\vdots & \ddots & \bar{a}_{d,0}\binom{m}{0} & \vdots & \ddots & \alpha\bar{b}_{d,0}\binom{n}{0} \\
\bar{a}_{d,m-1}\binom{m}{m-1}\theta^{m-1} & \ddots & \bar{a}_{d,1}\binom{m}{1}\theta & \alpha\bar{b}_{d,n-1}\binom{n}{n-1}\theta^{n-1} & \ddots & \alpha\bar{b}_{d,1}\binom{n}{1}\theta \\
\bar{a}_{d,m}\binom{m}{m}\theta^{m} & \ddots & \vdots & \alpha\bar{b}_{d,n}\binom{n}{n}\theta^{n} & \ddots & \vdots \\
 & \ddots & \bar{a}_{d,m-1}\binom{m}{m-1}\theta^{m-1} & & \ddots & \alpha\bar{b}_{d,n-1}\binom{n}{n-1}\theta^{n-1} \\
 & & \bar{a}_{d,m}\binom{m}{m}\theta^{m} & & & \alpha\bar{b}_{d,n}\binom{n}{n}\theta^{n}
\end{bmatrix},$$

$\bar{a}_{d,i}\binom{m}{i}\theta^i$, $i = 0, \ldots, m$, and $\alpha\bar{b}_{d,j}\binom{n}{j}\theta^j$, $j = 0, \ldots, n$, are the scaled coefficients of $\tilde{f}_d(w)$ and $\alpha\tilde{g}_d(w)$, which are defined in (7.34) and (7.35) respectively, and the matrices D_d^{-1} and Q_d are defined in (3.14) and (3.30) respectively. Then, (9.5) is written as

$$\left(D_d^{-1}U_d(\tilde{f}_d, \alpha\tilde{g}_d)Q_d\right) M_{d,q}x \approx \left(D_d^{-1}U_d(\tilde{f}_d, \alpha\tilde{g}_d)Q_d\right) e_{d,q}. \tag{9.6}$$

The inexact polynomials $\tilde{f}_d(w)$ and $\tilde{g}_d(w)$ are perturbed in order to induce a non-constant common divisor in their perturbed forms. If the perturbations of the coefficients of $\tilde{f}_d(w)$ and $\alpha\tilde{g}_d(w)$ are

$$z_i\theta^i, i = 0, \ldots, m \qquad \text{and} \qquad \alpha z_{m+1+j}\theta^j, j = 0, \ldots, n,$$

respectively, then $B_d = B_d(\alpha, \theta, z) \in \mathbb{R}^{(m+n-d+1)\times(m+n-2d+2)}$, the dth subresultant matrix of the perturbations, is

$$B_d = D_d^{-1} F_d Q_d,$$

where D_d^{-1} is defined in (3.14) and $F_d = F_d(\alpha, \theta, z) \in \mathbb{R}^{(m+n-d+1)\times(m+n-2d+2)}$ is equal to

$$\begin{bmatrix}
z_0\binom{m}{0} & & & \alpha z_{m+1}\binom{n}{0} & & \\
z_1\binom{m}{1}\theta & \ddots & & \alpha z_{m+2}\binom{n}{1}\theta & \ddots & \\
\vdots & \ddots & z_0\binom{m}{0} & \vdots & \ddots & \alpha z_{m+1}\binom{n}{0} \\
z_{m-1}\binom{m}{m-1}\theta^{m-1} & \ddots & z_1\binom{m}{1}\theta & \alpha z_{m+n}\binom{n}{n-1}\theta^{n-1} & \ddots & \alpha z_{m+2}\binom{n}{1}\theta \\
z_m\binom{m}{m}\theta^m & \ddots & \vdots & \alpha z_{m+n+1}\binom{n}{n}\theta^n & \ddots & \vdots \\
 & \ddots & z_{m-1}\binom{m}{m-1}\theta^{m-1} & & \ddots & \alpha z_{m+n}\binom{n}{n-1}\theta^{n-1} \\
 & & z_m\binom{m}{m}\theta^m & & & \alpha z_{m+n+1}\binom{n}{n}\theta^n
\end{bmatrix}, \tag{9.7}$$

and Q_d is defined in (3.30).

If $G_{d,q} = G_{d,q}(\alpha, \theta, z) \in \mathbb{R}^{(m+n-d+1)\times(m+n-2d+1)}$ is the matrix that results when the qth column $g_{d,q} \in \mathbb{R}^{m+n-d+1}$ of B_d is removed, then it follows from the definitions of $M_{d,q}$ and $e_{d,q}$ that

$$G_{d,q} = B_d M_{d,q} = D_d^{-1} F_d Q_d M_{d,q} \qquad \text{and} \qquad g_{d,q} = B_d e_{d,q} = D_d^{-1} F_d Q_d e_{d,q},$$

and thus (9.6) becomes

$$D_d^{-1}(U_d + F_d)Q_d M_{d,q} x = D_d^{-1}(U_d + F_d)Q_d e_{d,q}. \tag{9.8}$$

Since this equation is solved for α, θ, z and x, it is desirable to change the notation slightly. Thus, (9.8) is written as

$$D_d^{-1}\big(U_d(\alpha,\theta) + F_d(\alpha,\theta,z)\big)Q_d M_{d,q} x = c_d(\alpha,\theta) + h_d(\alpha,\theta,z), \tag{9.9}$$

where

$$c_d(\alpha,\theta) = D_d^{-1}U_d(\alpha,\theta)Q_d e_{d,q} \qquad \text{and} \qquad h_d(\alpha,\theta,z) = D_d^{-1}F_d(\alpha,\theta,z)Q_d e_{d,q}.$$

It is noted that depending on the column q, c_d and h_d may or may not be functions of α:

$$\begin{array}{lcll} c_d & = & c_d(\theta) & \text{if} \quad 1 \le q \le n-d+1 \\ c_d & = & c_d(\alpha,\theta) & \text{if} \quad n-d+2 \le q \le m+n-2d+2 \\ h_d & = & h_d(\theta,z) & \text{if} \quad 1 \le q \le n-d+1 \\ h_d & = & h_d(\alpha,\theta,z) & \text{if} \quad n-d+2 \le q \le m+n-2d+2. \end{array}$$

If $1 \le q \le n-d+1$, c_d and h_d have no dependence on α, and if $n-d+2 \le q \le m+n-2d+2$, c_d and h_d are functions of α. The following theory assumes that $n-d+2 \le q \le m+n-2d+2$.

Equation (9.9) is a non-linear equation that is solved by the Newton-Raphson method. In general, it has an infinite number of solutions, but the solution that is nearest the given inexact data is sought. The residual associated with an approximate solution of (9.9) is

$$r(\alpha,\theta,x,z) = c_d(\alpha,\theta) + h_d(\alpha,\theta,z) - D_d^{-1}(U_d(\alpha,\theta) + F_d(\alpha,\theta,z))Q_d M_{d,q} x, \tag{9.10}$$

and thus if $\tilde{r}$ is defined as

$$\tilde{r} := r(\alpha + \delta\alpha, \theta + \delta\theta, x + \delta x, z + \delta z),$$

then

$$\begin{aligned}
\tilde{r} &= c_d(\alpha+\delta\alpha,\theta+\delta\theta)+h_d(\alpha+\delta\alpha,\theta+\delta\theta,z+\delta z)\\
&\quad -D_d^{-1}(U_d(\alpha+\delta\alpha,\theta+\delta\theta)+F_d(\alpha+\delta\alpha,\theta+\delta\theta,z+\delta z))Q_dM_{d,q}(x+\delta x)\\
&= c_d+\frac{\partial c_d}{\partial\alpha}\delta\alpha+\frac{\partial c_d}{\partial\theta}\delta\theta+h_d+\frac{\partial h_d}{\partial\alpha}\delta\alpha+\frac{\partial h_d}{\partial\theta}\delta\theta+\sum_{i=0}^{m+n+1}\frac{\partial h_d}{\partial z_i}\delta z_i\\
&\quad -D_d^{-1}U_dQ_dM_{d,q}x-D_d^{-1}U_dQ_dM_{d,q}\delta x-\left(D_d^{-1}\frac{\partial U_d}{\partial\alpha}Q_dM_{d,q}x\right)\delta\alpha\\
&\quad -\left(D_d^{-1}\frac{\partial U_d}{\partial\theta}Q_dM_{d,q}x\right)\delta\theta-D_d^{-1}F_dQ_dM_{d,q}x-D_d^{-1}F_dQ_dM_{d,q}\delta x\\
&\quad -\left(D_d^{-1}\frac{\partial F_d}{\partial\alpha}Q_dM_{d,q}x\right)\delta\alpha-\left(D_d^{-1}\frac{\partial F_d}{\partial\theta}Q_dM_{d,q}x\right)\delta\theta\\
&\quad -D_d^{-1}\left(\sum_{i=0}^{m+n+1}\frac{\partial F_d}{\partial z_i}\delta z_i\right)Q_dM_{d,q}x,
\end{aligned}$$

to first order. It follows that

$$\begin{aligned}
\tilde{r} &= r(\alpha,\theta,x,z)-\left(D_d^{-1}\left(\frac{\partial U_d}{\partial\theta}+\frac{\partial F_d}{\partial\theta}\right)Q_dM_{d,q}x-\left(\frac{\partial c_d}{\partial\theta}+\frac{\partial h_d}{\partial\theta}\right)\right)\delta\theta\\
&\quad -\left(D_d^{-1}\left(\frac{\partial U_d}{\partial\alpha}+\frac{\partial F_d}{\partial\alpha}\right)Q_dM_{d,q}x-\left(\frac{\partial c_d}{\partial\alpha}+\frac{\partial h_d}{\partial\alpha}\right)\right)\delta\alpha-D_d^{-1}(U_d+F_d)Q_dM_{d,q}\delta x\\
&\quad +\sum_{i=0}^{m+n+1}\frac{\partial h_d}{\partial z_i}\delta z_i-D_d^{-1}\left(\sum_{i=0}^{m+n+1}\frac{\partial F_d}{\partial z_i}\delta z_i\right)Q_dM_{d,q}x. \qquad (9.11)
\end{aligned}$$

Example 9.1. If $q=n-d+3>n-d+1$, then $c_d=c_d(\alpha,\theta)$ and $h_d=h_d(\alpha,\theta,z)$,

and thus

$$c_d = \begin{bmatrix} 0 \\ \frac{\alpha\bar{b}_{d,0}\binom{n}{0}\binom{m-d}{1}}{\binom{m+n-d}{1}} \\ \frac{\alpha\bar{b}_{d,1}\binom{n}{1}\binom{m-d}{1}\theta}{\binom{m+n-d}{2}} \\ \vdots \\ \frac{\alpha\bar{b}_{d,n-1}\binom{n}{n-1}\binom{m-d}{1}\theta^{n-1}}{\binom{m+n-d}{n}} \\ \frac{\alpha\bar{b}_{d,n}\binom{n}{n}\binom{m-d}{1}\theta^{n}}{\binom{m+n-d}{n+1}} \\ 0_{m-d-1} \end{bmatrix}, \qquad \frac{\partial c_d}{\partial\theta} = \begin{bmatrix} 0 \\ 0 \\ \frac{\alpha\bar{b}_{d,1}\binom{n}{1}\binom{m-d}{1}}{\binom{m+n-d}{2}} \\ \vdots \\ \frac{\alpha\bar{b}_{d,n-1}\binom{n}{n-1}\binom{m-d}{1}(n-1)\theta^{n-2}}{\binom{m+n-d}{n}} \\ \frac{\alpha\bar{b}_{d,n}\binom{n}{n}\binom{m-d}{1}n\theta^{n-1}}{\binom{m+n-d}{n+1}} \\ 0_{m-d-1} \end{bmatrix},$$

where 0_{m-d-1} is a column vector of zeros of length $m-d-1$, and

$$\frac{\partial c_d}{\partial\alpha} = \begin{bmatrix} 0 \\ \frac{\bar{b}_{d,0}\binom{n}{0}\binom{m-d}{1}}{\binom{m+n-d}{1}} \\ \frac{\bar{b}_{d,1}\binom{n}{1}\binom{m-d}{1}\theta}{\binom{m+n-d}{2}} \\ \vdots \\ \frac{\bar{b}_{d,n-1}\binom{n}{n-1}\binom{m-d}{1}\theta^{n-1}}{\binom{m+n-d}{n}} \\ \frac{\bar{b}_{d,n}\binom{n}{n}\binom{m-d}{1}\theta^{n}}{\binom{m+n-d}{n+1}} \\ 0_{m-d-1} \end{bmatrix}.$$

The vectors h_d, $\frac{\partial h_d}{\partial \theta}$ and $\frac{\partial h_d}{\partial \alpha}$ have similar forms,

$$h_d = \begin{bmatrix} 0 \\ \frac{\alpha z_{m+1}\binom{n}{0}\binom{m-d}{1}}{\binom{m+n-d}{1}} \\ \frac{\alpha z_{m+2}\binom{n}{1}\binom{m-d}{1}\theta}{\binom{m+n-d}{2}} \\ \vdots \\ \frac{\alpha z_{m+n}\binom{n}{n-1}\binom{m-d}{1}\theta^{n-1}}{\binom{m+n-d}{n}} \\ \frac{\alpha z_{m+n+1}\binom{n}{n}\binom{m-d}{1}\theta^{n}}{\binom{m+n-d}{n+1}} \\ 0_{m-d-1} \end{bmatrix}, \quad \frac{\partial h_d}{\partial \theta} = \begin{bmatrix} 0 \\ 0 \\ \frac{\alpha z_{m+2}\binom{n}{1}\binom{m-d}{1}}{\binom{m+n-d}{2}} \\ \vdots \\ \frac{\alpha z_{m+n}\binom{n}{n-1}\binom{m-d}{1}(n-1)\theta^{n-2}}{\binom{m+n-d}{n}} \\ \frac{\alpha z_{m+n+1}\binom{n}{n}\binom{m-d}{1}n\theta^{n-1}}{\binom{m+n-d}{n+1}} \\ 0_{m-d-1} \end{bmatrix},$$

and

$$\frac{\partial h_d}{\partial \alpha} = \begin{bmatrix} 0 \\ \frac{z_{m+1}\binom{n}{0}\binom{m-d}{1}}{\binom{m+n-d}{1}} \\ \frac{z_{m+2}\binom{n}{1}\binom{m-d}{1}\theta}{\binom{m+n-d}{2}} \\ \vdots \\ \frac{z_{m+n}\binom{n}{n-1}\binom{m-d}{1}\theta^{n-1}}{\binom{m+n-d}{n}} \\ \frac{z_{m+n+1}\binom{n}{n}\binom{m-d}{1}\theta^{n}}{\binom{m+n-d}{n+1}} \\ 0_{m-d-1} \end{bmatrix}.$$

The partial derivatives $\frac{\partial U_d}{\partial \theta}$, $\frac{\partial U_d}{\partial \alpha}$, $\frac{\partial F_d}{\partial \theta}$ and $\frac{\partial F_d}{\partial \alpha}$ are calculated in a similar manner. $\square$

If $q > n - d + 1$, the general expression for h_d is

$$
\begin{aligned}
h_d &= \begin{bmatrix}
0_{q-n+d-2} \\
\frac{\alpha z_{m+1}\binom{n}{0}\binom{m-d}{q-n+d-2}}{\binom{m+n-d}{q-n+d-2}} \\
\frac{\alpha z_{m+2}\binom{n}{1}\binom{m-d}{q-n+d-2}\theta}{\binom{m+n-d}{q-n+d-1}} \\
\vdots \\
\frac{\alpha z_{m+n}\binom{n}{n-1}\binom{m-d}{q-n+d-2}\theta^{n-1}}{\binom{m+n-d}{q+d-3}} \\
\frac{\alpha z_{m+n+1}\binom{n}{n}\binom{m-d}{q-n+d-2}\theta^{n}}{\binom{m+n-d}{q+d-2}} \\
0_{m+n-2d-q+2}
\end{bmatrix} \\
&= \alpha D_d^{-1} \begin{bmatrix}
0_{q-n+d-2,m+1} & 0_{q-n+d-2,n+1} \\
0_{n+1,m+1} & G \\
0_{m+n-2d-q+2,m+1} & 0_{m+n-2d-q+2,n+1}
\end{bmatrix} \begin{bmatrix}
z_0 \\ \vdots \\ z_m \\ z_{m+1} \\ \vdots \\ z_{m+n+1}
\end{bmatrix} \\
&= \alpha D_d^{-1} P_d z,
\end{aligned}
$$

where D_d^{-1} is defined in (3.14), $G = G(\theta) \in \mathbb{R}^{(n+1)\times(n+1)}$,

$$
G = \operatorname{diag}\left[\begin{array}{cccc}
\binom{n}{0}\binom{m-d}{q-n+d-2} & \binom{n}{1}\binom{m-d}{q-n+d-2}\theta & \cdots & \binom{n}{n-1}\binom{m-d}{q-n+d-2}\theta^{n-1} \quad \binom{n}{n}\binom{m-d}{q-n+d-2}\theta^{n}
\end{array} \right],
$$

and

$$
P_d = P_d(\theta) = \begin{bmatrix}
0_{q-n+d-2,m+1} & 0_{q-n+d-2,n+1} \\
0_{n+1,m+1} & G \\
0_{m+n-2d-q+2,m+1} & 0_{m+n-2d-q+2,n+1}
\end{bmatrix} \in \mathbb{R}^{(m+n-d+1)\times(m+n+2)}.
$$

