The Faber polynomials for annular sectors and an application to the iterative solution of linear systems of equations

Myers, Nicholas John

The Faber Polynomials for Annular Sectors and an Application to the Iterative Solution of Linear Systems of Equations.

by

Nicholas John Myers

A thesis submitted in partial fulfilment
of the requirements for the degree of Doctor of Philosophy.

Department of Mathematical Sciences,
University of Durham,
Durham. DH1 3LE.
England.

September, 1994

To Paula, Mum and Dad. Thanks for the support.

Abstract

A conformal mapping of the exterior of the unit circle to the exterior of a region of the complex plane determines the Faber polynomials for that region. These polynomials are of interest in providing near-optimal polynomial approximations in a wide variety of contexts. The work of this thesis concerns the Faber polynomials for an annular sector $\{z : R \leq |z| \leq 1, \theta \leq |\arg z| \leq \pi\}$, with $0 < \theta \leq \pi$ and is contained in two main parts. In the first part the required conformal map is derived, and the first few Faber polynomials for the annular sector are given in terms of the transfinite diameter, ρ, of the region and two parameters a and b. These three numbers are determined numerically. We also give the Faber series for $1/z$ and improve upon a bound given in the literature for the norm of the Faber projection, $\|\chi_n\|$. In the second part of the thesis we give a new hybrid method for the iterative solution of linear systems of equations, $A\mathbf{x} = \mathbf{b}$, where the coefficient matrix, A, is large, sparse, nonsingular and non-Hermitian. The method begins with a few steps of the Arnoldi method to produce some information on the location of the spectrum of A. Our method then switches to an iterative method based on the Faber polynomials for an annular sector placed around these eigenvalue estimates. An annular sector is thought to be a useful region because it can be scaled and rotated to enclose any eigenvalue estimates bounded away from zero. Some examples will be exhibited and we will compare existing methods with ours.

Preface

The work in this thesis is based on research carried out, at Durham University, between October 1991 and October 1994. The material given has not been submitted previously for any other degree in this or any other University.

Chapters Two and Three mainly contain work given in a joint paper by my supervisor, Dr. John Coleman, and myself. The paper is to be published in Mathematics of Computation [12]. Chapter Five contains work based on a paper which was entered in the student paper competition at the Colorado Conference on Iterative Methods in April 1994 (see Myers [58]). Chapters One and Four contain reviews and no claim of originality is made for either of these chapters. Finally, in Chapter Six we give the conclusions and some ideas for future research.

I am extremely grateful to my supervisor, Dr. John Coleman, for his help and constant words of encouragement. In fact without his belief in me none of this would have been possible. I am also grateful to Dr. R. A. Smith, for helping to develop the conformal map given in Chapter Two, and the Engineering and Physical Sciences Research Council (previously the Science and Engineering Research Council), for the sponsorship I have received.

I would like to thank all those who have made the last three years in Durham so enjoyable. In particular, Wil McGhee and Thomas Moorhouse for putting up with me in an office for three years; David Bull, Richard Costambeys, Richard Hall, Ulrich Harder, David Hind, Alex Iskandar and Michael Young, to mention but a few; and James Blowey, Iain MacPhee, Andrew Perella and David Wooff for the squash and badminton that gave my mind a much needed rest from the mathematics. Finally, I would like to thank Mrs. Josephine Coleman for her kind words of encouragement.

Statement of Copyright

Contents

Chapter 1

Faber Polynomials

1.1 Introduction

In this opening chapter we will describe the Faber polynomials for a general region in the complex plane. We will also review most of the important properties of Faber polynomials (see Section 1.2.2). In Section 1.2.3 we will review the ideas of best and near-best approximation. Finally, in Section 1.2.4 we give some examples of regions in the complex plane where the Faber polynomials are known analytically.

1.2 A Review of Faber Polynomials

In 1903 Georg Faber published his solution to a classical complex approximation problem. The problem was to find, for a given region, a set of polynomials $\{p_n(z)\}_{n=0}^{\infty}$ such that an analytic function $f(z)$ could be expanded as a convergent series

$$\sum_{j=0}^{\infty} a_j p_j(z)$$

where the a_j depend on f and the region, but the $p_j(z)$ depend only on the region. His solutions are named after him and called Faber polynomials. They play an important role in complex approximation theory and have been used to provide both polynomial and rational approximations in a wide variety of different contexts. For example, near-minimax polynomial approximations may be obtained by truncating Faber series (Elliott [28], Ellacott [22], Chiu et al. [8] and Coleman and Myers [12]) and by economisation of Faber series (Ellacott and Gutknecht [25]). Faber polynomials have been used to produce approximations to the solution of ordinary differential equations by both the Lanczos τ-method (Coleman [9] and [10]) and by Clenshaw's method (Ellacott and Saff [27]). Rational approximations based on the Faber series were discussed in Ellacott [24], and Ellacott and Saff [26]. The Faber polynomials have also been used in applications in linear algebra, in particular they have been used to produce iterative methods. For example Starke and Varga [76] give a hybrid iterative method using the Faber polynomials for a polygonal region placed around some eigenvalue estimates bounded away from zero. Other examples in the applications of Faber polynomials to iterative methods include the idea of an asymptotically optimal semi-iterative method (see Eiermann [17] and Eiermann et al. [20] for the details).

1.2.1 The Faber Polynomials for a General Region

In this section we will describe how the Faber polynomials are defined for a general compact region, D, whose complement $(\mathbb{C}/D)$ is simply connected in the extended complex plane. For such a region D the Riemann mapping theorem tells us that there is a unique function ϕ such that

$$\lim_{z\to\infty} \frac{\phi(z)}{z} = 1 \tag{1.1}$$

and such that ϕ maps the complement of D conformally onto $\{w : |w| > \rho\}$, the complement of a closed disc of radius ρ (see Figure 1.1). The number ρ is called the transfinite diameter, or logarithmic capacity of D. The function ϕ has a Laurent expansion

$$\phi(z) = z + \alpha_0 + \frac{\alpha_1}{z} + \dots \tag{1.2}$$

about the point at infinity. To obtain $F_n(z)$, the Faber polynomial of degree n, we simply take the polynomial part of the Laurent expansion of $[\phi(z)]^n$. The difficult part in determining the Faber polynomials, for a region D, is to determine the above conformal map ϕ (or its inverse ψ). For this reason the Faber polynomials are known explicitly for only a few types of domain.

1.2.2 Some Properties of Faber Polynomials

From now on we will define ψ as the required conformal map from the complement of the unit disc, $\Delta = \{w : |w| \leq 1\}$, to the complement of the region D. We also define the transfinite diameter of D as,

$$\rho := \lim_{w\to\infty} \frac{\psi(w)}{w}. \tag{1.3}$$

Hence, the Laurent expansion of $\psi(w)$ about the point at infinity can be written in the form

$$z = \psi(w) = \rho\left(w + \beta_0 + \frac{\beta_1}{w} + \dots\right) \tag{1.4}$$

The required conformal mapping from the complement of D to the complement of $\{w : |w| \leq \rho\}$ is given by

$$\phi(z) = \rho\psi^{-1}(z),$$

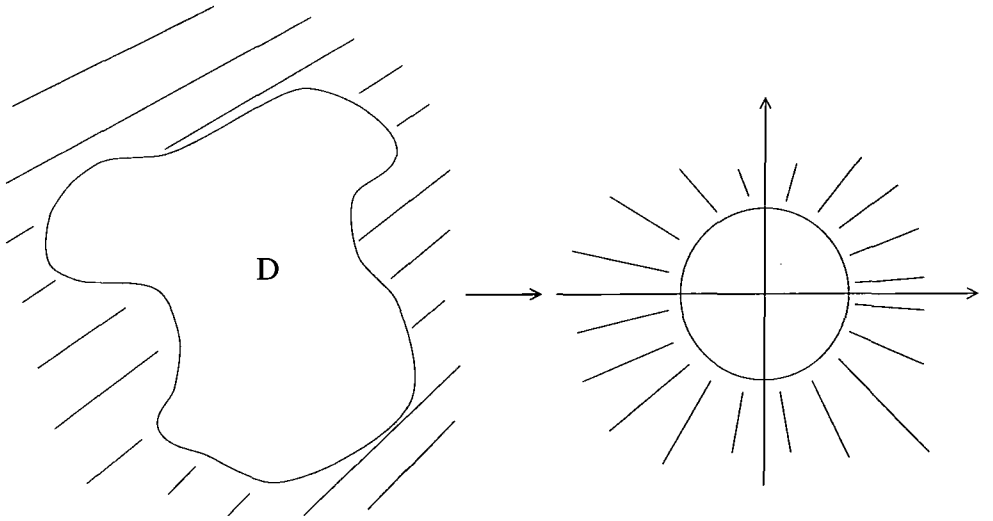

Figure 1.1: The mapping of a general region in the complex plane

where ψ^{-1} is the inverse mapping of ψ. We note that

$$\lim_{z\to\infty}\frac{\phi(z)}{z}=\lim_{z\to\infty}\frac{\rho\psi^{-1}(z)}{z}=\lim_{w\to\infty}\frac{\rho w}{\psi(w)}=1,$$

so that $\phi(z)$ is indeed the desired mapping. We will now describe some of the properties of the Faber polynomials for a general region D, when the mappings ϕ and ψ are defined as above.

From the above definition of the required conformal map ϕ, the following property is satisfied.

PROPERTY 1

$$\phi(\psi(w))^n=\rho^n w^n.$$

Previously we defined the Faber polynomial of degree n as the polynomial part of $(\phi(z))^n$, so that

$$\phi(z)^n=F_n(z)+H_n(z) \tag{1.5}$$

where $H_n(z)=O(1/|z|)$ as $|z|\to\infty$. Using Property 1 we find

$$F_n\left(\psi(w)\right)=\rho^n w^n-H_n\left(\psi(w)\right)$$

and $H_n(\psi(w)) = O(1/|w|)$ as $|w| \to \infty$, so that

PROPERTY 2

$$\frac{1}{2\pi i}\int_{|w|=1} \frac{F_m(\psi(w))}{w^{n+1}}dw = \begin{cases} \rho^n & \text{if } m = n \\ 0 & \text{if } m \neq n. \end{cases}$$

In the next property we need the concept of a level curve, so we define the level curve

$$\Gamma_R := \{z = \psi(w) : |w| = R > 1\}.$$

PROPERTY 3

$$F_n(z) = \frac{1}{2\pi i}\int_{\Gamma_R} \frac{\phi(\xi)^n}{\xi - z}d\xi = \frac{1}{2\pi i}\int_{|w|=R} \frac{\rho^n w^n \psi'(w)}{\psi(w) - z}dw$$

The first part of this property follows from (1.5), the residue theorem and the fact, $H_n(z) = O(1/|z|)$ as $|z| \to \infty$. The second part follows by a substitution of $\xi = \psi(w)$ in the first integral.

We now introduce the scaled Faber polynomials,

$$\tilde{F}_n(z) = \frac{F_n(z)}{\rho^n}, \quad n \geq 1. \tag{1.6}$$

Using Property 3 we are able to find a generating function for the scaled Faber polynomials and from these generate the actual Faber polynomials. This generating function for the scaled Faber polynomials is given by

PROPERTY 4

$$\frac{w\psi'(w)}{\psi(w) - z} = 1 + \sum_{j=1}^{\infty} \frac{\tilde{F}_j(z)}{w^j}.$$

To prove Property 4 we simply substitute this expression in Property 3 and use Cauchy's residue theorem to give $F_n(z) = \rho^n \tilde{F}_n(z)$.

Multiplying both sides of the expression in Property 4 by $\psi(w) - z$, then using equation (1.4) and equating coefficients of w^{-j} leads us to an important recurrence relation for the Faber polynomials, namely

PROPERTY 5

$$\begin{aligned} F_0(z) &= 1, \\ F_1(z) &= z - b_0, \\ F_2(z) &= (z - b_0)F_1(z) - 2b_1, \\ F_{n+1}(z) &= (z - b_0)F_n(z) - \sum_{k=1}^{n-1} b_k F_{n-k}(z) - (n+1)b_n, \quad \text{for } n \geq 2, \end{aligned}$$

where $b_j = \rho^{j+1}\beta_j$.

The next property of Faber polynomials is an important one. It appears in many articles in the literature (see Gaier [38], Ellacott [22], Elliott [28], Kövari and Pommerenke [52] and Pommerenke [65]) in a variety of different notations.

PROPERTY 6

If we define $v(t,s) = \arg(\psi(e^{it}) - \psi(e^{is}))$ then

(a) $$\int_0^{2\pi} |\, d_t v(t,s)| \leq V$$

(b) $$\widetilde{F}_j\left(\psi(e^{is})\right) = \frac{1}{\pi}\int_0^{2\pi} e^{itj}\, d_t v(t,s) \quad \text{for} \quad j \geq 1$$

$$\widetilde{F}_0\left(\psi(e^{is})\right) = \frac{1}{2\pi}\int_0^{2\pi} d_t v(t,s).$$

and V is the total rotation of the region D (see Section 3.3.3).

Proof

(a) Gaier [38], amongst others, mentions that this is proved in Radon [67].

(b) The first part of this result is given in Gaier [38]. It follows from differentiating, with respect to w, a result given in Pommerenke [65], namely

$$\log \frac{\psi(w) - \psi\left(e^{is}\right)}{\rho w} = \frac{1}{\pi}\int_0^{2\pi} \log\left(1 - \frac{e^{it}}{w}\right) d_t v(t,s),$$

and then using Property 4. The second part follows from $\widetilde{F}_0(z) = 1$. □

In Property 4 we gave a generating function for the Faber polynomials. If we divide the expression in Property 4 by w and then integrate with respect to w, we can derive another generating function for the scaled Faber polynomials.

PROPERTY 7

$$\log\left[\frac{\psi(w)-z}{\rho w}\right] = -\sum_{j=1}^{\infty}\frac{\tilde{F}_j(z)}{jw^j}.$$

The next property we shall describe is one that appeared in a paper by Ellacott and Saff [27]. The proof we shall give is one by Dr. J. P. Coleman [11].

PROPERTY 8

$$\begin{aligned}
\int F_0(z) &= F_1(z) + \text{ constant},\\
\int F_1(z) &= \frac{F_2(z)}{2} + \text{ constant},\\
\int F_n(z) &= \frac{F_{n+1}(z)}{n+1} - \sum_{m=1}^{n-1}\frac{n-m}{m}b_{n-m}F_m(z) + \text{ constant} \qquad \text{for } n \geq 2.
\end{aligned}$$

Proof

Using Property 7 we let

$$U(w,z) = \log\left[\frac{\psi(w)-z}{\rho w}\right] = -\sum_{j=1}^{\infty}\frac{\tilde{F}_j(z)}{jw^j}$$

Then we consider partial derivatives of U with respect to both w and z giving

$$\frac{\partial U}{\partial w} = \frac{\psi'(w)}{\psi(w)-z} - \frac{\rho}{\rho w} = \sum_{j=1}^{\infty}\frac{\tilde{F}_j(z)}{w^{j+1}}$$

and

$$\frac{\partial U}{\partial z} = \frac{-1}{\psi(w)-z} = -\sum_{j=1}^{\infty}\frac{\tilde{F}_j'(z)}{jw^j}.$$

So

$$\frac{\partial U}{\partial w} = -\psi'(w)\frac{\partial U}{\partial z} - \frac{1}{w}.$$

Using the previous three equations and equation (1.4) we find

$$\sum_{j=1}^{\infty}\frac{\tilde{F}_j(z)}{w^{j+1}} + \frac{1}{w} = \rho\left[1 - \sum_{k=1}^{\infty}k\beta_k w^{-(k+1)}\right]\left[\sum_{j=1}^{\infty}\frac{\tilde{F}_j'(z)}{jw^j}\right].$$

Finally equating powers of w^{-i} in this equation, integrating the resulting expressions and using equation (1.6) gives us the result. □

Eiermann and Varga [21] show that the zeros and local extreme points of Faber polynomials are the eigenvalues of certain matrices. Allowing for the different definition of the inverse mapping we shall now give their results. We start from Property 5 and rearrange it as

$$zF_n(z) = F_{n+1}(z) + \sum_{k=0}^{n} b_k F_{n-k}(z) + nb_n.$$

We then consider this equation for $n = 0, \ldots, m-1$ and write this system of equations in matrix-vector form as

$$z[F_0(z), F_1(z), \ldots, F_{m-1}(z)] = [F_0(z), F_1(z), \ldots, F_{m-1}(z)]F_m + [0, \ldots, 0, F_m(z)],$$

where F_m is the $m \times m$ principal submatrix of

$$F := \begin{pmatrix} b_0 & 2b_1 & 3b_2 & \cdots \\ 1 & b_0 & b_1 & \cdots \\ & 1 & b_0 & \ddots \\ & & \ddots & \ddots \end{pmatrix}$$

The next property follows immediately.

PROPERTY 9

$\lambda \in \mathbb{C}$ is a zero of $F_m(z)$ if and only if λ is an eigenvalue of F_m with corresponding left eigenvector $[F_0(\lambda), \ldots, F_{m-1}(\lambda)]$.

The last property that we shall describe in this section is one about the coefficients of the Faber polynomials (see Ellacott [22] or Coleman and Smith [13]). If we write

$$F_n(z) = \sum_{k=0}^{n} c_k^{(n)} z^k$$

then the coefficients satisfy

PROPERTY 10

$$c_k^{(n)} = \frac{\rho^n}{2\pi i} \int_{|w|=R} \frac{w^n (\, d\psi / \, dw)}{[\psi(w)]^{k+1}} \, dw.$$

Proof

From equation (1.5) and the residue theorem we find

$$c_k^{(n)} = \frac{1}{2\pi i}\int_{\Gamma_R} \frac{[\phi(z)]^n}{z^{k+1}}\,dz,$$

where Γ_R is the level curve introduced before Property 3. In this expression we make the substitution $z = \psi(w)$ and use Property 1 to find the result. □

Property 10 provides an alternative to generating the Faber polynomials by a recurrence relation (see Property 5). It allows the coefficients of the Faber polynomials to be calculated efficiently by the trapezium rule. If we write $w = R\exp[i\theta]$ in the above expression, then the integrand becomes periodic and we are integrating over a period therefore the integral, and hence the coefficients, can be calculated efficiently by the trapezium rule. The trapezium rule idea approximates the coefficients by

$$c_k^{(n)} \approx \frac{\rho^n}{N}\sum_{m=0}^{N-1} G_k\left(R\exp[i\theta_m]\right)$$

where

$$G_k(w) = \frac{\psi'(w)w^{n+1}}{\psi(w)^{k+1}},$$

$\theta_m = 2\pi m/N$ and N is the number of equal sub-intervals used.

1.2.3 Best and Near-Best Approximation

As the title of this section suggests we shall overview the ideas of both best approximation and near-best approximation. Both Kövari and Pommerenke [52] and Elliott [28] have shown, for practical values of the degree of the polynomial, that the truncated Faber series gives a near-best polynomial approximation. The ideas of Cheney and Price [7] and of Geddes and Mason [43] allow the concept of near-best approximation to be made more precise (see below). Geddes ([41] and [42]) developed this theory for the complex plane by considering the disc and the region bounded by an ellipse. He considers projections from the space of functions that are continuous on the boundary of the region and analytic on its interior onto P_n, the space of polynomials of degree less than or equal to n. For the disc he considers a projection given by truncating the Taylor series of the function; for a region

bounded by an ellipse he considers a similar projection given by truncating the Chebyshev series of the function.

In his thesis Elliott [28] generalised the results of Geddes([41] and [42]), to a general region D in the complex plane, by considering the partial sum of the Faber series of a given function. Also in this thesis, Elliott [28] gives a bound for the error in approximating a function by the n–th partial sum of the Faber expansion; a slight error in the bound was corrected by Ellacott [22].

We will now introduce the notation of Ellacott [22]. We denote by $A(D)$ the space of functions that are continuous at every point in D, and analytic in the interior of D. We also let $\hat{A}(D)$ denote the set of functions belonging to $A(D)$ that are analytic on the boundary of the domain. As above we let P_n denote the space of complex polynomials of degree less than or equal to n.

If we are given $f \in A(D)$, it is well known that there exists a unique best minimax approximation to f from P_n, that is, there exists $p_n \in P_n$ such that

$$\rho_n(f, D) := \|f - p_n\|_\infty \leq \|f - p\|_\infty \qquad \text{for all } p \in P_n,$$

where

$$\|g\|_\infty = \max_{z \in D} |g(z)|$$

and $g \in A(D)$.

In this section we will consider approximations to f which are truncated Faber series, that is we will consider a map from $A(D)$ to P_n, given by truncating the Faber series after $n+1$ terms. This map is a special case of a family of maps from $A(D)$ to P_n known as the projections. A projection $B_n : A(D) \to P_n$ is a bounded, idempotent linear operator. For a projection we have

$$f - B_n(f) = f - p_n + p_n - B_n(f) = f - p_n + B_n(p_n - f),$$

so that

$$\|f - B_n(f)\|_\infty \leq \|f - p_n\|_\infty + \|B_n(p_n - f)\|_\infty$$

$$\leq \|f - p_n\|_\infty + \left\| \frac{B_n(p_n - f)}{\|p_n - f\|_\infty} \right\|_\infty \|p_n - f\|_\infty$$

$$\leq \|f - p_n\|_\infty + \|B_n\| \, \|f - p_n\|_\infty$$

where

$$\|B_n\| = \sup_{g \in A(D) \,:\, \|g\|_\infty = 1} \|B_n(g)\|_\infty = \sup_{g \in A(D) \,:\, \|g\|_\infty = 1} \max_{z \in D} |B_n(g)(z)| .$$

This means, for a projection,

$$\|f - B_n\|_\infty \leq (1 + \|B_n\|) \, \|f - p_n\|_\infty .$$

Following Geddes and Mason [43] we call an approximation, $M(f)$, near-best within a relative distance τ if

$$\|f - M(f)\|_\infty \leq (1 + \tau)\rho_n(f, D),$$

where $\rho_n(f, D)$ is, as we defined earlier, the maximum error on D of the best approximation. From the previous expression we see a projection is near-best within a relative distance $\|B_n\|$. It is usual to require $\tau < 9$ so that not more than one decimal place is lost when replacing the best approximation by a near-best approximation. For this reason in later sections we require the norm of the Faber projection to be less than nine (see below and Section 3.4). Projections are not the only way of producing near-best approximations. For example, in the context of ordinary differential equations, Coleman ([9] and [10]) produces near-best approximations by the Lanczos-τ method, whereas Ellacott and Saff [27] find them by Clenshaw's method.

Given any $f \in A(D)$, the Faber series or Faber expansion is an expression of the form

$$\sum_{k=0}^{\infty} a_k F_k(z),$$

where the a_k are defined by

$$a_k = \frac{1}{2\pi i \rho^k} \int_{|w|=R_1} \frac{f\,(\psi(w))}{w^{k+1}} \, dw, \tag{1.7}$$

ρ is the transfinite diameter of the region and $R_1 > 1$ is sufficiently small that f can be extended analytically to the closed region bounded by the image under ψ of the circle $|w| = R_1$ (see Curtiss [15], Markushevich [57] or Gaier [38]).

As previously mentioned, we shall consider the Faber projection, namely

$$\chi_n \ : \ A(D) \to P_n,$$

given by truncating the Faber series after $n+1$ terms, that is

$$\chi_n(f) = \sum_{j=0}^{n} a_j F_j(z) \tag{1.8}$$

and a_j is given by equation (1.7).

Kövari and Pommerenke [52] give two bounds for the error incurred by approximating $f \in A(D)$ by its truncated Faber series $\chi_n(f)$. Their Theorem 3 states

Given any region D whose complement is connected and any function $f \in A(D)$ then

$$\|f - \chi_n(f)\|_\infty \leq An^\alpha \rho_n(f, D),$$

where A and $\alpha < 1/2$ are absolute constants.

The second result (their Theorem 4) says,

Given any region D whose complement is connected and any function $f \in A(D)$ then

$$\|f - \chi_n(f)\|_\infty \leq (A \log n + B)\rho_n(f, D)$$

and the constants A and B will only depend on the domain D.

Elliott [28] provided quantitative values for A and B, which were then corrected in 1983 by Ellacott [22]. The result from Ellacott is given in terms of a bound on $\|\chi_n\|$,

Let D be a Jordan region whose boundary Γ is of total rotation V. We have

$$\|\chi_n\| \leq \frac{V}{\pi}\left\{\frac{4}{\pi^2}\log n + B\right\} \qquad n \geq 1,$$

where B is a certain absolute constant which (from numerical values computed in Geddes and Mason [43]) has the value 1.773 to 3 decimal places.

The total rotation V of a region will be defined in Section 3.3.3. These results are useful because, for a convex curve ($V = 2\pi$) and practical values of n, they show the approximations are near-best within a relative distance 9. In fact when $V = 2\pi$ Ellacott's bound

leads to $\|\chi_n\| < 9$ for $n \leq 835$. This bound may not be as useful when V is much larger than 2π. For example, when $V = 3\pi$ the bound gives $\|\chi_n\| < 9$ for $n \leq 20$, and for $V = 4\pi$ the bound implies $\|\chi_n\| < 9$ for $n \leq 3$. It would be interesting to improve on Ellacott's bound, or at least to show when $V = 4\pi$ that $\|\chi_n\| < 9$ for reasonable values of n. This will be considered in Section 3.4.

1.2.4 Some Examples of Faber Polynomials

We now give some examples of regions in the complex plane where the Faber polynomials are known analytically. We will also exhibit some of the Faber polynomials of each region.

(a) The unit disc

For this region the mapping is given by $w = \phi(z) = z$; the transfinite diameter, ρ, is 1; and the Faber polynomials are given by $F_n(z) = z^n$. For this region the Faber series for a function f is simply its Taylor series. If better approximations, of a certain degree, to f are required on the unit disc then the disc must be subdivided and approximations to f sought on each sub–region. This lead Coleman and Smith [13] to consider the Faber polynomials for an obvious subdivision of the unit disc, namely the sector (see *(d)* below).

(b) The real interval $[a, b]$.

For this region the inverse mapping is a composition of two conformal maps. The first is the Joukowski function, $t = (w + 1/w)/2$, which maps the complement of the unit disc to the complement of the interval $[-1, 1]$. The second map is a linear transformation from $\mathbb{C}/[-1, 1]$ to $\mathbb{C}/[a, b]$. Hence, the inverse mapping is given by

$$z = \psi(w) = \frac{1}{4}(b - a)\left(w + \frac{1}{w}\right) + \frac{1}{2}(b + a).$$

We also find the transfinite diameter $\rho = \lim_{z\to\infty} \psi(w)/w = \frac{1}{4}(b - a)$. For this interval, the Faber polynomials are multiples of the Chebyshev polynomials for the interval $[-1, 1]$. In fact when $a = -1$ and $b = 1$ we have,

$$F_n(z) = \frac{T_n(z)}{2^{n-1}}, \qquad \text{for } n \geq 1, \quad F_0(z) = T_0(z).$$

For this reason Faber polynomials can be thought of as a way of extending the concept of

Chebyshev polynomials to other regions in the complex plane. The Faber polynomials for an ellipse with foci $\{-1, 1\}$ are also multiples of the Chebyshev polynomials.

(c) The circular arc

The third example we shall consider is that of the circular arc, which was first considered by Ellacott [23]. The exterior of the unit disc is mapped onto the exterior of the circular arc with end points $e^{\pm 2i\beta}$, containing the point $z = -1$, where $\cos(\beta) = 1/a$, by

$$z = \psi(w) = \frac{w(w-a)}{(aw-1)}, \quad \text{with} \quad \rho = \lim_{w\to\infty} \frac{\psi(w)}{w} = \frac{1}{a}.$$

If we now expand the inverse mapping in powers of w we can show that the coefficients defined in equation (1.4) are given by

$$\beta_i = \rho^{i-1}(\rho^2 - 1), \quad i \geq 0$$

so using Property 5 the first few Faber polynomials are

$$\begin{aligned} F_0(z) &= 1, \\ F_1(z) &= z + (1-\rho^2), \\ F_2(z) &= z^2 + 2(1-\rho^2)z + 1 - \rho^4, \\ F_3(z) &= z^3 + 3(1-\rho^2)z^2 + 3(1-\rho^2)z + 1 - \rho^6. \end{aligned}$$

(d) The sector of the disc

Elliott in his thesis describes how the mapping for the unit semi–disc can be derived. Then in 1987, Coleman and Smith [13], described how to find the inverse mapping and hence the Faber polynomials for a sector of the unit disc,

$$S_\alpha = \{z : |z| \leq 1, |\arg z| \leq \frac{\pi}{\alpha}\} \quad \text{with} \quad \alpha > 1.$$

In their paper they show that the unit disc $\{w : |w| \leq 1\}$ is mapped conformally onto the complement of the circular sector, S_α, by

$$\psi(w) = \frac{-\left[u + (u^2-1)^{\frac{1}{2}}\right]^2}{\left[v + (v^2-1)^{\frac{1}{2}}\right]^{p(a)}}$$

where

$$u(w) = \frac{i(w-1)}{2aw^{\frac{1}{2}}}, \quad v(w) = \frac{(2-a^2)(w-1)^2 + 4wa^2}{a^2(w+1)^2}$$

with

$$a = \alpha^{-1}(2\alpha - 1)^{\frac{1}{2}} \quad \text{and} \quad p(a) = (1-a^2)^{\frac{1}{2}}.$$

They found that the transfinite diameter of S_α is given by

$$\rho = \frac{\alpha^2}{(2\alpha-1)^{2-\frac{1}{\alpha}}}.$$

Coleman and Smith [13] also derive an extremely neat way of generating the coefficients, β_i, in the Laurent expansion of the inverse mapping,

$$\psi(w) = \rho\left(w + \beta_0 + \frac{\beta_1}{w} + \ldots\right).$$

They show that

$$(k+1)\beta_k = a_{k+1} - \sum_{\nu=0}^{k-1} \nu\beta_\nu a_{k-\nu},$$

where

$$a_k = P_k(x) + P_{k-1}(x)$$

and $P_n(x)$ is the Legendre polynomial of degree n and is generated from the following recurrence relation

$$(n+1)P_{n+1}(x) - (2n+1)xP_n(x) + nP_{n-1}(x) = 0, \text{ for } n \geq 1,$$

with

$$P_{-1}(x) = 0, \quad P_0(x) = 1.$$

This enables them to generate the Faber polynomials from the recurrence relation (Property 5). Coleman and Smith [13] find the coefficients of the Faber polynomial of degree n ($c_k^{(n)}$ for $k = 0, 1, \ldots, n$) from Property 10 which is given in Ellacott [22] and Gäier [38], and they were then able to compute the coefficients efficiently using the trapezium rule (see after Property 10). Properties of the Faber polynomials gave them ways of assessing the accuracy of computed coefficients, and in particular they mentioned that $c_k^{(n)}\rho^{k-n}$ was a rational number. Gatermann et al. [39], modified the above method by considering the scaled Faber polynomials

$$\tilde{F}_n(z) = \frac{F_n(z)}{\rho^n}.$$

They were able to show that the coefficients were rational, and that the scaled Faber polynomials could be generated algebraically in terms of a parameter, c, depending on the half angle of the sector (π/α).

The first few scaled Faber polynomials are given by,

$$\widetilde{F}_n(z) = \Phi_n\left(\frac{z}{\rho}\right),$$

where

$$\begin{aligned}
\Phi_0(z) &= 1 \\
\Phi_1(z) &= z - 2(1-c) \\
\Phi_2(z) &= z^2 - 4(1-c)z + (1-c)(2+2c) \\
\Phi_3(z) &= z^3 + (1-c)\left[-6z^2 + (9-3c)z - 2 - 4c^2\right]
\end{aligned}$$

and

$$c = \frac{1}{\alpha}\left(2 - \frac{1}{\alpha}\right).$$

(e) Other regions

As was mentioned in Section 1.2.1 the conformal mapping, and therefore the Faber polynomials, are not known for many regions in the complex plane. In this final part of the section we will review most, if not all, of the remaining regions for which the conformal mapping is known. Of course, we may generate the Faber polynomials for any region whose conformal map is known, but the regions we give in this section were considered so the Faber polynomials could be determined.

In his thesis Elliott [28] gives the mapping from the complement of the unit disc onto the complement of the square $\{z : |\mathrm{Re}(z)| \leq 1, |\mathrm{Im}(z)| \leq 1\}$. The mapping is a Schwarz–Christoffel map and is given by

$$z = D\int_1^w \left(1 + \frac{1}{w^4}\right)^{\frac{1}{2}} dw = D\left(w - \frac{1}{6w^3} + \frac{1}{8w^7} - \ldots\right),$$

where $D = 1.18034060$. Following on from this, Ellacott [22] describes the mapping for the rectangle $\{z : |\mathrm{Re}(z)| \leq A, |\mathrm{Im}(z)| \leq B\}$.

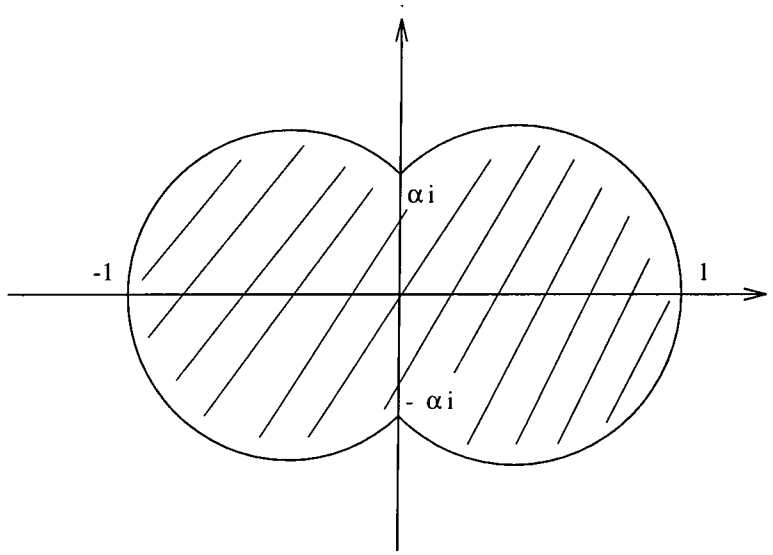

Figure 1.2: The complicated domain from Elliott's thesis.

Elliott also gives the mapping for a quite complicated region, that is, $|z \pm \alpha| = \alpha\sqrt{2}$, where $\alpha = \sqrt{2} - 1$ (see Figure 1.2). The mapping from the complement of this region onto $|w| \geq 2$ is given by

$$w = \phi(z) = z - \frac{\alpha^2}{z}.$$

The mapping for a lemniscate of the form

$$|z^k + A_{k-1}z^{k-1} + \ldots + A_0| = \rho^k,$$

is given in Markushevich [57].

Eiermann and Varga [21] give the mapping from $|w| > 1$ onto the complement of a hypocycloidal domain. The mapping is given by

$$\psi(w) = \alpha w + \beta w^{1-p},$$

where p is an integer greater than 1, $\alpha > 0$, $\beta \in \mathbb{C}$ and $\beta \neq 0$. The mapping is conformal in the region exterior to the unit disc if and only if $(p-1)|\beta|/\alpha \leq 1$.

In a recent paper Bartolomeo and He [4] study the Faber polynomials for a regular m-star, that is,

$$S_m = \{xw^k;\ 0 \leq x \leq 4^{\frac{1}{m}}, k = 0, 1, \ldots, m-1,\ w^m = 1\},$$

with $m \geq 2$. When $m = 2$, S_2 is the interval $[-2, 2]$ (see Section 1.2.4 example (b)). The mapping

$$z = w\left(1 + \frac{1}{w^m}\right)^{\frac{2}{m}}$$

maps $|w| > 1$ conformally onto the complement of S_m.

Finally we note that in a number of articles in the literature (see Ellacott [22], Starke and Varga [76], and Papamichael et al. [63]) a numerical conformal mapping package is used to generate the conformal map, and hence the Faber polynomials. The reason for this is an explicit formulae for the mapping is not known. However, the regions in question are polygonal and so a slight modification of Trefethen's SCPACK [77] can be used to numerically approximate the conformal map. From this approximate map we can then approximate the Faber polynomials. Driscoll [16] has recently adapted SCPACK for use in Matlab. He has also added some new features, that is, the ability to approximate exterior maps and being able to produce Faber polynomials.

Chapter 2

The Faber Polynomials for Annular Sectors.

2.1 Introduction

In this chapter we will investigate the Faber polynomials for the annular sector,

$$Q := \{z : R \leq |z| \leq 1, \theta \leq |\arg z| \leq \pi\}, \quad \text{with } 0 < \theta \leq \pi.$$

The motivation for considering an annular sector is an application to the solution of linear systems of equations (see Chapter 5). Eiermann et al. [17, 18, 19, 20], in their work on semi-iterative methods, consider the Faber polynomials as residual polynomials for the solution of linear systems of equations. Also, since our work began Starke and Varga [76] have given an iterative method using suitably normalised Faber polynomials as the residual polynomials (see Section 4.4.5). In their case the Faber polynomials are required for some bounded region which contains the estimated locations of matrix eigenvalues produced by the Arnoldi method or otherwise. Since one is working with a rough prediction of the eigenvalue spectrum it is not necessary for the chosen enclosing region to bear any specific relation to the estimated eigenvalues. Professor G. Opfer suggested to my supervisor, Dr. J. P. Coleman, that an annular sector would be a useful general–purpose region which, by scaling and rotation, could be adjusted to enclose any estimated eigenvalue cluster bounded away from the origin. This suggestion motivated the work in this chapter, and therefore the rest of the thesis. The main theoretical results of this chapter are summarised in the five following theorems.

THEOREM 1. The complement of the unit disc $\{w : |w| \leq 1\}$ is mapped conformally onto the complement of the annular sector

$$Q = \{z : R \leq |z| \leq 1, \theta \leq |\arg z| \leq \pi\}, \qquad 0 < \theta \leq \pi,$$

by

$$\psi(w) = -\exp\left[-\int_{a^2}^{-\zeta^{-1}} \frac{B(x)}{xA(x)}dx\right]$$

where

$$\zeta = \frac{1}{4w}\left[(w^2+1)(a^{-2}-a^2) - 2w(a^{-2}+a^2)\right]$$

and

$$A(x) = \left[(x-a^2)(x-a^{-2})\right]^{\frac{1}{2}}, \qquad B(x) = \left[(x-b^2)(x-b^{-2})\right]^{\frac{1}{2}}.$$

The parameters a and b satisfy the equations

$$\pi - \theta = \int_{a^2}^{b^2} \left[\frac{(b^2 - x)(b^{-2} - x)}{(x - a^2)(a^{-2} - x)}\right]^{\frac{1}{2}} \frac{dx}{x}$$

and

$$\log R = -\int_{b^2}^{b^{-2}} \left[\frac{(x - b^2)(b^{-2} - x)}{(x - a^2)(a^{-2} - x)}\right]^{\frac{1}{2}} \frac{dx}{x}.$$

THEOREM 2. The transfinite diameter of the annular sector Q defined in Theorem 1 is

$$\rho = \frac{(1 - a^4)}{4} \exp\left[\int_0^{a^2} \frac{a^2 + a^{-2} - b^2 - b^{-2}}{A(x)[A(x) + B(x)]}\, dx\right].$$

THEOREM 3. The coefficients of the Laurent expansion

$$\psi(w) = \rho(w + \beta_0 + \beta_1 w^{-1} + \cdots)$$

of the function defined in Theorem 1 may be generated recursively. Given a and b, in the notation of Theorem 1, let

$$u = \frac{2a^2(1 + b^4)}{b^2(1 - a^4)}, \qquad s = 2\frac{1 + a^4}{1 - a^4}.$$

Let $a_i = 0$ for $i < 0$, $a_0 = 1$ and, for $k \geq 0$,

$$\begin{aligned}(k + 1)a_{k+1} = (2k + 1)(s - u)a_k &- 2k(s^2 - su - 1)a_{k-1} \\ &+ (2k - 1)(s - u)a_{k-2} + (1 - k)a_{k-3}\end{aligned}$$

and

$$c_{k+1} = a_{k+1} - sa_k + a_{k-1}.$$

Then $\beta_0 = c_1$, $\beta_1 = \frac{1}{2}c_2$ and, for $n \geq 2$,

$$(n + 1)\beta_n = c_{n+1} - \sum_{l=1}^{n-1} l c_{n-l}\beta_l.$$

THEOREM 4. The Faber polynomial of degree n for the annular sector Q defined in Theorem 1 is $F_n(z) = \rho^n \tilde{F}_n(z)$, where ρ is the transfinite diameter of Q. The scaled Faber

polynomial may be written as $\tilde{F}_n(\rho z) = z^n + \phi_{n-1}(z)$ and the ancillary polynomials $\{\phi_n\}$ are generated recursively, in terms of the Laurent coefficients of Theorem 3, by the formula

$$\phi_n(z) = (z - \beta_0)\phi_{n-1}(z) - \sum_{k=1}^{n-1} \beta_k \phi_{n-k-1}(z) - \sum_{k=0}^{n-1} \beta_k z^{n-k} - (1+n)\beta_n.$$

THEOREM 5. The Faber series for z^{-1}, expressed in terms of the scaled Faber polynomials and the notation of Theorem 1, is

$$\frac{1}{z} = -\frac{4\rho a^2}{R(1-a^4)} \left[1 + \sum_{n=1}^{\infty} \left(\frac{1-a^2}{1+a^2} \right)^n \tilde{F}_n(z) \right].$$

Theorem 1, which is proved in Section 2.2, provides an expression for the mapping function ψ, whose inverse is a multiple of the function ϕ of (1.1); if $z = \psi(w)$ then $\phi(z) = \rho w$, where ρ is given by Theorem 2. Section 2.2 also contains checks on the formulae contained in Theorems 1 and 2. The checks involve verifying that the mapping and transfinite diameter formulae reduce to known results for the interval, the arc and the circular sector, which are all special cases of the annular sector. Section 2.3 establishes the formulae collected in Theorem 3, which allow the recursive evaluation of the Laurent coefficients essential for the computation of the Faber polynomials by the recurrence relation of Theorem 4. In addition to proving Theorem 4, Section 2.4 explores some of the properties of the ancillary polynomials $\{\phi_n\}$, and shows how explicit expressions for those polynomials may be obtained with the help of a computer algebra system such as REDUCE or Mathematica or Maple. In Section 2.5 we consider the mapping of the boundary of the annular sector. Section 2.6 considers how to obtain the conformal mapping, for an annular sector with arbitrary inner and outer radii, from the mapping given in Theorem 1, for the annular sector Q. As an example of a Faber series, the expansion of $1/z$ is investigated in Section 2.7 where we find the simple formula stated in Theorem 5.