Therefore, it follows that

$$
\delta h_d = \sum_{i=0}^{m+n+1} \frac{\partial h_d}{\partial z_i} \delta z_i = \alpha D_d^{-1} P_d \delta z,
$$

which enables the penultimate term in (9.11) to be simplified. Also, there exists a matrix $Y_d = Y_d(\alpha, \theta, x) \in \mathbb{R}^{(m+n-d+1)\times(m+n+2)}$ such that

$$(D_d^{-1}Y_d)z = (D_d^{-1}F_dQ_dM_{d,q})x,$$

for all α, θ, z, x, and this equation is obtained because polynomial multiplication is commutative. It therefore follows that on differentiating both sides of this equation with respect to z,

$$D_d^{-1}Y_d\delta z = D_d^{-1}\left(\delta F_d|_{\alpha,\theta:\text{ const.}}\right)Q_dM_{d,q}x = D_d^{-1}\left(\sum_{i=0}^{m+n+1}\frac{\partial F_d}{\partial z_i}\delta z_i\right)Q_dM_{d,q}x,$$

and thus (9.11) simplifies to

$$\begin{aligned}\tilde{r} \;=\;& r(\alpha,\theta,x,z) - \left(D_d^{-1}\left(\frac{\partial U_d}{\partial\theta}+\frac{\partial F_d}{\partial\theta}\right)Q_dM_{d,q}x - \left(\frac{\partial c_d}{\partial\theta}+\frac{\partial h_d}{\partial\theta}\right)\right)\delta\theta\\ &- \left(D_d^{-1}\left(\frac{\partial U_d}{\partial\alpha}+\frac{\partial F_d}{\partial\alpha}\right)Q_dM_{d,q}x - \left(\frac{\partial c_d}{\partial\alpha}+\frac{\partial h_d}{\partial\alpha}\right)\right)\delta\alpha - D_d^{-1}(U_d+F_d)Q_dM_{d,q}\delta x\\ &- D_d^{-1}(Y_d-\alpha P_d)\delta z. \qquad (9.12)\end{aligned}$$

Example 9.2. Let $m = 4, n = 3, d = 2$ and $q = 4$. Thus $D_2^{-1}U_2Q_2 \in \mathbb{R}^{6\times5}$ and $M_{2,4} \in \mathbb{R}^{5\times4}$,

$$D_2^{-1}U_2Q_2M_{2,4} = \begin{bmatrix} \frac{\bar{a}_{d,0}\binom{4}{0}\binom{1}{0}}{\binom{5}{0}} & 0 & \frac{\alpha\bar{b}_{d,0}\binom{3}{0}\binom{2}{0}}{\binom{5}{0}} & 0\\ \frac{\bar{a}_{d,1}\binom{4}{1}\binom{1}{0}\theta}{\binom{5}{1}} & \frac{\bar{a}_{d,0}\binom{4}{0}\binom{1}{1}}{\binom{5}{1}} & \frac{\alpha\bar{b}_{d,1}\binom{3}{1}\binom{2}{0}\theta}{\binom{5}{1}} & 0\\ \frac{\bar{a}_{d,2}\binom{4}{2}\binom{1}{0}\theta^2}{\binom{5}{2}} & \frac{\bar{a}_{d,1}\binom{4}{1}\binom{1}{1}\theta}{\binom{5}{2}} & \frac{\alpha\bar{b}_{d,2}\binom{3}{2}\binom{2}{0}\theta^2}{\binom{5}{2}} & \frac{\alpha\bar{b}_{d,0}\binom{3}{0}\binom{2}{2}}{\binom{5}{2}}\\ \frac{\bar{a}_{d,3}\binom{4}{3}\binom{1}{0}\theta^3}{\binom{5}{3}} & \frac{\bar{a}_{d,2}\binom{4}{2}\binom{1}{1}\theta^2}{\binom{5}{3}} & \frac{\alpha\bar{b}_{d,3}\binom{3}{3}\binom{2}{0}\theta^3}{\binom{5}{3}} & \frac{\alpha\bar{b}_{d,1}\binom{3}{1}\binom{2}{2}\theta}{\binom{5}{3}}\\ \frac{\bar{a}_{d,4}\binom{4}{4}\binom{1}{0}\theta^4}{\binom{5}{4}} & \frac{\bar{a}_{d,3}\binom{4}{3}\binom{1}{1}\theta^3}{\binom{5}{4}} & 0 & \frac{\alpha\bar{b}_{d,2}\binom{3}{2}\binom{2}{2}\theta^2}{\binom{5}{4}}\\ 0 & \frac{\bar{a}_{d,4}\binom{4}{4}\binom{1}{1}\theta^4}{\binom{5}{5}} & 0 & \frac{\alpha\bar{b}_{d,3}\binom{3}{3}\binom{2}{2}\theta^3}{\binom{5}{5}} \end{bmatrix},$$

$$c_2 = \begin{bmatrix} 0 \\ \frac{\alpha\bar{b}_{d,0}\binom{3}{0}\binom{2}{1}}{\binom{5}{1}} \\ \frac{\alpha\bar{b}_{d,1}\binom{3}{1}\binom{2}{1}\theta}{\binom{5}{2}} \\ \frac{\alpha\bar{b}_{d,2}\binom{3}{2}\binom{2}{1}\theta^2}{\binom{5}{3}} \\ \frac{\alpha\bar{b}_{d,3}\binom{3}{3}\binom{2}{1}\theta^3}{\binom{5}{4}} \\ 0 \end{bmatrix}, \qquad z^T = \begin{bmatrix} z_0 & z_1 & z_2 & z_3 & z_4 & z_5 & z_6 & z_7 & z_8 \end{bmatrix},$$

$$P_2 = \begin{bmatrix} 0_{1,5} & 0_{1,4} \\ 0_{4,5} & G(\theta) \\ 0_{1,5} & 0_{1,4} \end{bmatrix}, \qquad G(\theta) = \operatorname{diag}\begin{bmatrix} \binom{3}{0}\binom{2}{1} & \binom{3}{1}\binom{2}{1}\theta & \binom{3}{2}\binom{2}{1}\theta^2 & \binom{3}{3}\binom{2}{1}\theta^3 \end{bmatrix},$$

$$h_2 = \begin{bmatrix} 0 \\ \frac{\alpha z_5\binom{3}{0}\binom{2}{1}}{\binom{5}{1}} \\ \frac{\alpha z_6\binom{3}{1}\binom{2}{1}\theta}{\binom{5}{2}} \\ \frac{\alpha z_7\binom{3}{2}\binom{2}{1}\theta^2}{\binom{5}{3}} \\ \frac{\alpha z_8\binom{3}{3}\binom{2}{1}\theta^3}{\binom{5}{4}} \\ 0 \end{bmatrix}, \qquad x^T = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix},$$

it is easily verified that $h_2 = \alpha D_2^{-1} P_2 z$. Similarly, $D_2^{-1} F_2 Q_2 \in \mathbb{R}^{6\times 5}$,

$$D_2^{-1} F_2 Q_2 M_{2,4} = \begin{bmatrix} \frac{z_0\binom{4}{0}\binom{1}{0}}{\binom{5}{0}} & 0 & \frac{\alpha z_5\binom{3}{0}\binom{2}{0}}{\binom{5}{0}} & 0 \\ \frac{z_1\binom{4}{1}\binom{1}{0}\theta}{\binom{5}{1}} & \frac{z_0\binom{4}{0}\binom{1}{1}}{\binom{5}{1}} & \frac{\alpha z_6\binom{3}{1}\binom{2}{0}\theta}{\binom{5}{1}} & 0 \\ \frac{z_2\binom{4}{2}\binom{1}{0}\theta^2}{\binom{5}{2}} & \frac{z_1\binom{4}{1}\binom{1}{1}\theta}{\binom{5}{2}} & \frac{\alpha z_7\binom{3}{2}\binom{2}{0}\theta^2}{\binom{5}{2}} & \frac{\alpha z_5\binom{3}{0}\binom{2}{2}}{\binom{5}{2}} \\ \frac{z_3\binom{4}{3}\binom{1}{0}\theta^3}{\binom{5}{3}} & \frac{z_2\binom{4}{2}\binom{1}{1}\theta^2}{\binom{5}{3}} & \frac{\alpha z_8\binom{3}{3}\binom{2}{0}\theta^3}{\binom{5}{3}} & \frac{\alpha z_6\binom{3}{1}\binom{2}{2}\theta}{\binom{5}{3}} \\ \frac{z_4\binom{4}{4}\binom{1}{0}\theta^4}{\binom{5}{4}} & \frac{z_3\binom{4}{3}\binom{1}{1}\theta^3}{\binom{5}{4}} & 0 & \frac{\alpha z_7\binom{3}{2}\binom{2}{2}\theta^2}{\binom{5}{4}} \\ 0 & \frac{z_4\binom{4}{4}\binom{1}{1}\theta^4}{\binom{5}{5}} & 0 & \frac{\alpha z_8\binom{3}{3}\binom{2}{2}\theta^3}{\binom{5}{5}} \end{bmatrix},$$

and the matrix Y_2 is equal to $[\ y_l \mid y_r\ ]$, where

$$y_l = \begin{bmatrix} \binom{4}{0}\binom{1}{0}x_1 & 0 & 0 & 0 & 0 \\ \binom{4}{0}\binom{1}{1}x_2 & \binom{4}{1}\binom{1}{0}\theta x_1 & 0 & 0 & 0 \\ 0 & \binom{4}{1}\binom{1}{1}\theta x_2 & \binom{4}{2}\binom{1}{0}\theta^2 x_1 & 0 & 0 \\ 0 & 0 & \binom{4}{2}\binom{1}{1}\theta^2 x_2 & \binom{4}{3}\binom{1}{0}\theta^3 x_1 & 0 \\ 0 & 0 & 0 & \binom{4}{3}\binom{1}{1}\theta^3 x_2 & \binom{4}{4}\binom{1}{0}\theta^4 x_1 \\ 0 & 0 & 0 & 0 & \binom{4}{4}\binom{1}{1}\theta^4 x_2 \end{bmatrix},$$

and

$$y_r = \begin{bmatrix} \alpha\binom{3}{0}\binom{2}{0}x_3 & 0 & 0 & 0 \\ 0 & \alpha\binom{3}{1}\binom{2}{0}\theta x_3 & 0 & 0 \\ \alpha\binom{3}{0}\binom{2}{2}x_4 & 0 & \alpha\binom{3}{2}\binom{2}{0}\theta^2 x_3 & 0 \\ 0 & \alpha\binom{3}{1}\binom{2}{2}\theta x_4 & 0 & \alpha\binom{3}{3}\binom{2}{0}\theta^3 x_3 \\ 0 & 0 & \alpha\binom{3}{2}\binom{2}{2}\theta^2 x_4 & 0 \\ 0 & 0 & 0 & \alpha\binom{3}{3}\binom{2}{2}\theta^3 x_4 \end{bmatrix}.$$

It is easy to verify that $(D_2^{-1}Y_2)z = (D_2^{-1}F_2Q_2M_{2,4})x$. □

The initial values of α and θ are α_1 and θ_1, which are the solutions of (7.33). The initial value of z is $z^{(0)} = 0$ because the given data is inexact, and the initial value of x, is calculated from (9.10),

$$x_0 = \arg\min_x \left\| D_d^{-1}U_d(\alpha_1, \theta_1)Q_dM_{d,q}x - c_d(\alpha_1, \theta_1) \right\|. \tag{9.13}$$

The jth iteration in the Newton-Raphson method for the calculation of z, x, α, θ, is

obtained from (9.12),

$$\left[\begin{array}{cccc} H_z & H_x & H_\alpha & H_\theta \end{array}\right]^{(j)} \left[\begin{array}{c} \delta z \\ \delta x \\ \delta\alpha \\ \delta\theta \end{array}\right]^{(j)} = r^{(j)}, \tag{9.14}$$

where $r^{(j)} = r^{(j)}(\alpha, \theta, x, z)$,

$$\begin{aligned} H_z &= D_d^{-1}(Y_d - \alpha P_d) \in \mathbb{R}^{(m+n-d+1)\times(m+n+2)}, \\ H_x &= D_d^{-1}(U_d + F_d)Q_d M_{d,q} \in \mathbb{R}^{(m+n-d+1)\times(m+n-2d+1)}, \\ H_\alpha &= D_d^{-1}\left(\frac{\partial U_d}{\partial\alpha} + \frac{\partial F_d}{\partial\alpha}\right) Q_d M_{d,q} x - \left(\frac{\partial c_d}{\partial\alpha} + \frac{\partial h_d}{\partial\alpha}\right) \in \mathbb{R}^{m+n-d+1}, \\ H_\theta &= D_d^{-1}\left(\frac{\partial U_d}{\partial\theta} + \frac{\partial F_d}{\partial\theta}\right) Q_d M_{d,q} x - \left(\frac{\partial c_d}{\partial\theta} + \frac{\partial h_d}{\partial\theta}\right) \in \mathbb{R}^{m+n-d+1}, \end{aligned}$$

and the values of z, x, α, θ at the $(j+1)$th iteration are

$$\left[\begin{array}{c} z \\ x \\ \alpha \\ \theta \end{array}\right]^{(j+1)} = \left[\begin{array}{c} z \\ x \\ \alpha \\ \theta \end{array}\right]^{(j)} + \left[\begin{array}{c} \delta z \\ \delta x \\ \delta\alpha \\ \delta\theta \end{array}\right]^{(j)}.$$

Equation (9.14) is of the form

$$Cy = e, \tag{9.15}$$

where $C \in \mathbb{R}^{(m+n-d+1)\times(2m+2n-2d+5)}$, $y \in \mathbb{R}^{2m+2n-2d+5}$ and $e \in \mathbb{R}^{m+n-d+1}$,

$$C = \left[\begin{array}{cccc} H_z & H_x & H_\alpha & H_\theta \end{array}\right]^{(j)}, \qquad y = \left[\begin{array}{c} \delta z \\ \delta x \\ \delta\alpha \\ \delta\theta \end{array}\right]^{(j)}, \qquad e = r^{(j)}. \tag{9.16}$$

It is necessary to calculate the vector y of minimum magnitude that satisfies (9.15),

that is, the solution that is closest to the given inexact data is required.

Since

$$\left\|\begin{bmatrix} z^{(j+1)} - z^{(0)} \\ x^{(j+1)} - x_0 \\ \alpha^{(j+1)} - \alpha_1 \\ \theta^{(j+1)} - \theta_1 \end{bmatrix}\right\| = \left\|\begin{bmatrix} z^{(j)} + \delta z^{(j)} \\ x^{(j)} + \delta x^{(j)} - x_0 \\ \alpha^{(j)} + \delta\alpha^{(j)} - \alpha_1 \\ \theta^{(j)} + \delta\theta^{(j)} - \theta_1 \end{bmatrix}\right\| := ||Ey - p||, \qquad (9.17)$$

where $E = I_{2m+2n-2d+5}$, y is defined in (9.16), and p is equal to

$$p = -\begin{bmatrix} z^{(j)} \\ (x^{(j)} - x_0) \\ (\alpha^{(j)} - \alpha_1) \\ (\theta^{(j)} - \theta_1) \end{bmatrix}.$$

It is noted that E is constant and not updated between iterations.

The minimization of (9.17) subject to (9.15) is a least squares minimization with an equality constraint (the LSE problem),

$$\min_y ||Ey - p|| \qquad \text{subject to} \qquad Cy = e,$$

which can be solved by the QR decomposition [30]. This LSE problem is solved at each iteration, where C, e and p are updated between successive iterations.

Algorithm 9.1: SNTLN for a Sylvester matrix

Input Inexact Bernstein polynomials $f(x)$ and $g(x)$, which are of degrees m and n respectively and defined in (4.2), and the degree d of an AGCD of $f(x)$ and $g(x)$.

Output A structured low rank approximation of $\bar{S}_d(\tilde{f}_d, \alpha\tilde{g}_d)Q_d$.

Begin

1. Preprocess $f(x)$ and $g(x)$ to yield $\tilde{f}_d(w)$ and $\tilde{g}_d(w)$, which are defined in (7.34) and (7.35) respectively, using the preprocessing operations described in Section 7.3.

2. Calculate the integer q and the matrix $M_{d,q}$.

3. % Initialize the data

   - Calculate the diagonal matrices D_d^{-1} and Q_d.
   - Set $z = z^{(0)} = 0$, which yields $F_d = \frac{\partial F_d}{\partial \alpha} = \frac{\partial F_d}{\partial \theta} = 0$ and $h_d = \frac{\partial h_d}{\partial \alpha} = \frac{\partial h_d}{\partial \theta} = 0$.
   - Calculate $U_d, Y_d, P_d, c_d, \frac{\partial U_d}{\partial \alpha}, \frac{\partial U_d}{\partial \theta}, \frac{\partial c_d}{\partial \alpha}$ and $\frac{\partial c_d}{\partial \theta}$ for $\alpha = \alpha_1(d), \theta = \theta_1(d)$ and the initial value x_0 of x, which is defined in (9.13). Calculate the initial value of e, which is equal to the residual,

     $$\begin{aligned} r(\alpha_1(d), \theta_1(d), x_0, z^{(0)} = 0) &= c_d(\alpha_1(d), \theta_1(d)) \\ &\quad -D_d^{-1} U_d(\alpha_1(d), \theta_1(d)) Q_d M_{d,q} x_0, \end{aligned}$$

     and set the initial value of p, $p = 0$.
   - Define the matrices C and E.

4. % The loop for the iterations

   % Solve the LSE problem at each iteration using the QR decomposition

   repeat

   (a) Compute the QR decomposition of C^T,

   $$C^T = QR = Q \begin{bmatrix} R_1 \\ 0 \end{bmatrix}.$$

   (b) Set $w_1 = R_1^{-T} e$.

(c) Partition EQ as

$$EQ = \begin{bmatrix} E_1 & E_2 \end{bmatrix},$$

where

$E_1 \in \mathbb{R}^{(2m+2n-2d+5)\times(m+n-d+1)}$, $E_2 \in \mathbb{R}^{(2m+2n-2d+5)\times(m+n-d+4)}$.

(d) Compute

$$z_1 = E_2^{\dagger}(p - E_1 w_1).$$

(e) Compute the solution

$$y = Q \begin{bmatrix} w_1 \\ z_1 \end{bmatrix}.$$

(f) Set $z := z + \delta z$, $x := x + \delta x$, $\alpha := \alpha + \delta\alpha$ and $\theta := \theta + \delta\theta$.

(g) Update $U_d, \frac{\partial U_d}{\partial \alpha}, \frac{\partial U_d}{\partial \theta}, F_d, \frac{\partial F_d}{\partial \alpha}, \frac{\partial F_d}{\partial \theta}, Y_d, P_d, c_d, \frac{\partial c_d}{\partial \alpha}, \frac{\partial c_d}{\partial \theta}, h_d, \frac{\partial h_d}{\partial \alpha}, \frac{\partial h_d}{\partial \theta}$ from α, θ, x, z, and therefore C. Compute the residual

$$r(\alpha, \theta, x, z) = (c_d + h_d) - D_d^{-1}(U_d + F_d)Q_d M_{d,q} x,$$

and thus update e. Update p from α, θ, x and z.

until $\frac{||r(\alpha,\theta,x,z)||}{||c_d+h_d||} \leq 10^{-12}$

End

Algorithm 9.1 terminates when the residual $\frac{||r(\alpha,\theta,x,z)||}{||c_d+h_d||}$ is sufficiently small and yields α_*, θ_*, z^* and x^*, where α_* and θ_* are the optimal values of α and θ, and z^* is

the perturbation vector,

$$z^* = \begin{bmatrix} z_f^* \\ z_g^* \end{bmatrix} \in \mathbb{R}^{m+n+2},$$

where

$$z_f^* = \begin{bmatrix} z_0^* \\ z_1^* \\ \vdots \\ z_m^* \end{bmatrix} \in \mathbb{R}^{m+1} \qquad \text{and} \qquad z_g^* = \begin{bmatrix} z_{m+1}^* \\ z_{m+2}^* \\ \vdots \\ z_{m+n+1}^* \end{bmatrix} \in \mathbb{R}^{n+1}.$$

The elements $z_i^*, i = 0, \ldots, m$, in the vector z^*, occupy the first $n - d + 1$ columns of F_d defined in (9.7), and the elements $z_i^*, i = m+1, \ldots, m+n+1$, in the vector z^*, occupy the last $m - d + 1$ columns of F_d.

The elements of z_f^* are added to the coefficients $\bar{a}_{d,i}, i = 0, \ldots, m$ of $\tilde{f}_d(w)$, and the elements of z_g^* are added to the coefficients $\bar{b}_{d,j}, j = 0, \ldots, n$ of $\tilde{g}_d(w)$ respectively. Thus, the corrected forms of $\tilde{f}_d(w)$ and $\tilde{g}_d(w)$, which are defined in (7.34) and (7.35) respectively, are

$$\begin{aligned} \tilde{f}_d^*(w) &= \sum_{i=0}^{m} \left(\tilde{a}_i \theta_*^i\right) \binom{m}{i} (1 - \theta_* w)^{m-i} w^i \\ &= \sum_{i=0}^{m} \left((\bar{a}_{d,i} + z_i^*)\theta_*^i\right) \binom{m}{i} (1 - \theta_* w)^{m-i} w^i, \end{aligned} \tag{9.18}$$

and

$$\begin{aligned} \tilde{g}_d^*(w) &= \sum_{j=0}^{n} \left(\tilde{b}_j \theta_*^j\right) \binom{n}{j} (1 - \theta_* w)^{n-j} w^j \\ &= \sum_{j=0}^{n} \left((\bar{b}_{d,j} + z_{m+j+1}^*)\theta_*^j\right) \binom{n}{j} (1 - \theta_* w)^{n-j} w^j. \end{aligned} \tag{9.19}$$

The corrected forms $\tilde{f}_d^*(w)$ and $\alpha_* \tilde{g}_d^*(w)$ have a non-constant common divisor $\tilde{d}^*(w)$

of degree d, that is

$$\tilde{f}_d^*(w) = \tilde{u}^*(w)\tilde{d}^*(w) \qquad \text{and} \qquad \alpha_* \tilde{g}_d^*(w) = \tilde{v}^*(w)\tilde{d}^*(w), \tag{9.20}$$

where

$$\tilde{d}^*(w) = \sum_{i=0}^{d} \left(\tilde{d}_i \theta_*^i\right) \binom{d}{i} (1-\theta_* w)^{d-i} w^i,$$

the coefficients of the quotient polynomials $\tilde{u}^*(w)$ and $\tilde{v}^*(w)$ are obtained from the vector x^*,

$$\begin{bmatrix} \tilde{v}_0 \\ \vdots \\ \tilde{v}_{n-d}\theta_*^{n-d} \\ -\tilde{u}_0 \\ \vdots \\ -\tilde{u}_{m-d}\theta_*^{m-d} \end{bmatrix} = \begin{bmatrix} x_1^* \\ \vdots \\ x_{q-1}^* \\ -1 \\ x_{q+1}^* \\ \vdots \\ x_{m+n-2d+2}^* \end{bmatrix} \in \mathbb{R}^{m+n-2d+2},$$

and thus

$$\tilde{u}^*(w) = \sum_{i=0}^{m-d} \left(\tilde{u}_i \theta_*^i\right) \binom{m-d}{i} (1-\theta_* w)^{m-d-i} w^i,$$

and

$$\tilde{v}^*(w) = \sum_{j=0}^{n-d} \left(\tilde{v}_j \theta_*^j\right) \binom{n-d}{j} (1-\theta_* w)^{n-d-j} w^j.$$

The equations, $\tilde{f}_d^*(w) = \tilde{u}^*(w)\tilde{d}^*(w)$ and $\alpha_* \tilde{g}_d^*(w) = \tilde{v}^*(w)\tilde{d}^*(w)$, in (9.20) are written as

$$\sum_{j=\max(0,i-d)}^{\min(m-d,i)} \frac{\binom{m-d}{j}\binom{d}{i-j}}{\binom{m}{i}} \tilde{u}_j \theta_*^j \tilde{d}_{i-j} \theta_*^{i-j} = \tilde{a}_i \theta_*^i, \qquad i = 0, \ldots, m,$$

and

$$\sum_{j=\max(0,i-d)}^{\min(n-d,i)} \frac{\binom{n-d}{j}\binom{d}{i-j}}{\binom{n}{i}} \tilde{v}_j \theta_*^j \tilde{d}_{i-j} \theta_*^{i-j} = \alpha_* \tilde{b}_i \theta_*^i, \qquad i = 0, \ldots, n,$$

respectively, which can be expressed in the matrix form,

$$L^{-1} \begin{bmatrix} \tilde{\mathbf{u}}(\theta_*) \\ \tilde{\mathbf{v}}(\theta_*) \end{bmatrix} \tilde{\mathbf{d}}(\theta_*) = \begin{bmatrix} \tilde{\mathbf{f}}(\theta_*) \\ \alpha_* \tilde{\mathbf{g}}(\theta_*) \end{bmatrix}, \tag{9.21}$$

where $L^{-1} \in \mathbb{R}^{(m+n+2)\times(m+n+2)}$,

$$L^{-1} = \operatorname{diag}\begin{bmatrix} \frac{1}{\binom{m}{0}} & \frac{1}{\binom{m}{1}} & \cdots & \frac{1}{\binom{m}{m}} & \frac{1}{\binom{n}{0}} & \frac{1}{\binom{n}{1}} & \cdots & \frac{1}{\binom{n}{n}} \end{bmatrix},$$

$\tilde{\mathbf{u}}(\theta_*)$ and $\tilde{\mathbf{v}}(\theta_*)$ are Toeplitz matrices,

$$\tilde{\mathbf{u}}(\theta_*) = \begin{bmatrix} \tilde{u}_0\binom{m-d}{0} & & \\ \tilde{u}_1\binom{m-d}{1}\theta_* & \ddots & \\ \vdots & \ddots & \tilde{u}_0\binom{m-d}{0} \\ \tilde{u}_{m-d}\binom{m-d}{m-d}\theta_*^{m-d} & \ddots & \tilde{u}_1\binom{m-d}{1}\theta_* \\ & \ddots & \vdots \\ & & \tilde{u}_{m-d}\binom{m-d}{m-d}\theta_*^{m-d} \end{bmatrix} \in \mathbb{R}^{(m+1)\times(d+1)},$$

and

$$\tilde{\mathbf{v}}(\theta_*) = \begin{bmatrix} \tilde{v}_0\binom{n-d}{0} & & \\ \tilde{v}_1\binom{n-d}{1}\theta_* & \ddots & \\ \vdots & \ddots & \tilde{v}_0\binom{n-d}{0} \\ \tilde{v}_{n-d}\binom{n-d}{n-d}\theta_*^{n-d} & \ddots & \tilde{v}_1\binom{n-d}{1}\theta_* \\ & \ddots & \vdots \\ & & \tilde{v}_{n-d}\binom{n-d}{n-d}\theta_*^{n-d} \end{bmatrix} \in \mathbb{R}^{(n+1)\times(d+1)},$$

$\tilde{\mathbf{d}}(\theta_*) \in \mathbb{R}^{d+1}$ is the vector of the scaled coefficients of $\tilde{d}^*(w)$,

$$\tilde{\mathbf{d}}(\theta_*) = \begin{bmatrix} \tilde{d}_0\binom{d}{0} & \tilde{d}_1\binom{d}{1}\theta_* & \cdots & \tilde{d}_d\binom{d}{d}\theta_*^d \end{bmatrix}^T,$$

$\tilde{\mathbf{f}}(\theta_*) \in \mathbb{R}^{m+1}$ and $\tilde{\mathbf{g}}(\theta_*) \in \mathbb{R}^{n+1}$ are the coefficient vectors of $\tilde{f}_d^*(w)$ and $\tilde{g}_d^*(w)$,

$$\tilde{\mathbf{f}}(\theta_*) = \begin{bmatrix} \tilde{a}_0 & \tilde{a}_1\theta_* & \cdots & \tilde{a}_m\theta_*^m \end{bmatrix}^T,$$

and

$$\tilde{\mathbf{g}}(\theta_*) = \begin{bmatrix} \tilde{b}_0 & \tilde{b}_1\theta_* & \cdots & \tilde{b}_n\theta_*^n \end{bmatrix}^T.$$

The vector $\tilde{\mathbf{d}}(\theta_*)$ is unknown and can be calculated from (9.21) by solving the least squares problem, that is

$$\tilde{\mathbf{d}}(\theta_*) = \left(L^{-1} \begin{bmatrix} \tilde{\mathbf{u}}(\theta_*) \\ \tilde{\mathbf{v}}(\theta_*) \end{bmatrix} \right)^{\dagger} \begin{bmatrix} \tilde{\mathbf{f}}(\theta_*) \\ \alpha_*\tilde{\mathbf{g}}(\theta_*) \end{bmatrix},$$

which stores the scaled coefficients of common divisor $\tilde{d}^*(w)$ defined in the modified Bernstein basis. Then, the common divisor $d^*(x)$ in the Bernstein basis, which is an AGCD of the inexact polynomials $f(x)$ and $g(x)$, is obtained from the vector $\tilde{\mathbf{d}}(\theta_*)$,

$$\tilde{d}_i = \frac{\tilde{d}_i \binom{d}{i} \theta_*^i}{\binom{d}{i}\theta_*^i}, \qquad i = 0, \ldots, d, \tag{9.22}$$

where $\tilde{d}_i$, $i = 0, \ldots, d$, are the coefficients of $d^*(x)$.

In addition, the coefficients of the common divisor $d^*(x)$ can also be obtained using the QR decomposition applied to the Sylvester matrix of $\tilde{f}_d^*(w)$ and $\alpha_*\tilde{g}_d^*(w)$. It was shown in Chapter 3 that the coefficients of the GCD of two polynomials can be obtained from the last non-zero row of upper triangular form of their Sylvester matrix. Therefore, we can reduce the Sylvester matrix $\bar{S}(\tilde{f}_d^*, \alpha_*\tilde{g}_d^*)Q$ of $\tilde{f}_d^*(w)$ and $\alpha_*\tilde{g}_d^*(w)$ to its upper triangular form. Since the degree of the GCD of $\tilde{f}_d^*(w)$ and $\alpha_*\tilde{g}_d^*(w)$, which is equal to the rank loss of $\bar{S}(\tilde{f}_d^*, \alpha_*\tilde{g}_d^*)Q$, is known, the last non-zero row of its upper triangular form can be determined, which yields the coefficients of the GCD of $\tilde{f}_d^*(w)$ and $\alpha_*\tilde{g}_d^*(w)$. Thus, the coefficients of the GCD defined in the modified Bernstein basis are obtained, from which the coefficients of common divisor

$d^*(x)$ defined in the Bernstein basis are computed using (9.22).