2.2 The Conformal Mapping for the Annular Sector

In this section we will investigate the Faber polynomials for the previously defined annular sector

$$Q = \{z : R \leq |z| \leq 1, \theta \leq |\arg z| \leq \pi\},$$

where $0 < \theta \leq \pi$ (see Figure 2.1). There is no loss of generality in this choice since rotations and magnifications allow us to apply the results to any annular sector. For example, to work with the annular sector $\{z : R \leq |z| \leq 1, |\arg z| \leq \theta\}$ we make the transformations $z \to -z$ and $w \to -w$, the latter being required to maintain the form of (1.1). To investigate the Faber polynomials we will calculate an analytic function, $\psi(w)$, which maps the complement of the unit disc, $\Delta = \{w : |w| \leq 1\}$, conformally onto the exterior of the annular sector Q.

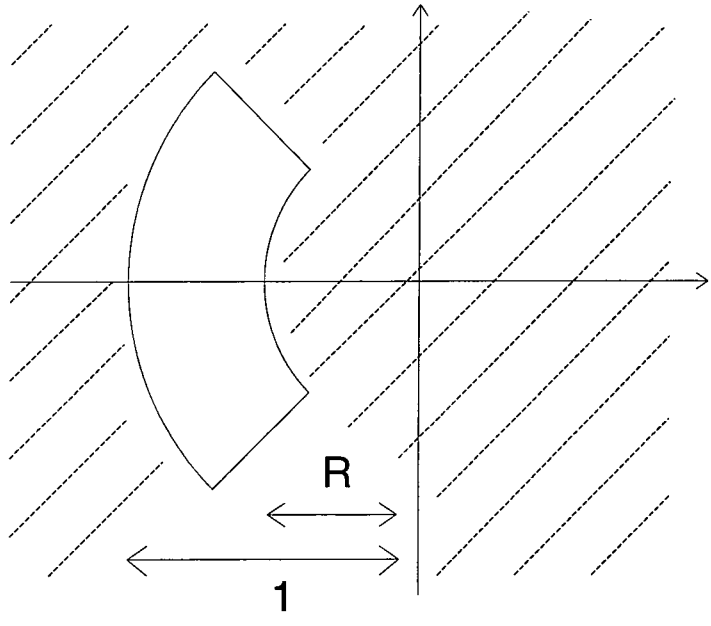

Figure 2.1: The annular sector Q

The domain obtained by cutting $\mathbb{C}\backslash Q$ along its intersection with the negative real axis is mapped conformally onto the shaded domain E of Figure 2.2 by the function $z \to \log z$,

where the principal value of the logarithm is taken. E is the interior of an infinite polygon with finite vertices at the points $\log R \pm i\pi$, $\log R \pm i\theta$, $\pm i\theta$ and $\pm i\pi$. Conversely, the function $z \to e^z$ maps E conformally onto the cut version of $\mathbb{C}\backslash Q$, and the infinite edges of the boundary of E are mapped onto the cut.

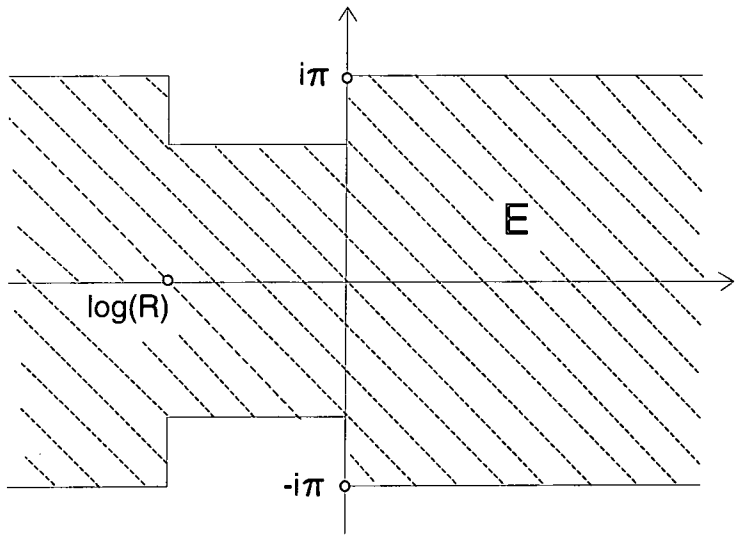

Figure 2.2: The domain E.

A Schwarz–Christoffel transformation may be used to map the upper half–plane $\Pi = \{v : \operatorname{Im} v > 0\}$ (see Figure 2.3) conformally onto the domain E, in such a way that the real axis is mapped onto the polygonal boundary in Figure 2.2 and the finite vertices of that polygon are the images of the points $\pm a^{-1}$, $\pm b^{-1}$, $\pm b$ and $\pm a$, with $0 < a \leq b \leq 1$. The Schwarz–Christoffel map has the form

$$u(v) = u(a) + K \int_a^v \left[\frac{(w-b)(w+b)(w-b^{-1})(w+b^{-1})}{(w-a)(w+a)(w-a^{-1})(w+a^{-1})} \right]^{\frac{1}{2}} \frac{dw}{w}$$

where a, b and K are constants to be determined.

Let

$$A(x) = [(x-a^2)(x-a^{-2})]^{\frac{1}{2}}, \quad B(x) = [(x-b^2)(x-b^{-2})]^{\frac{1}{2}}. \tag{2.1}$$

Then

$$u(v) - u(a) \;=\; K \int_a^v \frac{B(w^2)}{wA(w^2)}\, dw \tag{2.2}$$

$$= K\int_a^v \frac{B(w^2)-A(w^2)}{wA(w^2)}\,dw + K\log\left(\frac{v}{a}\right)$$
$$= K\int_{a^2}^{v^2} \frac{\cosh\alpha-\cosh\beta}{A(x)[B(x)+A(x)]}\,dx + K\log\left(\frac{v}{a}\right) \tag{2.3}$$

where we have written

$$a^{-2} = e^{\alpha} \quad \text{and} \quad b^{-2} = e^{\beta}. \tag{2.4}$$

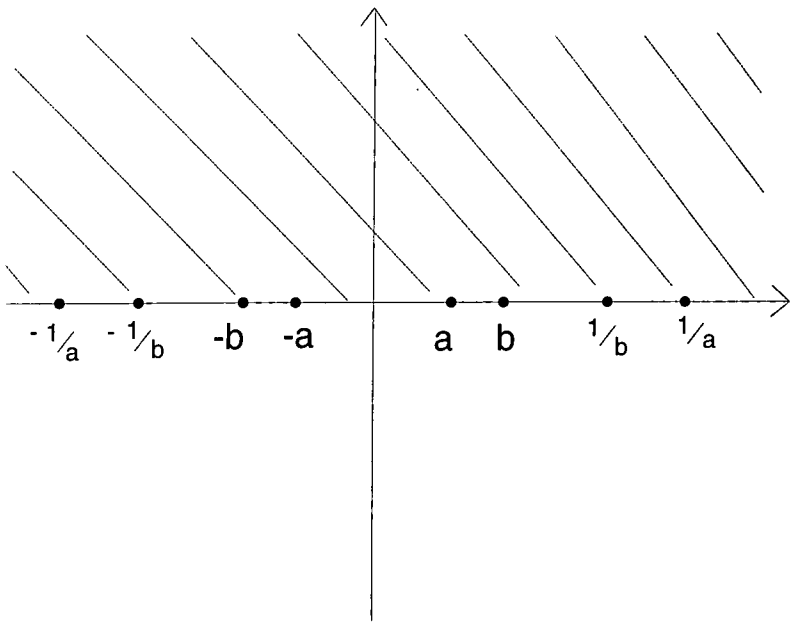

Figure 2.3: The upper half–plane Π

To determine the constant K we note that the positive real axis is mapped onto the upper boundary of the polygon in Figure 2.2 and the lower boundary is the image of the negative real axis. Therefore, for all $v \in (0, a)$,

$$2\pi i = u(v) - u(-v) = K\left[\log\left(\frac{v}{a}\right) - \log\left(-\frac{v}{a}\right)\right] = -K\pi i,$$

from which $K = -2$. Furthermore, we require

$$u(a) = i\pi,$$

and (2.2) becomes

$$u(v) = i\pi - \int_{a^2}^{v^2} \frac{B(x)}{xA(x)}\,dx.$$

Assuming that a and b can be determined for any given annular sector, (see Section 3.1), the rest of the argument is similar to that of Coleman and Smith [13]. A composition of the Joukowski function $w \to \frac{1}{2}(w^{-1}+w)$ and a linear transformation $w \to w\sinh\alpha - \cosh\alpha$ gives

$$\zeta = \frac{1}{2w}\left[(w^2+1)\sinh\alpha - 2w\cosh\alpha\right], \tag{2.5}$$

which maps $\mathbb{C}\backslash\Delta$, the complement of the unit disc Δ, conformally onto the slit plane $\mathbb{C}\backslash J$, where J is the interval $[-a^{-2}, -a^2]$ (see Figure 2.4). Let L denote the interval $(-\infty, 0]$ of the real axis. Then the function $\zeta \to i\zeta^{-\frac{1}{2}}$ maps the cut plane $\mathbb{C}\backslash L$ conformally onto the upper half-plane Π.

By composition of the mappings described here we obtain

$$z = \psi(w) = -\exp\left[-\int_{a^2}^{-\zeta^{-1}} \frac{B(x)}{xA(x)}\,dx\right] \tag{2.6}$$

which maps $\mathbb{C}\backslash L$ conformally onto the cut version of $\mathbb{C}\backslash Q$. At the end of Section 2.2.2 it will be shown that the cut introduced in $\mathbb{C}\backslash Q$ may be removed and that $\mathbb{C}\backslash L$ is then replaced by $\mathbb{C}\backslash J$.

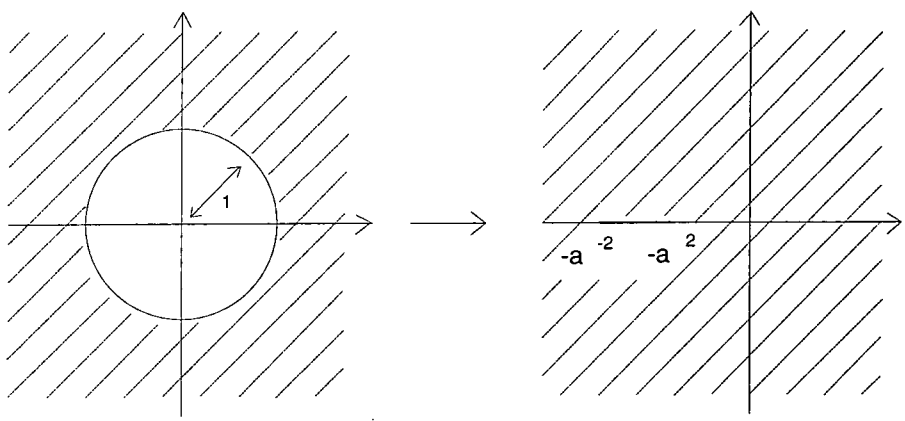

Figure 2.4: The mapping ζ

2.2.1 The defining equations for a and b

The symmetries inherent in our choice of the pre–images of the vertices of the polygon, under the Schwarz–Christoffel map, preserve the relationships, which are evident in Figure 2.2, between the lengths of the edges. The two distinct lengths which arise are

$$\pi - \theta = i\,[u(b) - u(a)]$$

and

$$-\log R = u(b) - u\left(b^{-1}\right).$$

These equations, which may be written as

$$\pi - \theta = \int_{a^2}^{b^2} \left[\frac{(b^2 - x)(b^{-2} - x)}{(x - a^2)(a^{-2} - x)}\right]^{\frac{1}{2}} \frac{dx}{x} \tag{2.7}$$

and

$$\log R = -\int_{b^2}^{b^{-2}} \left[\frac{(x - b^2)(b^{-2} - x)}{(x - a^2)(a^{-2} - x)}\right]^{\frac{1}{2}} \frac{dx}{x}, \tag{2.8}$$

uniquely determine θ and R for any given a and b such that $0 < a \leq b \leq 1$. Furthermore, as the geometrical interpretation in Figure 2.1 requires, $0 < \theta \leq \pi$. Since the integrand in (2.7) is non–negative the right–hand inequality is true and, since $b^2 + b^{-2}$ is a decreasing function of b,

$$\pi - \theta \leq \int_{a^2}^{1} \frac{1 - x}{[(x - a^2)(a^{-2} - x)]^{\frac{1}{2}}} \frac{dx}{x} = \pi - 2\sin^{-1}\left(\frac{2a}{a^2 + 1}\right); \tag{2.9}$$

in particular, $\theta > 0$. The integral in expression (2.9) will be calculated explicitly in Section 2.2.4. Equation (2.8) may be expressed in a form which is more useful for numerical computation, by regarding its right–hand side as a sum of integrals on $[b^2, 1]$ and $[1, b^{-2}]$. Making the transformation $x \to x^{-1}$ in the second integral we obtain

$$\log R = -2\int_{b^2}^{1} \left[\frac{(x - b^2)(b^{-2} - x)}{(x - a^2)(a^{-2} - x)}\right]^{\frac{1}{2}} \frac{dx}{x}. \tag{2.10}$$

The integrals in (2.7) and (2.8) may be expressed in terms of elliptic integrals of the first and third kinds. Considering (2.7) we multiply the numerator and the denominator of the integrand by the numerator itself to give

$$\pi - \theta = \int_{a^2}^{b^2} \frac{[1 - x\,(b^2 + b^{-2}) + x^2]}{x\,[(x - a^2)(a^{-2} - x)(b^2 - x)(b^{-2} - x)]^{\frac{1}{2}}}\, dx.$$

The numerator of the resulting integrand is a polynomial of degree two in x, so the above integral can be written as the sum of three integrals. Using a standard notation [66] we find

$$\pi - \theta = \frac{2ab}{1-a^2b^2}\left[C\,F\left(\frac{\pi}{2},k\right) + (a^2-a^{-2})\left\{\Pi\left(\frac{\pi}{2},-a^2k,k\right) - \Pi\left(\frac{\pi}{2},-\frac{k}{a^2},k\right)\right\}\right]$$

where

$$k = \frac{b^2-a^2}{1-a^2b^2}$$

and

$$C = 2(\cosh\alpha - \cosh\beta). \tag{2.11}$$

A similar method can be used to show that

$$\log R = \frac{-2ab}{1-a^2b^2}\left[C\,F\left(\frac{\pi}{2},k_1\right) + (b^2-a^2)\left\{\Pi\left(\frac{\pi}{2},k_2,k_1\right) - \frac{1}{a^2b^2}\Pi\left(\frac{\pi}{2},\frac{a^2}{b^2}k_2,k_1\right)\right\}\right]$$

where

$$k_1 = \frac{(1-a^4)^{\frac{1}{2}}(1-b^4)^{\frac{1}{2}}}{(1-a^2b^2)}, \qquad k_2 = \frac{(1-b^4)}{(1-a^2b^2)}.$$

The functions F and Π are the elliptic integrals of the first and third kinds, and are defined as,

$$F(\phi,k) = \int_0^\phi \frac{d\eta}{\sqrt{1-k^2\sin^2\eta}}$$

and

$$\Pi(\phi,\nu,k) = \int_0^\phi \frac{d\eta}{(1-\nu\sin^2\eta)\sqrt{1-k^2\sin^2\eta}}.$$

2.2.2 Formulae for $\psi(w)$

The mapping function ψ may be written in several different ways, which we list here for later reference.

By using the change of variable $x \to x^{-1}$ in the first formula quoted, by taking a^{-1} as reference point instead of a as in (2.2), and by using the substitution $x \to x^{-1}$ again we obtain

$$z = \psi(w) \;=\; -\exp\left[-\int_{a^2}^{-\zeta^{-1}} \frac{B(x)}{xA(x)}\,dx\right] \tag{2.12a}$$

$$= -\exp\left[\int_{a^{-2}}^{-\zeta} \frac{B(x)}{xA(x)}\,dx\right] \tag{2.12b}$$

$$= -R\exp\left[\int_{a^{2}}^{-\zeta} \frac{B(x)}{xA(x)}\,dx\right] \tag{2.12c}$$

$$= -R\exp\left[-\int_{a^{-2}}^{-\zeta^{-1}} \frac{B(x)}{xA(x)}\,dx\right] \tag{2.12d}$$

where

$$\zeta = \frac{1}{2w}\left[(w^2+1)\sinh\alpha - 2w\cosh\alpha\right].$$

In some cases, particularly if the integration interval may include the origin, it is preferable to remove the logarithmic term from the integral, as in the derivation of (2.3). Corresponding to the formulae (2.12) we have, with C as in (2.11),

$$z = a^2\zeta\exp\left[-\int_{a^{2}}^{-\zeta^{-1}} \frac{C}{A(x)[A(x)+B(x)]}\,dx\right] \tag{2.13a}$$

$$= a^2\zeta\exp\left[\int_{a^{-2}}^{-\zeta} \frac{C}{A(x)[A(x)+B(x)]}\,dx\right] \tag{2.13b}$$

$$= \frac{R\zeta}{a^2}\exp\left[\int_{a^{2}}^{-\zeta} \frac{C}{A(x)[A(x)+B(x)]}\,dx\right] \tag{2.13c}$$

$$= \frac{R\zeta}{a^2}\exp\left[-\int_{a^{-2}}^{-\zeta^{-1}} \frac{C}{A(x)[A(x)+B(x)]}\,dx\right]. \tag{2.13d}$$

Other forms of the mapping, which do not involve the intermediate variable ζ, may be derived from (2.12). We consider, for example, (2.12b) with the change of variable

$$x = \frac{-1}{2\mu}\left[(\mu^2+1)\sinh\alpha - 2\mu\cosh\alpha\right].$$

With this change of variable we find that when $x = -\zeta$ (the upper limit of the integral in (2.12b)), we have either $\mu = w$ or $\mu = 1/w$. Therefore, when $x = -\zeta$ we choose $\mu = w$ so that we are considering $|\mu| \geq 1$ when $|w| \geq 1$. We also note that $x = a^{-2}$ corresponds to $\mu = -1$ and similarly $x = a^2$ corresponds to $\mu = 1$.

Using the change of variable we have

$$\frac{dx}{d\mu} = \frac{(1-\mu^2)\sinh\alpha}{2\mu^2}$$

and

$$\begin{aligned} A(x)^2 &= (x-a^2)(x-a^{-2}) \\ &= \frac{1}{4\mu^2}\left[(\mu^2+1)\sinh\alpha + \mu(a^2-a^{-2})\right]\left[(\mu^2+1)\sinh\alpha + \mu(a^{-2}-a^2)\right] \\ &= \frac{(a^{-2}-a^2)^2}{16\mu^2}\left[\mu^2-2\mu+1\right]\left[\mu^2+2\mu+1\right]. \end{aligned}$$

so that

$$A(x) = \frac{a^{-2}-a^2}{4\mu}(\mu^2-1).$$

In a similar way

$$B(x) = \frac{\sinh\alpha}{2\mu}(\mu^2-2t\mu+1)^{\frac{1}{2}}(\mu^2-2\tau\mu+1)^{\frac{1}{2}},$$

with

$$t = \frac{\cosh\alpha - b^2}{\sinh\alpha}, \qquad \tau = \frac{\cosh\alpha - b^{-2}}{\sinh\alpha}. \tag{2.14}$$

Therefore, with the change of variable given above, the expression (2.12*b*) finally becomes

$$z = \psi(w) = -\exp\left[\int_{-1}^{w} \frac{(\mu^2-2t\mu+1)^{\frac{1}{2}}(\mu^2-2\tau\mu+1)^{\frac{1}{2}}}{\mu(\mu^2-2\mu\coth\alpha+1)}\,d\mu\right], \tag{2.15}$$

which, as mentioned previously, is an expression for the mapping which does not involve the intermediate variable ζ.

We note here that $t \in [0,1]$ and $\tau \in [-1,1]$. From before we remember that $0 < a \leq b \leq 1$ and this will be useful in the proof. Using $-1 \leq -b^2$ and $-b^2 \leq -a^2$ we find

$$0 \leq \frac{1-a^2}{1+a^2} = \frac{(1-a^2)^2}{1-a^4} \leq \frac{1+a^4-2a^2b^2}{1-a^4} = t \leq \frac{1-a^4}{1-a^4} = 1.$$

Using $-a^{-2} \leq -b^{-2}$ and $-b^{-2} \leq -1$ we can also show that

$$-1 = \frac{a^4-1}{1-a^4} \leq \frac{1+a^4-2a^2b^{-2}}{1-a^4} = \tau \leq \frac{1-2a^2+a^4}{1-a^4} = \frac{1-a^2}{1+a^2} \leq 1.$$

Finally we can also show that $t \geq \tau$. We begin by noting that $b^{-2} \geq b^2$, so $-2a^2b^2 \geq -2a^2b^{-2}$ and therefore

$$\frac{1+a^4-2a^2b^2}{1-a^4} \geq \frac{1+a^4-2a^2b^{-2}}{1-a^4}.$$

Hence $t \geq \tau$ for all a, b satisfying $0 < a \leq b \leq 1$.

To complete the proof that the function ψ provides the required mapping, from $\mathbb{C}\backslash\Delta$ to $\mathbb{C}\backslash J$, it is necessary to show that we may remove the cuts introduced in constructing ψ. Let

$$\begin{aligned}\chi(\zeta) &= a^2 \exp\left[\int_{a^{-2}}^{-\zeta} \frac{C}{A(x)\left[A(x)+B(x)\right]}\,dx\right] \\ &= a^2 \exp\left[-\int_{a^2}^{-\zeta^{-1}} \frac{C}{A(x)\left[A(x)+B(x)\right]}\,dx\right].\end{aligned}$$

The denominator of the integrand is a single–valued analytic function in $\mathbb{C}\backslash J$, the region of the complex plane exterior to the slit J, and it does not vanish in that region. Integration gives a single–valued analytic function in $\mathbb{C}\backslash J$ and the identity of the two integrals shows that it remains finite as $\zeta \to \infty$. It follows that $\chi(\zeta)$ is a single–valued analytic function in $\mathbb{C}\backslash J$ and, consequently, $\zeta\chi(\zeta)$ is also single–valued in $\mathbb{C}\backslash J$. Since the part of the interval L which lies in $\mathbb{C}\backslash J$ is mapped onto that part of the negative real axis which lies in $\mathbb{C}\backslash Q$, the function ψ given by (2.12) and (2.13) maps $\mathbb{C}\backslash\Delta$ conformally onto $\mathbb{C}\backslash Q$.

2.2.3 The transfinite diameter

Given the mapping, $z = \psi(w)$, from the outside of the unit disc to the outside of the annular sector Q, we define, as in Chapter 1,

$$\rho := \lim_{w\to\infty} \frac{\psi(w)}{w} \tag{2.16}$$

and $\phi(z) = \rho\psi^{-1}(z)$, where ψ^{-1} denotes the inverse mapping of ψ. Then $\phi(z)$ will map $\mathbb{C}\backslash Q$ conformally onto the complement of the disc $\{w : |w| \leq \rho\}$, so the number ρ is the transfinite diameter of the annular sector Q. As $w \to \infty$,

$$\zeta = \frac{1}{2}w\sinh\alpha\left[1+O\left(\frac{1}{w}\right)\right].$$

Therefore, from (2.13a) and (2.16),

$$\rho = \frac{(1-a^4)}{4} \exp\left[\int_0^{a^2} \frac{C}{A(x)[A(x)+B(x)]}\, dx\right]. \tag{2.17}$$

2.2.4 Special cases

There are three special cases of an annular sector which provide useful checks on the mapping, the formula for the transfinite diameter and the Faber polynomials themselves. In this section we shall show that the mapping and the transfinite diameter are correct for the three special cases namely the real interval, the circular arc and the circular sector.

Case (i) $b = a$. A real interval.

When $b = a$ the upper and lower limits in the integral of equation (2.7) are the same, so the integral is zero; the only possible exception to this is when the integrand is infinite, but this only occurs when $b = a = 0$ and we are considering $a > 0$ so we can ignore this. In this case equation (2.7) gives

$$\theta = \pi.$$

Also when $b = a$, equation (2.10) becomes

$$\begin{aligned} \log R &= -2\int_{a^2}^{1} \frac{dx}{x} \\ &= -2\left[\log(1) - \log(a^2)\right]. \end{aligned}$$

From this we have $R = a^4$ (again with the exception of $b = a = 0$). Consequently the annular sector Q becomes the interval $[-1, -R]$ of the real axis.

From (2.3) it is evident that

$$u(v) = i\pi - 2\log\left(\frac{v}{a}\right)$$

when $b = a$, because $\cosh\alpha = \cosh\beta$ and so the integrand vanishes. Therefore, as $v = i\xi^{-\frac{1}{2}}$,

$$z = e^u = a^2\zeta = \frac{1}{4}\left(w + \frac{1}{w}\right)(1-R) - \frac{1}{2}(1+R),$$

which, comparing with Section 1.2.4 example (b), correctly maps $\mathbb{C}\backslash\Delta$ onto the complement of the real interval $[-1, -R]$.

When $b = a$, (2.11) gives us $C = 0$, so the integrand in (2.17) vanishes, and the transfinite diameter in this case is given by

$$\rho = \frac{(1-a^4)}{4} = \frac{1}{4}(1-R),$$

which again agrees with Section 1.2.4 example (b).

Case (ii) $b = 1$. A circular arc.

THEOREM 6

When $b = 1$ we have $R = 1$ and $\theta = 2\sin^{-1}\left[2a(a^2+1)^{-1}\right]$.

Proof

The first part of Theorem 6 is easy to prove. When $b = 1$ the upper and lower limits of the integral in (2.8) are the same, so the integral vanishes and we have $R = 1$.

For the second part consider equation (2.7) with $b = 1$,

$$\pi - \theta = \int_{a^2}^{1} \frac{(1-x)\,dx}{x(x-a^2)^{\frac{1}{2}}(a^{-2}-x)^{\frac{1}{2}}}.$$

We then make the substitution $x = e^t$ and remember $a^{-2} = e^{\alpha}$, so when $x = a^2$ we have $t = \ln(a^2) = -\alpha$. The equation can now be written as

$$\begin{aligned}
\pi - \theta &= \int_{-\alpha}^{0} \frac{(1-e^t)\,dt}{[(a^2+a^{-2})e^t - 1 - e^{2t}]^{\frac{1}{2}}} \\
&= \int_{-\alpha}^{0} \frac{(e^{-\frac{t}{2}} - e^{\frac{t}{2}})\,dt}{[2\cosh\alpha - e^{-t} - e^{t}]^{\frac{1}{2}}} \\
&= \int_{-\alpha}^{0} \frac{-2\sinh\frac{t}{2}\,dt}{(2\cosh\alpha - 2\cosh t)^{\frac{1}{2}}} \\
&= \int_{-\alpha}^{0} \frac{-2\sinh\frac{t}{2}\,dt}{(2\cosh\alpha - 4\cosh^2\frac{t}{2} + 2)^{\frac{1}{2}}}
\end{aligned}$$

We now make the substitution $y = \cosh\frac{t}{2}$, and note that

$$\cosh\left(\frac{-\alpha}{2}\right) = \frac{1+a^2}{2a} = \gamma, \quad \cosh\alpha + 1 = \frac{(1+a^2)^2}{2a^2}.$$

Hence

$$\begin{aligned}\pi - \theta &= \int_\gamma^1 \frac{-4\,dy}{[(2\gamma)^2 - 4y^2]^{\frac{1}{2}}} \\ &= -2\left[\sin^{-1}\left(\frac{y}{\gamma}\right)\right]_\gamma^1 \\ &= -2\sin^{-1}\left(\frac{1}{\gamma}\right) + 2\sin^{-1}(1),\end{aligned}$$

and

$$\theta = 2\sin^{-1}\left(\frac{2a}{1+a^2}\right). \qquad \square \qquad (2.18)$$

The annular sector Q therefore degenerates, when $b = 1$, to an arc of the unit circle $|z| = 1$, of half–angle $\pi - \theta$. The question is do the formulae for the mapping and the transfinite diameter agree with the known results for the circular arc.

When $b = 1$, $\tau = t = \coth\alpha - 1/\sinh\alpha$ in (2.15) and

$$\begin{aligned}\psi(w) &= -\exp\left[\int_{-1}^w \frac{\mu^2 - 2t\mu + 1}{\mu(\mu^2 - 2\mu\coth\alpha + 1)}\,d\mu\right] \\ &= w\exp\left[\frac{2}{\sinh\alpha}\int_{-1}^w \frac{d\mu}{\mu^2 - 2\mu\coth\alpha + 1}\right].\end{aligned}$$

Using partial fractions we have

$$\begin{aligned}\frac{2}{\sinh\alpha}\int_{-1}^w \frac{d\mu}{\mu^2 - 2\mu\coth\alpha + 1} &= \int_{-1}^w \left[-\frac{\sinh\alpha}{\mu\sinh\alpha - \cosh\alpha + 1} + \frac{\sinh\alpha}{\mu\sinh\alpha - \cosh\alpha - 1}\right] \\ &= \left[\ln\left(\frac{\mu\sinh\alpha - \cosh\alpha - 1}{\mu\sinh\alpha - \cosh\alpha + 1}\right)\right]_{-1}^w \\ &= \ln\left(\left[\frac{(1-a^4)w - (1+a^2)^2}{(1-a^4)w - (1-a^2)^2}\right]\left[\frac{(1-a^2)}{(1+a^2)}\right]\right) \\ &= \ln\left[\frac{(1-a^2)w - (1+a^2)}{(1+a^2)w - (1-a^2)}\right]\end{aligned}$$

Noting that $\tanh\frac{\alpha}{2} = (1-a^2)/(1+a^2)$ the formula for the mapping becomes

$$z = \psi(w) = \frac{w(w\tanh\frac{\alpha}{2} - 1)}{w - \tanh\frac{\alpha}{2}}. \tag{2.19}$$

Comparing this with example (c) from Section 1.2.4, we see that this is the desired mapping provided that $\tanh\frac{\alpha}{2} = \cos\frac{\theta}{2}$, with θ given by (2.18). We note that

$$\begin{aligned}\cos\left(\frac{\theta}{2}\right) &= \cos\left[\sin^{-1}\left(\frac{2a}{1+a^2}\right)\right] \\ &= \cos\left[\cos^{-1}\left(\frac{1-a^2}{1+a^2}\right)\right] \\ &= \frac{1-a^2}{1+a^2}.\end{aligned}$$

We choose this value of the cosine because $\theta \in (0,\pi]$, so $\theta/2 \in (0,\pi/2]$, and hence $\cos(\theta/2) \geq 0$. Finally we have

$$\cos\left(\frac{\theta}{2}\right) = \tanh\frac{\alpha}{2}.$$

Hence the mapping function agrees with example (c) in Section 1.2.4 and correctly maps $\mathbb{C}\backslash\Delta$ onto the complement of the circular arc

$$\{z : |z| = 1,\ \theta \leq |\arg z| \leq \pi\},$$

where θ is given by equation (2.18).

From the expression (2.19) for $\psi(w)$, the transfinite diameter of the arc is

$$\rho = \tanh\frac{\alpha}{2} = \left(\frac{1-a^2}{1+a^2}\right) = \cos\frac{\theta}{2}$$

which, with allowance for the difference in notation, agrees with Ellacott [23] (see also example (c) in Section 1.2.4). As a check on formula (2.17) we should confirm that when $b = 1$ the formula gives the same result for the transfinite diameter. When $b = 1$ formula (2.17) becomes

$$\rho = \frac{(1-a^4)}{4}\exp\left[\int_0^{a^2}\frac{2(\cosh\alpha - 1)\,dx}{A(x)^2 + A(x)(x-1)}\right].$$

Making the substitution

$$x = -\frac{1}{2\mu}\left[(\mu^2+1)\sinh\alpha - 2\mu\cosh\alpha\right],$$

as in Section 2.2.2 and writing (see equation (2.38) in Section 2.7)

$$w^* = \frac{1+a^2}{1-a^2},$$

the transfinite diameter becomes

$$\begin{aligned}
\rho &= \frac{(1-a^4)}{4}\exp\left[\int_{w^*}^{1}\frac{4(\cosh\alpha-1)\,d\mu}{(1-\mu^2)\sinh\alpha-(1+\mu^2)\sinh\alpha+2\mu(\cosh\alpha-1)}\right] \\
&= \frac{(1-a^4)}{4}\exp\left[\int_{w^*}^{1}\frac{4(\cosh\alpha-1)\,d\mu}{-2\mu^2\sinh\alpha-2\mu(1-\cosh\alpha)}\right] \\
&= \frac{(1-a^4)}{4}\exp\left[\int_{1}^{w^*}\frac{2(\cosh\alpha-1)\,d\mu}{\mu(\mu\sinh\alpha+1-\cosh\alpha)}\right]
\end{aligned}$$

The integrand in the above expression is written as

$$-\frac{2}{\mu}+\frac{2\sinh\alpha}{\mu\sinh\alpha+1-\cosh\alpha},$$

so that the transfinite diameter can then be expressed as

$$\rho = \frac{(1-a^4)}{4}\exp\left[2\ln\left(\frac{w^*\sinh\alpha+1-\cosh\alpha}{w^*}\right)-2\ln\left(\sinh\alpha+1-\cosh\alpha\right)\right]$$

Finally we note that

$$\sinh\alpha+1-\cosh\alpha = 1-a^2$$

and

$$w^*\sinh\alpha+1-\cosh\alpha = 2,$$

so the transfinite diameter is now written as

$$\begin{aligned}
\rho &= \frac{(1-a^4)}{4}\exp\left[2\ln\left(\frac{2(1-a^2)}{1+a^2}\right)-2\ln(1-a^2)\right] \\
&= \frac{(1-a^2)}{(1+a^2)}
\end{aligned}$$

This agrees with the formula for the transfinite diameter on the previous page. Hence, when $b=1$, formula (2.17) gives the correct result for the transfinite diameter.

Case (iii) $a \to 0$ and $b \to 0$. A circular sector.

THEOREM 7

We have $\theta \to a\pi/b$ and $R \to 0$ in the limit as $a \to 0$ and $b \to 0$.

Proof

In the integrand of (2.7), $x << b^{-2} \leq a^{-2}$ when $0 < a \leq b << 1$. Expanding $(b^{-2} - x)^{\frac{1}{2}}(a^{-2} - x)^{-\frac{1}{2}}$, in terms of a and b, we obtain,

$$\begin{aligned}(b^{-2} - x)^{\frac{1}{2}}(a^{-2} - x)^{-\frac{1}{2}} &= \frac{a}{b}\left(1 - \frac{1}{2}b^2x + O(b^4)\right)\left(1 + \frac{1}{2}a^2x + O(a^4)\right) \\ &= \frac{a}{b}\left(1 + O(a^2) + O(b^2)\right),\end{aligned}$$

as $a \to 0$ and $b \to 0$. Substituting this in (2.7) we have

$$\pi - \theta = \int_{a^2}^{b^2} \frac{(b^2 - x)^{\frac{1}{2}}}{x(x - a^2)^{\frac{1}{2}}}\left(\frac{a}{b}\left(1 + O(a^2) + O(b^2)\right)\right) dx.$$

We now consider the integral

$$I = \frac{a}{b}\int_{a^2}^{b^2} \frac{(b^2 - x)^{\frac{1}{2}}}{x(x - a^2)^{\frac{1}{2}}}\, dx = \frac{a}{b}\int_{a^2}^{b^2} \frac{(b^2 - x)}{x\left[(x - a^2)(b^2 - x)\right]^{\frac{1}{2}}}\, dx;$$

with

$$2x = a^2 + b^2 + (b^2 - a^2)\sin t,$$

the integral becomes

$$I = \frac{a}{b}\int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \left[\frac{(b^2 - a^2) - (b^2 - a^2)\sin t}{(b^2 + a^2) + (b^2 - a^2)\sin t}\right] dt.$$

We now make the substitution $\tan t/2 = s$ so that

$$\sin t = \frac{2s}{1 + s^2}, \qquad \frac{dt}{ds} = \frac{2}{1 + s^2}$$

and the integral is now given by

$$I = \frac{a}{b}\int_{-1}^{1} \frac{2(b^2 - a^2)\,(s^2 - 2s + 1)\, ds}{(1 + s^2)\left[(b^2 + a^2)s^2 + 2s(b^2 - a^2) + (b^2 + a^2)\right]}$$

$$= \frac{2a}{b}\int_{-1}^{1}\left(\frac{2b^2}{(b^2+a^2)s^2+2s(b^2-a^2)+(b^2+a^2)}-\frac{1}{s^2+1}\right)ds.$$

$$= \frac{2a}{b}\int_{-1}^{1}\frac{2b^2}{(b^2+a^2)s^2+2s(b^2-a^2)+(b^2+a^2)}\,ds-\frac{2a}{b}\int_{-1}^{1}\frac{1}{s^2+1}\,ds$$

Making the substitution $s(b^2+a^2)=2(ab)y+(a^2-b^2)$ we have

$$\begin{aligned} I &= 2\int_{-\frac{a}{b}}^{\frac{b}{a}}\frac{dy}{1+y^2}-\frac{2a}{b}\int_{-1}^{1}\frac{ds}{1+s^2} \\ &= 2\left[\tan^{-1}(y)\right]_{-\frac{a}{b}}^{\frac{b}{a}}-\frac{2a}{b}\left[\tan^{-1}(s)\right]_{-1}^{1} \\ &= \pi-\frac{\pi a}{b} \end{aligned}$$

Substituting this integral into the expression given above for $\pi-\theta$ we find

$$\pi-\theta=\left(\pi-\frac{a\pi}{b}\right)\left[1+O(a^2)+O(b^2)\right]$$

so when $a\to 0$ and $b\to 0$ we have

$$\theta\to\frac{\pi a}{b}. \tag{2.20}$$

Similarly, from (2.10),

$$\log R=-\frac{2a}{b}\int_{b^2}^{1}\frac{(x-b^2)^{\frac{1}{2}}}{x(x-a^2)^{\frac{1}{2}}}\,dx\left[1+O(a^2)+O(b^2)\right].$$

Making the substitution $y^2=(x-b^2)/(x-a^2)$ we find

$$\frac{2a}{b}\int_{b^2}^{1}\frac{(x-b^2)^{\frac{1}{2}}}{x(x-a^2)^{\frac{1}{2}}}\,dx = \frac{2a}{b}\int_{0}^{y_0}\frac{2y^2(b^2-a^2)\,dy}{(b^2-a^2y^2)(1-y^2)}$$

with $y_0=(1-b^2)^{\frac{1}{2}}(1-a^2)^{-\frac{1}{2}}$. Using partial fractions we are able to evaluate this integral as

$$\frac{2a}{b}\left[\frac{b}{a}\log(b-ay)-\frac{b}{a}\log(b+ay)-\log(1-y)+\log(1+y)\right]_0^{y_0}.$$

Substituting this back into our expression for $\log R$ we find

$$\log R=-\frac{2a}{b}\left[\log\left(\frac{1+y_0}{1-y_0}\right)+\frac{b}{a}\log\left(\frac{b-ay_0}{b+ay_0}\right)\right]\left(1+O(a^2)+O(b^2)\right)$$

From (2.20) we see that as a and b both tend to zero we may write $a = \lambda b$. So for small a and b we find

$$\log R \approx 2\lambda \left[\log\left(\frac{1-y_0}{1+y_0}\right) + \lambda \log\left(\frac{1+\lambda y_0}{1-\lambda y_0}\right)\right]$$

So that

$$R \approx \left(\frac{1-y_0}{1+y_0}\right)^{2\lambda} \left(\frac{1+\lambda y_0}{1-\lambda y_0}\right)^{2\lambda^2}.$$

As $a \to 0$ and $b \to 0$ we have $y_0 = \sqrt{1-b^2}/\sqrt{1-a^2} \to 1$. Therefore $R \to 0$ as a and b both tend to zero, that is, with the exception of $\lambda = 1$ and $\lambda = 0$. When $\lambda = 0$ we note that a tends to zero independently of b (that is b may not be zero). When $\lambda = 1$ we have $a = b \to 0$, that is, we are considering the interval case (see case (i) in Section 2.2.4) in the limit as $a \to 0$. Referring to this case we see that $R \to 0$ as $a \to 0$.