A slight modification of the method of SNTLN for $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$ yields the method of SNTLN implemented on $\bar{S}_d(\acute{f}_d, \alpha_2\acute{g}_d)$. The method of SNTLN for $\bar{S}_d(\acute{f}_d, \alpha_2\acute{g}_d)$ also yields α_+, θ_+, z^+ and x^+. The values of α_+ and θ_+ are the optimal values of α and θ, which are computed using the method of SNTLN with the initial values of α and θ equal to $\alpha_2(d)$ and $\theta_2(d)$. The vector $z^+ \in \mathbb{R}^{m+n+2}$ is the perturbation vector,

$$z^+ = \begin{bmatrix} z_f^+ \\ z_g^+ \end{bmatrix} \in \mathbb{R}^{m+n+2},$$

where

$$z_f^+ = \begin{bmatrix} z_0^+ \\ z_1^+ \\ \vdots \\ z_m^+ \end{bmatrix} \in \mathbb{R}^{m+1} \qquad \text{and} \qquad z_g^+ = \begin{bmatrix} z_{m+1}^+ \\ z_{m+2}^+ \\ \vdots \\ z_{m+n+1}^+ \end{bmatrix} \in \mathbb{R}^{n+1}.$$

The corrected forms of $\acute{f}_d(w)$ and $\acute{g}_d(w)$, which are defined in (7.36) and (7.37) respectively, are obtained by adding the elements of z_f^+ and z_g^+ to the coefficients of $\acute{f}_d(w)$ and $\acute{g}_d(w)$ respectively, that is

$$\begin{aligned} \acute{f}_d^+(w) &= \sum_{i=0}^{m} \left(\acute{a}_i\theta_+^i\right)\binom{m}{i}(1-\theta_+ w)^{m-i}w^i \\ &= \sum_{i=0}^{m} \left((\ddot{a}_{d,i} + z_i^+)\theta_+^i\right)\binom{m}{i}(1-\theta_+ w)^{m-i}w^i, \end{aligned} \tag{9.23}$$

and

$$\begin{aligned} \acute{g}_d^+(w) &= \sum_{j=0}^{n} \left(\acute{b}_j\theta_+^{m+j+1}\right)\binom{n}{j}(1-\theta_+ w)^{n-j}w^j \\ &= \sum_{j=0}^{n} \left((\ddot{b}_{d,j} + z_{m+j+1}^+)\theta_+^j\right)\binom{n}{j}(1-\theta_+ w)^{n-j}w^j. \end{aligned} \tag{9.24}$$

The corrected forms $\acute{f}_d^+(w)$ and $\alpha_+\acute{g}_d^+(w)$ have a non-constant common divisor $\acute{d}^+(w)$

of degree d, and the coefficients of the quotient polynomials $\acute{u}^+(w)$ and $\acute{v}^+(w)$ are obtained from the vector x^+,

$$\begin{bmatrix} \acute{v}_0\binom{n-d}{0} \\ \vdots \\ \acute{v}_{n-d}\binom{n-d}{n-d}\theta_+^{n-d} \\ -\acute{u}_0\binom{m-d}{0} \\ \vdots \\ -\acute{u}_{m-d}\binom{m-d}{m-d}\theta_+^{m-d} \end{bmatrix} = \begin{bmatrix} x_1^+ \\ \vdots \\ x_{q-1}^+ \\ -1 \\ x_{q+1}^+ \\ \vdots \\ x_{m+n-2d+2}^+ \end{bmatrix} \in \mathbb{R}^{m+n-2d+2}.$$

Then, by following the same procedure described above, the common divisor $d^+(x)$, which is defined in the Bernstein basis, is computed.

9.2 Examples

This section shows the calculation of the coefficients of an AGCD of two inexact Bernstein polynomials using the method of SNTLN implemented on $\bar{S}_d(\acute{f}_d, \alpha_2\acute{g}_d)$ and $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$.

As stated earlier, the method of SNTLN involves choosing the optimal column of the subresultant matrix to move to the right hand side. Two criteria used here are the first principal angle and the residual described in Section 7.4. Experiments show that the method of SNTLN using different criteria returns the similar results, and therefore the examples in this section only show the results that are obtained from the method of SNTLN using the criterion based on the first principal angle to select the optimal column.

Furthermore, the coefficients of the computed AGCD $d(x)$ of $f(x)$ and $g(x)$ must be

compared with the coefficients of the GCD $\hat{d}(x)$ of their exact polynomials $\hat{f}(x)$ and $\hat{g}(x)$. The coefficients of $d(x)$ are considered correct when its coefficients are good approximations to the coefficients of $\hat{d}(x)$. This can be measured by computing the error between the normalized coefficients of $d(x)$ and the normalized coefficients of $\hat{d}(x)$, that is

$$\|d - \hat{d}\|,$$

where $\|d\| = \|\hat{d}\| = 1$ and $\|\cdot\|$ denotes the 2-norm.

In addition, as described earlier, $d(x)$ can be computed by solving the least squares problem or using the QR decomposition. We use $d_{ls}(x)$ to denote $d(x)$ computed by solving the least squares problem and use $d_{qr}(x)$ to denote $d(x)$ computed by using the QR decomposition.

Example 9.3. Consider the exact Bernstein polynomials $\hat{f}(x)$ and $\hat{g}(x)$, whose roots and multiplicities are specified in Table 9.1. It is seen that the degree of their GCD is $\hat{d} = 17$.

Root of $\hat{f}(x)$	Multiplicity
0.3279e+000	6
0.6134e+000	4
0.9792e+000	6
-1.3981e+000	3
-3.9166e+000	3
9.7133e+000	2

Root of $\hat{g}(x)$	Multiplicity
0.3279e+000	5
0.6134e+000	5
0.9792e+000	7
2.3296e+000	2
4.6798e+000	2
9.7133e+000	2

Table 9.1: The roots and multiplicities of $\hat{f}(x)$ and $\hat{g}(x)$ for Example 9.3.

Noise with componentwise signal-to-noise ratio 10^8 is added to each polynomial to yield the inexact polynomials $f(x)$ and $g(x)$. The degree of an AGCD, $d = 17$,

is determined using the previous methods introduced in Chapters 6 and 7, which is correct because $d = \hat{d}$.

In the method of SNTLN implemented on $\bar{S}_d(\acute{f}_d, \alpha_2 \acute{g}_d)$, the inexact polynomials $f(x)$ and $g(x)$ are firstly normalized to yield the forms $\dot{f}_d(x)$ and $\dot{g}_d(x)$, which are defined in (7.16) and (7.17) respectively. Then, $\dot{f}_d(x)$ and $\dot{g}_d(x)$ are transformed to their modified Bernstein polynomials $\acute{f}_d(w)$ and $\acute{g}_d(w)$ defined in (7.36) and (7.37) respectively, and the dth subresultant matrix $\bar{S}_d(\acute{f}_d, \alpha_2 \acute{g}_d)$ is computed. The method of SNTLN is then performed on $\bar{S}_d(\acute{f}_d, \alpha_2 \acute{g}_d)$ and yields the optimal values of α and θ, where $\alpha_+ = 7.7050$ and $\theta_+ = 2.5012$, and $\acute{f}_d^+(w)$ and $\acute{g}_d^+(w)$, which are the corrected forms of $\acute{f}_d(w)$ and $\acute{g}_d(w)$. The Sylvester matrix $\bar{S}(\acute{f}_d^+, \alpha_+ \acute{g}_d^+)$ of $\acute{f}_d^+(w)$ and $\alpha_+ \acute{g}_d^+(w)$ is then calculated.

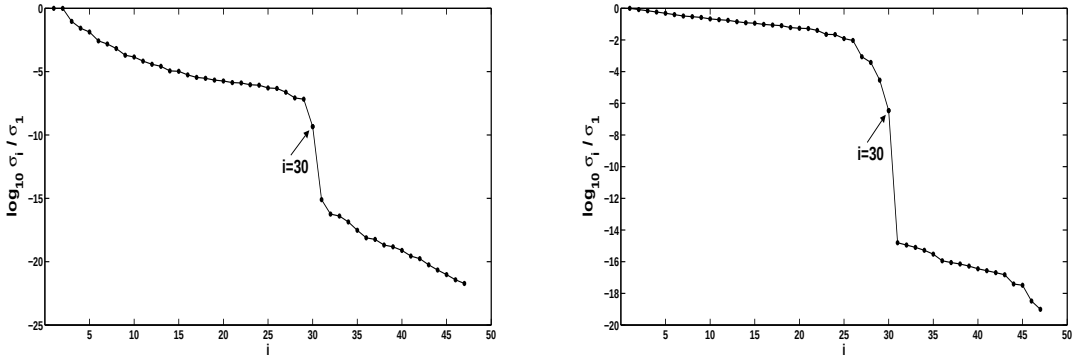
It is seen from Figure 9.1(a) that the rank loss of $\bar{S}(\acute{f}_d^+, \alpha_+ \acute{g}_d^+)$ is equal to 17, which implies that the degree of the GCD of the corrected polynomials $\acute{f}_d^+(w)$ and $\alpha_+ \acute{g}_d^+(w)$ is 17. Because $\acute{f}_d^+(w)$ and $\alpha_+ \acute{g}_d^+(w)$ have a non-constant common divisor, (9.21) is established, and then the coefficients of an AGCD $d_{ls}^+(x)$ of $f(x)$ and $g(x)$ are computed by solving the least squares problem. The error measure $\|d_{ls}^+ - \hat{d}\|$ is equal to 0.0030. In addition, the coefficients of an AGCD $d_{qr}^+(x)$ of $f(x)$ and $g(x)$ are computed using the QR decomposition, and the error measure $\|d_{qr}^+ - \hat{d}\|$ is equal to 0.4527.

In the method of SNTLN implemented on $\bar{S}_d(\tilde{f}_d, \alpha_1 \tilde{g}_d) Q_d$, the inexact polynomials $f(x)$ and $g(x)$ are initially normalized to obtain the forms $\breve{f}_d(x)$ and $\breve{g}_d(x)$, which are defined in (7.13) and (7.14) respectively. The polynomials $\breve{f}_d(x)$ and $\breve{g}_d(x)$ are then transformed to their modified Bernstein polynomials $\tilde{f}_d(w)$ and $\tilde{g}_d(w)$ defined in (7.34) and (7.35) respectively, and the dth subresultant matrix $\bar{S}_d(\tilde{f}_d, \alpha_1 \tilde{g}_d) Q_d$ is

computed. The method of SNTLN is then implemented on $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$ and yields the optimal values of α and θ, where $\alpha_* = 3.4840e+001$ and $\theta_* = 2.4940$, and $\tilde{f}_d^*(w)$ and $\tilde{g}_d^*(w)$, which are the corrected forms of $\tilde{f}_d(w)$ and $\tilde{g}_d(w)$.

Figure 9.1(b) shows the normalized singular values of the Sylvester matrix $\bar{S}(\tilde{f}_d^*, \alpha_*\tilde{g}_d^*)Q$ of $\tilde{f}_d^*(w)$ and $\alpha_*\tilde{g}_d^*(w)$, and it is seen that its numerical rank is equal to 30, which implies that the degree of the GCD of the corrected polynomials $\tilde{f}_d^*(w)$ and $\alpha_*\tilde{g}_d^*(w)$ is 17. Therefore, (9.21) is established, and then the coefficients of an AGCD $d_{ls}^*(x)$ of $f(x)$ and $g(x)$ are calculated by solving the least squares problem. Since the error measure $\|d_{ls}^* - \hat{d}\|$ is equal to $1.7322e-005$, it is much smaller than $\|d_{ls}^+ - \hat{d}\|$. This suggests that the coefficients of an AGCD of $f(x)$ and $g(x)$, which are obtained from the method of SNTLN implemented on $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$, are much closer to the coefficients of the GCD of their exact polynomials $\hat{f}(x)$ and $\hat{g}(x)$. Furthermore, the coefficients of an AGCD $d_{qr}^*(x)$ of $f(x)$ and $g(x)$ are computed using the QR decomposition, and the error measure $\|d_{qr}^* - \hat{d}\|$ is equal to 0.4490.

It is noted that the coefficients of an AGCD computed by solving the least squares problem are more accurate than those calculated using the QR decomposition because $\|d_{ls}^+ - \hat{d}\|$ and $\|d_{ls}^* - \hat{d}\|$ are much smaller than $\|d_{qr}^+ - \hat{d}\|$ and $\|d_{qr}^* - \hat{d}\|$ respectively.



(a) (b)

Figure 9.1: The normalized singular values of (a) $\bar{S}(\acute{f}_d^+, \alpha_+ \acute{g}_d^+)$ and (b) $\bar{S}(\tilde{f}_d^*, \alpha_* \tilde{g}_d^*)Q$ for Example 9.3.

□

Example 9.4. Consider the exact Bernstein polynomials $\hat{f}(x)$ and $\hat{g}(x)$, whose roots and multiplicities are specified in Table 9.2. It is seen that the degree of their GCD is $\hat{d} = 22$.

Root of $\hat{f}(x)$	Multiplicity
-0.3285e+000	5
0.3791e+000	6
-0.7113e+000	6
0.9214e+000	6
2.3125e+000	5
9.1474e+000	8

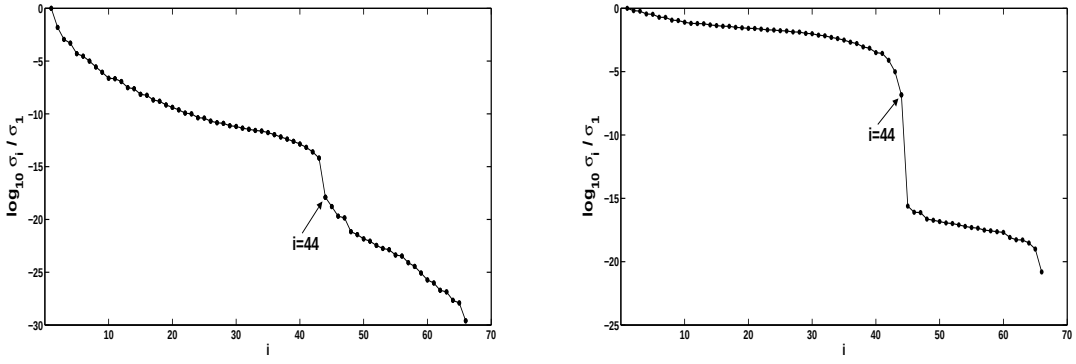
Root of $\hat{g}(x)$	Multiplicity
-0.3285e+000	3
0.3791e+000	7
0.5217e+000	3
0.9214e+000	7
1.4397e+000	3
9.1474e+000	7

Table 9.2: The roots and multiplicities of $\hat{f}(x)$ and $\hat{g}(x)$ for Example 9.4.

Uniformly distributed random noise with componentwise signal-to-noise ratio 10^8 is added to each polynomial to obtain the inexact polynomials $f(x)$ and $g(x)$. The degree of an AGCD, $d = 22$, is determined using the previous methods, which is correct because $d = \hat{d}$.

In the method of SNTLN implemented on $\bar{S}_d(\acute{f}_d, \alpha_2 \acute{g}_d)$, the method of SNTLN yields $\alpha_+ = 4.6807$, $\theta_+ = 1.1281$, and $\acute{f}_d^+(w)$ and $\acute{g}_d^+(w)$, which are the corrected forms of $\acute{f}_d(w)$ and $\acute{g}_d(w)$. The Sylvester matrix $\bar{S}(\acute{f}_d^+, \alpha_+ \acute{g}_d^+)$ of $\acute{f}_d^+(w)$ and $\alpha_+ \acute{g}_d^+(w)$ is computed. Figure 9.2(a) shows the normalized singular values of $\bar{S}(\acute{f}_d^+, \alpha_+ \acute{g}_d^+)$, and it is seen that its rank loss is equal to 23, which implies that the degree of the GCD of the corrected polynomials $\acute{f}_d^+(w)$ and $\alpha_+ \acute{g}_d^+(w)$ is 23. This is incorrect because the estimated degree of an AGCD, d, is equal to 22.

In the method of SNTLN implemented on $\bar{S}_d(\tilde{f}_d, \alpha_1 \tilde{g}_d) Q_d$, $\alpha_* = 1.0123e+001$, $\theta_* = 1.4386$, and $\tilde{f}_d^*(w)$ and $\tilde{g}_d^*(w)$, which are the corrected forms of $\tilde{f}_d(w)$ and $\tilde{g}_d(w)$, are obtained. Figure 9.2(b) shows the normalized singular values of the Sylvester matrix $\bar{S}(\tilde{f}_d^*, \alpha_* \tilde{g}_d^*) Q$ of $\tilde{f}_d^*(w)$ and $\alpha_* \tilde{g}_d^*(w)$, and its numerical rank is clearly defined and equal to 44, which implies that the corrected polynomials $\tilde{f}_d^*(w)$ and $\alpha_* \tilde{g}_d^*(w)$ have the GCD whose degree is equal to d. The coefficients of an AGCD $d_{ls}^*(x)$ of $f(x)$ and $g(x)$ are then calculated by solving the least squares problem, and $\|d_{ls}^* - \hat{d}\|$ is equal to $5.3456e-006$. This relatively small error suggests that the coefficients of $d_{ls}^*(x)$ are close to the coefficients of $\hat{d}(x)$. In addition, the coefficients of an AGCD $d_{qr}^*(x)$ of $f(x)$ and $g(x)$ can also be calculated using the QR decomposition, and $\|d_{qr}^* - \hat{d}\|$ is equal to 0.3099. The error measure $\|d_{ls}^* - \hat{d}\|$ is much smaller than $\|d_{qr}^* - \hat{d}\|$, which implies that the coefficients of an AGCD computed by solving the least squares problem are more accurate than those calculated using the QR decomposition.