Hence, $R \to 0$ as a and b both tend to zero. □

If we consider the limit $a \to 0$ and $b \to 0$, with $a = \lambda b$, then we can remove the small a and b argument from the previous proof (see Appendix A for the details).

Theorem 7 implies that in the limit as a and b both tend to zero the annular sector Q tends to the circular sector

$$\{z : |z| \leq 1,\ \theta \leq |\arg z| \leq \pi\}$$

where $\theta = \pi a/b$ and so our results in that limit should agree with those of Coleman and Smith [13] and of Gatermann et al. [39].

To find the transfinite diameter in the required limit we write $a = \lambda b$ in equation (2.17), and make the substitution $x = tb^2$, to obtain

$$\rho = \frac{(1-\lambda^4 b^4)}{4} \exp\left[I(\lambda, b)\right],$$

where

$$I(\lambda, b) = \int_0^{\lambda^2} \frac{[b^4(\lambda^2-1) + \lambda^{-2} - 1]\ dt}{(\lambda^2-t)^{\frac{1}{2}}(\lambda^{-2}-tb^4)^{\frac{1}{2}}\left[(\lambda^2-t)^{\frac{1}{2}}(\lambda^{-2}-tb^4)^{\frac{1}{2}} + (1-t)^{\frac{1}{2}}(1-b^4t)^{\frac{1}{2}}\right]}.$$

In the limit as $b \to 0$ this reduces to

$$\rho = \frac{1}{4} \exp\left[\int_0^{\lambda^2} \left\{\frac{\lambda(1-t)^{\frac{1}{2}}}{(\lambda^2-t)^{\frac{1}{2}}} - 1\right\} \frac{dt}{t}\right].$$

Substituting $y^2 = (1-t)/(\lambda^2 - t)$ we have

$$\rho = \frac{1}{4}\exp\left[\int_{\lambda^{-1}}^{\infty} \frac{2y(1-\lambda^2)\,dy}{(y^2-1)(\lambda y+1)}\right]$$

and using partial fractions we are able to convert the integral to a sum of elementary integrals and we find

$$\begin{aligned}
\rho &= \frac{1}{4}\exp\left[\int_{\lambda^{-1}}^{\infty}\left\{\frac{1-\lambda}{y-1} + \frac{1+\lambda}{1+y} - \frac{2\lambda}{\lambda y+1}\right\} dy\right] \\
&= \frac{1}{4}\exp\left[\log\left\{\frac{(y-1)^{1-\lambda}(y+1)^{1+\lambda}}{(\lambda y+1)^2}\right\}\right]_{\lambda^{-1}}^{\infty} \\
&= \frac{1}{4}\exp\left[\log\left(\frac{1}{\lambda^2}\right) - \log\left(\frac{(1+\lambda)^{1+\lambda}}{4\lambda^2(1-\lambda)^{\lambda-1}}\right)\right] \\
&= \frac{1}{(1-\lambda)^{1-\lambda}(1+\lambda)^{1+\lambda}}.
\end{aligned}$$

In this limit $\lambda = \theta/\pi$, where the half–angle of the sector is $\pi - \theta$. We now show that our expression for the transfinite diameter agrees with the transfinite diameter in Coleman and Smith [13]. In their notation π/α is the half–angle of the sector, so that in our expression

$$\lambda = 1 - \frac{1}{\alpha}$$

and the transfinite diameter is given by

$$\rho = \frac{\alpha^2}{(2\alpha-1)^{2-\frac{1}{\alpha}}}.$$

Comparing this expression with Theorem 2 of Coleman and Smith [13] (see example (d) of Section 1.2.4) shows that our expression is the transfinite diameter for a sector of the unit disc of half–angle $\pi - \theta$.

Similar reasoning applied to the integral in $(2.13a)$ allows us to compute ψ_0, the limit of the mapping function ψ as a and b both tend to 0. Substituting $x = tb^2$ into the integral in $(2.13a)$ we find

$$z = a^2\zeta \exp\left[-J(\lambda, b)\right]$$

where the integrand in $J(\lambda, b)$ is the same as the integrand in $I(\lambda, b)$; however, the lower limit on the integral is λ^2 and the upper limit is $-\zeta^{-1}/b^2$. Noting that

$$\zeta = \frac{1}{4a^2 w}\left[(w-1)^2 + O(a^4)\right]$$

as $a \to 0$, we find

$$\psi_0(w) = \frac{(w-1)^2}{4w} \exp\left[\int_{-x_0}^{\lambda^2} \left\{\frac{\lambda(1-t)^{\frac{1}{2}}}{(\lambda^2-t)^{\frac{1}{2}}} - 1\right\} \frac{dt}{t}\right]$$

where $x_0 = 4\lambda^2 w(w-1)^{-2}$. Once again we make the substitution $y^2 = (1-t)/(\lambda^2 - t)$ (see the transfinite diameter calculation on pages 43 and 44) and find

$$\begin{aligned} z &= \frac{(w-1)^2}{4w} \exp\left[\log\left\{\frac{(y-1)^{1-\lambda}(1+y)^{1+\lambda}}{(\lambda y+1)^2}\right\}\right]_{y_0}^{\infty} \\ &= \frac{(w-1)^2(\lambda y_0+1)^2}{4w\lambda^2(y_0-1)^{1-\lambda}(y_0+1)^{1+\lambda}} \end{aligned}$$

where $y_0 = \sqrt{(1+x_0)/(\lambda^2+x_0)}$.

We shall show that this expression for the mapping function is the same as the mapping function given by Coleman and Smith [13]. To avoid conflicting uses of a we shall slightly alter the notation of Coleman and Smith and write a_c for a. Coleman and Smith give the mapping function for a sector of the unit disc with $0 \leq |\arg Z| \leq \pi/\alpha$ as

$$Z = \psi(W) = \frac{-\left[u + \sqrt{u^2-1}\right]^2}{\left[v + \sqrt{v^2-1}\right]^{p(a_c)}},$$

with

$$\begin{aligned} u(W) &= \frac{i(W-1)}{2a_c W^{\frac{1}{2}}} \\ v(W) &= \frac{(2-a_c^2)(W-1)^2 + 4Wa_c^2}{a_c^2(W+1)^2} \end{aligned}$$

and

$$1 - a_c^2 = \left(1 - \frac{1}{\alpha}\right)^2, \qquad p(a_c) = 1 - \frac{1}{\alpha}.$$

In our case we have shown that $\lambda = (1 - 1/\alpha) = p(a_c)$ and $1 - a_c^2 = \lambda^2$.

Coleman and Smith place their sector symmetrically about the positive real axis and we place our sector symmetrically about the negative real axis. This means that to compare the mapping function of Coleman and Smith with ours we must make the transformations $Z \to -z$ and $W \to -w$, where the second transformation is used to preserve relation (1.1) (see the beginning of Section 2.2). When considering these transformations we must be careful to consider the correct branch of the square root in $u(W)$. We note that $\sqrt{W}$ is real and positive when W is real and positive, that is when w is real and negative. Now $\sqrt{w} = \sqrt{|w|}\exp i\theta/2$ where θ is the argument of w, so when w is real and negative we have $\sqrt{w} = i\sqrt{|w|}$. So $\sqrt{W} = \sqrt{-w} = \pm i\sqrt{w} = \mp\sqrt{|w|}$, and we must have $\sqrt{W} = -i\sqrt{w}$ to be on the correct branch of the square root function.

This means

$$u(W) = u(-w) = \frac{w+1}{2a_c\sqrt{w}},$$

and

$$v(W) = v(-w) = \frac{(2-a_c^2)(w+1)^2 - 4wa_c^2}{a_c^2(1-w)^2}.$$

Returning to our mapping, we see that it may be written as

$$z = \frac{(w-1)^2(1+\lambda y_0)^2}{4w\lambda^2(y_0^2-1)}\left(\frac{y_0-1}{y_0+1}\right)^{\lambda},$$

and because $p(a_c) = \lambda$ we now prove that

$$v + \sqrt{v^2-1} = \left(\frac{y_0+1}{y_0-1}\right) \quad \text{and} \quad (u+\sqrt{u^2-1})^2 = \frac{(w-1)^2(1+\lambda y_0)^2}{4w\lambda^2(y_0^2-1)}.$$

LEMMA 1

$$(u+\sqrt{u^2-1})^2 = \frac{(w-1)^2(1+\lambda y_0)^2}{4w\lambda^2(y_0^2-1)}.$$

Proof

From previous expressions for x_0 and y_0 (see page 45) we find

$$y_0^2 = \frac{1+x_0}{\lambda^2+x_0} = \frac{4w\lambda^2+(w-1)^2}{\lambda^2(w+1)^2}.$$

Hence

$$1+\lambda y_0 = \frac{w+1+\sqrt{4w\lambda^2+(w-1)^2}}{w+1}$$

and

$$y_0^2 - 1 = \frac{(1-\lambda^2)(w-1)^2}{\lambda^2(w+1)^2}.$$

Therefore

$$\begin{aligned}
\frac{(w-1)^2(1+\lambda y_0)^2}{4w\lambda^2(y_0^2-1)} &= \frac{\left[w+1+\sqrt{4w\lambda^2+(w-1)^2}\right]^2}{4w(1-\lambda^2)} \\
&= \frac{\left[w+1+\sqrt{(w+1)^2+4w(\lambda^2-1)}\right]^2}{4w(1-\lambda^2)} \\
&= \frac{\left[w+1+\sqrt{(w+1)^2-4a_c^2w}\right]^2}{4a_c^2w} \\
&= (u+\sqrt{u^2-1})^2. \qquad \square
\end{aligned}$$

LEMMA 2

$$v + \sqrt{v^2-1} = \left(\frac{y_0+1}{y_0-1}\right).$$

Proof

We note that $v + \sqrt{v^2-1}$ is one of the roots of

$$x^2 - 2xv + 1 = 0, \tag{2.21}$$

so for the lemma to be correct $(y_0+1)/(y_0-1)$ must satisfy

$$\left(\frac{y_0+1}{y_0-1}\right)^2 - 2v\left(\frac{y_0+1}{y_0-1}\right) + 1 = 0.$$

Rearranging this equation we find that it is true if and only if

$$v = \frac{y_0^2+1}{y_0^2-1}.$$

From before we know

$$y_0^2 - 1 = \frac{(1-\lambda^2)(w-1)^2}{\lambda^2(w+1)^2} \quad \text{and} \quad y_0^2 + 1 = \frac{(\lambda^2+1)(w+1)^2 - 4w(1-\lambda^2)}{\lambda^2(w+1)^2}$$

so

$$\begin{aligned}\frac{y_0^2+1}{y_0^2-1} &= \frac{(\lambda^2+1)(w+1)^2-4w(1-\lambda^2)}{(1-\lambda^2)(w-1)^2} \\ &= \frac{(2-a_c^2)(w+1)^2-4wa_c^2}{a_c^2(w-1)^2} \\ &= v.\end{aligned}$$

This means that $v=(y_0^2+1)/(y_0^2-1)$ so from (2.21)

$$\frac{y_0+1}{y_0-1} = v \pm \sqrt{v^2-1}.$$

We note that as $w \to 1$ then $v \to \infty$ so $v+\sqrt{v^2-1} \to \infty$ and $v-\sqrt{v^2-1} \to 0$. Finally we note that as $w \to 1$ then $x_0 \to \infty$ and $y_0 \to 1$ so $(y_0+1)/(y_0-1) \to \infty$ and hence

$$\frac{y_0+1}{y_0-1} = v+\sqrt{v^2-1}. \qquad \square$$

These two lemmas confirm that the map we have derived correctly reduces to that for a circular sector in the limit as $a \to 0$ and $b \to 0$ with $a/b = \theta/\pi$.

2.3 The Laurent expansion of $\psi(w)$ about the point at infinity

As in Chapter 1 (see equation (1.4)) the function $\psi(w)$ has a Laurent expansion of the form

$$\psi(w) = \rho(w+\beta_0+\beta_1 w^{-1}+\cdots)$$

about the point at infinity, and as before the coefficients of this expansion are required in a recurrence relation used to generate the Faber polynomials (property 5). In this section we will show how these coefficients, for the mapping $\psi(w)$, can be generated from simple recurrence relations.

Differentiation of (2.15) gives

$$\frac{d\psi}{dw} = \psi(w)\frac{M(w,t)M(w,\tau)}{w(w^2 - 2w\coth\alpha + 1)} \tag{2.22}$$

where

$$M(w,y) = (w^2 - 2yw + 1)^{\frac{1}{2}}.$$

If we now let $w = \xi^{-1}$ and $\Psi(\xi) = \psi(w)$, equation (2.22) may be rewritten as

$$\Psi(\xi) = -\frac{\xi(\xi^2 - 2\xi\coth\alpha + 1)}{M(\xi,t)M(\xi,\tau)}\frac{d\Psi(\xi)}{d\xi}. \tag{2.23}$$

Since $|t| \leq 1$ and $|\tau| \leq 1$, there is a convergent expansion of the form

$$[M(\xi,t)M(\xi,\tau)]^{-1} = \sum_{k=0}^{\infty} a_k\xi^k \tag{2.24}$$

for $|\xi| < 1$. Substitution in (2.23) gives

$$\Psi(\xi) = \left(\sum_{k=0}^{\infty} c_k\xi^k\right)\left(-\xi\frac{d\Psi}{d\xi}\right) \tag{2.25}$$

where

$$c_k = a_{k-2} - 2\coth\alpha\, a_{k-1} + a_k, \qquad k \geq 0, \tag{2.26}$$

with $a_{-2} = a_{-1} = 0$. Also, from (1.4),

$$\Psi(\xi) = \rho\left(\frac{1}{\xi} + \beta_0 + \beta_1\xi + \cdots\right)$$

for $|\xi| < 1$. We may substitute this in (2.25) and equate coefficients of the powers of ξ on both sides to obtain $\beta_0 = c_1$, $\beta_1 = \frac{1}{2}c_2$ and, for $n \geq 2$,

$$(1+n)\beta_n = c_{n+1} - \sum_{i=1}^{n-1} ic_{n-i}\beta_i. \tag{2.27}$$

The recurrence relation in (2.27) and the definition (2.26) allow us to generate $\beta_0, \beta_1, \ldots, \beta_n$, for a given positive integer n, when a_k is known for $k = 0, 1, \ldots, n+1$. In the case of circular sectors, Coleman and Smith [13] found a very simple expression for the corresponding coefficients in terms of Legendre polynomials. A similar approach can be used here; noting that $M(w,y)$ is the reciprocal of a generating function for the Legendre polynomials, that is,

$$M(w,y)^{-1} = (1 - 2yw + w^2)^{-\frac{1}{2}} = \sum_{n=0}^{\infty} P_n(y)w^n,$$

where $P_n(y)$ is the Legendre polynomial of degree n. Therefore

$$[M(\xi,t)M(\xi,\tau)]^{-1} = \left(\sum_{n=0}^{\infty} P_n(t)\xi^n\right)\left(\sum_{m=0}^{\infty} P_m(\tau)\xi^m\right)$$

$$= \sum_{n=0}^{\infty}\sum_{m=0}^{\infty} P_n(t)P_m(\tau)\xi^{n+m}.$$

The coefficient of ξ^k in this expression is

$$\sum_{l=0}^{k} P_l(t)P_{k-l}(\tau).$$

A comparison with equation (2.24) gives

$$a_k = \sum_{l=0}^{k} P_l(t)P_{k-l}(\tau).$$

This formula has the computational disadvantage that as k increases an increasing number of Legendre polynomials must be evaluated and stored. For this reason we have derived a five–term recurrence relation from which $\{a_k\}$ may be computed directly. Let $f(t,\tau,\xi)$ be the function in (2.24). Then, by differentiation and rearrangement,

$$[t+\tau-2\xi(1+2t\tau)+3(t+\tau)\xi^2-2\xi^3]f(t,\tau,\xi)$$
$$= (1-2t\xi+\xi^2)(1-2\tau\xi+\xi^2)\frac{\partial f(t,\tau,\xi)}{\partial\xi}.$$

Equating the coefficients of the powers of ξ on both sides of the equation we obtain

$$(k+1)a_{k+1} = (2k+1)(t+\tau)a_k - 2k(1+2t\tau)a_{k-1}$$
$$+ (2k-1)(t+\tau)a_{k-2} + (1-k)a_{k-3}$$

for $k \geq 0$, with $a_0 = 1$ and $a_i = 0$ for $i < 0$. With

$$s = 2\frac{\cosh\alpha}{\sinh\alpha} \quad \text{and} \quad u = 2\frac{\cosh\beta}{\sinh\alpha}$$

this becomes

$$(k+1)a_{k+1} = (2k+1)(s-u)a_k - 2k(s^2-su-1)a_{k-1}$$
$$+ (2k-1)(s-u)a_{k-2} + (1-k)a_{k-3}. \qquad (2.28)$$

2.4 The Faber Polynomials

The Faber polynomials, $F_n(z)$, satisfy the recurrence relation

$$F_{n+1}(z) = (z - b_0)F_n(z) - \sum_{k=1}^{n-1} b_k F_{n-k}(z) - (1+n)b_n, \quad n > 0 \tag{2.29}$$

where $b_k = \beta_k \rho^{k+1}$, ρ is the transfinite diameter of the region, and the β_k are generated from the recurrence relations above. Following Gatermann et al. [39], we introduce the scaled Faber polynomials

$$\tilde{F}_n(z) = F_n(z)\rho^{-n} = \Phi_n\left(\frac{z}{\rho}\right) \tag{2.30}$$

and let

$$\Phi_n(z) = z^n + \phi_{n-1}(z) \tag{2.31}$$

where ϕ_{n-1} is a polynomial of degree $n-1$ for $n \geq 1$, and $\phi_{-1}(z)$=0.

Substitution in (2.29) gives the recurrence relation

$$\phi_n(z) = (z - \beta_0)\phi_{n-1}(z) - \sum_{k=1}^{n-1} \beta_k \phi_{n-k-1}(z) - \sum_{k=0}^{n-1} \beta_k z^{n-k} - (1+n)\beta_n. \tag{2.32}$$

Our notation differs slightly from that used by Gatermann et al. [39], in their work on circular sectors, because no factor analogous to their $1-c$ is evident, except in the limit as $a \to 0$ and $b \to 0$, when the annular sector tends to a sector of the unit disc. Given the Schwarz–Christoffel parameters a and b corresponding to a particular annular sector, the Faber polynomials of degree up to $n_{\max}$ may be computed by the following algorithm.

Algorithm

$s = 2(1+a^4)/(1-a^4)$; $u = 2a^2b^{-2}(1+b^4)/(1-a^4)$;

$a_{-3} = a_{-2} = a_{-1} = 0$; $a_0 = 1$; $a_1 = s - u$;

$c_1 = a_1 - s$; $\beta_0 = c_1$; $\phi_0 = -\beta_0$;

$F_0(z) = 1$; $F_1(z) = z + \rho\phi_0$.

For $n = 1, n_{\max} - 1$

$$a_{n+1} = [(2n+1)(s-u)a_n - 2n(s^2 - su - 1)a_{n-1} + (2n-1)(s-u)a_{n-2} + (1-n)a_{n-3}]/(n+1);$$

$$c_{n+1} = a_{n+1} - sa_n + a_{n-1};$$

$\beta_n = (c_{n+1} - \sum_{l=1}^{n-1} l c_{n-l}\beta_l)/(n+1);$

$\phi_n = (z-\beta_0)\phi_{n-1} - \sum_{k=1}^{n-1} \beta_k(\phi_{n-k-1} + z^{n-k}) - (1+n)\beta_n - \beta_0 z^n;$

$\Phi_{n+1} = z^{n+1} + \phi_n; \quad F_{n+1}(z) = \rho^{n+1}\Phi_{n+1}\left(\frac{z}{\rho}\right).$

end.

Example With $n_{\max} = 2$ we obtain

$a_0 = 1; \quad a_1 = s - u;$

$c_1 = -u; \quad \beta_0 = -u;$

$\phi_0 = u;$

$a_2 = (s^2 - 4su + 3u^2 + 2)/2;$

$c_2 = (-s^2 - 2su + 3u^2 + 4)/2;$

$\beta_1 = (-s^2 - 2su + 3u^2 + 4)/4;$

$\phi_1 = 2uz + (s^2 + 2su - u^2 - 4)/2;$

$a_3 = (-s^3 - 3s^2u + 9su^2 - 5u^3 + 8s - 8u)/2;$

$c_3 = (-2s^3 + s^2u + 6su^2 - 5u^3 + 8s - 10u)/2;$

$\beta_2 = (-4s^3 + s^2u + 10su^2 - 7u^3 + 16s - 16u)/12;$

$\phi_2 = 3uz^2 + (3s^2 + 6su + 3u^2 - 12)z/4 + (2s^3 + s^2u - 2su^2 + u^3 + 2u - 8s)/2.$

Then

$$\begin{aligned}
\Phi_1(z) &= z + u, \\
\Phi_2(z) &= z^2 + \phi_1 = 2uz + (s^2 + 2su - u^2 - 4)/2, \\
\Phi_3(z) &= z^3 + 3uz^2 + (3s^2 + 6su + 3u^2 - 12)z/4 + (2s^3 + s^2u - 2su^2 + u^3 + 2u - 8s)/2.
\end{aligned}$$

2.4.1 Special Cases of Faber Polynomials

As mentioned previously there are a few special cases that provide checks on our work. In this section we will show that the first few Faber polynomials are in agreement with the first few known Faber polynomials for the special case regions.

Case (i) $b = a$

When $b = a$ we have

$$s = \frac{2(1+a^4)}{(1-a^4)} = u.$$

In this case

$$\begin{aligned}
\Phi_1(z) &= z + \frac{2(1+a^4)}{(1-a^4)} \\
\Phi_2(z) &= z^2 + \frac{4(1+a^4)}{(1-a^4)}z + \frac{4(1+a^4)^2}{(1-a^4)^2} - 2 \\
\Phi_3(z) &= z^3 + \frac{6(1+a^4)}{(1-a^4)}z^2 + 3\left[\frac{4(1+a^4)^2}{(1-a^4)^2} - 1\right]z + \frac{2(1+a^4)}{(1-a^4)}\left[\frac{4(1+a^4)^2}{(1-a^4)^2} - 3\right]
\end{aligned}$$

so that (see (2.30))

$$\begin{aligned}
F_1(z) &= z + \frac{(1+R)}{2} \\
F_2(z) &= z^2 + (1+R)z + \frac{1}{8}\left[2(1+R)^2 - (1-R)^2\right] \\
F_3(z) &= z^3 + \frac{3}{2}(1+R)z^2 + \frac{3}{16}\left[4(1+R)^2 - (1-R)^2\right]z \\
&\qquad + \frac{1}{32}\left[4(1+R)^3 - 3(1+R)(1-R)^2\right],
\end{aligned}$$

where we have used $\rho = (1-R)/4$ and $a^4 = R$.

As we saw earlier, in this case the annular sector becomes the real interval $[-1, -R]$. The Faber polynomials should then become scaled multiples of the Chebyshev polynomials for the interval $[-1, 1]$. For $[-1, 1]$ the first few Chebyshev polynomials are

$$T_1(x) = x, \quad T_2(x) = 2x^2 - 1, \quad T_3(x) = 4x^3 - 3x.$$

Mapping $[-1, 1]$ to $[-1, -R]$ using the transformation

$$z = \frac{1}{2}(1-R)x - \frac{1}{2}(1+R)$$

gives us

$$P_1(z) = \frac{2z}{(1-R)} + \frac{1+R}{1-R}$$

$$P_2(z) = \frac{8z^2}{(1-R)^2} + \frac{8z(1+R)}{(1-R)^2} + \frac{2(1+R)^2}{(1-R)^2} - 1$$

$$P_3(z) = \frac{32z^3}{(1-R)^3} + \frac{48z^2(1+R)}{(1-R)^3} + z\left[\frac{24(1+R)^2}{(1-R)^3} - \frac{6}{(1-R)}\right] + \frac{4(1+R)^3}{(1-R)^3} - \frac{3(1+R)}{(1-R)}$$

Finally, scaling these polynomials so that they are monic, we see that we have agreement with $F_1(z)$, $F_2(z)$, $F_3(z)$ above. So the first few generated Faber polynomials agree with the known results in this special case.

Case (ii) $b = 1$

When $b = 1$ the annular sector becomes an arc. From example (c) in Section 1.2.4 we know that the first few Faber polynomials for this region are given by

$$\begin{aligned} F_1(z) &= z + (1-\rho^2) \\ F_2(z) &= z^2 + 2(1-\rho^2)z + (1-\rho^4) \\ F_3(z) &= z^3 + 3(1-\rho^2)z^2 + 3(1-\rho^2)z + (1-\rho^6) \end{aligned}$$

where $\rho = (1-a^2)/(1+a^2)$. Using equation (2.30) we find that

$$\Phi_1(z) = z + \frac{(1-\rho^2)}{\rho}$$

$$\Phi_2(z) = z^2 + \frac{2(1-\rho^2)}{\rho}z + \frac{(1-\rho^4)}{\rho^2}$$

$$\Phi_3(z) = z^3 + \frac{3(1-\rho^2)}{\rho}z^2 + \frac{3(1-\rho^2)}{\rho^2}z + \frac{(1-\rho^6)}{\rho^3}.$$

When $b = 1$ we have

$$s = \frac{2(1+a^4)}{(1-a^4)}, \qquad u = \frac{4a^2}{(1-a^4)}$$

and the first few Φ_i's generated by our algorithm become

$$\Phi_1(z) = z + \frac{4a^2}{1-a^4}$$

$$\Phi_2(z) = z^2 + \frac{8a^2}{1-a^4}z + \frac{8a^2(1+a^4)}{(1-a^4)^2}$$

$$\Phi_3(z) = z^3 + \frac{12a^2}{1-a^4}z^2 + \frac{12a^2}{(1-a^2)^2}z + \frac{4a^2(3a^8+10a^4+3)}{(1-a^4)^3}.$$

Remembering that $\rho = (1-a^2)/(1+a^2)$, the coefficients of both forms of the above polynomials are all the same, for example

$$\frac{1-\rho^2}{\rho} = \frac{1+a^2}{1-a^2}\left[1 - \frac{(1-a^2)^2}{(1+a^2)^2}\right] = \frac{4a^2}{(1-a^4)}.$$

So that, once again, the first three generated Faber polynomials agree with the known special case.

Case (iii) $a \to 0, b \to 0, a/b = \lambda = \theta/\pi$

In the limit as $a \to 0$ and $b \to 0$, the annular sector becomes a sector of the unit disc. In Section 1.2.4 example (d) we give the first few Faber polynomials for this region in terms of a parameter c used by Gatermann et al. [39]. When the half–angle of the sector is written as π/α the parameter $c = (2 - 1/\alpha)/\alpha$.

In this case we have $s \to 2$ and

$$\begin{aligned} u \to 2\lambda^2 &= 2\left(\frac{\theta}{\pi}\right)^2 \\ &= 2\left(\frac{\pi - \pi/\alpha}{\pi}\right)^2 \\ &= 2\left[1 - \frac{1}{\alpha}\left(2 - \frac{1}{\alpha}\right)\right] \\ &= 2(1-c). \end{aligned}$$

The first few polynomials generated by our algorithm become

$$\begin{aligned} \Phi_1(z) &= z + 2(1-c) \\ \Phi_2(z) &= z^2 + 4(1-c)z + 2(1+c)(1-c) \\ \Phi_3(z) &= z^3 + 6(1-c)z^2 + (9-3c)(1-c)z + (1-c)(4c^2+2) \end{aligned}$$

Following Coleman and Smith [13], Gaterman et al. [39] considered a sector symmetric with respect to the positive real axis. Therefore, to compare the above polynomials with those generated by Gatermann et al. [39] we need to symmetrically position our annular sector on the positive real axis, this involves the mappings $z \to -z$ and $w \to -w$ (see Section 2.2). The overall effect of these changes is that instead of $\Phi_n(z)$ we must consider $(-1)^n\Phi_n(-z)$. A comparison with the polynomials given in Section 1.2.4, example (d), shows that our first few generated polynomials once again agree with the known results in this case.

2.4.2 The coefficients of $\phi_n(z)$

Letting

$$\phi_n(z) = \sum_{j=0}^{n} p_{n,j} z^{n-j} \tag{2.33}$$

in the recurrence relation (2.32) gives

$$\sum_{j=0}^{n} p_{n,j} z^{n-j} = \sum_{j=0}^{n-1} p_{n-1,j} z^{n-j} - \sum_{k=0}^{n-1} \beta_k \sum_{j=0}^{n-k-1} p_{n-k-1,j} z^{n-k-1-j} - \sum_{k=0}^{n-1} \beta_k z^{n-k} - (1+n)\beta_n.$$

In the second sum of the righthand side, we make the substitution $j = t - k - 1$ and therefore

$$\sum_{k=0}^{n-1} \beta_k \sum_{j=0}^{n-k-1} p_{n-k-1,j} z^{n-k-1-j} = \sum_{k=0}^{n-1} \sum_{t=k+1}^{n} \beta_k p_{n-k-1,t-k-1} z^{n-t}.$$

Writing $\sum_{k=0}^{n-1} \sum_{t=k+1}^{n}$ as $\sum_{t=1}^{n} \sum_{k=0}^{t-1}$ implies that

$$\sum_{j=0}^{n} p_{n,j} z^{n-j} = \sum_{j=0}^{n-1} p_{n-1,j} z^{n-j} - \sum_{t=1}^{n} \sum_{k=0}^{t-1} \beta_k p_{n-k-1,t-k-1} z^{n-t} - \sum_{k=0}^{n-1} \beta_k z^{n-k} - (1+n)\beta_n.$$

By equating coefficients of z^n we find that

$$p_{n,0} = p_{n-1,0} - \beta_0;$$

equating coefficients of z^0 implies that

$$\begin{aligned} p_{n,n} &= -\sum_{k=0}^{n-1} \beta_k p_{n-k-1,n-k-1} - (1+n)\beta_n \\ &= -\sum_{l=1}^{n} \beta_{n-l} p_{l-1,l-1} - (1+n)\beta_n; \end{aligned}$$

and for $i = 1, \ldots, n-1$ equating coefficients of z^i implies that

$$\begin{aligned} p_{n,n-i} &= p_{n-1,n-i} - \sum_{k=0}^{n-i-1} \beta_k p_{n-k-1,n-i-k-1} - \beta_{n-i} \\ &= p_{n-1,n-i} - \sum_{l=i+1}^{n} \beta_{n-l} p_{l-1,l-i-1} - \beta_{n-i}. \end{aligned}$$

Grouping these results together and using $p_{0,0} = u$, and $\beta_0 = -u$ we find that

$$p_{n,0} = p_{n-1,0} - \beta_0 = (n+1)u, \tag{2.34a}$$

$$p_{n,n} = -\sum_{k=1}^{n} \beta_{n-k} p_{k-1,k-1} - (1+n)\beta_n \tag{2.34b}$$

and, for $i = 1, \ldots, n-1$,

$$p_{n,n-i} = p_{n-1,n-i} - \sum_{k=i+1}^{n} \beta_{n-k} p_{k-1,k-1-i} - \beta_{n-i}. \tag{2.34c}$$

In the interest of brevity we shall use the term A–polynomial to describe a polynomial in two variables, s and u, which is invariant or changes sign, under the planar antipodal map $(s,u) \to (-s,-u)$, according as the degree of the polynomial is even or odd; in other words, such a polynomial of even (odd) degree contains only terms of the form $s^i u^j$ where $i+j$ is even (odd).

THEOREM 8. The coefficient $p_{n,j}$, for $n = 0, 1, \ldots$ and $j = 0, \ldots, n$, is an A–polynomial of degree $j+1$ in s and u.

Proof. An induction argument applied to the recurrence relation (2.28) shows that a_k is a polynomial in s and u of degree k. Furthermore, since $s-u$ and $s^2 - su - 1$ are A–polynomials of degree 1 and 2, respectively, the hypothesis that a_k is an A–polynomial of degree k for $k = 0, \ldots, n$ leads to the conclusion that the same property holds for $k = n+1$;

the induction hypothesis is readily confirmed for $n = 1$. It then follows from (2.26) that c_n is an A–polynomial of degree n, and an induction argument applied to equation (2.27) shows that β_n is an A–polynomial of degree $n + 1$.

Turning now to the equations (2.34), we assume, as an induction hypothesis, that for each n the coefficient $p_{n,j}$ is an A–polynomial of degree $j + 1$. Then each term on the right–hand side of (2.34c) is an A–polynomial of degree $n - i + 1$, and (2.34b) gives the corresponding result for $i = 0$; finally (2.34a) shows that $p_{n,0}$ is an A–polynomial of degree 1. Clearly the hypothesis is true for $n = 1$. □

In view of Theorem 8, we may write

$$\begin{aligned} p_{n-1,0} &= \gamma_{n1}u \\ p_{n-1,1} &= \gamma_{n2} + \gamma_{n3}s^2 + \gamma_{n4}su + \gamma_{n5}u^2 \\ p_{n-1,2} &= \gamma_{n6}s + \gamma_{n7}u + \gamma_{n8}s^3 + \gamma_{n9}s^2u + \gamma_{n10}su^2 + \gamma_{n11}u^3 \end{aligned}$$

etc. In keeping with the notation of Gatermann et al. [39] we regard the coefficients γ_{nk} as the elements of the n^{th} row of a matrix

$$\Gamma = (\gamma_{nk}), \qquad n = 1, 2, \ldots;\ k = 1, 2, \ldots, m(n)$$

where $m(n)$ is the number of terms in the n^{th} row of Γ.

THEOREM 9. The elements of the matrix Γ are rational numbers and the number of elements in the n^{th} row is

$$m(n) = \frac{1}{24}(2n^3 + 15n^2 + 37n - 30) - \frac{1}{4}\left[\frac{n-1}{2}\right]$$

where $[x]$ denotes the integer part of x.

Proof. It is clear from the various defining equations that the coefficients of the polynomials $p_{n,j}$ are rational numbers.

A homogeneous polynomial of degree j has $j + 1$ terms. An A–polynomial of degree $2r$ is a sum of homogeneous polynomials of even degree from 0 to $2r$ inclusive; it consists of

$$\sum_{l=0}^{r}(2l + 1) = (r + 1)^2$$

terms. Similarly an A–polynomial of degree $2r+1$ has

$$\sum_{l=1}^{r+1} 2l = r^2 + 3r + 2$$

terms.

Equation (2.34a) shows that $p_{n,0}$ consists of a single non–zero term. For $i \geq 2$, the polynomial $p_{n,i-1}$, being an A–polynomial of degree i, has $\frac{1}{4}(i+2)^2$ terms if i is even and $\frac{1}{4}(i+2)^2 - \frac{1}{4}$ terms if i is odd. For a given n the total number of terms is

$$\begin{aligned} m(n) &= 1 + \frac{1}{4}\sum_{i=2}^{n}(i+2)^2 - \frac{1}{4}\left[\frac{n-1}{2}\right] \\ &= \frac{1}{24}(2n^3 + 15n^2 + 37n - 30) - \frac{1}{4}\left[\frac{n-1}{2}\right]. \quad \square \end{aligned}$$

A computer algebra system may be used to compute the polynomials $p_{n,j}$ from (2.34), as polynomials in s and u. We have used Mathematica and REDUCE for this purpose. The polynomials $p_{n-1,j}$, for $j = 0(1)n-1$ and $n = 0(1)15$, are listed in Appendix B where $p_{n-1,j}$ is denoted by $p(j)$, for each value of n. The results given there may be used with (2.30), (2.31) and (2.33) to construct the Faber polynomials of degree ≤ 15.

The first ten rows of the matrix Γ have the form

$$\begin{pmatrix}
1 & & & & & & & & & & & \\
2 & -2 & 1/2 & 1 & -1/2 & & & & & & & \\
3 & -3 & 3/4 & 3/2 & 3/4 & -4 & 1 & 1 & 1/2 & -1 & 1/2 & \\
4 & -4 & 1 & 2 & 3 & -16/3 & -8/3 & 4/3 & 5/3 & 2/3 & 1/3 & \ldots \\
5 & -5 & 5/4 & 5/2 & 25/4 & -20/3 & -25/3 & 5/3 & 10/3 & 10/3 & 5/3 & \ldots \\
6 & -6 & 3/2 & 3 & 21/2 & -8 & -16 & 2 & 11/2 & 7 & 11/2 & \ldots \\
7 & -7 & 7/4 & 7/2 & 63/4 & -28/3 & -77/3 & 7/3 & 49/6 & 35/3 & 77/6 & \ldots \\
8 & -8 & 2 & 4 & 22 & -32/3 & -112/3 & 8/3 & 34/3 & 52/3 & 74/3 & \ldots \\
9 & -9 & 9/4 & 9/2 & 117/4 & -12 & -51 & 3 & 15 & 24 & 42 & \ldots \\
10 & -10 & 5/2 & 5 & 75/2 & -40/3 & -200/3 & 10/3 & 115/6 & 95/3 & 395/6 & \ldots
\end{pmatrix}$$

and other entries may be read from Appendix B. As Gatermann et al. [39] found for circular sectors, all entries in a column of Γ are expressible in terms of a polynomial in the row index. For example, we find

$$\gamma_{n1} = n, \qquad \gamma_{n11} = \frac{29}{12}n - \frac{5}{4}n^2 + \frac{1}{6}n^3,$$

$$\begin{aligned}
\gamma_{n2} &= -n, & \gamma_{n12} &= -\frac{3}{2}n + \frac{1}{2}n^2,\\
\gamma_{n3} &= \frac{1}{4}n, & \gamma_{n13} &= -\frac{1}{2}n - \frac{1}{4}n^2,\\
\gamma_{n4} &= \frac{1}{2}n, & \gamma_{n14} &= \frac{25}{3}n - \frac{11}{6}n^2,\\
\gamma_{n5} &= \frac{1}{2}n^2 - \frac{5}{4}n, & \gamma_{n15} &= \frac{-65}{6}n + \frac{55}{12}n^2 - \frac{1}{2}n^3,\\
\gamma_{n6} &= -\frac{4}{3}n, & \gamma_{n16} &= \frac{21}{96}n + \frac{1}{32}n^2,\\
\gamma_{n7} &= \frac{10}{3}n - n^2, & \gamma_{n17} &= -\frac{47}{24}n + \frac{11}{24}n^2,\\
\gamma_{n8} &= \frac{1}{3}n, & \gamma_{n18} &= \frac{43}{48}n - \frac{37}{48}n^2 + \frac{1}{8}n^3,\\
\gamma_{n9} &= -\frac{7}{12}n + \frac{1}{4}n^2, & \gamma_{n19} &= \frac{149}{24}n - \frac{59}{24}n^2 + \frac{1}{4}n^3,\\
\gamma_{n10} &= -\frac{11}{6}n + \frac{1}{2}n^2, & \gamma_{n20} &= -\frac{539}{96}n + \frac{307}{96}n^2 - \frac{5}{8}n^3 + \frac{1}{24}n^4.
\end{aligned}$$

2.5 The mapping of the boundary

Under the mapping ψ the boundary of the unit disc is mapped onto the boundary of the annular sector. It will be useful, particularly when we consider level curves (see Section 3.5), to know which points on the unit disc map to the corners of the annular sector. For this reason we consider the points $w = \exp(i\eta)$ with $0 \leq \eta < 2\pi$. The map ζ defined earlier (see equation (2.5)) maps these points onto

$$\zeta\left(e^{i\eta}\right) = \frac{1}{2}\left[\left(e^{i\eta} + e^{-i\eta}\right)\sinh\alpha\right] - \cosh\alpha = \frac{1}{2}\left[(a^{-2} - a^2)\cos\eta - (a^{-2} + a^2)\right].$$

We know that the points $\pm a$, $\pm a^{-1}$, $\pm b$, and $\pm\, b^{-1}$ play a key role in the conformal mapping and we also know that the mapping $\zeta = -v^{-2}$ maps the upper half-plane onto the cut plane $\mathbb{C}/L$ (see Section 2.2). Hence the points on the unit disc that map to the important points in the upper half-plane are given by

$$-v^{-2} = \frac{1}{2}\left[(a^{-2} - a^2)\cos\eta - (a^{-2} + a^2)\right],$$

where $v = \pm a$, $\pm a^{-1}$, $\pm b$, or $\pm\, b^{-1}$. When

$$v = \begin{cases} \pm a & \text{then} \quad \cos\eta = -1 \quad \text{and} \quad \eta = \pi \\ \pm a^{-1} & \text{then} \quad \cos\eta = 1 \quad \text{and} \quad \eta = 0. \end{cases}$$

Also when

$$v = \begin{cases} \pm b^{-1} & \text{we have} \quad \cos\eta = t \qquad \text{so} \quad \eta = t_1 \ \text{ or } \ 2\pi - t_1 \\ \pm b & \text{then} \quad \cos\eta = \tau \qquad \text{and} \quad \eta = \tau_1 \ \text{ or } \ 2\pi - \tau_1, \end{cases}$$

where $t_1,\ \tau_1 \in [0,\pi]$, $\cos(t_1) = t$ and $\cos(\tau_1) = \tau$. Previously we have shown that $|t| \leq 1$ and $|\tau| \leq 1$ so solutions to the equations $\cos\eta = t$ and $\cos\eta = \tau$ will exist. We have also shown that $t \geq \tau$ and therefore because $\cos\eta$ is a monotonic decreasing function of η on $[0,\pi]$ we have $t_1 \leq \tau_1$. We now remember that the upper half–plane is mapped onto the exterior of our annular sector by a combination of the exponential map and a Schwarz–Christoffel map. If as before we write $u(v)$ for the Schwarz–Christoffel map, then $\exp[u(\pm a^{-1})] = -R$, $\exp[u(\pm a)] = -1$, $\exp[u(b^{-1})] = Re^{i\theta}$, $\exp[u(-b^{-1})] = Re^{-i\theta}$, $\exp[u(b)] = e^{i\theta}$, and $\exp[u(-b)] = e^{-i\theta}$. So considering $w = \exp(i\eta)$ we find:

$\eta = 0$ is mapped to $-R$ in the z-plane;

$\eta = \pi$ is mapped to -1 in the z-plane;

$\eta = t_1$ is mapped to $R\exp(i\theta)$ in the z-plane;

$\eta = \tau_1$ is mapped to $\exp(i\theta)$ in the z-plane;

$\eta = 2\pi - \tau_1$ is mapped to $\exp(-i\theta)$ in the z-plane;

$\eta = 2\pi - t_1$ is mapped to $R\exp(-i\theta)$ in the z-plane.

Pictorially this can be represented by Figure 2.5.

This will be important when we consider level plots in Section 3.5, because when we consider a curve of radius slightly greater than 1 in the w-plane, then the part of the curve with $\eta \in (0, t_1)$ (that is region A in Figure 2.5) should be mapped under ψ to the arc $(-R, -Re^{-i\theta})$ in the z-plane (that is A' in Figure 2.5). A similar correspondence should occur for regions marked with the same letters in both the w and z planes.

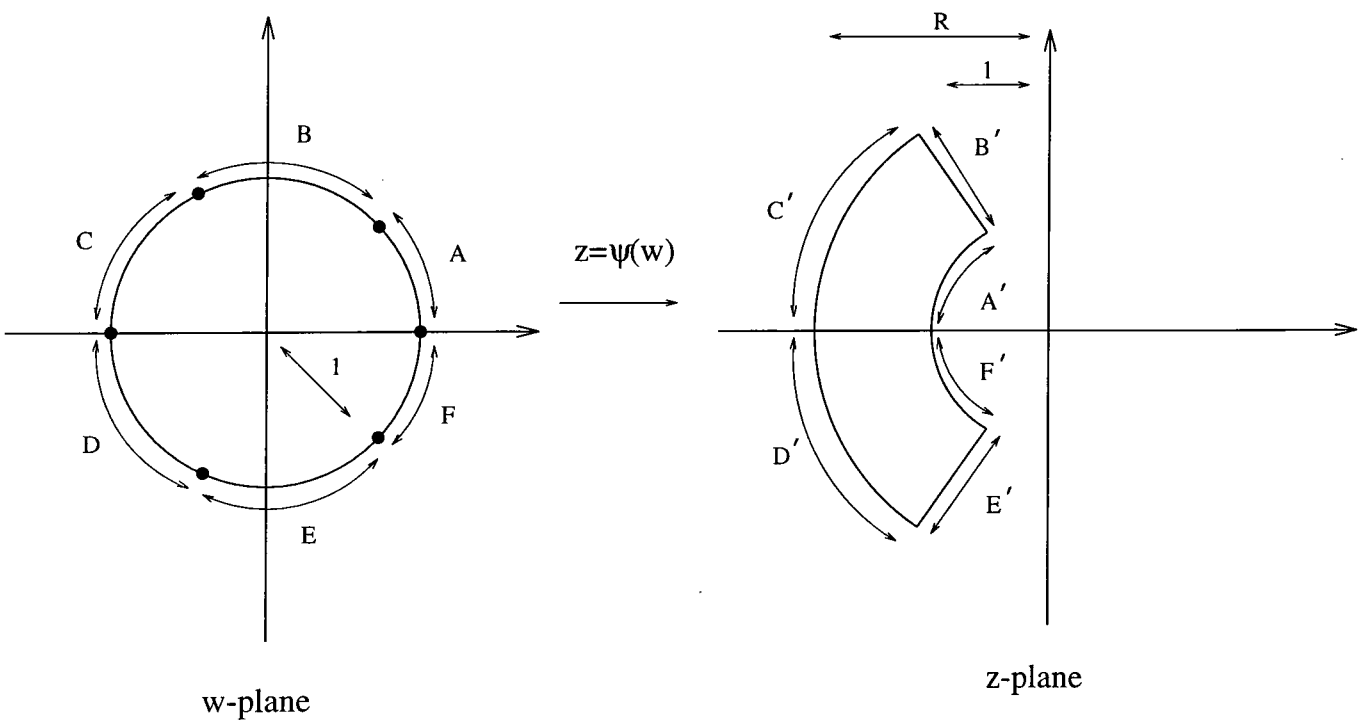

Figure 2.5: A pictorial representation of the mapping of the boundary of the unit disc.

2.6 Scaling the Annular Sector

In this section we shall consider the mapping of the complement of the unit disc onto the complement of the annular sector

$$Q(R_1, R_2, \theta) = \{R_1 \leq |Z| \leq R_2,\ \theta \leq |\arg Z| \leq \pi\}.$$

There are two main reasons for doing this. Firstly it may not be obvious that the ratio of the radii is the important quantity in determining the mapping and hence the Faber polynomials. The second reason is concerned with chapter 5 in which we will consider, amongst other things, how to find the Faber polynomials for any annular sector placed anywhere in the complex plane. The results given in this section are part of this, as is the use of the transformations $z \to -z$ and $w \to -w$ to find the Faber polynomials for an annular sector centred on the positive real axis from the annular sector centred on the negative real axis (see Section 2.2 and the end of Section 2.4.1).

We note that by writing $R_2 R = R_1$, the annular sector $Q(R_1, R_2, \theta)$ is simply the

annular sector Q scaled so that its outer radius is R_2; that is to the annular sector Q in the z-plane we apply the map $Z = R_2 z$ to get $Q(R_1, R_2, \theta)$. From Section 2.2, $\psi(w)$ maps the complement of the unit disc to the complement of the annular sector Q, so $\Psi(w) = R_2\psi(w)$ will map the exterior of the unit disc to the complement of $Q(R_1, R_2, \theta)$. Hence

$$\lim_{w\to\infty} \frac{\Psi(w)}{w} = R_2\rho,$$

and the transfinite diameter of $Q(R_1, R_2, \theta)$ is

$$\bar{\rho} := R_2\rho = \frac{R_2(1-a^4)}{4} \exp\left[\int_0^{a^2} \frac{C}{A(x)[A(x)+B(x)]}\, dx\right],$$

see equation (2.17).

In a similar manner to Section 1.2.2 we define $W = \Phi(Z) := R_2\rho\Psi^{-1}(Z)$; this maps the complement of $Q(R_1, R_2, \theta)$ to the complement of a disc of radius $R_2\rho$. Also,

$$\begin{aligned}
\lim_{Z\to\infty} \frac{\Phi(Z)}{Z} &= \lim_{Z\to\infty} \frac{R_2\rho\Psi^{-1}(Z)}{Z} \\
&= \lim_{W\to\infty} \frac{W}{\Psi\left(\frac{W}{R_2\rho}\right)} \\
&= \lim_{w\to\infty} \frac{R_2\rho w}{\Psi(w)} = 1;
\end{aligned}$$

$\Psi(Z)$ satisfies relation (1.1), and the mapping is the desired one. In this way the expressions (2.13) for the mapping become

$$\begin{aligned}
z &= a^2 R_2 \zeta \exp\left[-\int_{a^2}^{-\zeta^{-1}} \frac{C}{A(x)[A(x)+B(x)]}\, dx\right] \\
&= a^2 R_2 \zeta \exp\left[\int_{a^{-2}}^{-\zeta} \frac{C}{A(x)[A(x)+B(x)]}\, dx\right] \\
&= \frac{R_1\zeta}{a^2} \exp\left[\int_{a^2}^{-\zeta} \frac{C}{A(x)[A(x)+B(x)]}\, dx\right] \\
&= \frac{R_1\zeta}{a^2} \exp\left[-\int_{a^{-2}}^{-\zeta^{-1}} \frac{C}{A(x)[A(x)+B(x)]}\, dx\right].
\end{aligned}$$

The defining equations for a and b, that is (2.7) and (2.8), remain the same except that R is replaced by the ratio of the radii, R_1/R_2. In this way we see that the quantities determining a and b (and hence the mapping and the Faber polynomials), are the half-angle, $\pi - \theta$, of the sector and the ratio, R_1/R_2, of the inner and outer radii.

2.7 The Faber series for z^{-1} on an Annular Sector

The Faber series for a function f, analytic in the annular sector Q, is an expression of the form

$$\sum_{j=0}^{\infty} a_j F_j(z).$$

See Curtiss [15], Markushevich ([57], v. 3, p. 109), or Gaier ([38], p. 44). The coefficients are

$$a_j = \frac{1}{2\pi i \rho^j} \int_{|w|=R_1} \frac{f(\psi(w))}{w^{j+1}}\, dw \tag{2.35}$$

where $R_1 > 1$ is sufficiently small that f can be extended analytically to the closed region bounded by the image under ψ of the circle $|w| = R_1$. In particular, when $f(z) = z^{-1}$ the Faber series is

$$\frac{1}{z} = -\frac{1}{w^* \psi'(w^*)} \left[1 + \sum_{n=1}^{\infty} \frac{\tilde{F}_n(z)}{(w^*)^n} \right], \tag{2.36}$$

where $\tilde{F}_n(z)$ is the scaled Faber polynomial introduced in (2.30), and w^* is the root of magnitude greater than 1 of the equation $\psi(w) = 0$; in other words, w^* is the point which ψ maps to the origin in the z-plane. Equation (2.36) may be established either by applying Cauchy's residue theorem to (2.35) or, as in Chiu et al. [8], by using a generating function for Faber polynomials and the uniqueness of the Faber series.

We begin with Cauchy's residue theorem. We note that $f(z) = z^{-1}$ is analytic in any annular sector, Q, with inner radius $R > 0$, so if we have such an annular sector we may take $R_1 = 1$ in the above formula for the coefficients of the Faber series. Therefore the coefficients are given by

$$a_j = \frac{1}{2\pi i \rho^j} \int_{|w|=1} \frac{1}{\psi(w) w^{j+1}}\, dw.$$

In this equation we make the substitution $w = 1/u$ and note this maps the outside of the unit disc to the inside of the unit disc and causes us to integrate around the unit circle in the opposite direction (so introducing a negative sign into the calculation). The coefficients then become

$$a_j = \frac{1}{2\pi i \rho^j} \int_{|u|=1} \frac{u^{j-1}}{\psi\left(\frac{1}{u}\right)}\, du. \tag{2.37}$$

To use Cauchy's residue theorem we must calculate the residues at the poles of this expression. From the above we know that $\psi(w) = 0$ when $w = w^*$, so $\psi(1/u) = 0$ when $u = 1/w^*$. Hence at this point the residue of the integrand in equation (2.37) is given by

$$\lim_{u \to 1/w^*} \frac{\left(u - \frac{1}{w^*}\right) u^{j-1}}{\psi\left(\frac{1}{u}\right) - \psi\left(w^*\right)} = \lim_{u \to 1/w^*} -\frac{u^{j+1}}{\psi'\left(\frac{1}{u}\right)} = -\frac{1}{(w^*)^{j+1}\,\psi'\left(w^*\right)}.$$

From this and equation (2.37) we find the coefficients of the Faber series for $1/z$ are given by

$$a_j = \frac{-1}{\rho^j (w^*)^{j+1} \psi'\left(w^*\right)}.$$

It is straightforward from this expression to see that the Faber series for $1/z$ is given by equation (2.36).

Chiu et al. [8] give a different and simpler proof of this result. We start from Property 4 (see Section 1.2.2) and choose w^* so that $\psi(w) = 0$. This immediately gives us

$$-\frac{1}{z} = \frac{1}{w^* \psi'(w^*)} \left[1 + \sum_{n=1}^{\infty} \frac{\widetilde{F}_n(z)}{(w^*)^n}\right].$$

For an annular sector a simple formula can be found for the coefficients in (2.36). It is clear from (2.13) that $z = 0$ implies $\zeta = 0$ and therefore

$$w^* = \frac{1 + a^2}{1 - a^2}. \tag{2.38}$$

From (2.13c) and (2.22) we obtain

$$w\psi'(w) = \frac{R(1 - a^4)}{4wa^4} \exp\left[\int_{a^2}^{-\zeta} \frac{C\ dx}{A(x)[A(x) + B(x)]}\right] M(w, t) M(w, \tau). \tag{2.39}$$

Setting $\zeta = 0$ in this expression, and noting that

$$\rho = \frac{(1 - a^4)}{4} \exp\left[\int_0^{a^2} \frac{C\ dx}{A(x)[A(x) + B(x)]}\right]$$

and

$$M(w^*,t)M(w^*,\tau) = \frac{2w^*}{\sinh\alpha}$$

$$= \frac{4a^2}{(1-a^2)^2}$$

we obtain

$$w^*\psi'(w^*) = \frac{R(1-a^4)}{4a^2\rho}. \tag{2.40}$$

Then, from equation (2.36), the Faber series for z^{-1} is

$$\frac{1}{z} = -\frac{4\rho a^2}{R(1-a^4)}\left[1+\sum_{n=1}^{\infty}\left(\frac{1-a^2}{1+a^2}\right)^n \tilde{F}_n(z)\right]. \tag{2.41}$$

As a check on (2.41) it can be shown that it correctly gives a known Chebyshev expansion when $b = a$ (case (i) of Section 2.2.4). The Faber polynomial of degree $n \geq 1$ for the interval $[-1,1]$ is $2^{1-n}T_n(x)$, where T_n is the Chebyshev polynomial of degree n. The corresponding polynomial for $z \in [-1,-R]$ is

$$F_n(z) = 2\left(\frac{1-R}{4}\right)^n T_n\left(\frac{2z+1+R}{1-R}\right).$$

With the help of results from Section 2.2.4 for this particular case (that is $a^4 = R$ and $\rho = (1-R)/4$), the expansion (2.41) becomes

$$\frac{1}{z} = -\frac{1}{\sqrt{R}}\left\{1+2\sum_{n=1}^{\infty}\left(\frac{1-\sqrt{R}}{1+\sqrt{R}}\right)^n T_n\left(\frac{2z+1+R}{1-R}\right)\right\}. \tag{2.42}$$

The Chebyshev expansion for $1/(x-\delta)$, with $\delta > 1$ and $x \in [-1,1]$, may be established by a technique used, for example, by Fox and Parker ([34], p. 85). We begin by considering

$$\frac{1}{x-\delta} = \sum_{n=0}^{\infty}{}' a_n T_n(x),$$

where the prime denotes that the first term of the sum is $a_0/2$. Multiplying this equation by $x-\delta$ we find

$$\begin{aligned} 1 &= (x-\delta)\sum_{n=1}^{\infty} a_n T_n(x) + \frac{1}{2}a_0(x-\delta)T_0(x) \\ &= \frac{1}{2}\sum_{n=1}^{\infty} a_n\{T_{n+1}(x)+T_{n-1}(x)-2\delta T_n(x)\} + \frac{1}{2}a_0T_1(x) - \frac{1}{2}a_0\delta T_0(x). \end{aligned}$$

To obtain this expression we have used the recurrence relation

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x), \qquad n \geq 1$$

for the Chebyshev polynomials, and the fact that $xT_0(x) = x = T_1(x)$. Considering the sums separately we can rewrite our expression as

$$\begin{aligned} 1 &= \frac{1}{2}\sum_{k=2}^{\infty} a_{k-1}T_k(x) + \frac{1}{2}\sum_{k=0}^{\infty} a_{k+1}T_k(x) - \delta\sum_{k=1}^{\infty} a_kT_k(x) + \frac{1}{2}a_0T_1(x) - \frac{1}{2}a_0\delta T_0(x). \\ &= \sum_{n=1}^{\infty} T_n(x)\left[\frac{a_{n-1}}{2} - \delta a_n + \frac{a_{n+1}}{2}\right] + \left(\frac{1}{2}a_1 - \frac{1}{2}a_0\delta\right)T_0(x). \end{aligned}$$

The Chebyshev polynomials are linearly independent. Therefore, equating coefficients of the Chebyshev polynomial of degree n on both sides of this expression gives

$$a_1 - a_0\delta = 2 \tag{2.43a}$$

and for $n \geq 1$

$$a_{n+1} - 2\delta a_n + a_{n-1} = 0. \tag{2.43b}$$

The difference equation given by (2.43b) can be solved by standard methods, and the general solution is

$$a_n = A\left(\delta - \sqrt{\delta^2 - 1}\right)^n + B\left(\delta + \sqrt{\delta^2 - 1}\right)^n.$$

It is known, in this case, that the Chebyshev series converges (see Fox and Parker [34]), therefore $a_n \to 0$ as $n \to \infty$. This implies that B is zero (as $\delta + \sqrt{\delta^2 - 1} > \delta > 1$, see before). Substituting the expression for a_n into (2.43a) gives

$$A = -\frac{2}{\sqrt{\delta^2 - 1}}.$$

These results give the Chebyshev series as

$$\frac{1}{x - \delta} = -\frac{1}{\sqrt{\delta^2 - 1}}\left\{1 + 2\sum_{n=1}^{\infty}\left(\delta - \sqrt{\delta^2 - 1}\right)^n T_n(x)\right\}. \tag{2.44}$$

To prove this is the same as expression (2.42) we map $[-1, 1]$ to $[-1, -R]$ by the transformation

$$x = \frac{2z + 1 + R}{1 - R}.$$

This results in the left–hand side of equation (2.44) becoming

$$\frac{1-R}{2z+1+R-\delta(1-R)} = \frac{1-R}{2z},$$

if we choose $\delta = (1+R)/(1-R)$. From this choice of δ, we find

$$\sqrt{\delta^2-1} = \frac{2\sqrt{R}}{1-R}$$

and so

$$\delta - \sqrt{\delta^2-1} = \frac{1-\sqrt{R}}{1+\sqrt{R}}.$$

Substituting all these expressions into equation (2.44) gives equation (2.42) so verifying equation (2.41) in this case.

The maximum norm of the error in approximating z^{-1} on the domain Q by a truncated Faber series

$$q_n(z) = -\frac{1}{w^*\psi'(w^*)}\left(1+\sum_{k=1}^{n}(w^*)^{-k}\widetilde{F}_k(z)\right).$$

is easily bounded. From (2.36) we obtain

$$\begin{aligned}\left\|\frac{1}{z}-q_n(z)\right\|_\infty &= \max_{z\in D}\left|\frac{1}{w^*\psi'(w^*)}\sum_{k=n+1}^{\infty}(w^*)^{-k}\widetilde{F}_k(z)\right| \\ &\leq \frac{1}{|w^*\psi'(w^*)|}\sum_{k=n+1}^{\infty}|w^*|^{-k}\max_{z\in D}|\widetilde{F}_k(z)|\end{aligned}$$

We now use inequality (3.5) from the next chapter and note that the sum in the above expression is a geometric series with $1/|w^*| < 1$. Hence we find

$$\left\|\frac{1}{z}-q_n(z)\right\|_\infty \leq \frac{V|w^*|^{-n}}{\pi|w^*\psi'(w^*)|(|w^*|-1)}$$

(see also [8]) and from (2.38) and (2.40)

$$\left\|\frac{1}{z}-q_n(z)\right\|_\infty \leq \frac{2V\rho}{\pi R(1+a^2)}\left(\frac{1-a^2}{1+a^2}\right)^n.$$

Chapter 3

Numerics

3.1 Introduction

In this chapter we will discuss the numerical aspects of the Faber polynomials for annular sectors. To generate the Faber polynomials for an annular sector, by the algorithm in Section 2.4, we need three numbers, namely a, b and ρ. In Section 3.2 we describe our method to evaluate these numbers. The parameters a and b are found by combining modified Newton iteration with Kronrod-Patterson integration. Once a and b are known, ρ is found by Kronrod-Patterson integration. Any assessment of the accuracy of an approximation based on Faber polynomials requires some knowledge of a relevant norm. In Section 3.3 we introduce the area 2-norm, the line 2-norm and the maximum norm. Examples of these three norms are given for various annular sectors. In Section 3.4 we attempt to improve on a bound given in Ellacott[22] for the the norm of the Faber projection. Finally, in Section 3.5, we give a strategy to produce numerical level plots for an annular sector.

3.2 Numerical evaluation of a, b and ρ

To compute the Faber polynomials for a particular sector of an annulus we need to evaluate the Schwarz–Christoffel parameters (a and b) and the transfinite diameter ρ. Given R and θ defining a particular sector, a and b are found by solving the pair of non–linear equations (2.7) and (2.8). We used a modified Newton iteration in which the partial derivatives in the Jacobian were approximated by central–difference formulae of the form

$$\frac{\partial f(a,b)}{\partial a} \approx \frac{f(a+D,b)-f(a-D,b)}{2D}.$$

The convergence of the Newton iteration depends on having sufficiently good initial estimates of a and b. Table 3.1 gives suitable values for certain ranges of R and θ. The starting values, in this table, were obtained by trial and error. As θ tends to 0 or π and as R tends to 0 or 1 the convergence becomes much more sensitive to the choice of starting value, but the corresponding regions, which are close to known limits, are less likely to be of practical interest than those covered by Table 3.1. This sensitivity means that the accuracy of the starting values is more important for R tending to 0 or 1 and θ tending to 0 or π than for values in the centre of Table 3.1.

At each step of the Newton iteration it is necessary to evaluate the integrals in (2.7) and (2.8) numerically to an accuracy consistent with that required in the Newton iteration. Despite the square–root singularities at the end–points of the integration intervals, the NAG routine D01AHF, which is based on the Gauss–Kronrod–Patterson family of formulae, works satisfactorily. Figures 3.1 and 3.2 show a and b respectively, as functions of R and θ, for $0.001 \leq R \leq 0.999$ and $0.3 \leq \theta \leq 3.1414$.

To approximate the integrals in (2.7) and (2.8), integration methods other than the Kronrod-Patterson integration, in D01AHF, may be used. An obvious choice, because of the square-root singularities at the end points of these integrals, is Gauss-Chebyshev integration (see Section 3.4). This would have the advantage that the nodes and weights are fairly simple and easy to calculate. The idea would be to approximate the integral in question by Gauss-Chebyshev integration, with a certain number of nodes, and then to double the number of nodes and approximate the integral again. Comparing the two approximate values would reveal how close the approximation was to the value of the integral. If the difference was too large, then the number of nodes would be doubled until the difference between two successive approximations was satisfactory. Alternatively, a Kronrod-Patterson integration based on the Gauss-Chebyshev nodes may be used to evaluate the integrals. Both these ideas using Gauss-Chebyshev integration could prove more efficient that the Kronrod-Patterson integration employed in D01AHF, but the NAG routine works satisfactorily, so we decided to use it anyway.

The pictures in Figures 3.1 and 3.2 provide a useful visualisation of the special cases we have previously considered. For example as $R \to 1$ we see that $b \to 1$. Also as $\theta \to \pi$, a and b become the same function of R and we have $a \to b$. The case when $R \to 0$ is not so clear. Certainly for small θ we see that a and b both seem to tend towards zero (as they should). For larger values of θ, the vanishing of a and b as $R \to 0$ is not so obvious.

In Table 3.2 we give the a and b values (mostly to 9 decimal places) for some specific annular sectors. Most of the values in the table were produced numerically by the method described near the beginning of this section. The special cases when $\theta = \pi$, $R = 0$ and $R = 1$ have been added for completeness. These special cases can be found in Section 2.2.4, where the formulae used to calculate the values of a and b, in the table mentioned above,

can also be found. When $R = 0.001$ or $R = 0.0001$ and $\theta = 10°$ we found no starting values by our method of trial and error. For this reason there are blanks in Table 3.2. In Section 3.2.1 we discuss an interesting relationship between a, b and R which enables us to give approximate a and b for these annular sectors. The results are contained in Table 3.4.

Table 3.1. Starting values for modified Newton iteration.

R / θ	$0.999 - 0.1$	$0.1 - 0.02$	$0.02 - 0.01$	$0.01 - 0.005$	$0.005 - 0.001$
$3.1414 - 3.0$	$a = 0.489$ $b = 0.513$ $D = 1.0 \times 10^{-6}$	$a = 0.3760$ $b = 0.3761$ $D = 1.0 \times 10^{-6}$	$a = 0.3107$ $b = 0.3108$ $D = 1.0 \times 10^{-7}$	$a = 0.26591$ $b = 0.26592$ $D = 1.0 \times 10^{-8}$	$a = 0.17774$ $b = 0.17775$ 1.0×10^{-9}
$3.0 - 1.9$	$a = 0.49$ $b = 0.52$ $D = 0.001$	$a = 0.12$ $b = 0.24$ $D = 0.001$	$a = 0.09$ $b = 0.12$ $D = 0.001$	$a = 0.07$ $b = 0.12$ $D = 0.001$	$a = 0.07$ $b = 0.10$ $D = 0.001$
$1.9 - 1.57$	$a = 0.12$ $b = 0.24$ $D = 0.001$	$a = 0.12$ $b = 0.24$ $D = 0.001$	$a = 0.06$ $b = 0.10$ $D = 0.001$	$a = 0.06$ $b = 0.10$ $D = 0.0001$	$a = 0.01$ $b = 0.02$ $D = 0.0001$
$1.57 - 1.00$	$a = 0.05$ $b = 0.16$ $D = 0.001$	$a = 0.03$ $b = 0.08$ $D = 0.001$	$a = 0.01$ $b = 0.03$ $D = 0.001$	$a = 0.01$ $b = 0.03$ $D = 0.0001$	$a = 0.0005$ $b = 0.001$ $D = 0.0001$
$1.00 - 0.75$	$a = 0.05$ $b = 0.07$ $D = 1.0 \times 10^{-5}$	$a = 0.03$ $b = 0.08$ $D = 0.001$	$a = 6.0 \times 10^{-5}$ $b = 6.0 \times 10^{-4}$ $D = 1.0 \times 10^{-5}$	$a = 4.0 \times 10^{-5}$ $b = 3.0 \times 10^{-4}$ $D = 1.0 \times 10^{-5}$	$a = 4.0 \times 10^{-6}$ $b = 3.0 \times 10^{-5}$ $D = 1.0 \times 10^{-6}$
$0.75 - 0.50$	$a = 0.01$ $b = 0.06$ $D = 1.0 \times 10^{-5}$	$a = 0.03$ $b = 0.08$ $D = 0.001$	$a = 6.0 \times 10^{-5}$ $b = 6.0 \times 10^{-4}$ $D = 1.0 \times 10^{-5}$	$a = 4.0 \times 10^{-5}$ $b = 3.0 \times 10^{-4}$ $D = 1.0 \times 10^{-5}$	$a = 4.0 \times 10^{-6}$ $b = 3.0 \times 10^{-5}$ $D = 1.0 \times 10^{-6}$
$0.50 - 0.30$	$a = 1.1 \times 10^{-4}$ $b = 1.6 \times 10^{-3}$ $D = 1.0 \times 10^{-5}$	$a = 5.0 \times 10^{-6}$ $b = 5.0 \times 10^{-5}$ $D = 1.0 \times 10^{-6}$	$a = 1.0 \times 10^{-6}$ $b = 1.0 \times 10^{-5}$ $D = 1.0 \times 10^{-7}$	$a = 1.0 \times 10^{-7}$ $b = 1.0 \times 10^{-6}$ $D = 1.0 \times 10^{-8}$	$a = 1.0 \times 10^{-9}$ $b = 8.0 \times 10^{-9}$ $D = 1.0 \times 10^{-10}$

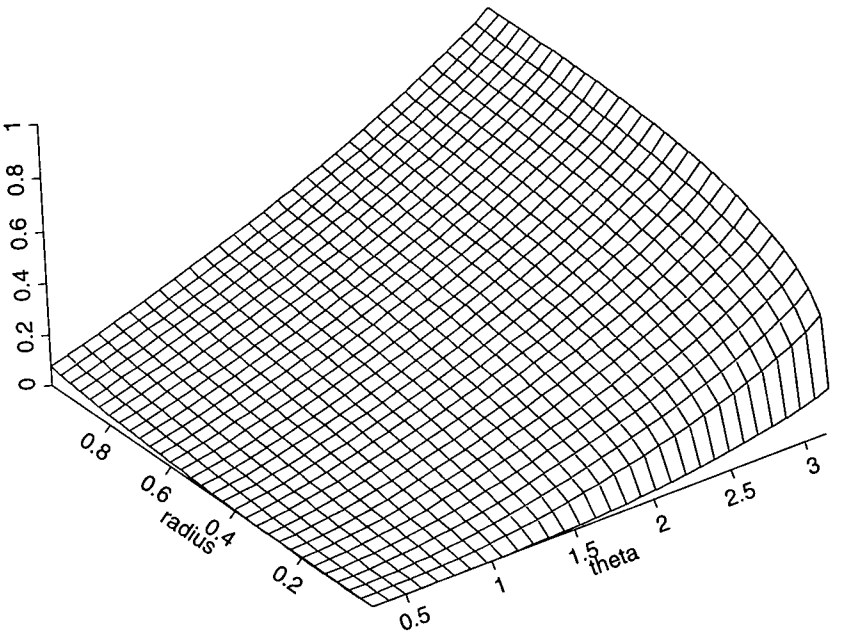

Figure 3.1: A graph of $a(R, \theta)$.

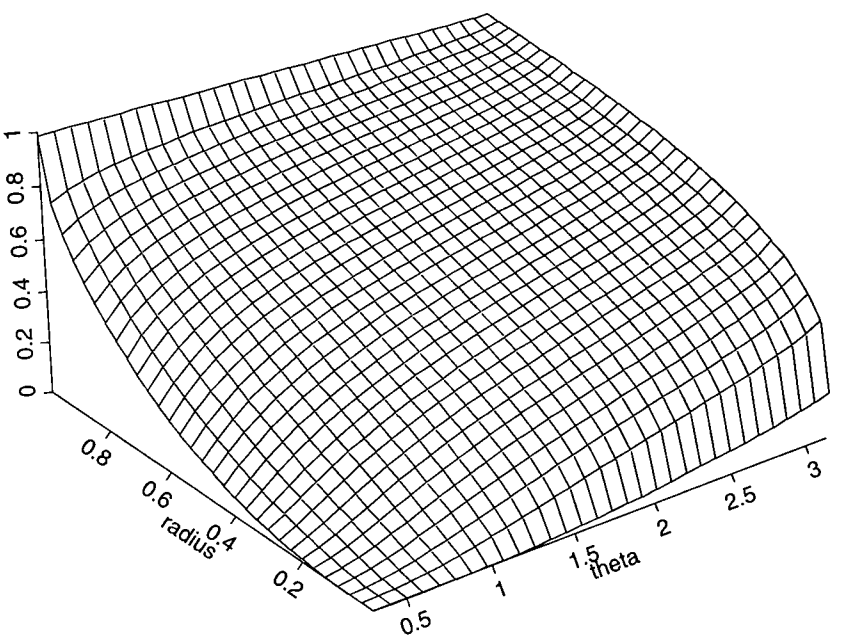

Figure 3.2: A graph of $b(R, \theta)$.

Table 3.2. Values of the parameters a and b for various annular sectors

R \ θ	180°	170°	135°	90°	45°	10°
1.0	$a = 1.000000000$ $b = 1.000000000$	$a = 0.916331174$ $b = 1.000000000$	$a = 0.668178637$ $b = 1.000000000$	$a = 0.414213562$ $b = 1.000000000$	$a = 0.198912367$ $b = 1.000000000$	$a = 0.043660942$ $b = 1.000000000$
0.9	$a = 0.974003746$ $b = 0.974003746$	$a = 0.878017363$ $b = 0.937060963$	$a = 0.631148942$ $b = 0.881912354$	$a = 0.382782478$ $b = 0.825095684$	$a = 0.172755451$ $b = 0.738067365$	$a = 0.025602384$ $b = 0.466272263$
0.5	$a = 0.840896415$ $b = 0.840896415$	$a = 0.736685430$ $b = 0.781324107$	$a = 0.500945881$ $b = 0.675703499$	$a = 0.275209691$ $b = 0.558217352$	$a = 0.093177072$ $b = 0.374320705$	$a = 0.001807389$ $b = 0.032533016$
0.1	$a = 0.562341325$ $b = 0.562341325$	$a = 0.476465167$ $b = 0.504594140$	$a = 0.288194124$ $b = 0.384667656$	$a = 0.121766367$ $b = 0.243653596$	$a = 0.018590391$ $b = 0.074362063$	$a = 0.000001293$ $b = 0.000023279$
0.01	$a = 0.316227766$ $b = 0.316227766$	$a = 0.258611523$ $b = 0.273828481$	$a = 0.133528561$ $b = 0.178046649$	$a = 0.038490176$ $b = 0.076980733$	$a = 0.001859032$ $b = 0.007436128$	$a = 4.09 \times 10^{-11}$ $b = 7.354 \times 10^{-10}$
0.001	$a = 0.177827941$ $b = 0.177827941$	$a = 0.140567495$ $b = 0.148836384$	$a = 0.061973392$ $b = 0.082631374$	$a = 0.012171613$ $b = 0.024343227$	$a = 0.000185903$ $b = 0.000743613$	
0.0001	$a = 0.100000000$ $b = 0.100000000$	$a = 0.076414197$ $b = 0.080909160$	$a = 0.028765391$ $b = 0.038353859$	$a = 0.003849002$ $b = 0.007698004$	$a = 0.000018590$ $b = 0.000074361$	
0.0	$a = b = 0.0$ $\lambda = a/b = 1.0$	$a = b = 0.0$ $\lambda = a/b = 0.944444$	$a = b = 0.0$ $\lambda = a/b = 0.75$	$a = b = 0.0$ $\lambda = a/b = 0.5$	$a = b = 0.0$ $\lambda = a/b = 0.25$	$a = b = 0.0$ $\lambda = a/b = 0.055556$

The integral in equation (2.17), which defines the transfinite diameter, must also be evaluated numerically to whatever accuracy we require. Again the NAG routine D01AHF is appropriate. In Table 3.3 we give the transfinite diameters for the annular sectors considered in Table 3.2. The transfinite diameters are mostly given to 9 d.p. and were calculated using the NAG routine we have just mentioned. Once again the results for the special cases were not calculated by this method, but were produced from the formulae given in Section 2.2.4. Table (3.3) again contains blanks when $\theta = 10°$ and $R = 0.001$ or $R = 0.0001$. The transfinite diameter in these cases is given after Table 3.4.

Table 3.3: Transfinite diameter (ρ) values for the sectors in Table 3.2.

θ / R	180°	170°	135°	90°	45°	10°
1.0	0.0	0.087155743	0.382683433	0.707106781	0.923879533	0.996194698
0.9	0.025	0.123238826	0.414254357	0.726271974	0.931163842	0.996846174
0.5	0.125	0.222485471	0.484814189	0.758648353	0.938181488	0.996916790
0.1	0.225	0.304659893	0.526040421	0.769377145	0.938784932	0.996918113
0.01	0.2475	0.320719703	0.530892545	0.769796162	0.938785885	0.996910086
0.001	0.24975	0.322114753	0.531116172	0.769800336	0.938786047	
0.0001	0.249975	0.322236536	0.531126553	0.769800381	0.938787498	
0.0	0.25	0.322248185	0.531127040	0.769800359	0.938785852	0.996916753

3.2.1 An interesting relationship between a, b and R

The numbers in Table 3.2 lead to an interesting and informative result. We notice that when $\theta = \pi/4$, there appears to be an approximate linear relationship between the radius and the values of a and b. For example, reducing the radius by a factor of 10 from $R = 0.1$ to $R = 0.01$ appears to have a similar affect on both the a and b values. When we consider $\theta = \pi/2$, we find a different relationship holds; for small a and b values it appears that when we reduce the radius by 10 we reduce the a and b values by a factor of $\sqrt{10}$. For both these cases it appears that, for small values of b, a reduction in the value of R by a certain factor, φ, reduces the value of a and b by a factor

$$(\varphi)^{\frac{1}{4\lambda}},$$

where $\lambda = \theta/\pi$ (1/4 in the first case, 1/2 in the second case). It would be nice if we could prove this result analytically.

In the first theorem of case (iii) in Section 2.2.4 we proved that for "small" b,

$$R = \left(\frac{1-y_0}{1+y_0}\right)^{2\lambda}\left(\frac{1+\lambda y_0}{1-\lambda y_0}\right)^{2\lambda^2}\left[1+O(b^2)\right] \quad \text{with } y_0 = \sqrt{\frac{1-b^2}{1-\lambda^2 b^2}}.$$

We now expand y_0 in powers of b so

$$\begin{aligned} y_0 &= (1-b^2)^{\frac{1}{2}}(1-\lambda^2 b^2)^{-\frac{1}{2}} \\ &= \left(1-\frac{1}{2}b^2+\ldots\right)\left(1+\frac{\lambda^2 b^2}{2}+\ldots\right) \\ &= 1+\frac{(\lambda^2-1)b^2}{2}+O(b^4). \end{aligned}$$

Using this we find

$$\begin{aligned} 1+y_0 &= 2+\frac{(\lambda^2-1)b^2}{2}+O(b^4) \\ 1-y_0 &= \frac{(1-\lambda^2)b^2}{2}+O(b^4) \end{aligned}$$

$$1 - \lambda y_0 = 1 - \lambda - \frac{\lambda(\lambda^2 - 1)b^2}{2} + O(b^4)$$

$$1 + \lambda y_0 = 1 + \lambda + \frac{\lambda(\lambda^2 - 1)b^2}{2} + O(b^4).$$

By substituting these expressions into our expression for R we can show that

$$R = b^{4\lambda}\left[\frac{(1-\lambda^2)^{2\lambda}(1+\lambda)^{2\lambda^2}}{2^{4\lambda}(1-\lambda)^{2\lambda^2}} + O(b^2)\right]\left[1 + O(b^2)\right],$$

and therefore for sufficiently small b we find

$$R \approx b^{4\lambda}\left[\frac{(1-\lambda^2)^{2\lambda}(1+\lambda)^{2\lambda^2}}{2^{4\lambda}(1-\lambda)^{2\lambda^2}}\right].$$

We are trying to prove for a fixed θ (or a fixed λ because $\lambda = \theta/\pi$) that when we reduce the radius by a certain factor we also reduce both the a and b values by a fixed amount. For this reason we will consider another annular sector with the same θ value, but a different radius, r, and hence a different "b" value, B. We know from the above argument that

$$r \approx B^{4\lambda}\left[\frac{(1-\lambda^2)^{2\lambda}(1+\lambda)^{2\lambda^2}}{2^{4\lambda}(1-\lambda)^{2\lambda^2}}\right],$$

so for small b and B we find

$$\frac{R}{r} \approx \left(\frac{b}{B}\right)^{4\lambda}.$$

Finally if $R = \varphi r$ then $b = \varphi^{1/4\lambda}B$ and the result is proven. We note the same result holds for a because for small b, $a = \lambda b$ and λ is a constant.

Whilst searching for starting values for the modified Newton iteration in Section 3.2, we found that it was difficult to find starting values when θ and R were quite small. The above result provides insight into why this is so. For example, if we consider $\theta = 1°$ and $R = 0.1$, and we reduce the radius by a factor of 10, then we will reduce the a and b values by a factor of $10^{180/4} = 1 \times 10^{45}$. So even if a and b were both 1 when $R = 0.1$, they would both be of the magnitude 10^{-45} by the time we reach a radius of 0.01. This explains why our search failed.

The result also helps us to improve upon some results in Table 3.2. In this table the last column is not finished. The result we have just proved allows us to produce estimates

for a and b when $\theta = 10°$ and $R = 0.001$ or $R = 0.0001$. We begin by presuming that the values for a and b when $R = 0.5$ and $\theta = 10°$, in Table 3.2, are adequate. We then reduce the radius by a factor of 5, to 0.1, and use the above formula to see that a and b must drop by a factor of approximately $5^{18/4} \approx 1398$. We then reduce the radius to 0.01 and once again use the above formula to find that these a and b values must reduce by a factor of about $10^{18/4} \approx 31623$. We can continue this argument to produce new estimates of the parameters a and b. In Table 3.4 we give the previous estimated values for a and b (where available) and we also give our new estimates for the parameters a and b. Given a pair of a and b values we integrate, using D01AHF, the integrals in equations (2.7) and (2.8). The results we obtained for R and θ also appear in Table 3.4.

Table 3.4: Previous and improved values of a and b.

R	previous a and b	R θ (degrees)	improved a and b	R θ (degrees)
0.1	1.293×10^{-6} 2.3279×10^{-5}	0.10005 9.9979	$1.293262293 \times 10^{-6}$ $2.327873129 \times 10^{-5}$	0.09999999 9.99999999
0.01	4.09×10^{-11} 7.354×10^{-10}	0.00995 10.011	$4.089654458 \times 10^{-11}$ $7.361381192 \times 10^{-10}$	0.01000002 9.999996
0.001	—— ——	—— ——	$1.293262293 \times 10^{-15}$ $2.327873129 \times 10^{-14}$	$9.99999998 \times 10^{-4}$ 9.99999999
0.0001	—— ——	—— ——	$4.089654458 \times 10^{-20}$ $7.361381192 \times 10^{-19}$	1.00004×10^{-4} 9.999996

The computed values of R and θ in this table show that our new estimates for a and b are improvements on the values given previously (see Table 3.2). The values of a and b given for $R = 0.001$ and $R = 0.0001$ give reasonable approximations to the angle and

radius in these cases. Therefore estimates of the parameters in these cases have been found. Using these new values for a and b we can complete Table 3.3. It turns out for $R = 0.1$, 0.01, 0.001, and 0.0001 that the transfinite diameters all agree to 9 decimal places and their numerical value is 0.996916756.