(a) (b)

Figure 9.2: The normalized singular values of (a) $\bar{S}(\acute{f}_d^+, \alpha_+ \acute{g}_d^+)$ and (b) $\bar{S}(\tilde{f}_d^*, \alpha_* \tilde{g}_d^*)Q$ for Example 9.4.

□

Example 9.5. Consider the exact Bernstein polynomials $\hat{f}(x)$ and $\hat{g}(x)$, whose roots and multiplicities are specified in Table 9.3. It is seen that the degree of their GCD is $\hat{d} = 26$.

Root of $\hat{f}(x)$	Multiplicity
0.1793e+000	10
0.5615e+000	5
0.7539e+000	9
0.8276e+000	3
1.3741e+000	5
1.4638e+000	4
-3.2719e+000	3

Root of $\hat{g}(x)$	Multiplicity
0.1793e+000	9
0.5615e+000	6
0.7539e+000	8
0.9913e+000	5
-1.2593e+000	4
1.3741e+000	4
2.1298e+000	3

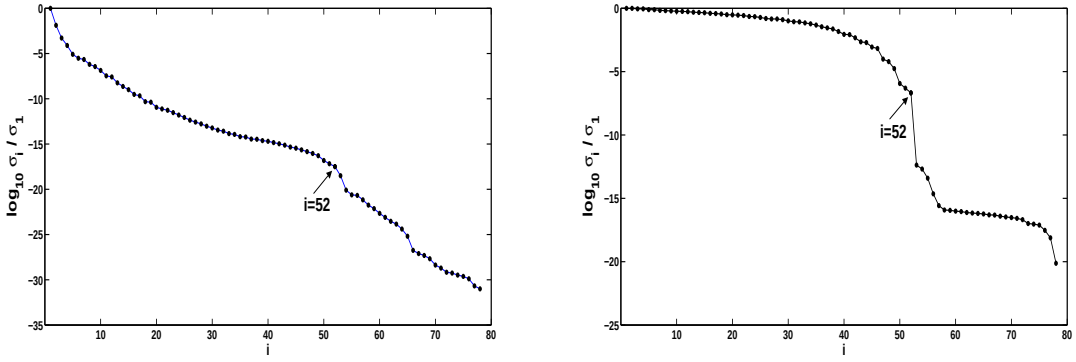
Table 9.3: The roots and multiplicities of $\hat{f}(x)$ and $\hat{g}(x)$ for Example 9.5.

Uniformly distributed random noise with componentwise signal-to-noise ratio 10^8 is added to each polynomial to obtain the inexact polynomials $f(x)$ and $g(x)$. The degree of an AGCD, $d = 26$, is determined using the previous methods, which is

correct because $d = \hat{d}$.

The values of $\alpha_+ = 0.2300$ and $\theta_+ = 1.9080$, and $\acute{f}_d^+(w)$ and $\acute{g}_d^+(w)$ are obtained from the method of SNTLN implemented on $\bar{S}_d(\acute{f}_d, \alpha_2 \acute{g}_d)$. The Sylvester matrix $\bar{S}(\acute{f}_d^+, \alpha_+ \acute{g}_d^+)$ of $\acute{f}_d^+(w)$ and $\alpha_+ \acute{g}_d^+(w)$ is computed. Figure 9.3(a) shows the normalized singular values of $\bar{S}(\acute{f}_d^+, \alpha_+ \acute{g}_d^+)$, and it is seen that its numerical rank is not clearly defined, which implies that the corrected polynomials $\acute{f}_d^+(w)$ and $\alpha_+ \acute{g}_d^+(w)$ are coprime. Therefore, the coefficients of an AGCD $d^+(x)$ can not be computed because $\acute{f}_d^+(w)$ and $\alpha_+ \acute{g}_d^+(w)$ do not have a non-constant common divisor.

The method of SNTLN implemented on $\bar{S}_d(\tilde{f}_d, \alpha_1 \tilde{g}_d) Q_d$ yields $\alpha_* = 2.5249e + 002$, $\theta_* = 1.6090$, $\tilde{f}_d^*(w)$ and $\tilde{g}_d^*(w)$. It is seen from Figure 9.3(b) that the rank loss of the Sylvester matrix $\bar{S}(\tilde{f}_d^*, \alpha_* \tilde{g}_d^*) Q$ of $\tilde{f}_d^*(w)$ and $\alpha_* \tilde{g}_d^*(w)$ is equal to 26, which suggests that the degree of the GCD of the corrected polynomials $\tilde{f}_d^*(w)$ and $\alpha_* \tilde{g}_d^*(w)$ is equal to d. Then, the coefficients of an AGCD $d_{ls}^*(x)$ of $f(x)$ and $g(x)$ are calculated by solving the least squares problem, and the error measure $\|d_{ls}^* - \hat{d}\|$ is equal to $8.6891e{-}007$. This very small error indicates that $d_{ls}^*(x)$, an AGCD of $f(x)$ and $g(x)$, is a good approximation to the GCD of their exact polynomials. Furthermore, the coefficients of an AGCD $d_{qr}^*(x)$ of $f(x)$ and $g(x)$ can also be calculated using the QR decomposition, and $\|d_{qr}^* - \hat{d}\|$ is equal to 1.3852. The coefficients of an AGCD computed by solving the least squares problem are more accurate than those calculated using the QR decomposition because $\|d_{ls}^* - \hat{d}\|$ is much smaller than $\|d_{qr}^* - \hat{d}\|$.



(a) (b)

Figure 9.3: The normalized singular values of (a) $\bar{S}(\acute{f}_d^+, \alpha_+ \acute{g}_d^+)$ and (b) $\bar{S}(\tilde{f}_d^*, \alpha_* \tilde{g}_d^*)Q$ for Example 9.5.

□

The three examples in this section compare the results obtained from $\bar{S}_d(\acute{f}_d, \alpha_2 \acute{g}_d)$ and $\bar{S}_d(\tilde{f}_d, \alpha_1 \tilde{g}_d)Q_d$ using the method of SNTLN, and it is shown that the method of SNTLN implemented on $\bar{S}_d(\tilde{f}_d, \alpha_1 \tilde{g}_d)Q_d$ yields significantly better results. The results shown in these three examples are consistent with other experiment results. In addition, the examples also show that the coefficients of an AGCD computed by solving the least squares problem are more accurate than those calculated using the QR decomposition.

9.3 Discussion

It was shown in Section 9.2 that when the method of SNTLN is implemented on $\bar{S}_d(\acute{f}_d, \alpha_2 \acute{g}_d)$ and $\bar{S}_d(\tilde{f}_d, \alpha_1 \tilde{g}_d)Q_d$ respectively, $\bar{S}_d(\tilde{f}_d, \alpha_1 \tilde{g}_d)Q_d$ always yields better results than $\bar{S}_d(\acute{f}_d, \alpha_2 \acute{g}_d)$. The advantage of $\bar{S}_d(\tilde{f}_d, \alpha_1 \tilde{g}_d)Q_d$ with respect to $\bar{S}_d(\acute{f}_d, \alpha_2 \acute{g}_d)$ is considered below. Furthermore, the coefficients of an AGCD can be computed by

solving the least squares problem and using the QR decomposition, and the examples in Section 9.2 showed that the QR decomposition returns the inferior results. It is therefore necessary to explain this phenomenon. These two issues are addressed in the following respectively.

Section 6.3 discussed the superiority of $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q$ with respect to $\bar{S}(\acute{f}, \alpha_2\acute{g})$ and explained the computational advantage of the inclusion of the diagonal matrix Q. This analysis is for $k = 1$ because $\bar{S}(\acute{f}, \alpha_2\acute{g}) = \bar{S}_1(\acute{f}_1, \alpha_2\acute{g}_1)$ and $\bar{S}(\tilde{f}, \alpha_1\tilde{g})Q = \bar{S}_1(\tilde{f}_1, \alpha_1\tilde{g}_1)Q_1$. The same analysis can be repeated for $\bar{S}_d(\acute{f}_d, \alpha_2\acute{g}_d)$ and $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$, and the diagonal matrix Q_d has the same effect on $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$, that is, the entries of Q_d mitigate the effects of the combinatorial factors $\binom{m}{i}$, $\binom{n}{j}$ and $\binom{m+n-d}{k}$, for large values of m and n, such that computations performed on $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$ are more stable.

The examples in Section 9.2 showed that in the method of SNTLN implemented on $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$, the coefficients of an AGCD are computed by solving the least squares problem of (9.21) and using the QR decomposition applied to the Sylvester matrix $\bar{S}(\tilde{f}_d^*, \alpha_*\tilde{g}_d^*)Q$ of the corrected polynomials $\tilde{f}_d^*(w)$ and $\alpha_*\tilde{g}_d^*(w)$, and the QR decomposition yields the inferior results than solving the least squares problem. The possible explanation is that the corrected polynomials $\tilde{f}_d^*(w)$ and $\alpha_*\tilde{g}_d^*(w)$ have a non-constant GCD, and thus the Sylvester matrix $\bar{S}(\tilde{f}_d^*, \alpha_*\tilde{g}_d^*)Q$ of $\tilde{f}_d^*(w)$ and $\alpha_*\tilde{g}_d^*(w)$ is rank deficient, which implies that several columns are linearly dependent on the other columns in $\bar{S}(\tilde{f}_d^*, \alpha_*\tilde{g}_d^*)Q$. It is shown in [16] that the QR decomposition applied to a matrix with linearly dependent columns is unstable.

9.4 Summary

This chapter considered the use of the method of SNTLN to calculate the coefficients of an AGCD of two inexact polynomials. In particular, if the degree d of an AGCD is determined initially, the method of SNTLN computes structured perturbations, such that the perturbed forms of the inexact polynomials have a non-constant common divisor of degree d.

The method of SNTLN is performed on two forms of the dth subresultant matrix, $\bar{S}_d(\acute{f}_d, \alpha_2\acute{g}_d)$ and $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$. Experiments show that the method of SNTLN implemented on $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$ returns much better results and recovers a good approximation to the GCD of exact polynomials. In addition, experiments also show that the method of SNTLN implemented on $\bar{S}_d(\tilde{f}_d, \alpha_1\tilde{g}_d)Q_d$ converges to the solution only after 4 or 5 iterations.

It was shown in this chapter that the method of SNTLN is efficient in the calculation of the coefficients of an AGCD, and therefore it is desirable to consider its other applications. The next chapter will discuss solving the deconvolution problem using the method of SNTLN.

Chapter 10

Deconvolution

Chapter 9 demonstrates the use of the method of SNTLN in the calculation of the coefficients of an AGCD of two inexact Bernstein polynomials. This chapter considers another application of the method of SNTLN. In particular, the method of SNTLN can be applied to compute an approximate deconvolution of two inexact Bernstein polynomials $h(x)$ and $f(x)$, that is, the division $h(x)/f(x)$ such that the result is a polynomial and not a rational function.

It is assumed that $\hat{f}(x)$ and $\hat{h}(x)$ are two exact Bernstein polynomials, and $\hat{f}(x)$ is an exact divisor of $\hat{h}(x)$. The presence of the random perturbations $\delta f(x)$ and $\delta h(x)$,

$$f(x) = \hat{f}(x) + \delta f(x) \qquad \text{and} \qquad h(x) = \hat{h}(x) + \delta h(x),$$

which implies that, with high probability, $f(x)$ is not an exact divisor of $h(x)$. Therefore, it is required to compute the polynomials $z_f(x)$ and $z_h(x)$ of minimum magnitude such that the function

$$\frac{h(x) + z_h(x)}{f(x) + z_f(x)},$$

is a polynomial, that is, $f(x)+z_f(x)$ is an exact divisor of $h(x)+z_h(x)$. This chapter demonstrates the use of the method of SNTLN for the computation of the polynomials $z_f(x)$ and $z_h(x)$.

10.1 The division of two Bernstein polynomials

Let $\hat{f}(x)$, $\hat{g}(x)$ and $\hat{h}(x)$ be exact Bernstein polynomials of degrees m, n and $m+n$ respectively,

$$\begin{aligned}\hat{f}(x) &= \sum_{i=0}^{m} \hat{a}_i \binom{m}{i} (1-x)^{m-i} x^i,\\ \hat{g}(x) &= \sum_{i=0}^{n} \hat{b}_i \binom{n}{i} (1-x)^{n-i} x^i,\\ \hat{h}(x) &= \sum_{i=0}^{m+n} \hat{c}_i \binom{m+n}{i} (1-x)^{m+n-i} x^i.\end{aligned}$$

It is assumed that $\hat{f}(x)$ is an exact divisor of $\hat{h}(x)$, and $\hat{g}(x)$ is the quotient polynomial, that is, $\hat{g}(x) = \hat{h}(x)/\hat{f}(x)$.

The product of $\hat{f}(x)$ and $\hat{g}(x)$ can be written as

$$\sum_{i=0}^{m}\sum_{j=0}^{n} \hat{a}_i \binom{m}{i} \hat{b}_j \binom{n}{j} (1-x)^{m+n-i-j} x^{i+j},$$

and thus the substitution $k = i + j$ yields

$$\sum_{k=0}^{m+n} \sum_{i=\max(0,k-n)}^{\min(m,k)} \hat{a}_i \binom{m}{i} \hat{b}_{k-i} \binom{n}{k-i} (1-x)^{m+n-k} x^k.$$

Therefore, the product of $\hat{f}(x)$ and $\hat{g}(x)$ equals to

$$\sum_{k=0}^{m+n} \sum_{i=\max(0,k-n)}^{\min(m,k)} \frac{\hat{a}_i \binom{m}{i} \hat{b}_{k-i} \binom{n}{k-i}}{\binom{m+n}{k}} \binom{m+n}{k} (1-x)^{m+n-k} x^k,$$

and thus the coefficients $\hat{c}_k$ of $\hat{h}(x)$ are given by

$$\hat{c}_k = \sum_{i=\max(0,k-n)}^{\min(m,k)} \frac{\hat{a}_i \binom{m}{i} \hat{b}_{k-i} \binom{n}{k-i}}{\binom{m+n}{k}}, \qquad k = 0, \ldots, m+n.$$

This equation can be written in matrix form as

$$\left(\bar{D}^{-1}\bar{T}(\hat{f})\right)\hat{b} = \hat{c}, \tag{10.1}$$

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

$$\bar{D}^{-1} = \operatorname{diag}\begin{bmatrix} \frac{1}{\binom{m+n}{0}} & \frac{1}{\binom{m+n}{1}} & \cdots & \frac{1}{\binom{m+n}{m+n-1}} & \frac{1}{\binom{m+n}{m+n}} \end{bmatrix},$$

$\bar{T}(\hat{f}) \in \mathbb{R}^{(m+n+1)\times(n+1)}$, $\hat{b} \in \mathbb{R}^{n+1}$, $\hat{c} \in \mathbb{R}^{m+n+1}$ and

$$\bar{T}(\hat{f}) = \begin{bmatrix} \hat{a}_0\binom{m}{0} & & \\ \hat{a}_1\binom{m}{1} & \ddots & \\ \vdots & \ddots & \hat{a}_0\binom{m}{0} \\ \hat{a}_m\binom{m}{m} & \ddots & \hat{a}_1\binom{m}{1} \\ & \ddots & \vdots \\ & & \hat{a}_m\binom{m}{m} \end{bmatrix}, \qquad \hat{b} = \begin{bmatrix} \hat{b}_0\binom{n}{0} \\ \hat{b}_1\binom{n}{1} \\ \vdots \\ \hat{b}_n\binom{n}{n} \end{bmatrix}, \qquad \hat{c} = \begin{bmatrix} \hat{c}_0 \\ \hat{c}_1 \\ \vdots \\ \hat{c}_{m+n} \end{bmatrix}.$$

Chapters 6, 7 and 9 have shown it is numerically advantageous to express the vector $\hat{b}$ as the product of a diagonal matrix $\bar{Q} \in \mathbb{R}^{(n+1)\times(n+1)}$ and a vector $\hat{p} \in \mathbb{R}^{n+1}$ of the coefficients $\hat{b}_i$,

$$\hat{b} = \bar{Q}\hat{p},$$

where

$$\bar{Q} = \operatorname{diag}\begin{bmatrix} \binom{n}{0} & \binom{n}{1} & \cdots & \binom{n}{n} \end{bmatrix}, \tag{10.2}$$

and

$$\hat{p} = \begin{bmatrix} \hat{b}_0 & \hat{b}_1 & \cdots & \hat{b}_n \end{bmatrix}^T,$$

and therefore (10.1) can be written as

$$\left(\bar{D}^{-1}\bar{T}(\hat{f})\bar{Q}\right)\hat{p} = \hat{c}. \tag{10.3}$$

Since (10.3) has a more complex form than (10.1), for simplicity, this chapter only considers (10.3). It is necessary to implement one preprocessing operation on (10.3) before the computation is performed on it. This preprocessing operation is considered in the next section.