3.3 Norms of Faber Polynomials on Annular Sectors

Any assessment of the accuracy of an approximation based on Faber polynomials requires some knowledge of a relevant norm. For the Faber polynomials on circular sectors Gatermann et al. [39] computed three different norms. In this section we shall introduce the three norms. Both the area and line versions of the 2-norm may be computed explicitly by a slight modification of the work of Gatermann et al. [39]. In Section 3.3.1 we will show how to generate the area 2-norm for the annular sector, and we shall give examples of this norm for Faber polynomials of various annular sectors. Similarly in Section 3.3.2 we will consider the line 2-norm. In Section 3.3.3 we will give a bound for $\|\widetilde{F}_n\|_\infty$ and also some examples of $\|\widetilde{F}_n\|_\infty$ for the annular sectors given in Table 3.2.

3.3.1 The area 2-norm

We begin this section by defining $Q_{\gamma,R,\rho}$, to be the annular sector symmetric with respect to the positive real axis with inner radius R/ρ, outer radius $1/\rho$ and half-angle γ. That is

$$Q_{\gamma,R,\rho} := \left\{ \frac{R}{\rho} \le |z| \le \frac{1}{\rho};\ \ |\arg z| \le \gamma \right\}.$$

Following Gatermann et al. [39], we define the square of the area 2–norm of the scaled Faber polynomial for an annular sector (with $\gamma > 0$) as

$$\left\|\widetilde{F}_n\right\|_2^2 := \int_{Q_{\gamma,R,1}} \left|\widetilde{F}_n(z)\right|^2 dx\, dy = \int_{Q_{\gamma,R,1}} \left|\Phi_n\left(\frac{z}{\rho}\right)\right|^2 dx\, dy \tag{3.1}$$

Here $z = x + iy$. Letting $z = \rho u$ where $u = v + iw$ we find

$$\|\widetilde{F}_n\|_2^2 = \rho^2 \int_{Q_{\gamma,r,\rho}} |\Phi_n(u)|^2\, dv\, dw.$$

We then consider

$$\begin{aligned}|\Phi_n(u)|^2 &= |u^n + \phi_{n-1}(u)|^2 \\ &= (u^n + \phi_{n-1}(u))\left(\bar{u}^n + \overline{\phi_{n-1}(u)}\right) \\ &= |u|^{2n} + 2\mathrm{Re}\left(\bar{u}^n \phi_{n-1}(u)\right) + |\phi_{n-1}(u)|^2.\end{aligned}$$

Using equation (2.33), which gives an expression for $\phi_n(z)$ in terms of the set of real coefficients $\{p_{n,j}\}_{j=0}^{n}$, we find

$$\begin{aligned}|\phi_{n-1}(u)|^2 &= \left|\sum_{j=0}^{n-1} p_{n-1,j} u^{n-1-j}\right|^2 \\ &= \sum_{j=0}^{n-1} |p_{n-1,j}|^2 |u|^{2(n-1-j)} + \sum_{\substack{i,j=0 \\ i\neq j}}^{n-1} p_{n-1,i}\, \bar{p}_{n-1,j}\, u^{n-1-i}\, \bar{u}^{n-1-j} \\ &= \sum_{j=0}^{n-1} p_{n-1,j}^2 |u|^{2(n-1-j)} + 2 \sum_{\substack{i,j=0 \\ i>j}}^{n-1} p_{n-1,i}\, p_{n-1,j}\, \mathrm{Re}(u^{n-1-i}\, \bar{u}^{n-1-j})\end{aligned}$$

By substituting these expressions into the formula for the area 2-norm we find

$$\begin{aligned}\left\|\widetilde{F}_n\right\|_2^2 = I_{n,n} + 2\sum_{j=0}^{n-1} p_{n-1,j} I_{n-1-j,n} &+ \sum_{j=0}^{n-1} p_{n-1,j}^2 I_{n-1-j,n-1-j} \\ &+ 2\sum_{j>k}^{n-1} p_{n-1,j} p_{n-1,k} I_{n-1-j,n-1-k} \qquad (3.2)\end{aligned}$$

where

$$I_{s,t} = \rho^2 \int_{Q_{\gamma,R,\rho}} \mathrm{Re}\left(u^s \bar{u}^t\right)\, dv\, dw.$$

To evaluate this norm for any annular sector we need to evaluate $I_{s,t}$ for any values of s and t. In the expression for $I_{s,t}$ we change to polar coordinates, that is we make the substitution $v = r\cos\vartheta$, $w = r\sin\vartheta$ so that $u = v + iw = r\exp(i\vartheta)$. Hence we find

$$\mathrm{Re}\left(u^s \bar{u}^t\right) = \mathrm{Re}\left[r^s e^{is\vartheta} r^t e^{-it\vartheta}\right]$$

and

$$I_{s,t} = \rho^2 \int_{\frac{R}{\rho}}^{\frac{1}{\rho}} r^{s+t+1}\, dr \int_{-\gamma}^{\gamma} \cos[(s-t)\vartheta]\, d\vartheta.$$

If we now evaluate the integrals in this expression we find

$$I_{s,t} = \begin{cases} 2\rho^{-s-t}\dfrac{\sin(s-t)\gamma}{(s-t)}\left[\dfrac{1-R^{s+t+2}}{s+t+2}\right] & \text{if } s \neq t \\ \\ \dfrac{\rho^{-2s}(1-R^{2s+2})\gamma}{s+1} & \text{if } s = t. \end{cases}$$

As $R \to 0$ these expressions reduce to those of Section 3.1 of [39], when account is taken of the difference in notation mentioned after equation (2.32) above.

Table 3.5: The area 2-norm, $\|\tilde{F}_{10}(z)\|_2$, for the annular sectors from Table 3.2. The significance of the symbols $*$ and $\dagger$ is explained in the text.

θ / R	180°	170°	135°	90°	45°	10°
0.9	0.44665283	0.08381187*	0.31934928†	0.56434232	0.59282140	0.49837668
0.5	0.99874608	0.14538911*	0.35388661†	0.62940677	0.66964105	0.53124163
0.1	1.33995848	0.21210244†	0.32372056†	0.47043032	0.71859599	0.53374537
0.01	1.40536038	0.22569208†	0.32993196†	0.47251876	0.72256602	0.53383978
0.001	1.41173386	0.22666794†	0.33061425†	0.47259666	0.72260876	0.53384072
0.0001	1.41236963	0.22674697†	0.33064103†	0.47259752	0.72260495	0.53384073
0.0	1.41244025	0.22675445†	0.33064231†	0.47259761	0.72260977	0.53384076

From the above expression for $I_{s,t}$ it is straightforward to produce numerical values for

the area 2-norm of the Faber polynomials for an annular sector. In Table 3.5 we show these calculated numerical values for the annular sectors from Table 3.2.

The only values in Table 3.5 that we have not explained how to derive are those for which $\gamma = \pi - \theta = 0$. In this case (see Section 2.2.4) the annular sector becomes the interval $[-1, -R]$ and the Faber polynomials become multiples of the Chebyshev polynomials for this interval. In fact for $[-1, -R]$ we find the monic Faber polynomials are given by

$$F_n(z) = (1-R)^n 2^{1-2n} \cos\left[n \cos^{-1}\left(\frac{2z+1+R}{1-R}\right)\right],$$

and $\rho = (1-R)/4$, so

$$\widetilde{F}_n(z) = 2\cos\left[n \cos^{-1}\left(\frac{2z+1+R}{1-R}\right)\right].$$

Hence we find

$$\begin{aligned}
\|\widetilde{F}_n\|_2^2 &= 4\int_{-1}^{-R} \cos^2\left[n \cos^{-1}\left(\frac{2z+1+R}{1-R}\right)\right] dz \\
&= 2(1-R)\int_0^{\pi} \cos^2(n\vartheta)\sin\vartheta \, d\vartheta \\
&= \frac{(1-R)}{2}\int_0^{\pi} \{\sin(2n+1)\vartheta + 2\sin\vartheta - \sin(2n-1)\vartheta\} \, d\vartheta \\
&= (1-R)\left(\frac{1}{2n+1} + 2 - \frac{1}{2n-1}\right) \\
&= 4(1-R)\left(\frac{2n^2-1}{4n^2-1}\right).
\end{aligned}$$

In what follows, by double precision we will mean double precision in IEEE notation, that is, fifteen or sixteen significant figures of a number will be stored. By quadruple precision we will mean that about thirty significant figures of a number are stored. The stars and daggers in Table 3.5 indicate that quadruple precision has been used to obtain 8 decimal places of accuracy. To calculate the area 2-norm by equation (3.2) we must first evaluate the three sums in this equation. If one or more of these sums is larger than 10^{16}, then in a double precision calculation vital information is lost. To check the

accuracy of our double precision calculations we carried out the same calculations using quadruple precision. Occasionally the sums mentioned above are so large that using double precision would give an obviously incorrect result. In this case the value in the table is that obtained from the quadruple precision calculation. We indicate this occurence with a star. Sometimes the double precision calculation was correct to a few decimal places of accuracy, though not to the 8 decimal places required in the table. In this case the result of the quadruple precision calculation is given in the table and it is marked with a dagger. For example, when $R = 0.1$ and $\theta = 170°$, the double precision calculation gives $\|\widetilde{F}_{10}(z)\|_2 = 0.21223559$ to 8 decimal places, whereas, the quadruple precision calculation gives $\|\widetilde{F}_{10}(z)\|_2 = 0.21210244$. For another example, we consider $R = 0.0001$ and $\theta = 135°$. To 8 decimal places, the double and quadruple precision calculations give the area 2-norm as 0.33064100 and 0.33064103, respectively. For the other results in the table, the double and quadruple precision calculations agree to 8 decimal places. It may be possible to rewrite equation (3.2) and evaluate the expression in a more stable manner, but for the purposes of the table our solution was to use quadruple precision.

Finally in this section, we note that Table 3.5 does not contain the values of the area 2-norm when $R = 1$. In this case the annular sector is an arc and the area and line 2-norms become the same norm, namely the standard 2-norm. We do not include these values in the table, and refer the reader to Table 3.6 for them.

3.3.2 The line 2-norm

In this section we consider the line 2-norm, $\hat{\|}\widetilde{F}_n\hat{\|}_2$. Again following Gatermann et al. [39] we define the square of the line 2-norm as

$$\hat{\|}\widetilde{F}_n\hat{\|}_2^2 := \int_{\partial Q_{\gamma,R,1}} |\widetilde{F}_n(z)|^2 \, ds = \int_{\partial Q_{\gamma,R,1}} \left|\Phi_n\left(\frac{z}{\rho}\right)\right|^2 ds, \tag{3.3}$$

where s is the arc length in the z-plane and $\partial Q_{\gamma,R,1}$ is the boundary of the annular sector $Q_{\gamma,R,1}$. Again we make the substitution $u = z/\rho = (x + iy)/\rho = v + iw$ and find

$$\hat{\|}\widetilde{F}_n\hat{\|}_2^2 = \rho \int_{\partial Q_{\gamma,R,\rho}} |\Phi_n(u)|^2 \, d\sigma,$$

where σ is the arc length in the u-plane. By similar reasoning to the arguments of Section 3.3.1 (the area 2-norm section) we can show

$$\hat{\|}\widetilde{F}_n\hat{\|}_2^2 = \hat{I}_{n,n} + 2\sum_{j=0}^{n-1} p_{n-1,j}\hat{I}_{n-1-j,n} + \sum_{j=0}^{n-1} p_{n-1,j}^2 \hat{I}_{n-1-j,n-1-j} + 2\sum_{j>k}^{n-1} p_{n-1,j}p_{n-1,k}\hat{I}_{n-1-j,n-1-k}, \tag{3.4}$$

where

$$\begin{aligned}\hat{I}_{s,t} &= \rho\int_{\partial Q_{\gamma,R,\rho}} \mathrm{Re}\left(u^s\bar{u}^t\right)\,d\sigma \\ &= \rho\int_{\partial Q_{\gamma,R,\rho}} r^{s+t}\cos(s-t)\vartheta\,d\sigma.\end{aligned}$$

To obtain the second integral from the first we have made the substitution $u = r\exp(i\vartheta)$. We now refer to Figure 3.3 where we consider the four line integrals that will make up this integral around the boundary. When the angle of the sector is held constant and equal to γ (line 1 in Figure 3.3) and we move along the arc in the direction of the arrow the radius is increasing and so $d\sigma = dr$. When we fix the radius equal to $1/\rho$ (arc 2), the angle is decreasing in the direction of the arrow and so $d\sigma = -(1/\rho)\,d\vartheta$. The other two regions are similar, when we fix the angle to be $-\gamma$ (line 3), $d\sigma = -dr$ and when we hold the radius equal to R/ρ (arc 4), $d\sigma = (R/\rho)\,d\vartheta$.

Using this information we find

$$\begin{aligned}\hat{I}_{s,t} &= \rho\int_{R/\rho}^{1/\rho} 2r^{s+t}\cos[(s-t)\gamma]\,dr + \rho\int_{-\gamma}^{\gamma}\left(\frac{R^{s+t+1}+1}{\rho^{s+t+1}}\right)\cos[(s-t)\vartheta]\,d\vartheta \\ &= \begin{cases} 2\rho^{-s-t}\left[\left(1-R^{s+t+1}\right)\left(\dfrac{\cos(s-t)\gamma}{s+t+1}\right) + \dfrac{\sin(s-t)\gamma}{(s-t)}\left(R^{s+t+1}+1\right)\right] & \text{if } s\neq t \\ 2\rho^{-2s}\left[\left(\dfrac{1-R^{2s+1}}{2s+1}\right)+\gamma\left(R^{2s+1}+1\right)\right] & \text{if } s=t.\end{cases}\end{aligned}$$

These formulae can be used to calculate the line 2-norm for all annular sectors except the arc (when $R = 1$) and the interval (when $\gamma = 0$). In these cases, the annular sector becomes a single line and if we used the above formulae we would have twice the desired result. Therefore when $R = 1$ or $\gamma = 0$ the correct result is obtained by dividing these

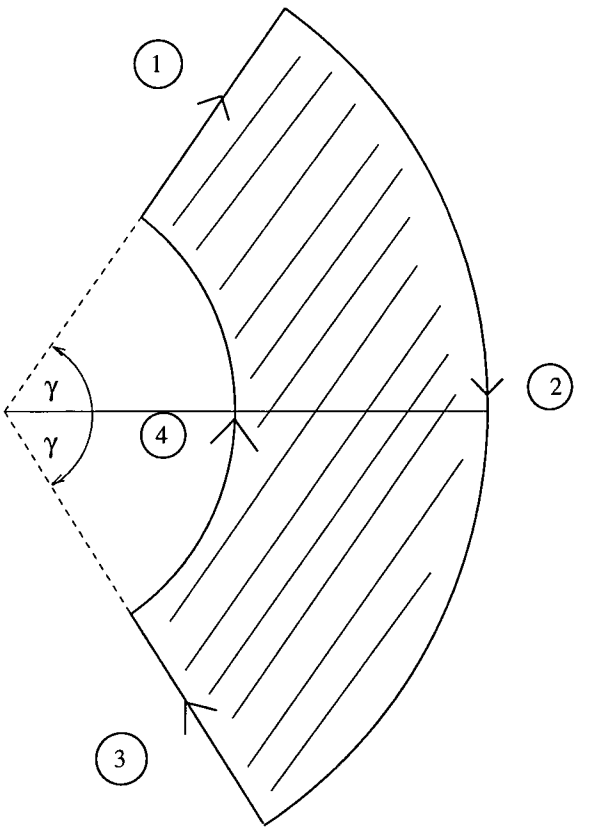

Figure 3.3: A plot of the boundary of the annular sector $Q_{\gamma,R,\rho}$.

formulae by 2. We see that as $R \to 0$ the expressions for $\hat{I}_{s,t}$ once again agree with the results of Gatermann et al. [39].

Table 3.6 shows the values of the line 2-norm for the annular sectors given in Table 3.2. As in Table 3.5, the stars indicate a result for which one or more of the sums in equation (3.4) become so large that a double precision calculation gives a silly result. In contrast the quadruple precision calculation gives the result shown in the table. As previously a dagger indicates a case where the double and quadruple precision calculations agree to a few decimal places. For this case the values in the table are those obtained from the quadruple precision calculation. For example, when $R = 0.1$ and $\theta = 170°$ the double precision calculation gives $\hat{\|}\tilde{F}_{10}(z)\hat{\|}_2 = 1.48165870$ to 8 decimal places, whereas, the quadruple precision calculation gives $\hat{\|}\tilde{F}_{10}(z)\hat{\|}_2 = 1.48267206$. For another example we consider $R = 0.01$ and $\theta = 135°$. The double precision calculation gives $\hat{\|}\tilde{F}_{10}(z)\hat{\|}_2 = 1.88913940$ to 8 decimal places, whereas, the quadruple precision calculation gives $\hat{\|}\tilde{F}_{10}(z)\hat{\|}_2 = 1.88913935$.

Once again, the rest of the table contains cases where the double and quadruple precision calculations agree to 8 decimal places.

Table 3.6: $\hat{\|}\tilde{F}_{10}(z)\hat{\|}_2$ for the sectors in Table 3.2.

θ / R	180°	170°	135°	90°	45°	10°
1.0		0.83449231*	1.77010715†	2.50101566	2.76170834	2.47397306
0.9		0.92914373*	1.86194054†	2.83880238	2.99729213	2.63381778
0.5	0.99874608*	1.23523861*	1.83976711†	2.40673907	2.60706091	2.51536336
0.1	1.33995848†	1.48267206†	1.87626564†	2.27527297	2.58849508	2.51455994
0.01	1.40536031†	1.54955152†	1.88913935†	2.27123691	2.57989399	2.5143340
0.001	1.41173386†	1.55988055†	1.89013081†	2.26930506	2.57865920	2.51430999
0.0001	1.41236963†	1.56101175†	1.89022818†	2.26907291	2.57850119	2.51430756
0.0	1.41244025†	1.56146771†	1.89018755	2.26905018	2.57852166	2.51430733

The spaces in the table are cases where the sums we must calculate are so large that even quadruple precision is not enough to produce a sensible result. We note that when $\theta = 180°$, the annular sector becomes an interval. Therefore the line 2-norm and the area 2-norm must become one and the same, namely the standard 2-norm for an interval. For

this reason, the values in tables 3.5 and 3.6 should agree when $\theta = \pi$. Comparing the results from the two tables, when $R = 0.5$, 0.1, 0.01, 0.001 and 0.0001, we see this is true. Therefore the blanks in Table 3.6 may be filled in from the results in Table 3.5.

3.3.3 The maximum norm

In this section we will exhibit upper and lower bounds on the maximum norm of the scaled Faber polynomials. The upper bound involves the total rotation of the boundary of the annular sector. We shall now introduce the idea of a curve of bounded rotation and then define the total rotation (see Ellacott[22], Elliott[28], Kövari and Pommerenke[52]).

We consider a smooth Jordan curve, Γ^* : $S(\tau)$, and denote the angle between the tangent to Γ^* at $S(\tau)$ and the positive real axis by $\varphi(\tau)$. Here τ is a parameter for the curve, not the constant given in equation (2.14). We then define the total rotation of Γ^* as the change in $\varphi(\tau)$ as we traverse the curve Γ^*. The total rotation of Γ^* is therefore given by

$$\int_{\Gamma^*} |\, d\varphi| = \int_{\Gamma^*} \left| \frac{d\varphi}{d\tau} \right| d\tau.$$

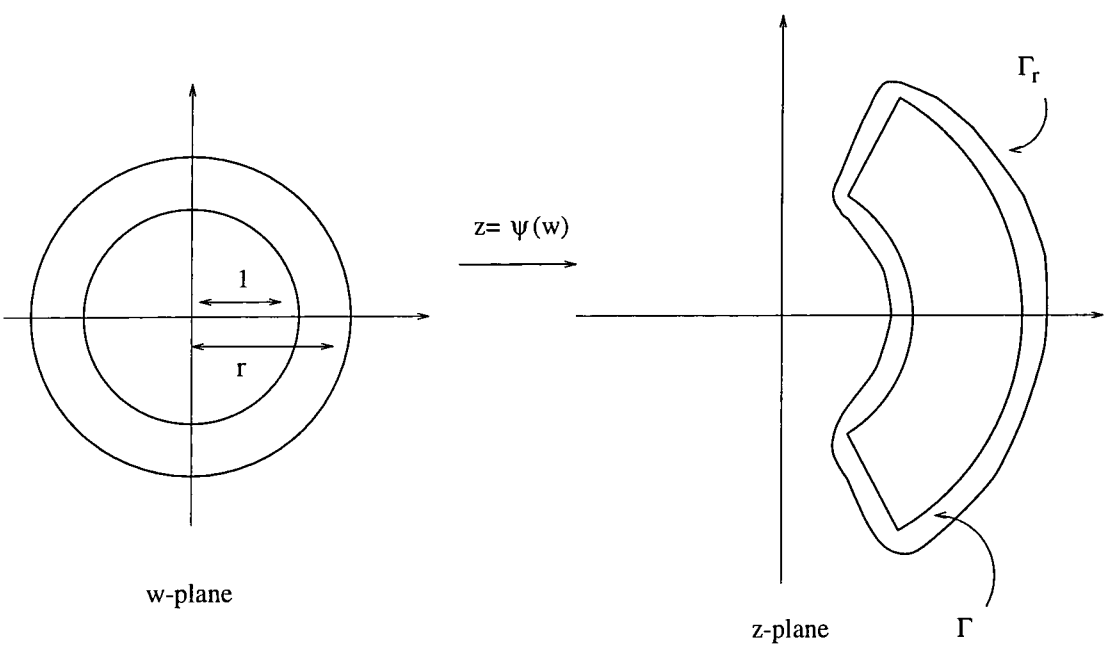

Figure 3.4: A level plot for an annular sector.

Let $\psi(w)$ map $|w| > 1$ onto the exterior of Γ, an arbitrary Jordan curve, and let Γ_r denote the level curves of the mapping, $\{z = \psi(w) : |w| = r\}$, (see Figure 3.4 where we show a level curve for an annular sector). Kövari and Pommerenke [52] state that $V(r)$, the total rotation of Γ_r, is a decreasing function of r. If $V(r)$ is bounded, Γ is said to be of bounded rotation, and

$$V = \sup_{r>1} V(r) = \lim_{r\to 1} V(r)$$

is called the total rotation of Γ.

The boundary of the annular sector is composed of simple arcs, therefore V is fairly easy to calculate for an annular sector. We now calculate V for an annular sector with a half–angle γ. To do this we consider two cases:

Case (i) $\gamma \leq \pi/2$.

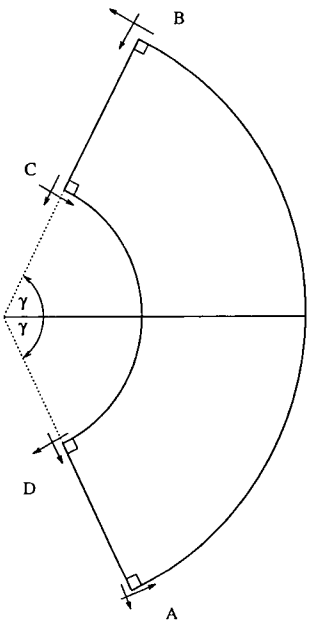

Figure 3.5: A general annular sector from case (i).

In figure 3.5 we show a general annular sector of this type. At the corners of the annular sector we have shown the tangent vector before and after the discontinuity. We see that at every corner the change in the tangent vector is $\pi/2$. To complete the calculation of V we must evaluate the change in the tangent vector along the arcs AB and CD. By elementary geometric considerations the tangent vector at A, along AB, makes an angle

of $\pi/2-\gamma$ with the positive real axis. (We note that if $\gamma > \pi/2$ then the angle would be $2\pi-(\gamma-\pi/2) = 5\pi/2-\gamma$.) Similarly the angle the tangent vector makes with the positive real axis at B is $\gamma+\pi/2$. So the change in the angle of the tangent vector along AB is

$$[\theta]_A^B = \left|\gamma+\frac{\pi}{2}-\left(\frac{\pi}{2}-\gamma\right)\right| = 2\gamma.$$

Considering the line CD we see that the angle the tangent vector makes with the positive real axis at C is $\gamma+3\pi/2$ and at D is $3\pi/2-\gamma$. Therefore

$$[\theta]_C^D = \left|\frac{3\pi}{2}-\gamma-\left(\gamma+\frac{3\pi}{2}\right)\right| = 2\gamma.$$

By combining the above results we find that for this annular sector

$$V = 4\left(\frac{\pi}{2}\right)+2\gamma+2\gamma = 2\pi+4\gamma.$$

Case (ii) $\gamma > \pi/2$.

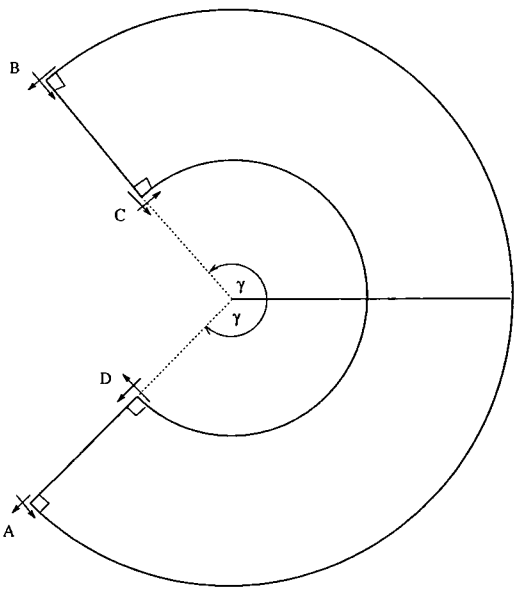

Figure 3.6: A general annular sector from case (ii).

In figure 3.6 we show a similar diagram to figure 3.5, but this time for a general annular sector of case (ii). As before we consider the tangent vector along AB. The tangent vector at A makes an angle of $5\pi/2-\gamma$ with the positive real axis, and the angle at B remains the same as in case (i). Considering the arc CD we see that the tangent vector at C makes an

angle of $\gamma - \pi/2$ with the positive real axis, and the angle the tangent vector at D makes with the positive real axis remains the same as case (i). Hence

$$[\theta]_A^B = \left|\gamma + \frac{\pi}{2} - \left(\frac{5\pi}{2} - \gamma\right)\right| = 2\pi - 2\gamma$$

and

$$[\theta]_C^D = \left|\frac{3\pi}{2} - \gamma - \left(\gamma - \frac{\pi}{2}\right)\right| = 2\pi - 2\gamma.$$

Remembering that the change in the angle of the tangent vector at each corner of the annular sector is $\pi/2$ we find

$$V = 4\left(\frac{\pi}{2}\right) + 2\pi - 2\gamma + 2\pi - 2\gamma = 6\pi - 4\gamma.$$

We now consider bounds on the maximum norm,

$$\left\|\tilde{F}_n\right\|_\infty = \max_{z\in Q}\left|\tilde{F}_n(z)\right|$$

of a scaled Faber polynomial; by the maximum principle that maximum value occurs on the boundary of the domain Q. Let

$$T_n(z) = z^n + a_{n-1}z^{n-1} + \cdots + a_0$$

be the Chebyshev polynomial of degree n for the annular sector Q, the monic polynomial of smallest maximum modulus on Q. It is known (Walsh [80], p.320) that $\|T_n\|_\infty \geq \rho^n$ and therefore, since no monic polynomial of degree n can have smaller norm, $\|\tilde{F}_n\|_\infty \geq 1$. For this norm Ellacott [23] provides an upper bound independent of the degree of the polynomial, n. His result is

$$\left\|\tilde{F}_n\right\|_\infty \leq \frac{V}{\pi}.$$

Combining these bounds we have

$$1 \leq \left\|\tilde{F}_n\right\|_\infty \leq \frac{V}{\pi}, \tag{3.5}$$

and for an annular sector of interior angle 2γ we have just shown that

$$V = \begin{cases} 2\pi + 4\gamma & \text{for } 0 \leq \gamma \leq \dfrac{\pi}{2} \\ \\ 6\pi - 4\gamma & \text{for } \dfrac{\pi}{2} \leq \gamma \leq \pi. \end{cases}$$

Starke and Varga [76], who used a different normalization for the Faber polynomials, provided bounds in terms of the norm of the corresponding Chebyshev polynomials. Their Theorem 3.4, for non–convex regions, is applicable to annular sectors, and $\phi(0)$, which appears in their bounds, is given in closed form by equation (2.38) (for their theorem see Section 5.3.3).

Table 3.7: $\|\widetilde{F}_{10}(z)\|_\infty$ for the sectors in Table 3.2.

θ / R	180°	170°	135°	90°	45°	10°
1.0		1.99960919	1.99995177	2.00759567 (2.01319547)	2.34237091	1.56576321
0.9	2.00006288*	1.53144187	1.56424045	1.78175708 (1.81060258)	1.80153021 (1.81802087)	1.44681530
0.5	2.00002296	1.51451514	1.51397749	1.71816450	1.33195921	1.43410186
0.1	2.00000000	1.55385355	1.49193299	1.48215499	1.38135082	1.43410173
0.01	2.00000000	1.48534007	1.50889957	1.49420633	1.38143109	1.43410173
0.001	2.00000000	1.81413232	1.51036632	1.49433074	1.38141090	1.43410173
0.0001	2.00000000	1.87656534	1.51214934	1.49433164	1.38122929	1.43410173
0.0	2.00000000	1.88579763	1.51534272	1.49433250	1.38143177	1.43410218

In Table 3.7 we show the value, given to 8 decimal places, of $\|\widetilde{F}_{10}\|_\infty$ for the annular sectors given in Table 3.2. The maximum value occurs on the boundary of the annular

sector and therefore to calculate the value of $\|\widetilde{F}_n\|_\infty$ we must just consider the value of $|\widetilde{F}_n(z)|$ over the boundary of the annular sector. The values in Table 3.7 were produced by sampling $|\widetilde{F}_{10}(z)|$ at 200 points placed around the boundary of the annular sector. Forty-nine points were placed along each edge. The remaining four points were placed at the corners. In Figure 3.7 we show, for a specific annular sector, a plot of the value of $|\widetilde{F}_{10}(z)|$ over the points we have just mentioned. The first point is positioned at $R\exp[i(\pi - \theta)]$, where θ is given in the definition of Q (see Section 2.2). From here we label the points in a clockwise sense until we return to this point again. If the maximum value occurs at a corner, which is the case for all but three of the examples in Table 3.7, then we will know the value of the maximum norm from our search. If, however, the maximum value does not occur at a corner, then we may not know the maximum value exactly, but we will have a reasonable approximation for both its value and the point(s) where it occurs. In Section 3.3.4 we will give a strategy for improving the approximate maximum value if the maximum does not occur at a corner. We have used this method to improve the approximate values of the maximum norm, in Table 3.7, which do not occur at corner. We show this improved value in brackets below the approximation we achieved from our initial search.

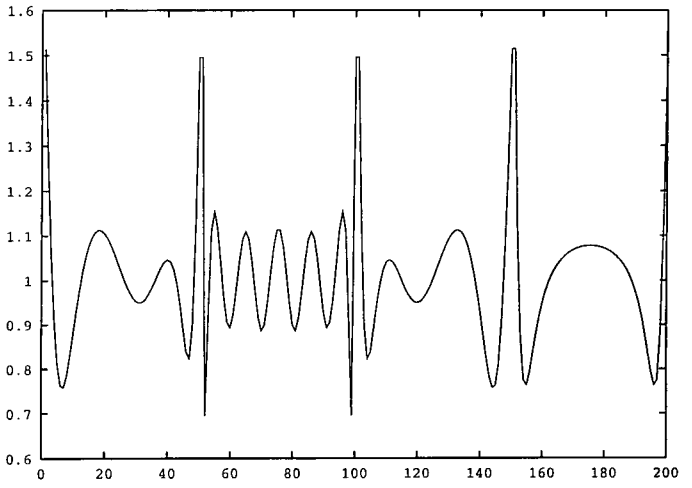

Figure 3.7: A plot of $|\widetilde{F}_{10}(z)|$ around the boundary of the annular sector when $R = 0.5$, and $\theta = 135°$, so $a = 0.500945881$, $b = 0.675703499$ and $\rho = 0.484814189$.

The star in Table 3.7 indicates that for this calculation quadruple precision was used.

When calculating the value of the Faber polynomial around the annular sector we first calculate the coefficients and then by a Hörner scheme we calculate the value of the polynomial at specific points around the annular sector. When $R = 0.9$ and $\theta = \pi$, intermediate values in the Hörner scheme become larger than 10^{16} and therefore using double precision causes vital information to be lost; our solution is to use quadruple precision.

3.3.4 Improving the estimate of the maximum norm

In this section we shall use Newton's method to improve the approximate value of the maximum norm when that maximum value does not occur at one of the corners of the annular sector. There are three examples of this in Table 3.7, and in all these examples the maximum value occurs on the boundary $|z| = 1$. Therefore in this section we will suppose that when the maximum value does not occur at a corner, it occurs on the boundary $|z| = 1$. If this is not the case, then modifications of the procedure we will describe would allow improvements in the approximate value of the maximum norm.

We begin by considering

$$\left|\widetilde{F}_n(z)\right|^2 = \left|\sum_{j=0}^{n} d_j z^{n-j}\right|$$

on the boundary $|z| = 1$, that is $z = \exp(i\theta)$. We have denoted the j–th coefficient of the scaled Faber polynomial, $\widetilde{F}_n(z)$, by d_j. Writing $z = \exp(i\theta)$ in this expression we find,

$$\begin{aligned}\left|\widetilde{F}_n\left(e^{i\theta}\right)\right|^2 &= \left|\sum_{j=0}^{n} d_j \cos[(n-j)\theta] + i\sum_{j=0}^{n} d_j \sin[(n-j)\theta]\right|^2 \\ &= C(\theta)^2 + S(\theta)^2\end{aligned}$$

where $C(\theta) = \sum_{j=0}^{n} d_j \cos[(n-j)\theta]$ and $S(\theta) = \sum_{j=0}^{n} d_j \sin[(n-j)\theta]$. We write $M(\theta)$ for $|\widetilde{F}_n(e^{i\theta})|$, and suppose that we have a reasonable initial approximation, θ_0, to the point where the maximum of $|\widetilde{F}_n(e^{i\theta})|$ occurs. This is usually possible from our search, and if it is not we could refine our search. From (3.5) we know that $\widetilde{F}_n(z)$ does not vanish near a maximum of $|\widetilde{F}_n(z)|$. Hence, with the above θ_0 we know that $M(\theta)$ is differentiable and not zero. We have

$$M^2 = C^2 + S^2$$

so

$$2MM' = 2CC' + 2SS',$$

where the prime means differentiation with respect to θ. For a stationary point (and in particular for a maximum) of $M(\theta)$ we require $M'(\theta) = 0$ and this gives

$$g(\theta) = CC' + SS' = 0,$$

where $C'(\theta) = -\sum_{j=0}^{n}(n-j)d_j \sin[(n-j)\theta]$ and $S'(\theta) = \sum_{j=0}^{n}(n-j)d_j \cos[(n-j)\theta]$. Substituting these expressions into $g(\theta)$ we find

$$\begin{aligned} g(\theta) &= -\sum_{k=0}^{n} d_k \cos[(n-k)\theta] \sum_{j=0}^{n}(n-j)d_j \sin[(n-j)\theta] \\ &\qquad + \sum_{k=0}^{n} d_k \sin[(n-k)\theta] \sum_{j=0}^{n}(n-j)d_j \cos[(n-j)\theta] \\ &= \sum_{j,k=0}^{n} d_k(n-j)d_j \{\cos[(n-j)\theta]\sin[(n-k)\theta] - \cos[(n-k)\theta]\sin[(n-j)\theta]\} \\ &= \sum_{j,k=0}^{n} d_k(n-j)d_j \sin[(j-k)\theta]. \end{aligned}$$

We split this double sum into three sums; one where $k < j$; one where $k = j$ and one where $k > j$ and observe several things. Firstly when $j = n$ the terms of the series vanish, secondly the second sum vanishes because $k = j$, and finally we can rewrite the third sum so that it becomes the negative of the first sum. These three observations imply

$$g(\theta) = \sum_{j=0}^{n-1}\sum_{k=0}^{j-1} d_j d_k(k-j)[\sin(j-k)\theta].$$

Given an initial approximation, θ_0, we use Newton's method to improve the approximate value, that is

$$\theta_{n+1} = \theta_n - \frac{g(\theta_n)}{g'(\theta_n)}, \quad n \geq 0.$$

From the above expression for $g(\theta)$ we find

$$g'(\theta) = \sum_{j=0}^{n-1}\sum_{k=0}^{j-1} d_j d_k(k-j)(j-k)[\cos(j-k)\theta].$$

For the three examples from Table 3.7, use of Newton's method with the above expressions produced improved approximate values for the maximum norm of the scaled Faber polynomials. In Table 3.7 these improved values are shown in brackets.

3.4 Improving the bound on $\|\chi_n\|$

As was mentioned in Section 1.2.3, Ellacott [22] derives a bound for $\|\chi_n\|$ where $\chi_n(f)$ is the truncated Faber series for f. This is a useful bound when the total rotation, V, is 2π (that is when the region in question is convex). For an annular sector $V \leq 4\pi$ (see Section 3.3.3). For values of V close to 4π the bound may not be as useful. For example, when $V = 4\pi$ the bound gives $\|\chi_n\| < 9$ for values of $n \leq 3$; as we will see $\|\chi_n\| < 9$ for larger values of n than 3. We will now give Ellacott's proof of his result. The reason for this is two-fold, firstly it is a nice proof and secondly we will improve on the result later in this section.

We recall from Section 1.2.3 that

$$\|\chi_n\| = \sup_{g\in A(D):\|g\|_\infty=1} \|\chi_n(g)\|_\infty = \sup_{g\in A(D):\|g\|_\infty=1} \max_{z\in D} |\chi_n(g)(z)|.$$

The function $\chi_n(g)(z)$ is simply a polynomial, so the maximum principle tells us that $|\chi_n(g)(z)|$ attains its maximum value at some point, $z_0 = \psi\left(e^{i\theta_0}\right)$, on the boundary of the region D. There is a difference in notation between this thesis and the paper by Ellacott [22]. Our map, ψ, maps the complement of the unit disc onto the complement of D, whereas Ellacott considers the mapping from the complement of a disc of radius ρ. Equations (1.7) and (1.8) give

$$\begin{aligned}
\|\chi_n(g)\|_\infty &= \left|\sum_{j=0}^{n} \frac{1}{2\pi i\rho^j} \int_{|w|=1} \frac{g(\psi(w))}{w^{j+1}}\, dw F_j(z_0)\right| \\
&\leq \frac{1}{2\pi}\left|\int_{|w|=1} \sum_{j=0}^{n} \frac{g(\psi(w))F_j(z_0)}{\rho^j w^{j+1}}\, dw\right| \\
&\leq \frac{1}{2\pi}\int_{|w|=1} \left|\sum_{j=0}^{n} \frac{\widetilde{F}_j(z_0)}{w^j}\right| \left|\frac{dw}{w}\right|
\end{aligned}$$

since $\|g\|_\infty = 1$ and $F_j(z) = \rho^j \widetilde{F}_j(z)$. Property 6$(b)$ in Section 1.2.2 now implies

$$\|\chi_n(g)\|_\infty \leq \frac{1}{2\pi}\int_{|w|=1} \left|{\sum_{j=0}^{n}}' \frac{1}{\pi w^j} \int_0^{2\pi} e^{ijs}\, d_s v(s,\theta_0)\right| \left|\frac{dw}{w}\right|$$

$$\leq \frac{1}{2\pi^2}\int_0^{2\pi}\int_{|w|=1}\left|{\sum_{j=0}^{n}}'\frac{e^{ijs}}{w^j}\right|\left|\frac{dw}{w}\right||\,d_s v(s,\theta_0)|\,.$$

In the second integral of this expression we make the substitution $w = \exp[-i(t-s)]$ giving

$$\begin{aligned}\int_{|w|=1}\left|{\sum_{j=0}^{n}}'\frac{e^{ijs}}{w^j}\right|\left|\frac{dw}{w}\right| &= \int_0^{2\pi}\left|{\sum_{j=0}^{n}}'e^{ijt}\right|dt\\ &= \int_0^{2\pi}\left|\sum_{j=0}^{n}e^{ijt}-\frac{1}{2}\right|dt \qquad (3.6)\\ &\leq \int_0^{2\pi}\left|\sum_{j=0}^{n}e^{ijt}\right|dt+\pi. \qquad (3.7)\end{aligned}$$

Therefore using property 6(a) in Section 1.2.2 we find

$$\|\chi_n\| \leq \frac{V}{\pi}\left(\tau_n+\frac{1}{2}\right) \qquad \text{where} \qquad \tau_n = \frac{1}{2\pi}\int_0^{2\pi}\left|\frac{e^{i(n+1)t}-1}{e^{it}-1}\right|dt.$$

The origins of τ_n lie with the classical Lebesgue constant γ_n (see Cheney [6]). Cheney defines S_n, the Fourier projection of degree n, for a real continuous periodic function, f, by a mapping of f onto its Fourier series. That is,

$$(S_n f)(x) = \frac{b_0}{2}+\sum_{k=1}^{n}a_k\sin kt + b_k\cos kt,$$

where

$$a_k = \frac{1}{\pi}\int_{-\pi}^{\pi}f(t)\sin(kt)\,dt \quad \text{and} \quad b_k = \frac{1}{\pi}\int_{-\pi}^{\pi}f(t)\cos(kt)\,dt.$$

Cheney [6] then shows that

$$\|S_n f\| \leq \|f\|\gamma_n, \quad \text{where} \quad \|g\| = \max_{x\in[-\pi,\pi]}|g(x)|,$$

and, as reported in Geddes and Mason [43], $\gamma_n = \tau_{2n}$. The value of τ_n is considered by Geddes and Mason [43]. In particular they show

$$\tau_n = \frac{4}{\pi^2}\log n + \eta + o(1),$$

where $\eta = 0.98941\ldots$ and $o(1)$ tends to zero monotonically as n increases. In Table 3.8 we give the value of τ_n for small n as computed by Geddes and Mason.