10.2 Preprocessing operation

It is seen from (10.3) that the coefficients of $\hat{f}(x)$ occupy the entries of the matrix $\bar{D}^{-1}\bar{T}(\hat{f})\bar{Q}$ and the coefficients of $\hat{h}(x)$ occupy the entries of the vector $\hat{c}$ respectively. If the coefficients of $\hat{f}(x)$ are much larger or smaller than the coefficients of $\hat{h}(x)$ in magnitude, this may cause both sides of (10.3) to be unbalanced. Therefore, it is necessary to normalize the entries of the matrix $\bar{D}^{-1}\bar{T}(\hat{f})\bar{Q}$ and the entries of the vector $\hat{c}$ respectively, such that both sides of (10.3) are better balanced.

10.2.1 Normalization

The entries of $\bar{D}^{-1}\bar{T}(\hat{f})\bar{Q}$ are normalized by their geometric mean. The computation of the geometric mean of the entries of $\bar{D}^{-1}\bar{T}(\hat{f})\bar{Q}$ can be easily obtained from the calculation of normalization constants for the Sylvester matrix $S(f,g)Q$ shown in Section 6.1.1.

If the geometric mean of all the terms that contain the coefficients of $\hat{f}(x)$ in $\bar{D}^{-1}\bar{T}(\hat{f})\bar{Q}$

is λ, then it follows that the normalized form of $\hat{f}(x)$ is

$$\ddot{f}(x) = \sum_{i=0}^{m} \ddot{a}_i \binom{m}{i} (1-x)^{m-i} x^i, \qquad \ddot{a}_i = \frac{\hat{a}_i}{\lambda}. \tag{10.4}$$

The entries of the vector $\hat{c}$ are normalized by their geometric mean, and therefore the normalized form of $\hat{h}(x)$ is

$$\ddot{h}(x) = \sum_{i=0}^{m+n} \ddot{c}_i \binom{m+n}{i} (1-x)^{m+n-i} x^i, \qquad \ddot{c}_i = \frac{\hat{c}_i}{\mu}, \tag{10.5}$$

where the geometric mean μ of the coefficients $\hat{c}_i$ is

$$\mu = \left(\prod_{i=0}^{m+n} |\hat{c}_i| \right)^{\frac{1}{m+n+1}}.$$

Therefore, it follows from (10.4) and (10.5) that (10.3) becomes

$$\left(\bar{D}^{-1} \bar{T}(\ddot{f}) \bar{Q} \right) \ddot{p} = \ddot{c}, \tag{10.6}$$

where $\ddot{p} \in \mathbb{R}^{n+1}$ is

$$\ddot{p} = \begin{bmatrix} \ddot{b}_0 & \ddot{b}_1 & \cdots & \ddot{b}_n \end{bmatrix}^T,$$

and the coefficients $\ddot{b}_i$ of the polynomial $\ddot{g}(x)$ are required to be computed,

$$\ddot{g}(x) = \sum_{i=0}^{n} \ddot{b}_i \binom{n}{i} (1-x)^{n-i} x^i. \tag{10.7}$$

10.3 The method of SNTLN

This section considers the method of SNTLN for the computation of the coefficients of $\ddot{g}(x)$, which are the solution of (10.6), when the inexact Bernstein polynomials $f(x)$ and $h(x)$ are specified, which are given by

$$f(x) = \sum_{i=0}^{m} a_i \binom{m}{i} (1-x)^{m-i} x^i, \tag{10.8}$$

and

$$h(x) = \sum_{i=0}^{m+n} c_i \binom{m+n}{i} (1-x)^{m+n-i} x^i. \tag{10.9}$$

It is therefore assumed that $f(x)$ and $h(x)$ have been preprocessed using the operation discussed in Section 10.2.

The inexact nature of $f(x)$ and $h(x)$ implies that $f(x)$ is not an exact divisor of $h(x)$, and thus (10.6) is replaced by

$$\left(\bar{D}^{-1}\bar{T}(\ddot{f})\bar{Q}\right)\ddot{p} \approx \ddot{c}, \tag{10.10}$$

where $\ddot{p} \in \mathbb{R}^{n+1}$ and $\ddot{c} \in \mathbb{R}^{m+n+1}$ are, respectively,

$$\ddot{p} = \begin{bmatrix} \ddot{b}_0 & \ddot{b}_1 & \cdots & \ddot{b}_n \end{bmatrix}^T \qquad \text{and} \qquad \ddot{c} = \begin{bmatrix} \ddot{c}_0 & \ddot{c}_1 & \cdots & \ddot{c}_{m+n} \end{bmatrix}^T. \tag{10.11}$$

The approximation (10.10) is converted to an equation by the addition of a matrix $B = B(z) \in \mathbb{R}^{(m+n+1)\times(n+1)}$ to $\bar{T}(\ddot{f})$, and a vector $d \in \mathbb{R}^{m+n+1}$ to $\ddot{c}$,

$$\left(\bar{D}^{-1}\left(\bar{T}(\ddot{f}) + B(z)\right)\bar{Q}\right)\ddot{p} = \ddot{c} + d. \tag{10.12}$$

The matrix $B(z)$ has the same structure as $\bar{T}(\ddot{f})$, and $B(z)$ and d are given by

$$B(z) = \begin{bmatrix} z_0\binom{m}{0} & & & \\ z_1\binom{m}{1} & \ddots & & \\ \vdots & \ddots & z_0\binom{m}{0} & \\ z_m\binom{m}{m} & \ddots & z_1\binom{m}{1} & \\ & \ddots & \vdots & \\ & & z_m\binom{m}{m} & \end{bmatrix}, \qquad d = \begin{bmatrix} d_0 \\ d_1 \\ \vdots \\ d_{m+n} \end{bmatrix},$$

where the coefficients z_i and d_i of the polynomials

$$s(x) = \sum_{i=0}^{m} z_i \binom{m}{i} (1-x)^{m-i} x^i,$$

and

$$t(x) = \sum_{i=0}^{m+n} d_i \binom{m+n}{i} (1-x)^{m+n-i} x^i,$$

are computed by the method of SNTLN, such that $\ddot{f}(x) + s(x)$ is an exact divisor of $\ddot{h}(x) + t(x)$.

Equation (10.12) is a non-linear equation that is solved by the Newton-Raphson method. In general, it has an infinite number of solutions, but the solution that is nearest the given inexact data is sought. The residual associated with an approximate solution of this non-linear equation is

$$r(z, \ddot{p}, d) = (\ddot{c} + d) - \left(\bar{D}^{-1}\left(\bar{T}(\ddot{f}) + B(z)\right)\bar{Q}\right)\ddot{p}, \tag{10.13}$$

and thus if $\tilde{r}$ is defined as

$$\tilde{r} := r(z + \delta z, \ddot{p} + \delta\ddot{p}, d + \delta d),$$

then

$$\begin{aligned} \tilde{r} &= (\ddot{c} + (d + \delta d)) - \left(\bar{D}^{-1}\big(\bar{T}(\ddot{f}) + B(z + \delta z)\big)\bar{Q}\right)(\ddot{p} + \delta\ddot{p}) \\ &= (\ddot{c} + (d + \delta d)) - \left(\bar{D}^{-1}\Big(\bar{T} + B + \sum_{i=0}^{m} \frac{\partial B}{\partial z_i}\delta z_i\Big)\bar{Q}\right)(\ddot{p} + \delta\ddot{p}). \end{aligned}$$

It follows that to first order

$$\tilde{r} = r(z, \ddot{p}, d) + \delta d - \left(\bar{D}^{-1}(\bar{T} + B)\bar{Q}\right)\delta\ddot{p} - \left(\bar{D}^{-1}\left(\sum_{i=0}^{m} \frac{\partial B}{\partial z_i}\delta z_i\right)\bar{Q}\right)\ddot{p}. \tag{10.14}$$

The simplification of the last term of this expression requires that the polynomial multiplication

$$\left(\sum_{i=0}^{n} \ddot{b}_i \binom{n}{i} (1-x)^{n-i} x^i\right)\left(\sum_{i=0}^{m} z_i \binom{m}{i} (1-x)^{m-i} x^i\right), \tag{10.15}$$

which can also be expressed as

$$\left(\sum_{i=0}^{m} z_i \binom{m}{i} (1-x)^{m-i} x^i\right)\left(\sum_{i=0}^{n} \ddot{b}_i \binom{n}{i} (1-x)^{n-i} x^i\right), \tag{10.16}$$

be considered. It follows from (10.3) that the multiplications (10.15) and (10.16) can be expressed in matrix form as, respectively,

$$\bar{D}^{-1}Y(\ddot{p})(Rz) \qquad \text{and} \qquad \bar{D}^{-1}B(z)(\bar{Q}\ddot{p}), \tag{10.17}$$

where $Y = Y(\ddot{p}) \in \mathbb{R}^{(m+n+1)\times(m+1)}$ is a Toeplitz matrix, and

$$\begin{aligned} z &= \begin{bmatrix} z_0 & z_1 & \cdots & z_m \end{bmatrix}^T \in \mathbb{R}^{m+1}, \\ R &= \operatorname{diag}\begin{bmatrix} \binom{m}{0} & \binom{m}{1} & \cdots & \binom{m}{m} \end{bmatrix} \in \mathbb{R}^{(m+1)\times(m+1)}. \end{aligned}$$

It therefore follows from (10.17) that

$$Y(Rz) = B(\bar{Q}\ddot{p}), \tag{10.18}$$

and the differentiation of both sides of this equation with respect to z yields

$$Y(R\delta z) = \left(\sum_{i=0}^{m} \frac{\partial B}{\partial z_i}\delta z_i\right)(\bar{Q}\ddot{p}),$$

and thus (10.14) simplifies to

$$\tilde{r} = r(z, \ddot{p}, d) + \delta d - \left(\bar{D}^{-1}(\bar{T} + B)\bar{Q}\right)\delta\ddot{p} - (\bar{D}^{-1}YR)\delta z. \tag{10.19}$$

Example 10.1. Let $m = 4$ and $n = 3$, and thus $\bar{D}^{-1} \in \mathbb{R}^{8\times 8}$, $B \in \mathbb{R}^{8\times 4}$, $\bar{Q} \in \mathbb{R}^{4\times 4}$ and $\ddot{p} \in \mathbb{R}^4$. The matrices $\bar{D}^{-1}$ and B, and the vector $\bar{Q}\ddot{p}$ are equal to

$$\bar{D}^{-1} = \operatorname{diag}\begin{bmatrix} \frac{1}{\binom{7}{0}} & \frac{1}{\binom{7}{1}} & \frac{1}{\binom{7}{2}} & \frac{1}{\binom{7}{3}} & \frac{1}{\binom{7}{4}} & \frac{1}{\binom{7}{5}} & \frac{1}{\binom{7}{6}} & \frac{1}{\binom{7}{7}} \end{bmatrix},$$

$$B = \begin{bmatrix} z_0\binom{4}{0} & & & \\ z_1\binom{4}{1} & z_0\binom{4}{0} & & \\ z_2\binom{4}{2} & z_1\binom{4}{1} & z_0\binom{4}{0} & \\ z_3\binom{4}{3} & z_2\binom{4}{2} & z_1\binom{4}{1} & z_0\binom{4}{0} \\ z_4\binom{4}{4} & z_3\binom{4}{3} & z_2\binom{4}{2} & z_1\binom{4}{1} \\ & z_4\binom{4}{4} & z_3\binom{4}{3} & z_2\binom{4}{2} \\ & & z_4\binom{4}{4} & z_3\binom{4}{3} \\ & & & z_4\binom{4}{4} \end{bmatrix} \quad \text{and} \quad \bar{Q}\ddot{p} = \begin{bmatrix} \ddot{b}_0\binom{3}{0} \\ \ddot{b}_1\binom{3}{1} \\ \ddot{b}_2\binom{3}{2} \\ \ddot{b}_3\binom{3}{3} \end{bmatrix},$$

respectively, and thus $\bar{D}^{-1}B(\bar{Q}\ddot{p})$ is the vector of coefficients of the polynomial formed from the multiplication

$$\left(\sum_{i=0}^{4} z_i\binom{4}{i}(1-x)^{4-i}x^i\right)\left(\sum_{i=0}^{3} \ddot{b}_i\binom{3}{i}(1-x)^{3-i}x^i\right).$$

This polynomial multiplication can also be expressed as

$$\left(\sum_{i=0}^{3} \ddot{b}_i\binom{3}{i}(1-x)^{3-i}x^i\right)\left(\sum_{i=0}^{4} z_i\binom{4}{i}(1-x)^{4-i}x^i\right),$$

and thus the vector of coefficients of the product can also be expressed as

$$\bar{D}^{-1}\begin{bmatrix} \ddot{b}_0\binom{3}{0} & & & & \\ \ddot{b}_1\binom{3}{1} & \ddot{b}_0\binom{3}{0} & & & \\ \ddot{b}_2\binom{3}{2} & \ddot{b}_1\binom{3}{1} & \ddot{b}_0\binom{3}{0} & & \\ \ddot{b}_3\binom{3}{3} & \ddot{b}_2\binom{3}{2} & \ddot{b}_1\binom{3}{1} & \ddot{b}_0\binom{3}{0} & \\ & \ddot{b}_3\binom{3}{3} & \ddot{b}_2\binom{3}{2} & \ddot{b}_1\binom{3}{1} & \ddot{b}_0\binom{3}{0} \\ & & \ddot{b}_3\binom{3}{3} & \ddot{b}_2\binom{3}{2} & \ddot{b}_1\binom{3}{1} \\ & & & \ddot{b}_3\binom{3}{3} & \ddot{b}_2\binom{3}{2} \\ & & & & \ddot{b}_3\binom{3}{3} \end{bmatrix}\begin{bmatrix} z_0\binom{4}{0} \\ z_1\binom{4}{1} \\ z_2\binom{4}{2} \\ z_3\binom{4}{3} \\ z_4\binom{4}{4} \end{bmatrix}. \tag{10.20}$$

Since

$$R = \text{diag}\left[\binom{4}{0} \quad \binom{4}{1} \quad \binom{4}{2} \quad \binom{4}{3} \quad \binom{4}{4} \right],$$

it follows that (10.20) can also be written as

$$\bar{D}^{-1}Y(Rz) = \bar{D}^{-1} \begin{bmatrix} \ddot{b}_0\binom{3}{0} & & & & \\ \ddot{b}_1\binom{3}{1} & \ddot{b}_0\binom{3}{0} & & & \\ \ddot{b}_2\binom{3}{2} & \ddot{b}_1\binom{3}{1} & \ddot{b}_0\binom{3}{0} & & \\ \ddot{b}_3\binom{3}{3} & \ddot{b}_2\binom{3}{2} & \ddot{b}_1\binom{3}{1} & \ddot{b}_0\binom{3}{0} & \\ & \ddot{b}_3\binom{3}{3} & \ddot{b}_2\binom{3}{2} & \ddot{b}_1\binom{3}{1} & \ddot{b}_0\binom{3}{0} \\ & & \ddot{b}_3\binom{3}{3} & \ddot{b}_2\binom{3}{2} & \ddot{b}_1\binom{3}{1} \\ & & & \ddot{b}_3\binom{3}{3} & \ddot{b}_2\binom{3}{2} \\ & & & & \ddot{b}_3\binom{3}{3} \end{bmatrix} \times R \begin{bmatrix} z_0 \\ z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix}.$$

It follows from (10.18) that this vector is equal to $\bar{D}^{-1}B(\bar{Q}\ddot{p})$, and it is seen that

$$Y = \begin{bmatrix} \ddot{b}_0\binom{3}{0} & & & & \\ \ddot{b}_1\binom{3}{1} & \ddot{b}_0\binom{3}{0} & & & \\ \ddot{b}_2\binom{3}{2} & \ddot{b}_1\binom{3}{1} & \ddot{b}_0\binom{3}{0} & & \\ \ddot{b}_3\binom{3}{3} & \ddot{b}_2\binom{3}{2} & \ddot{b}_1\binom{3}{1} & \ddot{b}_0\binom{3}{0} & \\ & \ddot{b}_3\binom{3}{3} & \ddot{b}_2\binom{3}{2} & \ddot{b}_1\binom{3}{1} & \ddot{b}_0\binom{3}{0} \\ & & \ddot{b}_3\binom{3}{3} & \ddot{b}_2\binom{3}{2} & \ddot{b}_1\binom{3}{1} \\ & & & \ddot{b}_3\binom{3}{3} & \ddot{b}_2\binom{3}{2} \\ & & & & \ddot{b}_3\binom{3}{3} \end{bmatrix}.$$

□

The jth iteration in the Newton-Raphson method for the calculation of $z, \ddot{p}$ and d is obtained from (10.19),

$$\begin{bmatrix} H_z & H_{\ddot{p}} & H_d \end{bmatrix}^{(j)} \begin{bmatrix} \delta z \\ \delta \ddot{p} \\ \delta d \end{bmatrix}^{(j)} = r^{(j)}, \tag{10.21}$$

where $r^{(j)} = r^{(j)}(z, \ddot{p}, d)$,

$$\begin{aligned} H_z &= \bar{D}^{-1}YR \in \mathbb{R}^{(m+n+1)\times(m+1)}, \\ H_{\ddot{p}} &= \bar{D}^{-1}(\bar{T} + B)\bar{Q} \in \mathbb{R}^{(m+n+1)\times(n+1)}, \\ H_d &= -I \in \mathbb{R}^{(m+n+1)\times(m+n+1)}, \end{aligned}$$

and the matrix I is an identity matrix. The values of $z, \ddot{p}$ and d at the $(j+1)$th

iteration are

$$\begin{bmatrix} z \\ \ddot{p} \\ d \end{bmatrix}^{(j+1)} = \begin{bmatrix} z \\ \ddot{p} \\ d \end{bmatrix}^{(j)} + \begin{bmatrix} \delta z \\ \delta\ddot{p} \\ \delta d \end{bmatrix}^{(j)}.$$

The initial values of z and d are $z^{(0)} = 0$ and $d^{(0)} = 0$ because the given data is inexact, the initial value $\ddot{p}_0$ of $\ddot{p}$ is calculated from (10.13),

$$\ddot{p}_0 = \arg\min_w \|\bar{D}^{-1}\bar{T}(\ddot{f})\bar{Q}w - \ddot{c}\|_2, \tag{10.22}$$

where $\ddot{c}$ is defined in (10.11).

Equation (10.21) is of the form

$$Cy = q, \tag{10.23}$$

where $C \in \mathbb{R}^{(m+n+1)\times(2m+2n+3)}$, $y \in \mathbb{R}^{2m+2n+3}$, $q \in \mathbb{R}^{m+n+1}$, and

$$C = \begin{bmatrix} H_z & H_{\ddot{p}} & H_d \end{bmatrix}^{(j)}, \qquad y = \begin{bmatrix} \delta z \\ \delta\ddot{p} \\ \delta d \end{bmatrix}^{(j)}, \qquad q = r^{(j)}. \tag{10.24}$$

It is necessary to calculate the vector y with minimum magnitude that satisfies (10.23), that is, the solution that is closest to the given inexact data is required. The objective function is

$$\left\| \begin{bmatrix} z^{(j+1)} - z^{(0)} \\ \ddot{p}^{(j+1)} - \ddot{p}_0 \\ d^{(j+1)} - d^{(0)} \end{bmatrix} \right\| = \left\| \begin{bmatrix} z^{(j)} + \delta z^{(j)} \\ \ddot{p}^{(j)} + \delta\ddot{p}^{(j)} - \ddot{p}_0 \\ d^{(j)} + \delta d^{(j)} \end{bmatrix} \right\| := ||Ey - h||, \tag{10.25}$$

where

$$E = I_{2m+2n+3}, \qquad h = -\begin{bmatrix} z^{(j)} \\ \ddot{p}^{(j)} - \ddot{p}_0 \\ d^{(j)} \end{bmatrix}, \tag{10.26}$$

and y is defined in (10.24). It is noted that E is constant and not updated between iterations.

The minimization of (10.25) subject to (10.23) is a least squares minimization with an equality constraint (the LSE problem),

$$\min_y ||Ey - h|| \qquad \text{subject to} \qquad Cy = q,$$

which can be solved by the QR decomposition [30]. This LSE problem is the same type of problem considered in Chapter 9, and it is solved at each iteration, where C, q and h are updated between successive iterations.

Algorithm 10.1: Deconvolution of two Bernstein polynomials

Input Inexact Bernstein polynomials $f(x)$ and $h(x)$, which are of degrees m and $m+n$ respectively.

Output The polynomial $g(x) = h(x)/f(x)$.

Begin

1. Process $f(x)$ and $h(x)$ to yield $\ddot{f}(x)$ and $\ddot{h}(x)$, which are defined in (10.4) and (10.5) respectively, using the preprocessing operation described in Section 10.2.

2. % Initialize the data

   - Calculate the diagonal matrices $\bar{D}^{-1}$ and $\bar{Q}$.
   - Set $z = z^{(0)} = 0$, which yields $B = 0$, and $d = d^{(0)} = 0$.
   - Calculate $\bar{T}, Y$ and the initial value $\ddot{p}_0$ of $\ddot{p}$, which is defined in (10.22). Calculate the initial value of q,

$$q^{(0)} = r^{(0)},$$

where $r^{(0)}$ is equal to the initial value of the residual,

$$\begin{aligned} r^{(0)} &= r(z^{(0)} = 0, \ddot{p}_0, d^{(0)} = 0) \\ &= \ddot{c} - \left(\bar{D}^{-1}\bar{T}(\ddot{f})\bar{Q}\right)\ddot{p}_0. \end{aligned}$$

- Define the matrices C and E.

3. % The loop for the iterations

   % Solve the LSE problem at each iteration using the QR decomposition

   repeat

   (a) Compute the QR decomposition of C^T,

   $$C^T = QR = Q \begin{bmatrix} R_1 \\ 0 \end{bmatrix}.$$

   (b) Set $w_1 = R_1^{-T} q$.

   (c) Partition EQ as

   $$EQ = \begin{bmatrix} E_1 & E_2 \end{bmatrix},$$

   where

   $E_1 \in \mathbb{R}^{(2m+2n+3)\times(m+n+1)}$, $\qquad E_2 \in \mathbb{R}^{(2m+2n+3)\times(m+n+2)}$.

   (d) Compute

   $$z_1 = E_2^{\dagger}(h - E_1 w_1),$$

   where h is defined in (10.26).

   (e) Compute the solution

   $$y = Q \begin{bmatrix} w_1 \\ z_1 \end{bmatrix}.$$

(f) Set $z := z + \delta z$, $\ddot{p} := \ddot{p} + \delta\ddot{p}$ and $d := d + \delta d$.

(g) Update B and Y, and therefore C from z, $\ddot{p}$ and d. Compute the residual

$$r(z, \ddot{p}, d) = (\ddot{c} + d) - \bar{D}^{-1}(\bar{T} + B)\bar{Q}\ddot{p},$$

and thus update $q = r(z, \ddot{p}, d)$. Update h from z, $\ddot{p}$ and d.

until $\frac{||r(z,\ddot{p},d)||}{||\ddot{c}+d||} \leq 10^{-12}$

End

Algorithm 10.1 terminates when the residual $\frac{||r(z,\ddot{p},d)||}{||\ddot{c}+d||}$ is sufficiently small and it yields z^*, p^* and d^*. The vector z^*

$$z^* = \begin{bmatrix} z_0^* & z_1^* & \cdots & z_m^* \end{bmatrix}^T \in \mathbb{R}^{m+1},$$

is the perturbation vector for the coefficients of the polynomial $\ddot{f}(x)$, and the vector d^*

$$d^* = \begin{bmatrix} d_0^* & d_1^* & \cdots & d_{m+n}^* \end{bmatrix}^T \in \mathbb{R}^{m+n+1},$$

is the perturbation vector for the coefficients of the polynomial $\ddot{h}(x)$, such that the perturbed form of $\ddot{f}(x)$ is an exact divisor of the perturbed form of $\ddot{h}(x)$. The corrected forms of $\ddot{f}(x)$ and $\ddot{h}(x)$ are therefore given by

$$\begin{aligned} f^*(x) &= \sum_{i=0}^{m} a_i^* \binom{m}{i} (1-x)^{m-i} x^i \\ &= \sum_{i=0}^{m} (\ddot{a}_i + z_i^*) \binom{m}{i} (1-x)^{m-i} x^i, \end{aligned}$$

and

$$\begin{aligned} h^*(x) &= \sum_{i=0}^{m+n} c_i^* \binom{m+n}{i} (1-x)^{m+n-i} x^i \\ &= \sum_{i=0}^{m+n} (\ddot{c}_i + d_i^*) \binom{m+n}{i} (1-x)^{m+n-i} x^i, \end{aligned}$$

respectively. The quotient polynomial $g^*(x)$ is obtained from the vector p^*,

$$p^* = \begin{bmatrix} b_0^* & b_1^* & \cdots & b_n^* \end{bmatrix}^T \in \mathbb{R}^{n+1},$$

that is

$$g^*(x) = \sum_{i=0}^{n} b_i^* \binom{n}{i} (1-x)^{n-i} x^i.$$

10.4 Examples

This section shows the results obtained from the method of SNTLN implemented on (10.10), which are compared with the results obtained using the method of least squares. The method of least squares is now described.

Consider the inexact Bernstein polynomials $f(x)$ and $h(x)$, which are defined in (10.8) and (10.9) respectively. They are preprocessed using the operation described in Section 10.2 to yield $\ddot{f}(x)$ and $\ddot{h}(x)$, which are defined in (10.4) and (10.5) respectively. The inexact nature of $\ddot{f}(x)$ and $\ddot{h}(x)$ implies that $\ddot{f}(x)$ is not an exact divisor of $\ddot{h}(x)$, and thus the approximation (10.10) is established. The approximate solution $\ddot{p}$ is then computed using the method of least squares, that is

$$\ddot{p} \approx \left(\bar{D}^{-1} \bar{T}(\ddot{f}) \bar{Q} \right)^{\dagger} \ddot{c}, \tag{10.27}$$

and the coefficients of the quotient polynomial $\ddot{g}(x)$ are obtained from $\ddot{p}$.

The criteria are required to be established in order to compare the method of SNTLN and the method of least squares. The following criteria are developed for the method

of SNTLN, but they are also applied to the method of least squares.

The method of SNTLN yields the corrected forms $f^*(x)$, $g^*(x)$ and $h^*(x)$. If $f^*(x)$ is an exact divisor of $h^*(x)$ and $g^*(x)$ is the quotient polynomial, then

$$f^*(x)g^*(x) = h^*(x),$$

should be satisfied, which is easily checked by computing

$$\frac{\|f^*g^* - h^*\|}{\|h^*\|}.$$

Furthermore, the coefficients of the quotient polynomial $g^*(x)$ are compared with the coefficients of the exact quotient polynomial $\hat{g}(x)$, which can be achieved by computing the error between the normalized coefficients of $g^*(x)$ and the normalized coefficients of $\hat{g}(x)$, that is

$$\|g^* - \hat{g}\|,$$

where $\|g^*\| = \|\hat{g}\| = 1$ and $\|\cdot\|$ denotes the 2-norm. These criteria are also applied when the method of least squares is used to solve the deconvolution problem.

Example 10.2. Consider the exact Bernstein polynomials $\hat{f}(x)$ and $\hat{h}(x)$, whose roots and multiplicities are specified in Table 10.1.

Root of $\hat{f}(x)$	Multiplicity
0.4327e+000	5
0.5479e+000	6
1.0000e+003	5
-1.2147e+000	2
7.3125e+000	8

Root of $\hat{h}(x)$	Multiplicity
0.4327e+000	6
0.5479e+000	7
1.0000e+003	8
-1.2147e+000	3
1.2793e-004	5
7.3125e+000	9

Table 10.1: The roots and multiplicities of $\hat{f}(x)$ and $\hat{h}(x)$ for Example 10.2.

It is shown in Table 10.1 that $\hat{f}(x)$ is an exact divisor of $\hat{h}(x)$, and thus the exact

quotient polynomial $\hat{g}(x)$ is easily computed.

Noise with componentwise signal-to-noise ratio 10^4 is added to the coefficients of $\hat{f}(x)$ and $\hat{h}(x)$ to obtain their inexact forms $f(x)$ and $h(x)$.

In the method of least squares, the approximation (10.10) is established, and it follows from (10.27) that the approximate solution $\ddot{p}$ of (10.10) is computed. The coefficients of the quotient polynomial $\ddot{g}(x)$ are obtained from $\ddot{p}$. The error measure $\|\ddot{g} - \hat{g}\|$ is equal to $1.0089e-004$ and $\frac{\|\ddot{f}\ddot{g}-\ddot{h}\|}{\|\ddot{h}\|}$ is equal to $5.1646e-008$.

In the method of SNTLN, the method of SNTLN performed on (10.10) yields the vectors z^*, p^* and d^*. The perturbation vectors z^* and d^* allow the corrected forms of $\ddot{f}(x)$ and $\ddot{h}(x)$, $f^*(x)$ and $h^*(x)$, to be obtained. The coefficients of the quotient polynomial $g^*(x)$ are obtained from the vector p^*. The error measure $\frac{\|f^*g^*-h^*\|}{\|h^*\|}$ is equal to $2.8713e-017$ and $\|g^* - \hat{g}\|$ is equal to $7.8794e-005$. Compared with the results obtained from the method of least squares, the relatively small error between $f^*(x)g^*(x)$ and $h^*(x)$ indicates that $f^*(x)g^*(x) = h^*(x)$ is more exactly satisfied. □

Example 10.3. Consider the exact Bernstein polynomials $\hat{f}(x)$ and $\hat{h}(x)$, whose roots and multiplicities are specified in Table 10.2.

Root of $\hat{f}(x)$	Multiplicity
0.3178e+000	6
0.4431e+000	4
0.5979e+000	4
0.6129e+000	5
0.7189e+000	4

Root of $\hat{h}(x)$	Multiplicity
0.3178e+000	8
0.4431e+000	6
0.5979e+000	6
0.6129e+000	6
0.7189e+000	6
0.8251e+000	3
0.9134e+000	4
0.9998e+000	4

Table 10.2: The roots and multiplicities of $\hat{f}(x)$ and $\hat{h}(x)$ for Example 10.3.

It follows from Table 10.2 that $\hat{f}(x)$ is an exact divisor of $\hat{h}(x)$, and thus the exact quotient polynomial $\hat{g}(x)$ is easily computed.

The addition of noise with componentwise signal-to-noise ratio 10^4 to the coefficients of $\hat{f}(x)$ and $\hat{h}(x)$ yields their inexact forms $f(x)$ and $h(x)$.

In the method of least squares, the coefficients of the quotient polynomial $\ddot{g}(x)$ are obtained from the approximate solution $\ddot{p}$ of the approximation (10.10), which is computed using (10.27). The error measure $\|\ddot{g} - \hat{g}\|$ is equal to 0.1105 and $\frac{\|\ddot{f}\ddot{g}-\ddot{h}\|}{\|\ddot{h}\|}$ is equal to $2.4953e-005$.