Table 3.8: The value of τ_n for $1 \leq n \leq 10$.

n	1	2	3	4	5	6	7	8	9	10
τ_n	1.273	1.436	1.552	1.642	1.716	1.778	1.832	1.880	1.923	1.961

In Ellacott's bound the constant, B (see Section 1.2.3), is chosen so the bound holds for all n; therefore this constant is chosen to be $1/2 + \tau_1 = 1.773\ldots$. This is restrictive, in the sense that the value of n, given by the bound, to obtain an approximation that is near-best within a relative distance τ is less than the actual value of n required. It is restrictive because $4\log(n)/\pi^2 + 1.773$ is bigger than $\tau_n + 1/2$ for all $n \geq 2$. The values in Table 3.8 imply for $V = 4\pi$ that $\|\chi_n\| < 9$ when $n \leq 5$; this is an improvement on $n \leq 3$ given in Section 1.2.3. We can, however, do better than this. The problem with the derivation of Ellacott's bound occurs in deriving the bound in (3.7) from the bound in (3.6). When the region is convex, Ellacott's bound gives $n \leq 835$ to achieve an approximation that is near-best within a relative distance 9. In most, if not all, practical situations we would be happy with this value of n. An annular sector is not convex, in fact V can be as large as 4π; as we have mentioned with such an annular sector the bound is restrictive and we need $n \leq 3$ to achieve an approximation that is near-best within a relative distance 9. This provides motivation to improve Ellacott's bound. To improve the bound we consider a numerical calculation of the integral (which we define to be τ_n^*) given in equation (3.6). The sum in this expression is a geometric series and therefore in a standard way (Geddes and Mason [43] and Ellacott [22]) we find

$$\tau_n^* = \int_0^{2\pi} \left| \sum_{j=0}^{n} e^{ijt} - \frac{1}{2} \right| dt = \int_0^{2\pi} \left| \frac{e^{i(n+1)t} - 1}{e^{it} - 1} - \frac{1}{2} \right| dt$$

$$= \int_0^{2\pi} \left| \exp\left(\frac{int}{2}\right) \left[\frac{\sin \frac{n+1}{2} t}{\sin \frac{t}{2}} \right] - \frac{1}{2} \right| dt.$$

We now make the substitution $t = 2u$ so

$$\int_0^{2\pi} \left| \sum_{j=0}^{n} e^{ijt} - \frac{1}{2} \right| dt = 2 \int_0^{\pi} \left| \frac{\sin(n+1)u}{\sin u} - \frac{1}{2} e^{-inu} \right| du$$

$$= 2\int_0^\pi \sqrt{\left(\frac{\sin(n+1)u}{\sin u} - \frac{1}{2}\cos(nu)\right)^2 + \frac{1}{4}\sin^2(nu)}\, du$$

$$= 2\int_0^\pi \sqrt{\frac{\sin(n+1)u\sin(nu)\cos(u)}{\sin^2 u} + \frac{1}{4}}\, du.$$

We could now integrate this expression numerically for different values of n. It is possible, however, to write this in a form where Gauss-Chebyshev quadrature can be used to approximate this integral. We begin by writing

$$\sin(nu)\cos(u) = \frac{1}{2}[\sin(n+1)u + \sin(n-1)u]$$

and then collect the fractions inside the square-root over a common denominator. Finally in the numerator we substitute

$$\begin{aligned}\sin(n+1)u\sin(n-1)u &= \frac{1}{2}[\cos 2u - \cos(2nu)],\\ \sin^2(n+1)u &= \frac{1}{2}[1 - \cos(2n+2)u]\end{aligned}$$

and

$$\sin^2 u = \frac{1-\cos 2u}{2},$$

so the integral becomes

$$\int_0^\pi \sqrt{\frac{\frac{3}{2} - \cos(2n+2)u + \frac{1}{2}\cos(2u) - \cos(2nu)}{\sin^2 u}}\, du.$$

In this expression we substitute $u = \cos^{-1}(x)$ and note that

$$\frac{du}{dx} = -\frac{1}{\sqrt{1-x^2}}, \quad \sin u = \sqrt{1-x^2}, \quad \text{and} \quad \cos(mu) = T_m(x),$$

where $T_m(x)$ is the Chebyshev polynomial of degree m. Remembering that

$$T_2(x) = 2x^2 - 1$$

and

$$T_{2n+2}(x) + T_{2n}(x) = 2xT_{2n+1}(x),$$

the expression finally yields

$$\int_0^{2\pi} \left| \sum_{j=0}^{n} e^{ijt} - \frac{1}{2} \right| dt = \int_{-1}^{1} \frac{f(x)}{\sqrt{1-x^2}}\, dx, \tag{3.8}$$

where

$$f^2(x) = \frac{1 - 2xT_{2n+1}(x) + x^2}{(1-x^2)}.$$

As we have mentioned the expression (3.8) is in an ideal form to use Gauss-Chebyshev quadrature. We note that the apparent singularities, in the expression for $f(x)$, at $x = \pm 1$ are removable. This is because $T_{2n+1}(1) = 1$ and $T_{2n+1}(-1) = (-1)^{2n+1}$, and therefore the numerator in $f^2(x)$ also has a factor of $(1-x^2)$. Gauss-Chebyshev quadrature produces an approximation to the righthand side of (3.8) of the form

$$\int_{-1}^{1} \frac{f(x)}{\sqrt{1-x^2}}\, dx \approx \sum_{i=1}^{m} H_i f(x_i),$$

where $H_i = \pi/m$ and $x_i = \cos[(2i-1)\pi/(2m)]$ (see Fox and Parker [34]). Hence

$$\begin{aligned} \int_0^{2\pi} \left| \sum_{j=0}^{n} e^{ijt} - \frac{1}{2} \right| dt &\approx \frac{\pi}{m} \sum_{i=1}^{m} \left[\frac{1 - 2x_i T_{2n+1}(x_i) + x_i^2}{1 - x_i^2} \right]^{\frac{1}{2}} \\ &= \frac{\pi}{m} \sum_{i=1}^{m} \left[\frac{1 - 2\cos\left\{\frac{(2i-1)\pi}{2m}\right\} \cos\left\{\frac{(2n+1)(2i-1)\pi}{2m}\right\} + \cos^2\left\{\frac{(2i-1)\pi}{2m}\right\}}{1 - \cos^2\left\{\frac{(2i-1)\pi}{2m}\right\}} \right]^{\frac{1}{2}}. \end{aligned}$$

In all cases (that is for all values of n used) we began with $m = 16$ and produced an approximation to the integral by the above formula. We then doubled m to 32 and produced another approximation. If the difference between these two approximations was not less than some tolerance (10^{-9}) multiplied by the initial approximation, then we continued to double the number of points (m) until two successive approximations differed by less than the tolerance multiplied by the initial approximation. When this was so we used the latter approximation as our value for the integral. The results for $1 \leq n \leq 30$ are shown in Table 3.9 to 4 decimal places. In Table 3.9 we also give the value of $\tau_n^*/2\pi$. This is important because our new bound for $\|\chi_n\|$ is

$$\|\chi_n\| \leq \frac{V\tau_n^*}{2\pi^2}.$$

Table 3.9: The value of the integral given by equation (3.6) for $1 \leq n \leq 30$. We also give the value of this integral divided by 2π.

n	τ_n^*	$\tau_n^*/2\pi$	n	τ_n^*	$\tau_n^*/2\pi$	n	τ_n^*	$\tau_n^*/2\pi$
1	6.6824	1.0635	11	11.8684	1.8889	21	13.4620	2.1425
2	7.9804	1.2701	12	12.0809	1.9227	22	13.5778	2.1610
3	8.8377	1.4066	13	12.2769	1.9539	23	13.6885	2.1786
4	9.4782	1.5085	14	12.4589	1.9829	24	13.7947	2.1955
5	9.9895	1.5899	15	12.6287	2.0099	25	13.8965	2.2117
6	10.4151	1.6576	16	12.7879	2.0352	26	13.9945	2.2273
7	10.7797	1.7156	17	12.9378	2.0591	27	14.0888	2.2423
8	11.0985	1.7664	18	13.0793	2.0816	28	14.1798	2.2568
9	11.3819	1.8115	19	13.2134	2.1030	29	14.2676	2.2708
10	11.6368	1.8521	20	13.3407	2.1232	30	14.3525	2.2843

When $V = 4\pi$ we find, from Table 3.9, that $\|\chi_n\| < 9$ for $n \leq 27$. This is a much better value than the one given by Ellacott's bound ($n \leq 3$), and shows that for practical values of n the truncated Faber series is near-best within a relative distance 9.

3.4.1 The improved constant

We shall now improve the constant in Ellacott's bound (see Section 1.2.3), that is, $B = 1.773\ldots$. We begin by bounding $\tau_n^*/2\pi$ above and below. The upper bound is given in Ellacott [22]. Using $|a| - |b| \leq |a - b| \leq |a| + |b|$ we find

$$\tau_n - \frac{1}{2} = \tau_n - \frac{\pi}{2\pi} \leq \frac{1}{2\pi} \int_0^{2\pi} \left| \frac{e^{i(n+1)t} - 1}{e^{it} - 1} - \frac{1}{2} \right| dt = \frac{\tau_n^*}{2\pi} \leq \tau_n + \frac{\pi}{2\pi} = \tau_n + \frac{1}{2}.$$

The results of Geddes and Mason [43] imply that

$$\frac{4}{\pi^2}\log n + 0.98941\ldots \leq \frac{4}{\pi^2}\log n + \eta + O(1) \leq \frac{4}{\pi^2}\log n + 1.273\ldots.$$

Combining the two previous expressions therefore gives

$$\frac{4}{\pi^2}\log n + 0.48941\ldots \leq \frac{\tau_n^*}{2\pi} \leq \frac{4}{\pi^2}\log n + 1.773\ldots.$$

From this we see that $\tau_n^*/2\pi$ behaves like $4\log n/\pi^2$, at least as $n \to \infty$. In Table 3.10 we consider the value of $\tau_n^*/2\pi - 4\log n/\pi^2$ for small n, specifically $1 \leq n \leq 30$.

Table 3.10: The value of $\tau_n^*/2\pi - 4\log n/\pi^2$ for $1 \leq n \leq 30$.

n	$\tau_n^*/2\pi - 4\log n/\pi^2$	n	$\tau_n^*/2\pi - 4\log n/\pi^2$	n	$\tau_n^*/2\pi - 4\log n/\pi^2$
1	1.0635	11	0.9171	21	0.9087
2	0.9892	12	0.9156	22	0.9082
3	0.9613	13	0.9144	23	0.9078
4	0.9467	14	0.9133	24	0.9074
5	0.9376	15	0.9124	25	0.9071
6	0.9314	16	0.9116	26	0.9068
7	0.9270	17	0.9109	27	0.9066
8	0.9236	18	0.9102	28	0.9063
9	0.9210	19	0.9096	29	0.9060
10	0.9189	20	0.9091	30	0.9058

The values in Table 3.10 suggest that $\tau_n^*/2\pi - 4\log n/\pi^2$ is a monotonically decreasing function of n. If this is true then

$$\frac{\tau_n^*}{2\pi} \leq \frac{4}{\pi^2}\log n + 1.0635\ldots.$$

Hence, we have

$$\|\chi_n\| \leq \frac{V}{\pi}\left(\frac{\tau_n^*}{2\pi}\right) \leq \frac{V}{\pi}\left(\frac{4}{\pi^2}\log n + 1.064\ldots\right),$$

and we have improved the constant in Ellacott's bound (see Section 1.2.3 and [22]) from 1.773 to 1.0635. When the region is convex ($V = 2\pi$), this new bound implies that $\|\chi_n\| < 9$ for $n \leq 4813$. When $V = 3\pi$ we find $\|\chi_n\| < 9$ for $n \leq 118$, and when $V = 4\pi$ we find $\|\chi_n\| < 9$ for $n \leq 18$. The new bound is still pessimistic, but it is an improvement on the one given in Ellacott [22].

3.5 Level plots

In this section we describe a numerical calculation of level plots for an annular sector. In Section 1.2.2 we defined a level curve, Γ_r, to be

$$\Gamma_r = \{z = \psi(w) : |w| = r > 1\},$$

that is, the image of ψ applied to the circle of radius r in the w-plane. In Section 2.2.2 we derived from (2.12b) an expression for the conformal mapping given by

$$z = \psi(w) = -\exp\left[\int_{-1}^{w} \frac{(\mu^2 - 2t\mu + 1)^{\frac{1}{2}}(\mu^2 - 2\tau\mu + 1)^{\frac{1}{2}}}{\mu(\mu^2 - 2\mu\coth\alpha + 1)}\, d\mu\right]. \tag{3.9}$$

A similar strategy can be applied to equation (2.12c) producing another expression for the mapping,

$$z = \psi(w) = -R\exp\left[\int_{1}^{w} \frac{(\mu^2 - 2t\mu + 1)^{\frac{1}{2}}(\mu^2 - 2\tau\mu + 1)^{\frac{1}{2}}}{\mu(\mu^2 - 2\mu\coth\alpha + 1)}\, d\mu\right]. \tag{3.10}$$

There are a vast number of different forms for the conformal map (see Section 2.2.2), so many different formulae may be used to approximate the level curves numerically. The branch points that exist when evaluating the integrand in the expressions above can cause computational problems because a computer may calculate the value of the square root function on a different branch to the desired one. This may cause some of the aforementioned formulae to give poor level plots and for this reason we describe a method we found to work satisfactorily. We begin with (3.9) and (3.10), substitute $\mu = 1/\nu$, and produce

two more forms of the mapping, that is

$$z = \psi(w) = -\exp\left[-\int_{-1}^{1/w} \frac{(\nu^2 - 2t\nu + 1)^{\frac{1}{2}}(\nu^2 - 2\tau\nu + 1)^{\frac{1}{2}}}{\nu(\nu^2 - 2\nu\coth\alpha + 1)}\, d\nu\right] \tag{3.11}$$

and

$$z = \psi(w) = -R\exp\left[-\int_{1}^{1/w} \frac{(\nu^2 - 2t\nu + 1)^{\frac{1}{2}}(\nu^2 - 2\tau\nu + 1)^{\frac{1}{2}}}{\nu(\nu^2 - 2\nu\coth\alpha + 1)}\, d\nu\right]. \tag{3.12}$$

We then consider certain points on the curve $|w| = r$, $r > 1$, in particular the points $w = r\exp(i\eta)$ where $\eta = 2\pi k/n$ with $k = 0, 1, \ldots, n$. We note that $k = n$ gives us the same point in the w-plane as when $k = 0$. Our strategy is to use (3.11) if $\cos\eta < 0$ and (3.12) if $\cos\eta \geq 0$. We chose to use two formulae to calculate the mapping because it seems natural to choose the formula which is best suited for the point we are considering at that time, that is, we choose to use (3.11) if the point is closer to -1 and (3.12) if the point is nearer 1. Once we have chosen which expression to use, that is whether $\cos\eta \geq 0$ or $\cos\eta < 0$, then we use Gauss-Kronrod-Patterson integration to approximate the integral. In doing this we consider the integral along a straight line from -1 (if we use (3.11)) or 1 (if we use (3.12)) to the point $1/w$. The integrand in expressions (3.11) and (3.12) has poles at $\nu = 0$, $\nu = (1+a^2)/(1-a^2)$ and $\nu = (1-a^2)/(1+a^2)$; this gives rise to a problem in our straight line integration. The problem occurs when the radius, r, of our level curve is greater than $(1 + a^2)/(1 - a^2)$ and the angle, η, of the point in the w-plane is close to 0 or 2π. When this occurs the straight line integration will pass close to or go through a singularity in the integrand and our numerical image point will be a poor approximation to the true image point. In this case it would be possible to find a reasonable approximation to the image point by considering a different path to integrate along. For example, we could use an arc joining the points 1 and $1/w$. In figure (3.8) we show the level plots produced, as we have described, for the annular sector of half-angle $\pi/4$ radians and inner radius 1/2. Referring to Table (3.2) we see that for this annular sector $a = 0.500945881$ and $b = 0.675703499$. Hence, $(1 + a^2)/(1 - a^2) = 1.6700372$ and we only have a problem in our level plot s if $r > 1.6700372$. In figure (3.8) we show the level plots with radii, 1.67, 1.5, 1.2, 1.1, 1.01, and 1.001. The level plot with radius 1.001 provides, at least as far as the eye can see, a decent approximation to the desired annular sector.

To get a good plot of the annular sector and its level curves it would be nice to have the same number of points along each arc making up the boundary of the annular sector. The method we have just described does not have this property. However, referring to Section 2.5 it would be possible to do this. We would simply consider which parts of the curve $|w| = r$ map to which parts of the annular sector and distribute the points evenly over each of these.

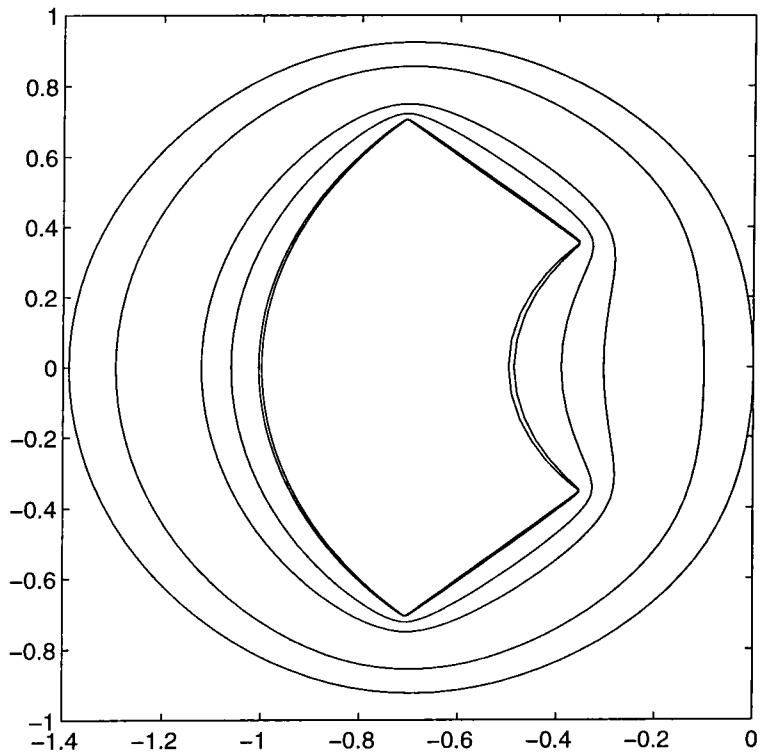

Figure 3.8: Level plots for an annular sector of half-angle $\pi/4$ and inner radius $1/2$.

3.5.1 The Féjer points

The boundary of the annular sector is a Jordan curve (simple closed curve), therefore (see Gaier [38], pg. 67) the mapping ψ from $\{w : |w| > 1\}$ onto $\mathbb{C}/Q$ can be extended continuously to a mapping from $\{w : |w| \geq 1\}$. The images under ψ of the $(n+1)$-st roots of unity are called the Féjer points of order n on D. The Féjer points are therefore given

by

$$z_k^{(n)} = \psi\left(\exp\left\{\frac{2\pi ik}{n+1}\right\}\right) \qquad \text{for } k = 0, 1, \ldots, n.$$

When interpolating a function, $f(z)$ analytic on D (in this case the annular sector Q), at an arbitrary set of points in D, then as we take more and more points the interpolating polynomial, $L_n(z)$, need not tend to the function we are trying to approximate. Gaier [38] introduces the idea of a set of uniformly distributed points. He shows that the Féjer points are uniformly distributed, and that the interpolating polynomials, $L_n(z)$, tend to $f(z)$ as $n \to \infty$ if and only if we interpolate f at a set of uniformly distributed points. The set of Féjer points are not the only uniformly distributed set of points, but they are easy to calculate if the conformal map, ψ, is known. In figure (3.9) we show approximations to the Féjer points $z_k^{(n)}$, $k = 0, 1, \ldots, n$ where $n = 50$ for the annular sector, Q, of half-angle $\pi/4$ and inner radius $1/2$. The Féjer points shown are only approximate because we actually found the points

$$\psi\left(r\exp\left\{\frac{2\pi ik}{n+1}\right\}\right)$$

where r was chosen to be 1.001. To do this we used exactly the same technique as we used in Section 3.5.

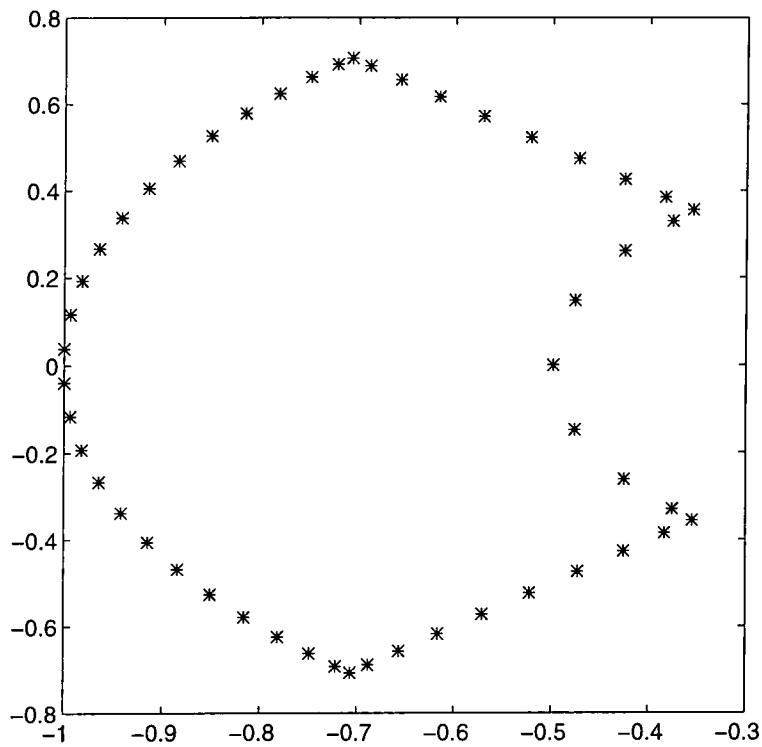

Figure 3.9: The approximate Féjer points, $z_k^{(50)}$ with $k = 0, 1, \ldots, 50$ for an annular sector of half-angle $\pi/4$ and inner radius $1/2$.

Chapter 4

A review of Iterative Methods.

4.1 Introduction

One of the most important problems in numerical computation is the solution of nonsingular systems of linear equations,

$$A\mathbf{x} = \mathbf{b}, \tag{4.1}$$

where $A \in \mathbb{C}^{N\times N}$, $\mathbf{x}$ and $\mathbf{b} \in \mathbb{C}^N$. Linear systems of this type occur in many different areas. For example they occur when solving partial differential equations by finite element or finite difference methods. When the size of the matrix in (4.1) is small, the usual approach is to use direct methods such as Gaussian Elimination. Gaussian Elimination, like other direct methods, involves a factorisation of the matrix A. For this reason when the size of the linear system is large, direct methods can become costly in terms of both computing time and storage. It is often better (and sometimes the only option) to use iterative techniques to solve (4.1), especially when the coefficient matrix, A, is sparse. The coefficient matrix is certainly sparse in the examples we have already mentioned, that is when the linear system arises from discretising a partial differential equation; for these examples an iterative method may work well.

The aim of this chapter is to review iterative methods for the solution of (4.1) and to describe which methods are suitable for various classes of coefficient matrix. We will start with the very basic iterative techniques known as stationary methods, an example of which is the Jacobi iteration. Stationary methods involve splittings of the coefficient matrix and usually exhibit slow convergence at least compared with the methods we will describe in the later sections of this chapter. After a very brief discussion of stationary methods we will then describe some of the most popular nonstationary methods, such as CG (for symmetric, positive, definite matrices), GMRES, QMR and Bi-CG (for nonsymmetric, nonsingular matrices). Most of the methods we describe belong to a class of methods known as Krylov subspace methods. In the last section of this chapter we will review hybrid methods. Hybrid methods involve two stages. In the first stage they use a method to produce some information about the coefficient matrix. In the second stage they use a parameter dependent method and the information obtained from the first stage, to improve the estimate of the solution.

4.2 Stationary Iterative Methods

Some of the oldest and easiest to implement iterative techniques are the stationary iterative methods. For these methods the coefficient matrix, A, is split in the form

$$A = M - Q,$$

where the matrix M is nonsingular. The problem (4.1) is then equivalent to $M\mathbf{x} = Q\mathbf{x} + \mathbf{b}$ and this suggests the iterative scheme

$$\mathbf{x}_{k+1} = M^{-1}Q\mathbf{x}_k + M^{-1}\mathbf{b}. \tag{4.2}$$

The methods are called stationary because $M^{-1}Q$ and $M^{-1}\mathbf{b}$ do not depend on the iteration number k, and so the methods do not vary from iteration to iteration. Methods of this type include Jacobi's method, the Gauss-Seidel method and the successive overrelaxation methods.

We define the error at the k-th step as

$$\mathbf{e}_k := \mathbf{x} - \mathbf{x}_k, \tag{4.3}$$

and say that the method (4.2) is convergent if $\lim_{k\to\infty} \mathbf{e_k} = 0$. We now define $T = M^{-1}Q$, and therefore

$$\begin{aligned} \mathbf{e}_k &= \mathbf{x} - \mathbf{x}_k \\ &= T\mathbf{e}_{k-1} \\ &= T^k\mathbf{e}_0. \end{aligned}$$

Consequently for any consistent matrix norm, the error satisfies

$$\|\mathbf{e}_k\| = \left\|T^k\mathbf{e}_0\right\| \leq \left\|T^k\right\| \|\mathbf{e}_0\|.$$

It is well known that $\|T^k\| \to 0$ as $k \to \infty$ if and only if $\rho(T) < 1$ (see most numerical analysis undergraduate texts, for example Atkinson [2], Burden and Faires [5] and Kreysig [51]). This result means the iterative method converges, provided the eigenvalue of T with largest modulus, is less than one. We will now briefly consider four standard iterative methods.

4.2.1 Jacobi's Method

To describe Jacobi's method we begin with the i-th equation of the linear system (4.1) that is,

$$\sum_{j=1}^{N} a_{ij}x^{(j)} = b^{(i)}.$$

If $\prod_{i=1}^{n} a_{ii} \neq 0$, we can produce an iterative method whereby the k-th iterate of the i-th component of $\mathbf{x}$ is found from

$$x_k^{(i)} = \left(b^{(i)} - \sum_{j \neq i} a_{ij} x_{k-1}^{(j)} \right) / a_{ii}.$$

This is Jacobi's method for solving the system (4.1). Following a standard notation we write $A = D - L - U$, where the matrices D, L and U contain the diagonal, strictly lower triangular and strictly upper triangular parts of the matrix A, respectively. In the notation of Section 4.2 we find $M = D$ and $Q = L + U$. Consequently Jacobi's method may be written as

$$\mathbf{x}_k = D^{-1}(L + U)\mathbf{x}_{k-1} + D^{-1}\mathbf{b}.$$

4.2.2 The Gauss-Seidel Method

The Gauss-Seidel method is a modification of the Jacobi method. The idea is to use the new approximation to $x_k^{(i)}$ as soon as it becomes available. Therefore the Gauss-Seidel method is written

$$x_k^{(i)} = \left(\mathbf{b}^{(i)} - \sum_{j<i} a_{ij} x_k^{(j)} - \sum_{j>i} a_{ij} x_{k-1}^{(j)} \right) / a_{ii},$$

provided $\prod_{i=1}^{n} a_{ii} \neq 0$ once again. If we split the coefficient matrix as we did for Jacobi's method we find $M = D - L$ and $Q = U$ for the Gauss-Seidel method. So in matrix notation the Gauss-Seidel method can be written

$$\mathbf{x}_k = (D - L)^{-1}U\mathbf{x}_{k-1} + (D - L)^{-1}\mathbf{b}.$$

4.2.3 The Overrelaxation Methods

In this section we consider the Successive Overrelaxation Method (SOR) and Symmetric Successive Overrelaxation Method (SSOR). SOR is a stationary iterative method that contains a parameter, w, which can be set by the user. In the notation of Section 4.2 this method is represented by $M = (D - wL)/w$ and $Q = [(1 - w)D + wU]/w$. Therefore in matrix notation SOR is written

$$\mathbf{x}_k = (D - wL)^{-1} [(1 - w)D + wU] \mathbf{x}_{k-1} + w(D - wL)^{-1}\mathbf{b}.$$

When $w = 1$ this reduces to the Gauss-Seidel method.

The SSOR method combines two SOR-like iterations, the difference being in the second iteration where the roles of L and U are interchanged. We define

$$\begin{aligned} M_1 &= \frac{1}{w}(D - wL), \quad Q_1 = \frac{1}{w}[(1 - w)D + wU], \\ M_2 &= \frac{1}{w}(D - wU), \quad Q_2 = \frac{1}{w}[(1 - w)D + wL], \end{aligned}$$

and then the SSOR method is given by the following two equations,

$$\mathbf{x}_{k+\frac{1}{2}} = M_1^{-1} Q_1 \mathbf{x}_k + M_1^{-1}\mathbf{b}$$

and

$$\mathbf{x}_{k+1} = M_2^{-1} Q_2 \mathbf{x}_{k+\frac{1}{2}} + M_2^{-1}\mathbf{b}.$$

This may also be written as

$$\mathbf{x}_{k+1} = M_2^{-1} Q_2 M_1^{-1} Q_1 \mathbf{x}_k + w(2 - w)(D - wU)^{-1} D (D - wL)^{-1}\mathbf{b}.$$

A good review of all these stationary iterative methods is given in Barrett et al. [3]. According to this and the references therein, the SSOR convergence rate with an optimal w is slower than the SOR convergence rate with an optimal w. For this reason the SSOR method is rarely used, except maybe as a preconditioner for symmetric matrices (see [3] for the details).

Stationary iterative methods were once popular. However, nowadays this is not the case because more powerful techniques have been developed. For most matrices, stationary

iterative methods suffer from slow convergence. The SOR method with an optimal w, usually converges quicker than the others, but compared to the methods of the next section it is slow. Stationary iterative methods do have the advantages of being easy to explain and implement.

4.3 Nonstationary Iterative Methods

4.3.1 Krylov Subspace Methods

In this section we will introduce Krylov subspace methods. Most of the modern iterative methods for solving (4.1) belong to this set of methods. Krylov subspace methods begin with an initial approximation $\mathbf{x}_0$ to the solution of (4.1), $\mathbf{x} = A^{-1}\mathbf{b}$. From this initial approximation the method produces a sequence of iterates of the form

$$\mathbf{x}_n \in \mathbf{x}_0 + K_n(\mathbf{r}_0, A), \tag{4.4}$$

where $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$ and $K_n(\mathbf{r}_0, A)$ is the n-th Krylov subspace generated by $\mathbf{r}_0$ and A, that is,

$$K_n(\mathbf{r}_0, A) \equiv \text{span}\left\{\mathbf{r}_0, A\mathbf{r}_0, \ldots, A^{n-1}\mathbf{r}_0\right\}. \tag{4.5}$$

A general element of this Krylov subspace is

$$\alpha_0\mathbf{r}_0 + \alpha_1 A\mathbf{r}_0 + \ldots + \alpha_{n-1}A^{n-1}\mathbf{r}_0 = q_{n-1}(A)\mathbf{r}_0, \tag{4.6}$$

where $q_{n-1} \in P_{n-1}$, the set of polynomials with degree less than or equal to $n-1$ (as defined in Section 1.2.3). The n-th residual is defined as

$$\mathbf{r}_n = \mathbf{b} - A\mathbf{x}_n = A\mathbf{e}_n, \tag{4.7}$$

that is the difference between the right-hand side $\mathbf{b}$ and the approximation $A\mathbf{x}_n$. In this equation we have also noted the n-th residual is the n-th error, $\mathbf{e}_n$ (as defined by (4.3)), multiplied by A. In view of equations (4.4) and (4.6) we find

$$\begin{aligned} \mathbf{r}_n &= A(\mathbf{x} - \mathbf{x}_n) = \mathbf{r}_0 - Aq_{n-1}(A)\mathbf{r}_0 \\ &= p_n(A)\mathbf{r}_0, \end{aligned} \tag{4.8}$$

where

$$p_n \in P_n \quad \text{with} \quad p_n(0) = 1. \tag{4.9}$$

The polynomials q_{n-1} are known as iteration polynomials, whereas the p_n are known as residual polynomials. We note that $p_n(z) = 1 - zq_{n-1}(z)$.

As a motivation for Krylov subspace methods, we suppose A has a set of orthonormal eigenvectors $\{\mathbf{v}_j\}_{j=1}^N$ (that is A is a normal matrix) and corresponding eigenvalues $\{\lambda_j\}$, and we write the initial residual as a linear combination of these eigenvectors that is

$$\mathbf{r}_0 = \sum_{j=1}^{N} \gamma_j \mathbf{v}_j.$$

Then

$$\mathbf{r}_n = p_n(A)\mathbf{r}_0 = \sum_{j=1}^{N} \gamma_j p_n(A)\mathbf{v}_j = \sum_{j=1}^{N} \gamma_j p_n(\lambda_j)\mathbf{v}_j,$$

and so

$$\begin{aligned} \|\mathbf{r}_n\|_2^2 &= \sum_{j=1}^{N} \gamma_j^2 p_n^2(\lambda_j) \\ &\leq \max_{\lambda \in \sigma(A)} |p_n(\lambda)|^2 \|\mathbf{r_0}\|_2^2. \end{aligned}$$

Therefore finally

$$\|\mathbf{r}_n\|_2 \leq \max_{\lambda \in \sigma(A)} |p_n(\lambda)| \, \|\mathbf{r}_0\|_2, \tag{4.10}$$

where $\sigma(A)$ is the spectrum of the matrix A. This means the n–th residual is small provided $|p_n|$ is small on the spectrum of A. It seems therefore that Krylov subspace methods could be a good idea.

The above result can be extended to non-normal matrices (see for example [37]). We begin by supposing that the coefficient matrix, A, is diagonalisable, that is there exists a matrix, X, such that $X^{-1}AX = D = \mathrm{Diag}(\lambda_1, \ldots, \lambda_N)$. Therefore

$$\begin{aligned} \|\mathbf{r}_n\|_2 &= \|p_n(XDX^{-1})\mathbf{r}_0\|_2 \\ &= \|Xp_n(D)X^{-1}\mathbf{r}_0\|_2 \\ &\leq \|X\|_2 \|\mathrm{Diag}(p_n(\lambda_1), \ldots, p_n(\lambda_N))\|_2 \|X^{-1}\|_2 \|\mathbf{r}_0\|_2 \\ &\leq \|X\|_2 \|X^{-1}\|_2 \max_{\lambda \in \sigma(A)} |p_n(\lambda)| \, \|\mathbf{r}_0\|_2. \end{aligned} \tag{4.11}$$

This again shows that the n-th residual is small provided $|p_n|$ is small on the spectrum of A, that is, provided $K_2(X)$ is small. The inequality (4.11) differs from (4.10) by a factor of $K_2(X) = \|X\|_2\|X^{-1}\|_2$. For normal matrices $K_2(X) = 1$ and we again obtain the inequality (4.10). For highly non-normal matrices $K_2(X)$ can be extremely large and the bound given in (4.11) may not be a useful one.

One of the aims of this chapter is to arrive at hybrid methods as a sensible idea for solving the linear system (4.1). As previously mentioned, hybrid methods involve two stages. In the first stage a parameter–free scheme is used to gain some information about A. In the second stage a parameter–dependent scheme is used, where the residual polynomial, p_n, or iteration polynomial, q_{n-1}, is constructed from the information gained in the first stage. This second stage is usually, if not always, a Krylov subspace method. An example of the information gained in the first stage, would be the knowledge of a compact set G which contains the eigenvalues of A, but does not contain the origin, that is

$$\lambda(A) \subseteq G \subset \mathbb{C}, \quad 0 \notin G.$$

Given such a set G, Freund et al. [37] suggest a good set of polynomials to choose as the residual polynomials would be those satisfying

$$\max_{\lambda \in G} |p_n(\lambda)| = \min_{p \in P_n : p(0)=1} \max_{\lambda \in G} |p(\lambda)|,$$

that is the best approximations to zero on the set G. They also note that this approximation problem can only be solved analytically in a few special cases, such as the case of the interval, where the Chebyshev polynomials are optimal. Usually when these polynomials are not known exactly, the way forward is to choose polynomials that are near-best approximations to zero, or to choose polynomials which are asymptotically optimal (for semi-iterative methods see Eiermann et al. [20]). For a review of hybrid methods see Section 4.4.

4.3.2 The Method of Conjugate Gradients (CG)

The method of conjugate gradients (CG) is one of the oldest and most popular methods used for a linear system (4.1) whose matrix A is Hermitian and positive definite. Hestenes

and Stiefel [50] introduced this method in 1952 as a direct method (see Barrett et al. [3]). In the 1960's CG became less popular because on a computer the search directions (see below) will eventually lose their A-orthogonality. In the seventies, however, CG was shown to be an effective iterative procedure and once again interest grew in the method. There have been many papers and articles written about CG; an excellent introduction to the method and its ideas is given in Shewchuk [71], a good review of the method is contained in Barrett et al. [3] and a nice section on the subject is contained in Golub and Van Loan [44].

CG was introduced to improve upon the ideas of steepest descent, where the new iterate is the old iterate plus a particular multiple of the residual. The idea of CG is not to use the residual, $\mathbf{r}_i$, to improve the iterate, but to use some other search direction, $\mathbf{p}_i$. The search directions in CG are chosen to be A-orthogonal ($\mathbf{p}_j^H A\mathbf{p}_i = 0$ for $i \neq j$). This is an important property because it means, in exact arithmetic, CG will converge in at most N steps (where N is the size of the matrix A). Given an initial approximation to the solution, $\mathbf{x}_0$, and hence an initial residual, $\mathbf{r}_0$, the method begins by setting the first search direction to be $\mathbf{p}_0 = \mathbf{r}_0$, and then updates the iterates by

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \alpha_i \mathbf{p}_i, \quad \text{where} \quad \alpha_i = \frac{\mathbf{r}_i^H \mathbf{r}_i}{\mathbf{p}_i^H A\mathbf{p}_i}.$$

The residuals are updated by

$$\mathbf{r}_{i+1} = \mathbf{r}_i - \alpha_i A\mathbf{p}_i.$$

Finally the new search direction is found from

$$\mathbf{p}_{i+1} = \mathbf{r}_{i+1} + \beta_{i+1}\mathbf{p}_i,$$

where

$$\beta_{i+1} = \frac{\mathbf{r}_{i+1}^H \mathbf{r}_{i+1}}{\mathbf{r}_i^H \mathbf{r}_i}.$$

The beauty of CG is contained in two wonderful properties. Firstly it is based on three simple two-term recurrences. Secondly the n-th iterate, $\mathbf{x}_n \in \mathbf{x}_0 + K_n(\mathbf{r}_0, A)$, found by running the CG algorithm for n steps minimises the A-norm of the error, where the A-norm is defined as $\|u\|_A := (u, Au)^{\frac{1}{2}}$. By (4.7) this result may be written in terms of the

minimisation of the residual in a different norm, that is

$$\|\mathbf{r}_n\|_{A^{-1}} = \|\mathbf{b} - A\mathbf{x}_n\|_{A^{-1}} = \min_{\mathbf{x}\in\mathbf{x}_0+K_n(\mathbf{r}_0,A)} \|\mathbf{b} - A\mathbf{x}\|_{A^{-1}}. \tag{4.12}$$

Freund et al. [37] state for a general non-Hermitian matrix that $\|\ \|_{A^{-1}}$ is not a norm, so usually the minimisation in expression (4.12) is replaced by one of two conditions. They suggest replacing (4.12) by

$$\|\mathbf{r}_n\|_2 = \min_{\mathbf{x}\in\mathbf{x}_0+K_n(\mathbf{r}_0,A)} \|\mathbf{b} - A\mathbf{x}\|_2$$

and this gives rise to methods known as minimum residual (MR) methods. As an alternative they suggest replacing (4.12) with a Galerkin condition,

$$\mathbf{s}^H\mathbf{r}_n = 0 \quad \text{for all} \quad \mathbf{s} \in K_n(\mathbf{r}_0, A),$$

that is the n-th residual is made orthogonal to all vectors from the n-th Krylov subspace. Methods satisfying (or using) this condition are called orthogonal residual (OR) methods. For Hermitian indefinite linear systems, Paige and Saunders [62] gave two algorithms, MINRES (based on the MR property) and SYMMLQ (based on the OR property). They also showed that these methods can be implemented using short recurrences. For a general non-Hermitian matrix it would be interesting to ask when is it possible for a CG-like scheme, characterised by an MR or OR property, to be implemented using short recurrences. A result due to Faber and Manteuffel ([32] and [33]) provides the answer. Their result states that

A CG-like scheme satisfying an MR or OR property can be implemented using short recurrences only if the matrix A is of the form

$$A = e^{i\theta}(T + \sigma I), \quad \textit{where} \quad T = T^H, \theta \in \mathbb{R}, \ \textit{and} \ \sigma \in \mathbb{C}.$$

Finally in this section we note that a convergence bound for the CG algorithm can be derived (see Elman [29], Golub and Van Loan [44] or the references in Barrett et al. [3] for the details). The bound depends on the spectral condition number of the matrix A and from this it can be shown that the number of iterations required to achieve a set accuracy is proportional to this condition number (see for example Golub and Van Loan [44]).