In the method of SNTLN, the method of SNTLN implemented on (10.10) yields the vectors z^*, p^* and d^*. The corrected forms $f^*(x)$, $g^*(x)$ and $h^*(x)$ are then obtained. The error measure $\frac{\|f^*g^*-h^*\|}{\|h^*\|}$ is equal to $2.6666e-013$ and $\|g^* - \hat{g}\|$ is equal to 0.1096. Compared with the results obtained from the method of least squares, the significantly smaller error between $f^*(x)g^*(x)$ and $h^*(x)$ means that $f^*(x)g^*(x) = h^*(x)$ is more precisely satisfied. □

10.5 Summary

This chapter considered the use of the method of SNTLN to solve the approximate deconvolution of two inexact Bernstein polynomials $f(x)$ and $h(x)$. It has been shown that the method is effective in computing the perturbations applied to the coefficients of $f(x)$ and $h(x)$, such that the perturbed form of $f(x)$ is an exact divisor of the perturbed form of $h(x)$. The typical examples shown in Section 10.4 demonstrate that the method of SNTLN yields significantly better results than the method of least squares. Furthermore, experiments also show that the method of SNTLN converges to the solution only after 4 or 5 iterations.

Chapter 11

Conclusion and future work

This thesis considered the application of structure preserving matrix methods for some ill-posed operations on Bernstein polynomials. In particular, the operations of greatest common divisor computations and polynomial division were considered.

Three algorithms to compute the GCD of Bernstein polynomials, Euclid's algorithm, and operations on the Bézout and Sylvester resultant matrices were introduced. It was shown in Chapter 3 that when exact polynomials are specified, these three algorithms provide an unambiguous and correct result in a symbolic computing environment. However, when the GCD computation is performed in a floating point environment and the polynomials are inexact because of added noise, these algorithms fail to calculate the GCD of polynomials because noise makes the inexact forms of polynomials coprime, and therefore the computation of the GCD becomes an ill-posed problem. Thus, an AGCD of inexact polynomials must be considered. Different definitions of an AGCD may be specified for different problems. In this thesis, the degree of an AGCD of two inexact polynomials is defined to be correct when it is equal to the degree of the GCD of their exact forms because this reproduces in the given noisy

polynomials a property of their theoretically exact forms.

The computation of an AGCD of inexact polynomials requires the degree of an AGCD to be determined initially. This can be achieved by calculating the normalized singular values of the Bézout and Sylvester resultant matrices when the preprocessing operations are implemented on them. In particular, since the Sylvester matrix has a partitioned structure, three preprocessing operations are implemented on the Sylvester matrix, which are the normalization of the polynomials, the introduction of a parameter α, and a transformation of the independent variable x to a new independent variable w. However, due to the bilinear nature of the Bézout matrix, only the third preprocessing operation, a transformation of the independent variable x to a new independent variable w, is required to be implemented for the Bézout matrix. Experiments show that these preprocessing operations allow the improved and correct determination of the degree of an AGCD to be obtained. In addition, it is noted that compared with the conventional form of Sylvester matrix, its modified form obtained by post-multiplying its conventional form with a diagonal matrix yields significantly better results. The importance of the inclusion of this diagonal matrix is discussed in Chapter 6.

Furthermore, the degree of an AGCD can also be determined using the first principal angle and the residual of an approximate linear algebraic equation, and these methods involve Sylvester subresultant matrices. In particular, for each subresultant matrix, the three preprocessing operations mentioned above are required to be implemented, its optimal column is then selected using the criteria based on the first principal angle and the residual. For each subresultant matrix, the first principal angle and the residual between its optimal column and its remaining matrix after the removal of

optimal column are recorded. The degree of an AGCD is determined by observing the maximum change of the first principal angle or the residual.

After the degree of an AGCD is determined, the perturbations of minimum magnitude applied to the coefficients of the inexact polynomials are calculated using the method of SNTLN, such that the perturbed forms of the inexact polynomials have a non-constant common divisor of the determined degree. Experiments demonstrate that few iterations are required for the method of SNTLN to converge to a solution, and similarly, the subresultant matrices with the inclusion of diagonal matrices recover a much better approximation to the coefficients of the GCD. In addition, it has also been shown in this thesis that the method of SNTLN can be used to solve the deconvolution problem of inexact polynomials.

This thesis has shown that structured matrix methods allow excellent computational results to be obtained to ill-posed problems in which the coefficients of Bernstein polynomials are corrupted by noise. It is therefore appropriate to apply them to some practical problems.

The method introduced in this thesis has shown that reliable results can be obtained from ill-posed operations on univariate Bernstein polynomials. It is therefore desirable to consider applying this method to bivariate and trivariate Bernstein polynomials. Since the size of resultant matrices of bivariate and trivariate Bernstein polynomials is much larger, it is necessary to consider computationally efficient algorithms. This includes the calculation of the displacement rank of resultant matrices. In addition, there exists an important difference between univariate polynomials, and bivariate and trivariate polynomials. In particular, a univariate polynomial of degree d has exactly d linear factors, but a bivariate polynomial and a trivariate polynomial of

degree $d > 1$ may not have d factors. For example, $x^2 + y + 1 = 0$ does not have any linear factors. Furthermore, it should be noted that the Sylvester matrix of multivariate polynomials is rectangular, not square. These issues have not been addressed and will be considered in the future work.

This method can also be used to solve some practical problems in CAGD. For example, the intersection points of Bézier curves are frequently considered in CAGD. Since the Bézier curve is represented by Bernstein polynomials, the computation of the intersection points of Bézier curves is reduced to calculating the common roots of Bernstein polynomials. The operations considered in this thesis are required for the robust solution of these intersection problems.

Finally, it is shown in [65] that a polynomial root solver that is explicitly designed for the computation of multiple roots of a polynomial requires the computation of the GCD of a polynomial and its derivative. This thesis has presented a reliable method to compute the GCD of Bernstein polynomials, and thus the computation of multiple roots of Bernstein polynomial using the algorithm in [65] is practical. In addition, the algorithm in [65] also involves the division of Bernstein polynomials, which has been addressed in this work.

Bibliography

[1] A. V. Aho, J. E. Hopcroft, and J. D. Ullman. *The Design and Analysis of Computer Algorithms*. Addison-Wesley, 1974.

[2] V. B. Anand. *Computer Graphics and Geometric Modeling for Engineers*. John Wiley and Sons, Inc, 1993.

[3] S. Barnett. *Polynomials and Linear Control Systems*. Marcel Dekker, New York, USA, 1983.

[4] B. Beckermanna and G. Labahn. A fast and numerically stable Euclidean-like algorithm for detecting relatively prime numerical polynomials. *Journal of Symbolic Computation*, 26(6):691–714, 1998.

[5] D. A. Bini and L. Gemignani. Bernstein-Bezoutian matrices. *Theoretical Computer Science*, 315:319–333, 2004.

[6] D. A. Bini and A. Marco. Computing curve intersection by means of simultaneous iterations. *Numerical Algorithms*, 43:151–175, 2006.

[7] B. Buchberger. Gröbner bases: An algorithm method in polynomial ideal theory. In *Multidimensional Systems Theory(N.K.Boze ed.)*, chapter 6, pages 184–232. Reidel Publishing Company, Dodrecht - Boston - Lancaster, 1985.

[8] J. F. Canny. *The Complexity of Robot Motion Planning*. The MIT Press, 1988.

[9] P. Chin, R. M. Corless, and G. F. Corliss. Optimization strategies for the approximate GCD problem. In *ISSAC' 98 Proceedings of the 1998 international symposium on Symbolic and algebraic computation*, pages 228–235. ACM Press, 1998.

[10] A. Clebsch. *J. für Math*, 64:43–65, 1865.

[11] R. M. Corless, P. M. Gianni, B. M. Trager, and S. M. Watt. The singular value decomposition for polynomial systems. In *Proceedings of the* 1995 *International Symposium on Symbolic and Algebraic Computation*, pages 195–207. ACM Press, 1995.

[12] R. M. Corless, S. M. Watt, and L. Zhi. Qr factoring to compute the GCD of univariate approximate polynomials. *IEEE transactions on signal processing*, 52(12):3394–3402, 2004.

[13] P. de Casteljau. Courbes et surfaces *à* pôles. Technical report, A. Citroën, Paris, 1963.

[14] A. L. Dixon. The eliminant of three quantics in two independent variables. *Proc. London Math Soc*, 6:49–69, 1908.

[15] I. Emiris, A. Galligo, and H. Lombardi. Certified approximate univariate GCDs. *Journal of Pure and Applied Algebra*, 117/118:229–251, 1997.

[16] László Erdős. Linear algebra for MATH2601 Numerical methods, August 2000.

[17] G. Farin and D. Hansford. *The Essentials of CAGD*. A. K. Peters, Ltd, 2000.

[18] G. Farin, J. Hoschek, and M.-S. Kim. *Handbook of Computer Aided Geometric Design*. North Holland, 2002.

[19] Gerald Farin. *Curves and Surfaces for Computer Aided Geometric Design*. Academic press, INC, 1988.

[20] R. T. Farouki and T. N. T. Goodman. On the optimal stability of the Bernstein basis. *Mathematics of Computation*, 65(216):1553–1566, 1996.

[21] R. T. Farouki and V. T. Rajan. On the numerical condition of polynomials in Bernstein form. *Computer Aided Geometric Design*, 4(3):191–216, 1987.

[22] R. T. Farouki and V. T. Rajan. Algorithms for polynomials in Bernstein form. *Computer Aided Geometric Design*, 5:1–26, 1988.

[23] O. Faugeras. *Three-dimensional computer vision: a geometric viewpoint*. The MIT Press, 1993.

[24] D. R. Fulkerson and P. Wolfe. An algorithm for scaling matrices. *SIAM Review*, 4:142–146, 1962.

[25] J. Gallier. *Curves and Surfaces in Geometric Modeling: Theory and Algorithms*. Morgan Kaufmann publishers, 1999.

[26] S. Gao, E. Kaltofen, J. May, Z. Yang, and L. Zhi. Approximate factorization of multivariate polynomials via differential equations. In *Proceedings of the* 2004 *International Symposium on Symbolic and Algebraic Computation*, pages 167–174. ACM Press, 2004.

[27] C. B. Garcia and W. I. Zangwill. Finding all solutions to polynomial systems and other systems of equations. *Mathematical Programming*, 16:159–176, 1979.

[28] R. Goldman. *Pyramid Algorithms*. Morgan Kaufmann publishers, 2002.

[29] R. N. Goldman, T. W. Sederberg, and D. C. Anderson. Vector elimination: A technique for the implicitization, inversion, and intersection of planar parametric rational polynomial curves. *Computer Aided Geometric Design*, 1:327–356, 1984.

[30] G. H. Golub and C. F. Van Loan. *Matrix computations*. John Hopkins University Press, Baltimore, USA, 1996.

[31] J. Hoschek and D. Lasser. *Computer Aided Geometric Design*. A K Peters, 1993.

[32] V. Hribernig and H. J. Stetter. Detection and validation of clusters of polynomial zeros. *Journal of Symbolic Computation*, 24(6):667–681, 1997.

[33] N. K. Karmarkar and Y. N. Lakshman. Approximate polynomial greatest common divisors and nearest singular polynomials. In *ISSAC' 96 Proceedings of the 1996 international symposium on Symbolic and algebraic computation*, pages 35–42. ACM Press, 1996.

[34] P. A. Koparkar and S. P. Mudur. A new class of algorithms for the processing of parametric curves. *Computer-Aided Design*, 15(1):41–45, 1983.

[35] J. M. Lane and R. F. Riesenfeld. A theoretical development for the computer generation and display of piecewise polynomial surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(1):35–46, 1980.

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

[37] B. Liang and S. Pillai. Blind image deconvolution using a robust 2-D GCD approach. *IEEE International Symposium on Circuits and Systems*, pages 1185–1188, 1997.

[38] B. Liang and S. Pillai. Blind image deconvolution using a robust GCD approach. *IEEE Transactions on Image Processing*, 8(2):295–301, 1999.

[39] M.-T. Noda and T. Sasaki. Approximate GCD and its application to ill-conditioned equations. *Journal of Computational and Applied Mathematics*, 38:335–351, 1991.

[40] S. Petitjean. Algebraic geometry and computer vision: Polynomial systems, real and complex roots. *Journal of Mathematical Imaging and Vision*, 10:191–220, 1999.

[41] J. Ben Rosen, H. Park, and J. Glick. Total least norm formulation and solution for structured problems. 17(1):110–128, 1996.

[42] D. Rupprecht. An algorithm for computing certified approximate GCD of n univariate polynomials. *Journal of Pure and Applied Algebra*, 139:255–284, 1999.

[43] A. Schönhage. Quasi-GCD computations. *Journal of Complexity*, 1:118–137, 1985.

[44] T. Sederberg and G. Chang. Best linear common divisors for approximate degree reduction. *Computer Aided Design*, 25:163–168, 1993.

[45] T. W. Sederberg. Algorithm for algebraic curve intersection. *Computer-Aided Design*, 21(9):547–555, 1989.

[46] T. W. Sederberg. Applications to computer aided geometric design. In *Proceedings of Symposia in Applied Mathematics*, volume 53, pages 67–89, 1997.

[47] T. W. Sederberg, D. C. Anderson, and R. N. Goldman. Implicit representation of parametric curves and surfaces. *Computer Vision,Graphics and Image Processing*, 28:72–84, 1984.

[48] T. W. Sederberg and S. R. Parry. Comparison of three curve intersection algorithms.

[49] J. M. Snyder. Interval analysis for computer graphics. In *Proceedings of ACM Siggraph*, pages 121–130, 1992.

[50] E. Staerk. *Mehrfach differenzierbare Bézierkurven und Bézierflächen*. PhD thesis, T. U. Braunschweig, 1976.

[51] P. Stoica and T. Söderström. Common factor detection and estimation. *Automatica*, 33(5):985–989, 1997.

[52] D. Sun and L. Zhi. Structured low rank approximation of a Bezout matrix. *Mathematics in Computer Science*, 1:427–437, 2007.

[53] Yi-Feng Tsai and Rida T. Farouki. Algorithm 812: Bpoly: An object-oriented library of numerical algorithms for polynomials in Bernstein form. *Computer Aided Design*, 27(2):267–296, 2001.

[54] J. V. Uspensky. *Theory of Equations*. McGraw-Hill, New York, USA, 1948.

[55] R. J. Walker. *Algebraic Curves*. Springer-Verlag, 1978.

[56] D. S. Watkins. *Fundamentals of Matrix Computations*. John Wiley and Sons, New York, USA, 1991.

[57] J. R. Winkler. A resultant matrix for scaled Bernstein polynomials. *Linear Algebra and its Applications*, 319:179–191, 2000.

[58] J. R. Winkler. A unified approach to resultant matrices for Bernstein basis polynomials. *Computer Aided Geometric Design*, 25:529–541, 2008.

[59] J. R. Winkler and J. D. Allan. Structured low rank approximations of the Sylvester resultant matrix for approximate GCDs of Bernstein basis polynomials. *Electronic Transactions on Numerical Analysis*, 31:141–155, 2008.

[60] J. R. Winkler and J. D. Allan. Structured total least norm and approximate GCDs of inexact polynomials. *Journal of Computational and Applied Mathematics*, 215:1–13, 2008.

[61] J. R. Winkler and R. N. Goldman. The Sylvester resultant matrix for Bernstein polynomials. In M. Mazure T. Lyche and L. L. Schumaker, editors, *Curve and Surface Design: Saint-Malo 2002*, pages 407–416. Nashboro Press, 2003.

[62] J. R. Winkler and 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*, 234:3226–3242, 2010.

[63] J. R. Winkler, M. Hasan, and X. Y. Lao. Two methods for the calculation of the degree of an approximate greatest common divisor of two inexact polynomials. *Calcolo*, 49(4):241–267, 2012.

[64] J. R. Winkler and X. Y. Lao. The calculation of the degree of an approximate greatest common divisor of two polynomials. *Journal of Computational and Applied Mathematics*, 235(6):1587–1603, 2011.

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

[66] C. K. Yap. Complete subdivision algorithms, i: intersection of Bezier curves. In *Proceedings of the twenty-second annual symposium on Computational geometry*, pages 217–226, 2006.

[67] C. J. Zarowski, X. Ma, and F. W. Fairman. QR-factorization method for computing the greatest common divisor of polynomials with inexact coefficients. *IEEE transactions on signal processing*, 48(11):3042–3051, 2000.

[68] Z. Zeng. The approximate GCD of inexact polynomials. Part 1: A univariate algorithm. Preprint. 2004.

[69] Z. Zeng. Computing multiple roots of inexact polynomials. *Mathematics of Computation*, 74:869–903, 2005.