4.3.3 CG on the Normal Equations

We now consider general non-Hermitian linear systems of equations, that is linear systems (4.1), whose matrix A is a non-Hermitian matrix. An initially attractive idea is to use CG to solve a linear system of equations with A^HA (or AA^H) as its matrix. We note A^HA is an Hermitian positive semi-definite matrix (see Atkinson [2], *p.*478), therefore if A is nonsingular A^HA is positive definite and we can use CG.

There are two approaches to obtain a linear system of this kind (known as the normal equations), firstly we multiply (4.1) throughout by A^H, that is we solve,

$$A^HA\mathbf{x} = \mathbf{b}_1 \quad \text{with} \quad \mathbf{b}_1 = A^H\mathbf{b};$$

this approach is known as CGNR. The second approach is to write $\mathbf{x} = A^H\mathbf{y}$ and so

$$AA^H\mathbf{y} = \mathbf{b},$$

solving this equation by CG is known as CGNE.

In general both of these approaches are not favored because the convergence rate of CG is governed by the condition number of the matrix A, so for these systems the convergence depends on the square of the condition number of this coefficient matrix. Freund et al. [37] review some cases that are optimal in the sense that CGNR and CGNE are equivalent mathematically to CG-type methods based on the MR or OR conditions. Nachtigal et al. [59] suggest that solving the normal equations is often ignored as a method to solve a linear system, when in fact this method may converge quite quickly.

4.3.4 The General Minimum Residual Method (GMRES)

CG, SYMMLQ and MINRES are all based on the Hermitian Lanczos process (see Barrett et al. [3] and Elman [29]). The general minimum residual method (GMRES) is a method for general non-Hermitian matrices which is based on the Arnoldi algorithm [1]. The result of Faber and Manteuffel ([32] and [33]) means that, for a general non-Hermitian matrix, a method which minimises some norm cannot contain short recurrences. So in GMRES all

the previously calculated orthogonal vectors must be stored and therefore the amount of work and storage will grow linearly with the number of iterations.

We will begin by describing Arnoldi's method. Given an initial vector $\mathbf{v}_1$ such that $\|\mathbf{v}_1\|_2 = 1$, Arnoldi's method produces an orthonormal basis for the Krylov subspace span $\{\mathbf{v}_1, \ldots, A^{m-1}\mathbf{v}_1, \ldots\}$. This is constructed via the Gram-Schmidt process, that is for $j = 1, 2, \ldots$ we set

$$\begin{aligned} h_{i,j} &:= \mathbf{v}_i^H A \mathbf{v}_j \qquad\qquad i = 1, \ldots j \\ \widehat{\mathbf{v}}_{j+1} &:= A\mathbf{v}_j - \sum_{i=1}^{j} h_{i,j}\mathbf{v}_i \\ h_{j+1,j} &:= \|\widehat{\mathbf{v}}_{j+1}\|_2, \quad \text{and} \quad \mathbf{v}_{j+1} := \frac{\widehat{\mathbf{v}}_{j+1}}{h_{j+1,j}}, \end{aligned} \tag{4.13}$$

provided $h_{j+1,j}$ is different from zero. These equations may be written more succinctly in matrix form. For example, if we consider m steps of this algorithm (that is take $j = 1, \ldots, m$) and if we define $V_m := [\mathbf{v}_1, \ldots, \mathbf{v}_m]$ (so $V_m \in \mathbb{C}^{N\times m}$) and $H_m := [h_{i,j}]_{1\le i\le j\le m}$ (therefore $H_m \in \mathbb{C}^{m\times m}$ is an upper Hessenberg matrix), then

$$AV_m = V_m H_m + \widehat{\mathbf{v}}_{m+1}\mathbf{e}_m^H \tag{4.14}$$

with $\mathbf{e}_m = (0, \ldots, 0, 1)^H \in \mathbb{C}^m$. The Gram-Schmidt process produces an orthonormal basis, and so from (4.14) we find

$$V_m^H A V_m = H_m. \tag{4.15}$$

The Arnoldi method for eigenvalues is to use the eigenvalues of H_m as approximations to the eigenvalues of A. Most of the hybrid methods we shall describe later in this chapter (see Section 4.4) produce eigenvalue estimates by Arnoldi's method. Indeed, in Chapter 5 we will describe our hybrid method which also uses Arnoldi's method to produce eigenvalue estimates.

To enable us to describe the GMRES algorithm neatly we define $\widehat{H}_m \in \mathbb{C}^{(m+1)\times m}$ to be the matrix containing H_m in its first m rows and $(0, 0, \ldots, 0, h_{m+1,m})$ as its last. Any iterate of the form $\mathbf{x}_m \in \mathbf{x}_0 + K_m(\mathbf{r}_0, A)$ can be written in terms of the orthonormal basis for $K_m(\mathbf{r}_0, A)$ found by Arnoldi's method, starting with $\mathbf{v}_1 = \mathbf{r}_0/\|\mathbf{r}_0\|_2$. That is, any such

iterate can be written

$$\mathbf{x}_m \in \mathbf{x}_0 + V_m \mathbf{y}_m, \tag{4.16}$$

for some $\mathbf{y}_m \in \mathbb{C}^{m\times 1}$. Therefore,

$$\mathbf{r}_m = \mathbf{b} - A\mathbf{x}_m = \mathbf{r}_0 - AV_m\mathbf{y}_m.$$

From (4.14) we find

$$\begin{aligned}
\mathbf{r}_m &= \mathbf{r}_0 - (V_m H_m + h_{m+1,m}\mathbf{v}_{m+1}\mathbf{e}_m^H)\mathbf{y}_m \\
&= \|\mathbf{r}_0\|_2\mathbf{v}_1 - V_{m+1}\widehat{H}_m\mathbf{y}_m \\
&= V_{m+1}\left[\|\mathbf{r}_0\|_2\epsilon_1 - \widehat{H}_m\mathbf{y}_m\right],
\end{aligned}$$

where $\epsilon_1 = (1,0,\ldots,0) \in \mathbb{C}^{(m+1)\times 1}$. The GMRES algorithm produces iterates $\mathbf{x}_m \in \mathbf{x}_0 + K_m(\mathbf{r}_0, A)$ such that $\|\mathbf{r}_m\|_2$ is minimum over all such choices. The $\mathbf{v}_i$ are orthonormal, and therefore

$$\|\mathbf{r}_m\|_2 = \left\| \|\mathbf{r}_0\|_2\epsilon_1 - \widehat{H}_m\mathbf{y}_m \right\|_2. \tag{4.17}$$

Hence, minimising $\|\mathbf{r}_m\|_2$ is equivalent to choosing $\mathbf{y}_m$ to solve a least-squares problem involving the right hand side of this expression. The GMRES algorithm solves this least-squares system by first finding a QR factorisation of the matrix $\widehat{H}_m$. That is

$$Q\widehat{H}_m = \begin{pmatrix} R_m \\ 0 \end{pmatrix},$$

where Q is a unitary matrix and R_m is upper triangular, that is,

$$R_m = \begin{pmatrix} r_{11} & r_{21} & \ldots & r_{1m} \\ 0 & r_{22} & \ldots & r_{2m} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \ldots & 0 & r_{mm} \end{pmatrix}.$$

As Q is unitary we find

$$\|\mathbf{r}_m\|_2 = \left\| Q\|\mathbf{r}_0\|_2\epsilon_1 - \begin{pmatrix} R_m \\ 0 \end{pmatrix}\mathbf{y}_m \right\|_2.$$

Writing $Q\|\mathbf{r}_0\|_2\epsilon_1 = (z_1,\ldots,z_{m+1})^T$ then $\|\mathbf{r}_m\|_2$ is minimised when

$$y_m = z_m/r_{mm},\ \ldots,\ y_1 = \left(z_1 - \sum_{i=2}^{m} r_{1i}y_i\right)/r_{11}.$$

Once $\mathbf{y}_m$ is known then a new iterate can be found from equation (4.16). We note that $\widehat{H}_m$ contains $\widehat{H}_{m-1}$ as a submatrix. Therefore in principle R_k can be updated from R_{k-1} and it can be shown (see [29]) that $\mathbf{x}_m$ and $\|\mathbf{r}_m\|_2$ can be obtained at essentially no cost, at least compared to the cost of producing the next vector in the orthonormal basis for $K_n(\mathbf{r}_0, A)$.

We know all the previous orthonormal vectors are needed to calculate the next vector in the basis. Hence, the work and storage requirements, for GMRES, grow linearly with the number of iterations. Therefore if a large number of iterations is required to obtain convergence, then there are large computational costs and storage requirements. One way around this problem is to restart the algorithm after s steps, replacing the old initial iterate with the new iterate obtained from the GMRES algorithm after s steps. The skill in this idea, is deciding when to restart, that is the value of s. If s is too small then the method will take a long time to converge; if s is too large then more work will be done than is necessary. Another problem with this idea is that there exist matrices for which GMRES has very slow convergence until the last step (see Nachtigal et al. [59]). For such matrices restarting before the last step will result in slow convergence.

Finally in this section we once again mention that the Arnoldi/GMRES idea is the main method used in stage 1 of hybrid algorithms (see Section 4.4). The idea is to use Arnoldi's method for eigenvalues to produce some eigenvalue estimates and at the same time to use GMRES to produce a new initial iterate for the second stage.

4.3.5 BiConjugate Gradient Method (BCG)

For a general non-Hermitian matrix, as we have already mentioned, the conjugate gradient method is not suitable because the residual vectors cannot be made orthogonal with short recurrences (see Faber and Manteuffel [32] and [33], and Section 4.3.2). The GMRES algorithm (see Section 4.3.4) produces a minimisation of residuals, but at the expense of using long recurrences. Another approach is that used by the biconjugate gradient (BCG) method. This method begins with two non-zero vectors, $\mathbf{r}_0$ and $\tilde{\mathbf{r}}_0$; usually one sets $\tilde{\mathbf{r}}_0$ to be $\mathbf{r}_0$, $\bar{\mathbf{r}}_0$, or a random vector. From these initial vectors the BCG algorithm produces two

mutually orthogonal sequences of vectors $\mathbf{r}_n$ and $\tilde{\mathbf{r}}_n$, that is,

$$\tilde{\mathbf{r}}_i^T \mathbf{r}_j = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise.} \end{cases} \tag{4.18}$$

The BCG algorithm does this using short recurrences, but does not provide a minimisation. Once we have $\mathbf{r}_0$ and $\tilde{\mathbf{r}}_0$, the algorithm, taken from Freund et al. [37], is as follows:

Set : $\mathbf{q}_0 = \mathbf{r}_0, \tilde{\mathbf{q}}_0 = \tilde{\mathbf{r}}_0, \rho_0 = \tilde{\mathbf{r}}_0^T \mathbf{r}_0.$

For $n = 1, 2, \ldots$

$\sigma_{n-1} = \tilde{\mathbf{q}}_{n-1}^T A \mathbf{q}_{n-1}$

$\alpha_{n-1} = \rho_{n-1} / \sigma_{n-1}$

$\mathbf{x}_n = \mathbf{x}_{n-1} + \alpha_{n-1} \mathbf{q}_{n-1}$

$\mathbf{r}_n = \mathbf{r}_{n-1} - \alpha_{n-1} A \mathbf{q}_{n-1}$

$\tilde{\mathbf{r}}_n = \tilde{\mathbf{r}}_{n-1} - \alpha_{n-1} A^T \tilde{\mathbf{q}}_{n-1}$

$\rho_n = \tilde{\mathbf{r}}_n^T \mathbf{r}_n$

$\beta_n = \rho_n / \rho_{n-1}$

$\mathbf{q}_n = \mathbf{r}_n + \beta_n \mathbf{q}_{n-1}$

$\tilde{\mathbf{q}}_n = \tilde{\mathbf{r}}_n + \beta_n \tilde{\mathbf{q}}_{n-1}.$

The choices

$$\alpha_{n-1} = \frac{\tilde{\mathbf{r}}_{n-1}^T \mathbf{r}_{n-1}}{\tilde{\mathbf{q}}_{n-1}^T A \mathbf{q}_{n-1}} \quad \text{and} \quad \beta_n = \frac{\tilde{\mathbf{r}}_n^T \mathbf{r}_n}{\tilde{\mathbf{r}}_{n-1}^T \mathbf{r}_{n-1}},$$

ensure the bi-orthogonality relations

$$\tilde{\mathbf{r}}_i^T \mathbf{r}_j = 0 = \tilde{\mathbf{q}}_i^T A \mathbf{q}_j \quad \text{if } i \neq j.$$

The main advantages of BCG are its short recurrences and the fact that it is often as accurate as GMRES, at the cost of two matrix vector products per iteration. BCG, however, suffers from a few drawbacks. Firstly, it often exhibits erratic convergence behaviour. Secondly, each iteration of the method involves a multiplication by both A and A^T. This would be a problem if the matrix A^T was not available, for example, if we were given a

routine to calculate a matrix vector product, but not the matrix itself. Finally the BCG algorithm can suffer from breakdowns. Looking at the above algorithm this occurs when

$$\sigma_{n-1} = 0 \quad \text{and} \quad \tilde{\mathbf{r}}_{n-1} \neq 0, \mathbf{r}_{n-1} \neq 0,$$

or if

$$\rho_{n-1} = 0 \quad \text{and} \quad \tilde{\mathbf{r}}_{n-1} \neq 0, \mathbf{r}_{n-1} \neq 0.$$

Freund et al. [37] discuss the sources of these breakdowns and also the look-ahead Lanczos algorithm, which is a method that handles most breakdowns in the BCG algorithm. The next few methods we describe try to improve on one, or more, of the aforementioned problems of the BCG algorithm. Good reviews of BCG, and related algorithms, are contained in Freund et al. [37] and Barrett et al. [3].

4.3.6 The Conjugate Gradient Squared Method (CGS)

In 1989, Sonneveld [75] introduced his conjugate gradients squared algorithm (CGS). This was the first method, based on the BCG algorithm, avoiding the multiplications by A^T. It is relatively straightforward to prove that the vectors $\mathbf{r}_n$ and $\tilde{\mathbf{r}}_n$, in the BCG algorithm, satisfy $\mathbf{r}_n = p_n(A)\mathbf{r}_0$ and $\tilde{\mathbf{r}}_n = p_n(A^T)\tilde{\mathbf{r}}_0$ for some $p_n \in P_n$. Therefore, we find that

$$\rho_n = \tilde{\mathbf{r}}_0^T \left[p_n(A)\right]^2 \mathbf{r}_0,$$

and ρ_n can be calculated without the A^T multiplication. In a similar way we find that

$$\sigma_n = \tilde{\mathbf{r}}_0^T A \left[q_n(A)\right]^2 \mathbf{r}_0,$$

for some $q_n \in P_n$. Sonneveld [75] showed that both ρ_n and σ_n can be updated by using short recurrences. Algorithms for the implementation of CGS are contained in Nachtigal et al. [60] and Barrett et al. [3]. It turns out, for the CGS algorithm, that

$$\mathbf{r}_{2n} = \left(p_n(A)\right)^2 \mathbf{r}_0,$$

where $p_n(A)$ is the residual polynomial for the BCG algorithm. That is, the CGS residual polynomials are just the squares of the BCG residual polynomials. For this reason the CGS algorithm shows even more erratic convergence than the BCG algorithm. When BCG diverges CGS will diverge, but when BCG converges, CGS may still diverge.

4.3.7 BiCGSTAB and BiCGSTAB2

In this section we will discuss BiCGSTAB, the method of Van der Vorst [79], and BiCGSTAB2, the method of Gutknecht [47]. These methods were developed to avoid the erratic convergence of both the CGS and BCG algorithms. For the CGS algorithm one obtains iterates of the form

$$\mathbf{x}_{2n} = \mathbf{x}_0 + K_{2n}(\mathbf{r}_0, A), \tag{4.19}$$

where as mentioned above

$$\mathbf{r}_{2n} = (p_n(A))^2 \mathbf{r}_0, \tag{4.20}$$

and $p_n(A)$ is the BCG residual polynomial. BiCGSTAB generates iterates satisfying equation (4.19), but in contrast to (4.20), the residuals are found from

$$\mathbf{r}_{2n} = p_n(A)G_n(A)\mathbf{r}_0,$$

where G_n is a polynomial of degree n, with $G_n(0) = 1$. The G_n satisfy $G_n(x) = (1 - g_n x)G_{n-1}(x)$, where the parameters g_n are found by determining local steepest descent solutions. We refer the reader to Freund et al. [37] and Van der Vorst [79] for the details. BiCGSTAB often exhibits smoother convergence than both BCG and CGS.

Instead of linear factors, Gutknecht [47] introduces quadratic factors into the polynomial G_n. His reason for doing so is to allow $G_n(x)$ to have complex as well as real roots. For real A, the BiCGSTAB method will only give rise to real g_n, even if the eigenvalues of A are complex. Therefore, the method of Gutknecht could be a useful one. The method is known as BiCGSTAB2.

4.3.8 The Quasi-Minimal Residual algorithm (QMR)

The QMR algorithm of Freund and Nachtigal [36] is a method designed to overcome some of the short falls in the BCG algorithm; in particular its breakdowns. As in the BCG algorithm, it produces two sets of mutually orthogonal vectors, $\{\mathbf{v}_j\}$ and $\{\mathbf{w}_j\}$, (see Equation (4.18)). To avoid breakdowns that occur in the BCG algorithm, the QMR algorithm uses

the look-ahead Lanczos process. The idea of this procedure is to get around cases when $\mathbf{w}_n^T\mathbf{v}_n = 0$, but neither $\mathbf{w}_n$, nor $\mathbf{v}_n$ are zero. It does this by relaxing the biorthogonality condition (see Equation (4.18)) when a breakdown occurs (see Freund et al. [37] for the details). As in Section 4.3.4, if we start the method with $\mathbf{v}_1 = \mathbf{r}_0/\|\mathbf{r}_0\|_2$, then, after n steps of the QMR method, the set $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ spans the Krylov subspace $K_n(\mathbf{r}_0, A)$. Therefore

$$\mathbf{x}_n = \mathbf{x}_0 + K_n(\mathbf{r}_0, A) = \mathbf{x}_0 + V_n\mathbf{y}_n, \tag{4.21}$$

for some $\mathbf{y}_n \in \mathbb{C}^n$, with $V_n = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$ as in Section 4.3.4. The look-ahead Lanczos process produces vectors $\mathbf{v}_i$ such that

$$AV_n = V_{n+1}\widehat{H}_n, \tag{4.22}$$

where $\widehat{H}_n \in \mathbb{C}^{(n+1)\times n}$ is an upper Hessenberg block tridiagonal matrix (see Freund et al. [37] or Freund and Nachtigal [36]). From Equations (4.21) and (4.22) we find

$$\begin{aligned}\mathbf{r}_n &= \mathbf{b} - A\mathbf{x}_n \\ &= \mathbf{r}_0 - AV_n\mathbf{y}_n \\ &= V_{n+1}\left(\|\mathbf{r}_0\|_2\epsilon_1 - \widehat{H}_n\mathbf{y}_n\right),\end{aligned}$$

where ϵ_1 is given in Section 4.3.4. Unlike in the GMRES algorithm (see Section 4.3.4), V_{n+1} is not a unitary matrix. To find $\mathbf{y}_n$, however, the QMR algorithm still solves

$$\left\|\|\mathbf{r}_0\|_2\epsilon_1 - \widehat{H}_n\mathbf{y}_n\right\|_2 = \min_{\mathbf{y}}\left\|\|\mathbf{r}_0\|_2\epsilon_1 - \widehat{H}_n\mathbf{y}\right\|_2.$$

As mentioned above $\widehat{H}_n$ is an upper Hessenberg matrix. Therefore this minimisation may be solved efficiently by the QR algorithm as in Section 4.3.4. The residual, $\mathbf{r}_n$, is not actually minimised by this process. Therefore the solution is viewed as a quasi-minimal residual and this gives rise to the method's name. Like the BCG algorithm, the QMR algorithm involves multiplications by A and A^T. There exist transpose free methods based on the QMR algorithm, such as TFQMR (see Freund et al. [37]).

4.4 Hybrid Methods

In this section we will review hybrid algorithms for the solution of linear systems of equations (4.1). Hybrid methods were introduced to overcome problems with Krylov subspace

methods. For example, the GMRES algorithm (see Section 4.3.4) can be very costly in terms of both storage and the number of vector operations required to converge. As is mentioned in Section 4.3.4 a way around this problem is to restart the GMRES algorithm; unfortunately this can lead to slow convergence. A completely different approach is that taken by hybrid methods. A hybrid method contains two stages. In the first stage a method like CGS, GMRES or QMR is used. The purpose of this stage is to gain some information about the coefficient matrix, A. Once a few steps of Stage 1 have been completed then the method switches to the second stage. In this stage a method is used that applies the information obtained in the first stage. An example of such a method is Chebyshev iteration (see Section (4.4.1)). A good review of Hybrid methods is contained in Nachtigal et al. [60]. In this review they summarise hybrid methods as follows:

Stage 1. Acquire information about A via an iteration requiring no a priori information about A.

Stage 2. Apply this information via a method requiring a priori information about A.

As mentioned previously hybrid methods were designed to overcome the high cost of both work and storage in methods like GMRES. The methods used in stage 2 tend to cost less per step than those used in stage 1. Therefore the change from a stage 1 type method to a stage 2 type method could be useful. In methods like BCG (see Section (4.3.5)) multiplications by A^H (or A^T) are required. Sometimes this matrix is either not available or costly to compute. In these cases hybrid methods will have an advantage over methods such as BCG. If A^H (or A^T) is available or easy to compute then hybrid methods may or may not compete with such methods. To quote Nachtigal et al. [60], "the natural realm of applicability of hybrid methods is to problems where Krylov subspace methods take fewer steps than the alternatives." By this they mean that hybrid methods might be used successfully for matrices where methods like GMRES outperform methods like CGS, BCG and CGNR.

We will now review the existing hybrid methods, beginning with Manteuffel's algorithm which appeared in 1978.

4.4.1 Manteuffel's algorithm

The first hybrid method to appear in the literature is one by Manteuffel [56]. In his paper he gives three methods for estimating a few of the extreme eigenvalues of the matrix A, one of which is a modified power iteration. An ellipse is then placed around these eigenvalue estimates. From this ellipse two parameters, c and d, are found. Given these parameters the algorithm then switches to an iteration based on the Chebyshev polynomials (Chebyshev iteration). The iteration is carried out as follows:

$$\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0, \quad \boldsymbol{\Delta}_0 = \frac{1}{d}\mathbf{r}_0, \quad \mathbf{x}_1 = \mathbf{x}_0 + \boldsymbol{\Delta}_0,$$

and then for $n = 1, 2, \ldots$

$$\mathbf{r}_n = \mathbf{b} - A\mathbf{x}_n, \quad \boldsymbol{\Delta}_n = \alpha_n \mathbf{r}_n + \beta_n \boldsymbol{\Delta}_{n-1}, \quad \mathbf{x}_{n+1} = \mathbf{x}_n + \boldsymbol{\Delta}_n,$$

where

$$\alpha_n = \frac{2}{c}\frac{T_n\left(\frac{d}{c}\right)}{T_{n+1}\left(\frac{d}{c}\right)}, \quad \beta_n = \frac{T_{n-1}\left(\frac{d}{c}\right)}{T_{n+1}\left(\frac{d}{c}\right)}$$

and $T_n(z)$ is the n-th Chebyshev polynomial. From a property of Chebyshev polynomials, that is, the recurrence relation (see Section 2.7), these parameters may be calculated recursively by

$$\alpha_1 = \frac{2d}{2d^2 - c^2}, \quad \beta_1 = d\alpha_1 - 1, \quad \alpha_n = \left[d - \left(\frac{c}{2}\right)^2 \alpha_{n-1}\right]^{-1}, \quad \beta_n = d\alpha_n - 1.$$

We define

$$p_n(\lambda) = T_n\left(\frac{d-\lambda}{c}\right) \Big/ T_n\left(\frac{d}{c}\right),$$

and then using the recurrence relation for the Chebyshev polynomials, and the definitions of α_n and β_n, it is relatively straightforward to see that the Chebyshev iteration gives rise to an iterative method with residual polynomials given by the scaled and shifted Chebyshev polynomials, $p_n(\lambda)$. That is $\mathbf{r}_n = p_n(A)\mathbf{r}_0$.

The Chebyshev iteration is simple to implement provided that c and d can be calculated. Manteuffel [56] gives an adaptive procedure to estimate the convex hull of the spectrum of A. It is based on estimating extremal eigenvalues by the power method (or a modified

version of it). The estimate of the convex hull is then placed inside an ellipse and the parameters, c and d, for this ellipse can be found by a strategy given in Manteuffel [55]. In the notation of Nachtigal et al. [60], Manteuffel's algorithm may be represented by

Modified Power method $\rightarrow$ eigenvalue estimates $\rightarrow$ ellipse $\rightarrow$ Chebyshev iteration.

The Chebyshev iteration, and therefore Manteuffel's hybrid method, can be used to solve nonsymmetric linear systems where the matrix A has eigenvalues that lie in the right (left) half plane. This means that Manteuffel's method, like many other hybrid iterative methods, will fail when the origin belongs to the convex hull of the spectrum of A. The method will work well when the spectrum of the matrix can be well approximated by an ellipse not containing the origin.

4.4.2 The method of Smolarski and Saylor

A problem with Manteuffel's algorithm is that it only works well for matrices whose spectra may be well approximated by an ellipse. In 1981 Smolarski and Saylor [74] proposed a modification of Manteuffel's algorithm which replaces the ellipse with a more general region in the complex plane. Again they begin with the modified power iteration and obtain some eigenvalue estimates. They then place a polygonal region around these estimates and to find a residual polynomial they solve a least-squares approximation problem on the polygon.

They define an L_2-optimal polynomial, R_n, to be a polynomial of degree n for which $\|R_n\|_w$ is a minimum, where

$$\|f\|_w^2 = (f,f)_w = \frac{1}{L}\int_\gamma f(\lambda)\overline{f(\lambda)}w(\lambda)|d\lambda|,$$

γ is a curve in the complex plane, $L = \int_\gamma |d\lambda|$ is the length of the curve γ, and w is some weight function. The contour γ is chosen to be the above polygonal region enclosing the eigenvalue estimates. If $R_n(\lambda) = 1 + \eta_1\lambda + \ldots + \eta_n\lambda^n$, then

$$\|R_n\|_w^2 = (\boldsymbol{\eta}, B\boldsymbol{\eta}), \tag{4.23}$$

where $\boldsymbol{\eta} = (1, \eta_1, \ldots, \eta_n)^T$, $B = (\beta_{ij})$ is the modified moment matrix associated with the basis $\{1, \lambda, \ldots, \lambda^n\}$ and $\beta_{ij} = (\lambda^i, \lambda^j)_w$. We refer the reader to Smolarski and Saylor [74]

for a procedure to minimise expression (4.23). Once this minimum is found, the residual polynomial, R_n, is implemented by Richardson iteration.

In 1991 Saylor and Smolarski [70] modified their method to include the Arnoldi/ GMRES algorithm. Their new method uses the link between the Arnoldi and GMRES algorithms to obtain both a good initial iterate for stage 2 of the hybrid method, and eigenvalue estimates. In the notation of Nachtigal et al. [60] it may be represented as

$$\text{Arnoldi/GMRES} \rightarrow \text{eigenvalue estimates} \rightarrow \text{polygon} \rightarrow L^2\text{-optimal polynomial} \rightarrow \text{Richardson iteration.}$$

4.4.3 The method of Elman, Saad and Saylor

In 1986 a very important modification of Manteuffel's algorithm was introduced by Elman et al. [30]. They were the first people to replace the modified power iteration of stage 1 by the Arnoldi algorithm. Most of the hybrid methods introduced after this algorithm use this idea, including the method of Saylor and Smolarski [70] which we have just described. The big advantage of using the Arnoldi algorithm to obtain eigenvalue estimates is its link with the GMRES algorithm (see Section 4.3.4). Therefore at the end of stage 1 of the hybrid method one has a better initial solution to the linear system, as well as eigenvalue estimates. The rest of this algorithm proceeds as Manteuffel's did. That is, Elman et al. [30] place an ellipse around the eigenvalue estimates, and then use the Chebyshev polynomials for this ellipse as the residual polynomials. Once again their method will only work well when the spectrum of the coefficient matrix can be well approximated by an ellipse. In the notation of Nachtigal et al. [60] this method is represented by

$$\text{Arnoldi/GMRES} \rightarrow \text{eigenvalue estimates} \rightarrow \text{ellipse} \rightarrow \text{Chebyshev iteration.}$$

4.4.4 The method of Saad

The hybrid method of Youcef Saad [69] was published in 1987. It is based on the method of Smolarski and Saylor [74] (see Section (4.4.2)) with two notable differences. Firstly, he obtains eigenvalue estimates by the Arnoldi/GMRES algorithm, and secondly, he improves

the stability of the method by constructing a well-conditioned basis of shifted Chebyshev polynomials, before finding the L^2-optimal polynomial.

As mentioned above the method begins with a few steps of the Arnoldi/GMRES algorithm to produce both a good initial solution for stage 2 and eigenvalue estimates. A polygonal region, H, is then placed around these estimates. Smolarski and Saylor [74] give their method with respect to the basis $\{1, \lambda, \ldots, \lambda^j\}$. Unfortunately, for this basis, the modified moment matrix, B (see Section (4.4.2)), is ill-conditioned and therefore the method is unstable. Saad [69] considered a more stable basis, namely $\{1, \ldots, t_j(\lambda)\}$, where the $t_j(\lambda)$ are scaled and shifted Chebyshev polynomials. Along each edge, E_ν, of H he considers a weight function w_ν (see Saad [69] for the details). He then defines an inner product,

$$\langle p, q \rangle_w = \int_{\partial H} p(\lambda)\overline{q(\lambda)}w(\lambda)|d\lambda| = \sum_{\nu=1}^{\mu} \int_{E_\nu} p(\lambda)\overline{q(\lambda)}w_\nu(\lambda)|d\lambda|,$$

where $\mu + 1$ is the number of edges of the domain H.

In the paper he describes how to generate the modified moment matrix for this more stable basis. Along each edge E_ν for $\nu = 1, \ldots, \mu$ he expresses the polynomials $t_j(\lambda)$ in terms of the Chebyshev polynomials, $T_i(\xi)$, for the edge E_ν. That is he writes

$$t_j(\lambda) = \sum_{i=0}^{j} \gamma_{i,j}^{\nu} T_i(\xi),$$

where ξ depends on λ. By using the recurrence relation for Chebyshev polynomials, Saad is able to calculate, for each $\nu = 1, 2, \ldots, \mu$, the coefficients $\gamma_{i,j}^{\nu}$ (see Proposition 2 in Saad [69] for the details). He is then able to calculate the coefficients, $m_{ij} = \langle t_i, t_j \rangle_w$, of the modified moment matrix, M_n, from these $\gamma_{i,j}^{\nu}$ (see Proposition 3 in Saad [69]). The beauty of this method, for calculating the coefficients of M_n, is that it explicitly calculates the required integrals. As in Smolarski and Saylor [74], once the matrix M_n is known, the residual polynomial, $p_n(\lambda)$, (or iteration polynomial, $q_n(\lambda)$) is found by minimising

$$\|p_n(\lambda)\|_w = \|1 - \lambda q_n(\lambda)\|_w = \left\langle 1 - \lambda q_n(\lambda), 1 - \lambda q_n(\lambda) \right\rangle_w^{\frac{1}{2}}.$$

Writing $p(\lambda) = \sum_{i=0}^{n} \eta_i t_i(\lambda)$ and $q(\lambda) = \sum_{j=0}^{n} \theta_j t_j(\lambda)$ then the inner product, given above, becomes

$$\langle p, q \rangle_w = (M_n \boldsymbol{\eta}, \boldsymbol{\theta}),$$

where $\boldsymbol{\eta} = (\eta_0, \ldots, \eta_n)^T$ and $\boldsymbol{\theta} = (\theta_0, \ldots, \theta_n)^T$. Using this expression for the inner product and writing $q_n(\lambda) = \sum_{i=0}^{n-1} \eta_i t_i(\lambda)$ then

$$\|p_n(\lambda)\|_w^2 = [\mathbf{e}_1 - T_n\boldsymbol{\eta}]^H M_n [\mathbf{e}_1 - T_n\boldsymbol{\eta}],$$

where $\mathbf{e}_1 = (1, 0, \ldots, 0)^T$, $\boldsymbol{\eta} = (\eta_0, \ldots, \eta_{n-1})^T$,

$$T_n = \begin{pmatrix} \alpha_0 & \delta_1 & & & \\ \beta_1 & \alpha_1 & \delta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \delta_{n-1} \\ & & & \beta_{n-1} & \alpha_{n-1} \\ & & & & \beta_n \end{pmatrix},$$

and α_i, β_{i+1} and δ_i are the coefficients in the recurrence relation for the polynomial $t_{i+1}(\lambda)$. The matrix M_n is a symmetric positive definite matrix and therefore there exists a Choleski factorisation $M_n = LL^T$ (see Golub and Van Loan [44]). Noting that $L^T\mathbf{e}_1 = l_{11}\mathbf{e}_1$ and writing $L^T T_n = F_n$, where F_n is an $(n+1) \times n$ upper Hessenberg matrix we find

$$\|p_n(\lambda)\|_w = \|l_{11}\mathbf{e}_1 - F_n\boldsymbol{\eta}\|_2.$$

This expression can be minimised, by using the QR algorithm, in a similar way to minimising expression (4.17) in the GMRES algorithm (see Section 4.3.4). Once the coefficients $\boldsymbol{\eta}$ are known the iteration polynomial, $q_n(\lambda)$, is implemented using a second-order Richardson iteration (see Saad [69] for the details). Nachtigal et al. [60] summarise this method as

Arnoldi/GMRES $\rightarrow$ eigenvalue estimates $\rightarrow$ polygon $\rightarrow$ Chebyshev basis $\rightarrow$
L^2-optimal polynomial $\rightarrow$ second-order Richardson iteration.

4.4.5 The Arnoldi/Faber method of Starke and Varga

In this section we will review the Arnoldi/Faber method of Starke and Varga [76]. This method is the major motivation for our work in Chapter 5. In fact the only essential difference between the two methods is that Starke and Varga use the Faber polynomials for a polygonal region, whereas we use the Faber polynomials for an annular sector. Starke and Varga's method begins, as usual, with a few steps of the Arnoldi/GMRES algorithm.

Therefore, at the start of stage 2 we have eigenvalue estimates and a good initial solution. The method, as set up in their paper, is only suitable for matrices whose eigenvalues are symmetric with respect to the real axis. Starke and Varga mention the possibility of implementing their method for a general non-Hermitian matrix, but they do not give the details and it would seem that their ideas must change to be able to do this. Once some eigenvalue estimates are known, then a polygonal region is placed around these estimates (see Starke and Varga [76] for the details) and a numerical conformal mapping package is used to find the required conformal map. The mapping package used is SCPACK (see Trefethen [77]) which is only set up to find interior mapping functions. The polygonal region they obtain is symmetric with respect to the real axis. Therefore, they are able to use a "trick" given by Li in his thesis, (see Starke and Varga and the references therein) to find the exterior conformal map. From this numerical conformal map they are able to find the Faber polynomials for the polygonal region. These Faber polynomials, suitably normalised, are then used as the residual polynomials in stage 2 of Starke and Varga's method. The iteration polynomials, obtained from this choice of residual polynomials, are implemented using a Horner iteration. In the notation of Nachtigal et al. [60] this method is represented by

$$\text{Arnoldi/GMRES} \rightarrow \text{eigenvalue estimates} \rightarrow \text{polygon} \rightarrow \text{Faber polynomials} \rightarrow \text{conformal map} \rightarrow \text{Horner iteration.}$$

Starke and Varga [76] choose the Faber polynomials as residual polynomials for two main reasons. Firstly, they, amongst others, show that the Faber polynomials, for a region D in the complex plane, are near-best approximation to zero (see Sections 1.2.3 and 5.3.3). Secondly, the Faber polynomials are small on level sets (see Section 3.5) of the domain D. As Starke and Varga point out this is an important property when dealing with non-normal matrices, because for these matrices the convergence behaviour of polynomial iterations does not only depend on the spectrum of A, but also on neighbourhoods of the domain (for example, see Trefethen's ϵ-pseudospectra [78]).

We would like to finish this review of the Arnoldi/Faber method by pointing out some of its disadvantages. Firstly, the method requires a numerical conformal mapping package to produce the Faber polynomials. It would be better if we could avoid this by considering a

region in the complex plane whose Faber polynomials are known analytically (see Chapter 5 and the Faber polynomials for an annular sector). Secondly, as it is set up in Starke and Varga's paper, the method can only be applied to matrices whose spectra are symmetric with respect to the real axis. They do mention that it is possible to apply their method to matrices with eigenvalue estimates situated almost anywhere in the complex plane. Their method, however, would have to change to enable this. For example, their method for placing a polygonal region around the eigenvalue estimates would change if the eigenvalues were not symmetrically placed about the real axis. It would be better if we could have a general purpose region which could be scaled and rotated to anywhere in the complex plane. Therefore, the method would be the same no matter where the eigenvalues were situated. The annular sector would seem to be such a region (see Chapter 5).

4.4.6 Hybrid GMRES

This section contains a review of the hybrid method of Nachtigal et al. [60]. Their method was the first hybrid method not to use an eigenvalue estimator. They give two reasons for dropping the eigenvalue estimates from their method. Firstly, for non-normal matrices considering only the eigenvalues may not provide enough information about the matrix. Nachtigal et al. [60] give a simple example of a polynomial which is small on the spectrum of a particular matrix, but when used in stage 2 of a hybrid method it leads to divergence. Instead of considering the spectrum of a matrix they suggest considering its pseudospectrum. The second problem with eigenvalues is the existence of matrices whose eigenvalues and pseudospectra are far from zero, but whose eigenvalue estimates are small or even zero (see Nachtigal et al. [60] for an example). Nachtigal et al. [60] note that the first problem is not really a problem, because eigenvalue estimates tend to estimate the pseudospectrum of a matrix rather than its spectrum. As mentioned above, it is usually the pseudospectrum that governs the convergence of the method, not the spectrum. The second problem is more important and is something one must be aware of when using an eigenvalue estimator such as the the Arnoldi algorithm. For these two reasons, Nachtigal et al. [60] propose their hybrid GMRES algorithm, without an eigenvalue estimator. In

their notation the method may be represented as

$$\text{GMRES} \rightarrow \text{GMRES polynomial} \rightarrow \text{Richardson iteration.}$$

Their method begins with a few steps of the GMRES algorithm. They then calculate the GMRES polynomial explicitly and implement it iteratively. After n steps of the GMRES algorithm they have a matrix, V_n, whose columns span the Krylov subspace $K_n(\mathbf{r}_0, A)$ (see Section 4.3.4). They write K_n for the matrix whose columns are the Krylov vectors, that is,

$$K_n := [\mathbf{r}_0, \ldots, A^{n-1}\mathbf{r}_0].$$

Therefore, as the columns of V_n and K_n span the same space, they find that $V_n = K_n C_n$ for some upper triangular matrix

$$C_n = \begin{pmatrix} c_{11} & \cdots & c_{1n} \\ & \ddots & \vdots \\ & & c_{nn} \end{pmatrix}.$$

Equation (4.16) gives,

$$\mathbf{x}_n = \mathbf{x}_0 + V_n \mathbf{y}_n = \mathbf{x}_0 + K_n C_n \mathbf{y}_n.$$

As the iteration proceeds, Nachtigal et al. [60] generate the elements of C_n column by column. Once the matrix C_n and the vector $\mathbf{y}_n$ are known, they form the vector $C_n\mathbf{y}_n = (\alpha_0, \ldots, \alpha_{n-1})^T$ and then

$$\mathbf{x}_n = \mathbf{x}_0 + \left(\alpha_0 I + \ldots + \alpha_{n-1} A^{n-1}\right) \mathbf{r}_0.$$

The GMRES residual polynomial is then found from

$$p_n(z) = 1 - z q_{n-1}(z), \quad \text{where} \quad q_{n-1}(z) = \alpha_0 + \ldots + \alpha_{n-1} z^{n-1}.$$

Nachtigal et al. [60] choose to implement this polynomial in a Richardson iteration. To do this they begin by numerically finding the roots of the polynomial and factorising it as $p_n(z) = \prod_{i=1}^{n} (1 - z/\xi_i)$. This polynomial is then implemented in a Richardson iteration as in Smolarski and Saylor [74], that is

For $j = 1$ to n

$$\mathbf{x}_j = \mathbf{x}_{j-1} + \mathbf{r}_{j-1}/\xi_j, \quad \text{where} \quad \mathbf{r}_{j-1} = \mathbf{b} - A\mathbf{x}_{j-1}.$$

Out of an interest in stability, they choose to apply the roots in an order governed by the weighted Leja ordering (see Nachtigal et al. [60] and the references therein). Assuming the roots are distinct then this ordering is defined by

$$|\xi_j| \prod_{i=1}^{j-1} |\xi_j - \xi_i| = \max_{j \le l \le n} |\xi_l| \prod_{i=1}^{j-1} |\xi_l - \xi_i|, \qquad j = 1, 2, \ldots, n-1,$$

where the first step is $|\xi_1| = \max_{1 \le 1 \le n} |\xi_l|$.

This is the gist of the hybrid GMRES method. They do, however, discuss when to switch from the first to the second stage of the hybrid algorithm, and the possibility of cycling back to the first stage if the second stage convergence rate is slow. We refer the reader to their paper for the details.

4.4.7 Other hybrid methods

There are two other hybrid methods in the literature. The first is a method proposed by Elman and Streit [31]. It begins, as most recent hybrid methods do, with a few steps of the Arnoldi/GMRES algorithm. After obtaining a polygonal region around some eigenvalue estimates they solve an L^∞-approximation problem on this domain and then implement the polynomial using Horner iteration. In 1991, Li [53] gave a hybrid iterative method based on a conformal map. The steps to produce a polygonal region are the same as in most of the hybrid methods we have described, that is,

$$\text{Arnoldi/GMRES} \rightarrow \text{eigenvalue estimates} \rightarrow \text{polygon}.$$

He then uses a conformal mapping package, SCPACK (see Trefethen [77]), to find the conformal map from the exterior of the polygonal region onto the exterior of the unit disc. Next he produces a rational approximation based on the conformal map and implements it in a (k, l)-step iteration (see Li [53] for the details). In this paper, Li [53] also reviews the semi-iterative methods (SIM's) of Eiermann et al. [17, 18, 19, 20], for which the generalised Faber series play an important role. In Eiermann [17] three SIM's are given; SIM 1, $q_{n-1}(z)$ is the truncated generalised Faber series for $1/(1-z)$; SIM 2 where $q_{n-1}(z)$ is the truncated Faber series for $1/(1-z)$; and SIM 3 where $p_n(z)$ is chosen to be $F_n(z)/F_n(1)$. The Faber polynomials in these cases are defined for a region D in the complex plane such that

$1 \notin D$. Our Faber polynomials are defined on a domain D such that $0 \notin D$. Allowing for this difference the three SIM's become; SIM 1, $q_{n-1}(z)$ is the truncated generalised Faber series for $1/z$; SIM 2 where $q_{n-1}(z)$ is the truncated Faber series for $1/z$; and SIM 3 where $p_n(z)$ is chosen to be $F_n(z)/F_n(0)$.

Finally in this section, we would like to note the existence of methods which become hybrid methods when we add an eigenvalue estimation routine, such as Arnoldi/GMRES, to them. We have already mentioned the SIM's given in Eiermann [17] based on Faber polynomials and Faber series. For other such methods, we refer the reader to the references given in Nachtigal et al. [60].

Chapter 5

A Hybrid Method for Solving Nonsymmetric Systems of Linear Equations Using the Faber Polynomials for Annular Sectors.

5.1 Introduction

In this chapter we will give a hybrid method to solve the linear system (4.1), based on the Faber polynomials for an annular sector. The coefficient matrix, A, for this linear system will in general be, a large, sparse, non-Hermitian, complex matrix. Our method will be similar to Starke and Varga's method (see Section 4.45). In the first stage, like the Arnoldi-Faber method of Starke and Varga [76], and many other hybrid iterative methods, we will use Arnoldi's method to produce some eigenvalue estimates for A. In their method, Starke and Varga placed a polygonal region around these estimates and from the conformal mapping, produced by a numerical conformal mapping package, they were able to find the Faber polynomials for this polygonal region. They then use these polynomials as the iteration polynomials in the second stage of their method. Here, in contrast, we will place an annular sector around the eigenvalue estimates. Then in the second stage we will use the Faber polynomials for this annular sector as the iteration polynomials. There are a few reasons for this choice: firstly an annular sector can easily be placed around any eigenvalue estimates bounded away from zero, secondly the Faber polynomials are known analytically for an annular sector, and finally using an annular sector allows us to consider cases of matrices that Starke and Varga's method could not deal with (for real matrices, specifically when we get two eigenvalue estimates λ_1, λ_2 such that $\lambda_1 < 0 < \lambda_2$). Once we have found the desired Faber polynomials we shall implement them in a Horner-type iteration. Therefore in the notation of Nachtigal et al. [60] our method is given by

$$\text{Arnoldi/GMRES} \rightarrow \text{eigenvalue estimates} \rightarrow \text{annular sector} \rightarrow \text{Faber polynomials} \rightarrow \text{Horner iteration.}$$

5.2 The Faber Polynomials for any Annular Sector

In Chapter 2 we described how to generate the conformal mapping and hence the Faber polynomials for an annular sector of the form

$$Q = \{z : R \leq |z| \leq 1, \theta \leq |\arg z| \leq \pi\}, \quad 0 < \theta \leq \pi.$$

In this section we will describe how to generate the Faber polynomials for any annular sector placed anywhere in the complex plane. The preliminaries to this were given in

Chapter 2. In Sections 2.2 and 2.4.1 we described how to find the Faber polynomials for an annular sector placed symmetrically about the positive real axis from an annular sector Q. In Section 2.6 we described how to determine the mapping from the complement of the unit disc onto the complement of $Q(R_1, R_2, \theta)$, an annular sector placed symmetrically about the negative real axis, with inner radius R_1, outer radius R_2 and half-angle $\pi - \theta$. We found that the mapping is given by $Z = R_2 z = R_2 \psi(w)$, where $\psi(w)$ maps the complement of the unit disc onto the complement of the annular sector $Q = Q(R, 1, \theta)$, with $R = R_1/R_2$. We also found that the transfinite diameter of $Q(R_1, R_2, \theta)$ is given by $\rho^* = R_2\rho$, where ρ is the transfinite diameter of Q (see Section 2.2.3). From equation (1.4) we find

$$Z = \rho^* \left(w + \beta_0 + \frac{\beta_1}{w} + \ldots \right).$$

Therefore the Laurent coefficients of this mapping are the same as those for the mapping $\psi(w)$ and the polynomial $\Phi_n(z)$ is unaltered (see equation (2.30)). Hence, the Faber polynomial, of degree n, for the annular sector $Q(R_1, R_2, \theta)$ is given by

$$F_n(Z) = (\rho^*)^n \Phi_n \left(\frac{Z}{\rho^*} \right) = R_2^n \rho^n \Phi_n \left(\frac{z}{\rho} \right) = R_2^n F_n(z) = R_2^n F_n \left(\frac{Z}{R_2} \right). \tag{5.1}$$

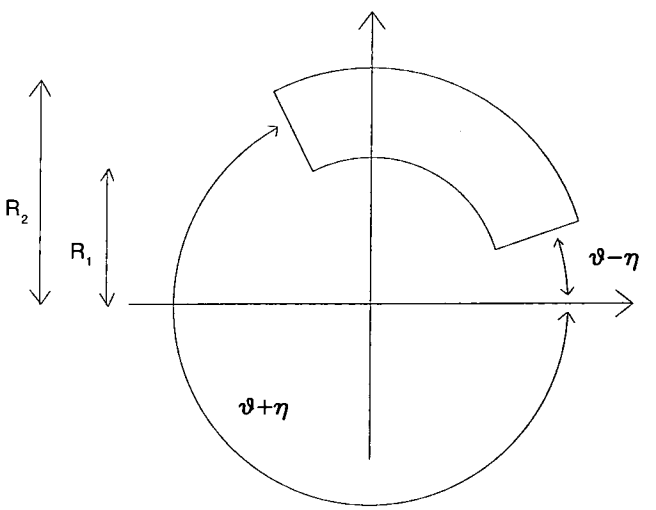

Figure 5.1: The annular sector $\bar{Q}$.

When considering matrices whose eigenvalue estimates are not symmetric with respect to the negative real axis, it will be important that we can find the Faber polynomials for

the annular sector

$$\bar{Q} = \{Z : R_1 \leq |Z| \leq R_2,\ \theta - \eta \leq \arg Z \leq 2\pi - \theta - \eta\}. \tag{5.2}$$

Figure 5.1 shows the annular sector $\bar{Q}$, which is simply the annular sector $Q(R_1, R_2, \theta)$ rotated through an angle $-\eta$ where $\eta \in [0, 2\pi)$ is the angle, measured in the anti-clockwise direction, which the bisecting ray of the sector makes with the negative real axis. We note that the exterior of $Q(R, 1, \theta)$ is mapped onto the exterior of $\bar{Q}$ by

$$Z = R_2 e^{-i\eta} z.$$

Then $F_n(z) = F_n(Ze^{i\eta}/R_2)$ and the corresponding monic Faber polynomial for $\bar{Q}$ is

$$\bar{F}_n(Z) = R_2^n e^{-in\eta} F_n\left(\frac{Ze^{i\eta}}{R_2}\right). \tag{5.3}$$

As a check on this expression we note that it gives the same polynomials as equation (5.1) when $\eta = 0$.

5.3 The hybrid method

In this section we will introduce our hybrid method which uses the Faber polynomials for an annular sector as the iteration polynomials. In Section 5.3.1 we find some eigenvalue estimates with the use of Arnoldi's method. Then in Section 5.3.2 we place an annular sector around these estimates. Section 5.3.3 gives, amongst other things, reasons for choosing the Faber polynomials as a sensible choice for the residual polynomials. Finally in Section 5.3.4 we give some implementation details.

5.3.1 Eigenvalue estimation by Arnoldi's method

As we have mentioned, in the first stage of the hybrid method we will use Arnoldi's method (see Section 4.3.4) to compute m eigenvalue estimates of the non-Hermitian matrix, $A \in \mathbb{C}^{N\times N}$. After running Arnoldi's algorithm for m steps we obtain an orthonormal basis for the Krylov subspace $K_m := \text{span}\{\mathbf{v}_1, \ldots, A^{m-1}\mathbf{v}_1\}$, where $\mathbf{v}_1 = \mathbf{r}_0/\|\mathbf{r}_0\|_2$. Referring to equation (4.15) we find that

$$V_m^H A V_m = H_m,$$

where H_m is an upper Hessenberg matrix and $V_m = [\mathbf{v}_1, \ldots, \mathbf{v}_m]$ (see Section 4.3.4). We will use the eigenvalues of H_m as approximations to the eigenvalues of A. The NAG library contains routines to approximate the eigenvalues of an upper Hessenberg matrix. For example, the NAG routine F02APF calculates all the eigenvalues of a real upper Hessenberg matrix, whereas the routine F02ANF calculates all the eigenvalues of a complex upper Hessenberg matrix. H_m will be real for some of the examples in Section 5.4 and we will use F02APF, otherwise we use F02ANF. In this way we find m eigenvalue estimates, $\{\lambda_i\}_{i=1}^m$, for the matrix A.

5.3.2 Sector determination

Given some eigenvalue estimates $\{\lambda_i\}_{i=1}^m$, the next step is to place an annular sector around the eigenvalue estimates, and use the known, suitably scaled, Faber polynomials for this region as iteration polynomials.

The modulus and argument of each eigenvalue are found, with the arguments defined on $(-\pi, \pi]$. We set R_{min} to be the smallest eigenvalue modulus and R_{max} to be the largest eigenvalue modulus. We then place the eigenvalue arguments in increasing order, $\{\mu_i\}_{i=1}^m$, so that the smallest is first (μ_1) and the largest is last (μ_m), and look for the largest separation between adjacent arguments. That is we set x to be the largest of $2\pi - (\mu_m - \mu_1)$ and $\mu_{i+1} - \mu_i$ for $i = 1, \ldots, m-1$. The half-angle of the required sector is $\mu = (2\pi - x)/2$. Finally we must determine η, which as mentioned previously is the angle the bisecting ray of the sector makes with the negative real axis. If $x = \mu_{j+1} - \mu_j$ for some $j \in \{1, \ldots, m-1\}$ then

$$\eta \equiv [\pi - \mu_{j+1} - \mu](\mathrm{mod} 2\pi) \in [0, 2\pi),$$

otherwise if $x = 2\pi - (\mu_m - \mu_1)$ then

$$\eta \equiv [\pi - \mu_1 - \mu](\mathrm{mod} 2\pi) \in [0, 2\pi).$$

We have now found an annular sector of the form $\bar{Q}$ (see equation (5.2)) containing the eigenvalue estimates (see Figure (5.2)). As stated above, the Faber polynomials for this region are used as iteration polynomials. To determine these Faber polynomials we begin by finding the Faber polynomials for an annular sector $Q(R, 1, \theta)$, that is, the annular sector

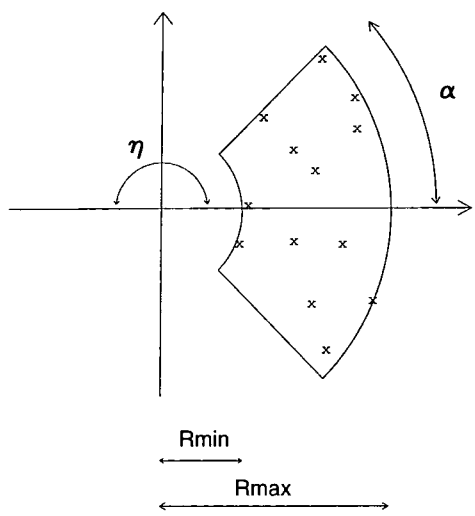

Figure 5.2: An annular sector placed around some eigenvalue estimates with the radii, the half–angle and the rotation from the negative real axis marked.

symmetric with respect to the negative real axis, having inner radius $R = R_{min}/R_{max}$, outer radius 1 and angle $\theta = \pi - \mu$. Referring back to Chapter 2 (specifically Section 2.4) we find that the Faber polynomials for $Q(R, 1, \theta)$ are determined by two parameters, a and b, and the transfinite diameter ρ. As we described in Section 3.2 the parameters, a and b, are found by modified Newton iteration combined with Kronrod-Patterson integration. The transfinite diameter may also be found by numerical integration (see Section 3.2). Equation (5.3) generates the Faber polynomials for $\bar{Q}$ from the Faber polynomials for $Q(R, 1, \theta)$. Therefore once a, b and ρ are determined for $Q(R, 1, \theta)$ we can easily obtain the desired Faber polynomials for $\bar{Q}$ from equation (5.3).

5.3.3 The iteration polynomial

As we have already mentioned in Chapter 4, many iterative methods for solving (4.1) can be written as

$$\mathbf{x}_m = \mathbf{x}_0 + q_{m-1}(A)\mathbf{r}_0,$$

where q_{m-1} is a polynomial of degree $m-1$ (see Sections 4.3 and 4.4). The error and residual of such methods are defined by equations (4.3) and (4.7), respectively. The equations are

$$\mathbf{e}_m = \mathbf{x} - \mathbf{x}_m, \quad \text{and} \quad \mathbf{r}_m = A\mathbf{e}_m = \mathbf{b} - A\mathbf{x}_m.$$

Then

$$\mathbf{r}_m = \mathbf{b} - A(\mathbf{x}_0 + q_{m-1}(A)\mathbf{r}_0) = (I - Aq_{m-1}(A))\mathbf{r}_0$$

and

$$\mathbf{e}_m = A^{-1}\mathbf{r}_m = (I - Aq_{m-1}(A))\mathbf{e}_0.$$

With the residual polynomial $p_m(z) = 1 - zq_{m-1}(z)$, so that $p_m(0) = 1$, these two equations can be written as $\mathbf{r}_m = p_m(A)\mathbf{r}_0$ and $\mathbf{e}_m = p_m(A)\mathbf{e}_0$ (see Section 4.3.1). Therefore, for any consistent pair of matrix and vector norms on $\mathbb{C}^N$

$$\|\mathbf{r}_m\| \leq \|p_m(A)\| \, \|\mathbf{r}_0\| \quad \text{and} \quad \|\mathbf{e}_m\| \leq \|p_m(A)\| \, \|\mathbf{e}_0\| \qquad (m \geq 1),$$

and the aim is to choose polynomials p_m, in the set

$$\Pi_m = \{ \text{ polynomials of degree } m \mid p_m(0) = 1\},$$

such that $\|p_m(A)\|$ is as small as possible. For our residual polynomial, we will choose

$$p_m(z) = \frac{F_m(z)}{F_m(0)},$$

where $F_m(z)$ is the Faber polynomial, of degree m, for the annular sector, $\bar{Q}$, determined in Section (5.3.2).

We have now described our choice of residual polynomials, but why do we think this will be a successful choice? Eiermann et al. [20] show that

$$\lim_{m\to\infty} \|p_m(A)\|^{\frac{1}{m}} = \lim_{m\to\infty} \left(\max_{\lambda\in\sigma(A)} |p_m(\lambda)| \right)^{\frac{1}{m}},$$

where $\sigma(A)$ is the spectrum of A, and $\|\ \|$ is any matrix norm. In other words the convergence of the iterative method asymptotically depends on the spectrum of A. The annular sector chosen in the previous section, contains the eigenvalue estimates, and therefore this annular sector may be considered as an approximation to the spectrum of A. In choosing a residual polynomial we wish to find a polynomial of degree m, with $p_m(0) = 1$, so that $|p_m(z)|$ is small on the annular sector.

For a general region, D, in the complex plane, Starke and Varga [76], amongst others, have shown that the Faber polynomials can be very useful for polynomial matrix iterations.

They show that the suitably normalised Faber polynomials, $F_m(z)/F_m(0)$, are near-best approximation to zero with respect to the maximum norm (see their theorem below). Indeed they show for large m that the maximum value of $F_m(z)/F_m(0)$ is no larger than $V\rho_m(0,D)/\pi$, where V is the total boundary rotation of D and $\rho_m(f,D)$ is the smallest possible error, over D, obtained when approximating a function f by a polynomial of degree m (see Section 1.2.3). When considering non-normal matrices, it is often found that the convergence of a polynomial iterative method does not depend only on the spectrum of A (compare the above asymptotic result), but on larger sets containing $\sigma(A)$, for example, the ϵ-pseudospectra of Trefethen ([78] and [68]). For this reason Starke and Varga [76] wanted their residual polynomials to be small on larger regions than D. They proved that the Faber polynomials are not only nearly optimal on D, but also that they are nearly optimal on level sets of D (see Section 3.5). In particular their Theorem 3.4, in our notation, states,

Let D be of bounded boundary rotation with $0 \notin D$, and let V denote the total boundary rotation of D. Then, the normalised Faber polynomials, associated with D, satisfy

$$\begin{aligned}\rho_m(0,D) &\leq \max_{z\in D}\left|\frac{F_m(z)}{F_m(0)}\right| \leq \frac{V/\pi}{|w^*|^m-(1+V/\pi)} \\ &\leq \frac{V/\pi}{1-\frac{(1+V/\pi)}{|w^*|^m}}\ \rho_m(0,D),\end{aligned}$$

for all $m > \log(1+V/\pi)/(\log|w^*|)$. Moreover, if $1 \leq r < |w^*|$, then

$$\rho_m(0,D_r) \leq \max_{z\in D_r}\left|\frac{F_m(z)}{F_m(0)}\right| \leq \frac{V(r)/\pi}{1-(1+V(r)/\pi)\left(\frac{r}{|w^*|}\right)^m}\ \rho_m(0,D_r),$$

is valid for all $m > \log(1+V(r)/\pi)/(\log|w^*|-\log r)$.

By $V(r)$ we mean the total rotation of a level curve Γ_r (see Section 3.3.3) and by D_r we mean the set bounded by Γ_r. The point in the w–plane that $\psi(w)$ maps to the origin in the z–plane is denoted by w^* (for an annular sector see equation (2.38)).

This theorem is in two parts. The first part shows that $F_m(z)/F_m(0)$ is near-best to zero. The second part shows that the maximum value of $F_m(z)/F_m(0)$ is also small on level sets of the region. This is an extremely useful property of Faber polynomials, especially when dealing with non-normal matrices. The main reasons for choosing the Faber polynomials as residual polynomials lie in the results of this theorem.

5.3.4 Implementation

We define a vector operation to be N scalar multiplications and N scalar additions, where N is the dimension of the linear system we are trying to solve. As in Nachtigal et al. [60] and Starke and Varga [76] we will use the number of vector operations as an indication of the speed of convergence of the method. If we let l denote the average number of non-zero elements per row in the matrix A, then a matrix-vector multiplication costs l vector operations.

Firstly we must consider the cost of Arnoldi's method (see Section 4.3.4), that is, how many vector operations it takes to obtain m_1 eigenvalue estimates for the matrix A. For the j-th step, $j = 1, \ldots, m_1$, of Arnoldi's method, calculating $\mathbf{w}_j = A\mathbf{v}_j$ involves l vector operations, calculating $h_{i,j} = \mathbf{v}_i^T \mathbf{w}_j$ for $i = 1, \ldots j$ involves j vector operations, computing $\hat{\mathbf{v}}_{j+1} = \mathbf{w}_j - \sum_{i=1}^{j} h_{i,j}\mathbf{v}_i$ requires j vector operations, computation of the norm requires one vector operation and finally the calculation of $\mathbf{v}_{j+1}$ requires about one vector operation. So in total the number of vector operations involved in Arnoldi's method is

$$\sum_{j=1}^{m_1}(l + j + j + 2) = m_1\,[l + 3 + m_1]\,.$$

Secondly we consider how many vector operations are involved in implementing a polynomial iterative method using a Faber polynomial of degree m. As in Starke and Varga [76] we will implement the iteration polynomial, $q_{m-1}(z) = (F_m(0) - F_m(z))/F_m(0)z = \alpha_{0,m} + \alpha_{1,m}z + \ldots + \alpha_{m-1,m}z^{m-1}$ in a Horner-type iteration. This requires $m(l+1)$ vector operations. The Horner iteration is of the form

$$\mathbf{w}_0 = \alpha_{m-1,m}\mathbf{r}_{\text{old}},$$

$$\mathbf{w}_j = A\mathbf{w}_{j-1} + \alpha_{m-1-j,m}\mathbf{r}_{\text{old}}, \qquad j = 1, \ldots, m-1,$$

$$\mathbf{x}_{\text{new}} = \mathbf{x}_{\text{old}} + \mathbf{w}_{m-1},$$

$$\mathbf{r}_{\text{new}} = \mathbf{b} - A\mathbf{x}_{\text{new}}.$$

In the examples, we neglect the work involved in choosing the particular sector of the annulus, as this should be negligible compared to the work involved in the iterations.

5.4 Examples

In this section we consider a number of examples of linear systems which we will solve by our method. In all these examples, except where stated, we will choose the degree of the residual polynomial to be the same as the number of steps taken in the Arnoldi algorithm.

Example 1

We consider discretising the boundary value problem

$$\begin{aligned} -\Delta u + \tau u_x &= f(x,y), \quad (x,y) \in S \\ u(x,y) &= g(x,y), \quad (x,y) \in \partial S, \end{aligned} \tag{5.4}$$

by central differences on the unit square $S := (0,1) \times (0,1)$ with boundary ∂S. This leads to solving a system of equations

$$A\mathbf{x} = (B \otimes I_n + I_n \otimes C)\mathbf{x} = \mathbf{b} \tag{5.5}$$

with $n = 1 + 1/h$,

$$B = \begin{pmatrix} 2 & -1 & & \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & 2 \end{pmatrix} \quad \text{and} \quad C = \begin{pmatrix} 2 & -1+\mu & & \\ -1-\mu & \ddots & \ddots & \\ & \ddots & \ddots & -1+\mu \\ & & -1-\mu & 2 \end{pmatrix},$$

and $\mu = \tau(h/2)$, where h is the meshsize of the discretisation.

The eigenvalues of a tridiagonal matrix,

$$\begin{pmatrix} a & b & & \\ c & a & b & \\ & \ddots & \ddots & \ddots \\ & & c & a \end{pmatrix} \in \mathbb{C}^{n \times n},$$

are given by $\lambda = a + 2\sqrt{bc}\cos(s\pi/(n+1))$, for $s = 1, 2, \ldots, n$, with corresponding eigenvectors,

$$\mathbf{u} = [u_j] = \left[\left(\frac{c}{b}\right)^{\frac{j}{2}} \sin\left(\frac{j\pi s}{n+1}\right) \right] \tag{5.6}$$

(see for example Smith [73]). In these formulae care must be taken when finding the square–root. If we write $c = |c|\exp(i\theta)$ and $b = |b|\exp(i\phi)$ then

$$\sqrt{\frac{c}{b}} = \sqrt{\left|\frac{c}{b}\right|}\exp\left[\frac{i(\theta-\phi)}{2}\right]. \tag{5.7}$$

Now $\mathbf{u} = [u_j]$, given by equation (5.6), is an eigenvector of the tridiagonal matrix if $cu_{j-1} + au_j + bu_{j+1} = \lambda u_j$. From equation (5.6) we find

$$cu_{j-1} + au_j + bu_{j+1} = c\left(\sqrt{\frac{c}{b}}\right)^{j-1}\sin\left[\frac{(j-1)\pi s}{n+1}\right] + au_j + b\left(\sqrt{\frac{c}{b}}\right)^{j+1}\sin\left[\frac{(j+1)\pi s}{n+1}\right].$$

Using equation (5.7) and some algebra gives

$$cu_{j-1} + au_j + bu_{j+1} = \left[a + 2c\sqrt{\left|\frac{b}{c}\right|}\exp\left[\frac{i(\phi-\theta)}{2}\right]\cos\left(\frac{s\pi}{n+1}\right)\right]u_j. \tag{5.8}$$

As stated above the eigenvalues of the tridiagonal matrix are $a + 2\sqrt{bc}\cos(s\pi/(n+1))$, with care taken over the square–root. For the matrix B, we have $a = 2$, and $b = c = -1$. Therefore equations (5.6), (5.7) and (5.8) give the eigenvalues of the matrix B as $\lambda = 2 - 2\cos(s\pi/(n+1))$ with corresponding eigenvector $\mathbf{u} = [u_j] = [\sin(j\pi s/(n+1))]$. The matrix C has $a = 2$, $b = -1+\mu$ and $-1-\mu$. With $\mu > 1$, the same equations similarly give the eigenvalues of C as $\lambda = 2+2i\sqrt{|1-\mu^2|}\cos(t\pi/(n+1)) = 2+2\sqrt{1-\mu^2}\cos(t\pi/(n+1))$, with corresponding eigenvector $\mathbf{v} = [v_l]$, where

$$v_l = \left(\frac{-1-\mu}{-1+\mu}\right)^{\frac{l}{2}}\sin\left(\frac{l\pi t}{n+1}\right).$$

The matrices $B \otimes I_n$ and $I_n \otimes C$ have a common set of eigenvectors, namely $\mathbf{w}$, whose entry in index $l + (j-1)n$ is

$$w_{l+(j-1)n} = v_l\, u_j = \left(\frac{-1-\mu}{-1+\mu}\right)^{\frac{l}{2}}\sin\left(\frac{l\pi t}{n+1}\right)\sin\left(\frac{j\pi s}{n+1}\right).$$

The eigenvalues of the matrix $B \otimes I_n + I_n \otimes C$ are therefore given by

$$4 - 2\cos\left(\frac{s\pi}{n+1}\right) + 2\sqrt{1-\mu^2}\cos\left(\frac{t\pi}{n+1}\right) \tag{5.9}$$

where $k = 1, 2, \ldots n$, $l = 1, 2, \ldots n$.

This is Example 2 in Starke and Varga [76]. They give this as an example of a linear system, not as a method for solving (5.4). By this we mean that they fix the mesh size for the discretisation, by setting $h = 1/31$, that is, $n = 32$. Starke and Varga then consider this linear system with different μ values, and therefore different τ. By varying the value of μ they are therefore considering the solution of different partial differential equations. We remark that to solve equation (5.4), by our method, one would fix τ and then discretise the system with different h values. Hence, the size of the linear system will vary with the different h values chosen. For this example, we will consider Starke and Varga's approach, that is fix $h = 1/31$. We do this for two reasons, firstly, like Starke and Varga we use this as an example of a linear system not as a means to solve equation (5.4), and secondly our approach will allow us to compare the results of Starke a nd Varga [76].

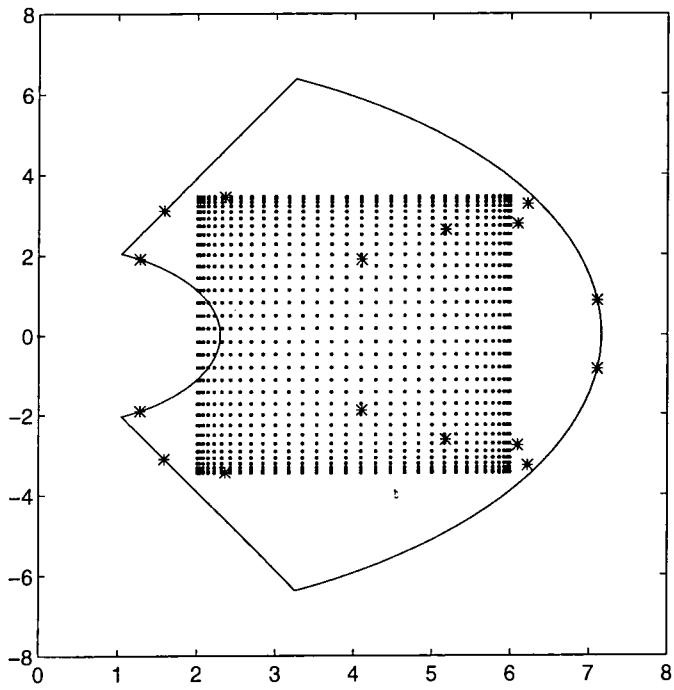

Figure 5.3: A plot of 16 eigenvalue estimates (stars), the actual eigenvalues (dots) and the annular sector for example 1, with random starting vectors and $\mu = 2$.

To allow this comparison with the results of Starke and Varga [76], we consider solving this linear system with a random righthand side $\mathbf{b}$ and a random initial solution $\mathbf{x}_0$. The random vectors were produced from the normal distribution, $N(0,1)$, using MATLAB. We run 16 steps of the Arnoldi process and therefore obtain 16 eigenvalue estimates for the

matrix $A = B \otimes I_n + I_n \otimes C$. We then place an annular sector around these estimates. In Figure (5.3) we give a plot of the chosen annular sector when $\mu = 2$, we also show the 16 eigenvalue estimates as stars and the actual eigenvalues (see Equation (5.9)) as dots. In Figure (5.5) we give a similar plot for $\mu = 4$.

Table 5.1: The parameters for example 1 with random starting vectors, $\mu = 2$, and $4, 8, 12$, and 16 eigenvalue estimates.

eigenvalue estimates	4	8	12	16
R_{max}	6.260696017	6.748047954	7.054200525	7.158911226
R_{min}	4.411842781	3.591748348	2.982998352	2.296100365
α	0.676716469	1.003314661	1.024454495	1.098733207
a	0.607634387	0.443468572	0.398907110	0.337930265
b	0.789967884	0.661702264	0.598031851	0.523064535
ρ	0.407267775	0.564861966	0.583701576	0.617875205

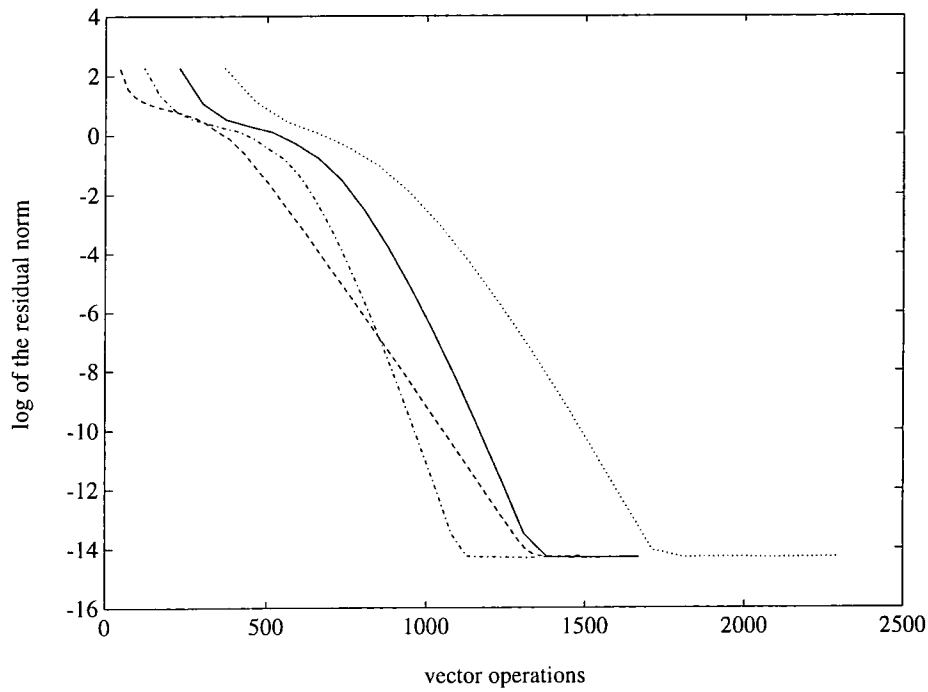

Figure 5.4: Convergence curves for example 1 with random starting vectors, $\mu = 2$, and $4, 8, 12$ and 16 eigenvalue estimates.

In the next stage of our algorithm we use the Faber polynomial of degree 16, for this

annular sector, as the residual polynomial. We need an initial solution to start this stage of the algorithm. It would be possible to use the links between the GMRES algorithm and Arnoldi's algorithm (see Section 4.3.4) to produce an improved initial solution. However, in this example, we chose to use the same initial solution $\mathbf{x}_0$ as we did in the first stage of our algorithm. In Figure (5.4) we plot, for $\mu = 2$, the log of the Euclidean norm of the residual, $\|\mathbf{r}_k\|_2$, versus the number of vector operations. The dotted line represents our method using 16 eigenvalue estimates. In this figure we also show the convergence curves for the method with 4 (dashed line), 8 (dash-dot line) and 12 (solid line) eigenvalue estimates. We note that with 8 eigenvalue estimates the method converges to 10^{-14} in the least number of vector operations, whereas with 16 eigenvalue estimates the method converges, again to 10^{-14}, in the greatest number of vector operations. The parameters $a, b, \rho, R_{max}, R_{min}$, and μ required in the second stage of the algorithm are given in Table 5.1. For this example the eigenvalues and their estimates are symmetric with respect to the positive real axis and therefore $\eta = \pi$.

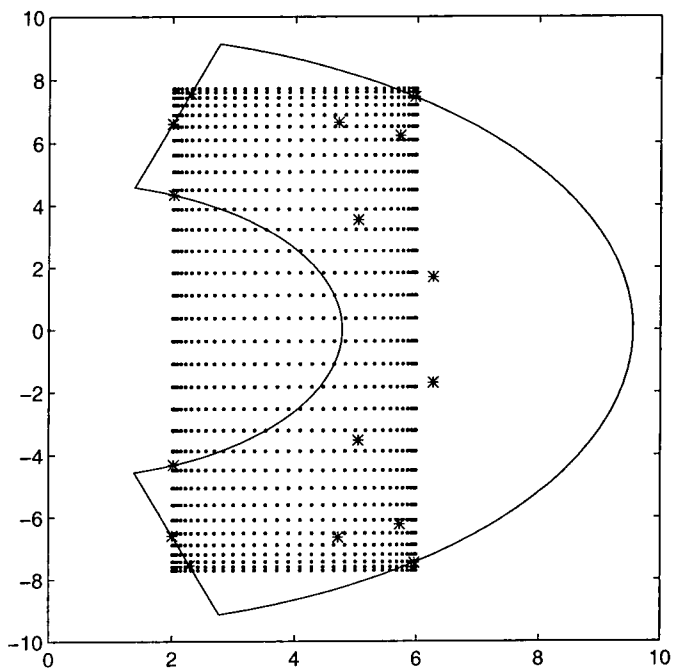

Figure 5.5: A plot of 16 eigenvalue estimates (stars), the actual eigenvalues (dots) and the annular sector for example 1, with random starting vectors and $\mu = 4$.

A similar convergence curve is shown in Figure (5.6) this time for $\mu = 4$. Once again the

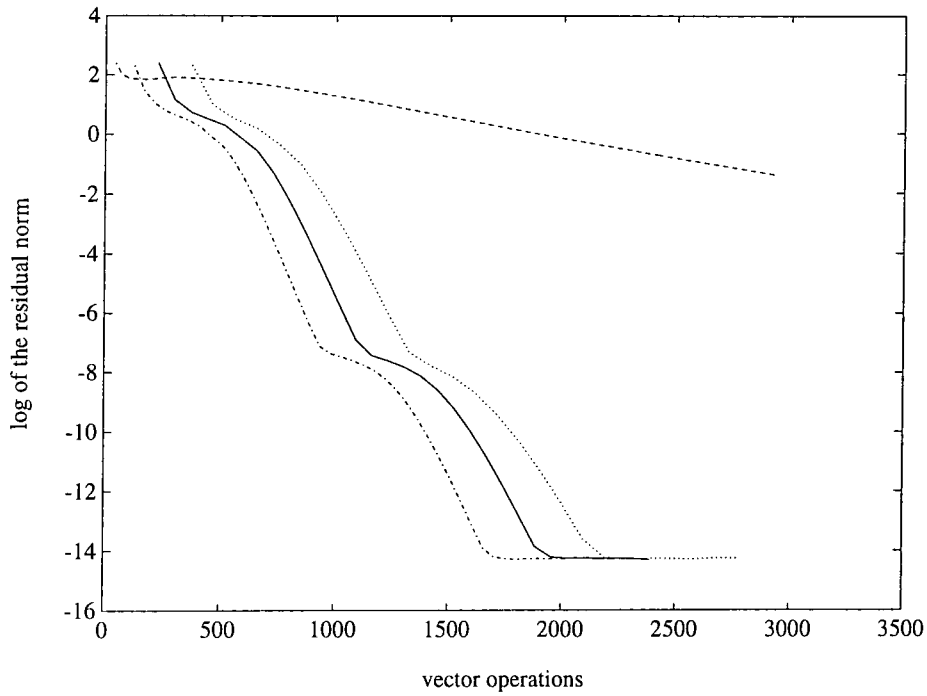

Figure 5.6: Convergence curves for example 1 with random starting vectors, $\mu = 4$, and $4, 8, 12$ and 16 eigenvalue estimates.

method with 16 eigenvalue estimates is represented by dots, and the methods with 12, 8 and 4 eigenvalue estimates are represented by solid, dash-dot and dashed lines, respectively. This time the method with 4 eigenvalue estimates converges very slowly compared to the others. This is because not enough information about A is obtained with only 4 eigenvalue estimates. The convergence curves with 8, 12 and 16 eigenvalue estimates all contain regions where the convergence slows for a short time and then accelerates again. Observing Figure (5.5) we see that even with 16 eigenvalue estimates the annular sector only contains part of the spectrum. If we consider 32 eigenvalue estimates then the annular sector contains much more of the spectrum and the convergence curve is smoother. However, because we have taken more eigenvalue estimates, the method takes more vector operations to converge to 10^{-14} than the method with 16 eigenvalue estimates. The parameters used in the methods, whose convergence curves are contained in Figure (5.6), are given in Table 5.2 Once again $\eta = \pi$.

We now compare the results we have just given with those of Starke and Varga [76].

Table 5.2: The parameters for example 1 with random starting vectors, $\mu = 4$, and $4, 8, 12$, and 16 eigenvalue estimates.

eigenvalue estimates	4	8	12	16
R_{max}	7.699739638	9.081252758	9.348770592	9.548432785
R_{min}	7.107633830	6.422866965	6.127775211	4.778122352
μ	1.046397859	1.234935314	1.269103190	1.276539785
a	0.549738053	0.427390580	0.402484467	0.353756280
b	0.881078425	0.726933361	0.693535419	0.604823601
ρ	0.522259695	0.631252220	0.649191755	0.665315835

A summary of these results when $\mu = 2$ and $\mu = 4$ is contained in Table 5.3. The table contains results from five different methods; Arnoldi/Faber, which is the method of Starke and Varga; GMRES, the generalised minimum residual method restarted after so many steps; Hybrid GMRES, the method of Nachtigal et al. [60]; Arnoldi/Chebyshev, the method of Elman et al. [30]; and finally our method. When using the Hybrid GMRES algorithm,

Table 5.3: The approximate number of vector operations to converge to the specified tolerance, for example 1, with random starting vectors.

Method	$\mu = 2$	$\mu = 4$
Arnoldi/Faber(16)	$1800(10^{-13})$	$2100(10^{-13})$
GMRES(16)	$4200(10^{-10})$	$4800(10^{-10})$
Hybrid GMRES(16)	$1800(10^{-13})$	$2200(10^{-13})$
Arnoldi/Chebyshev(16)	$2300(10^{-13})$	$2600(10^{-13})$
Our Method(16)	$1700(10^{-13})$	$2100(10^{-13})$

Starke and Varga fix the number of steps used in the GMRES part of the algorithm. They do this to allow comparison between the methods. The actual Hybrid GMRES algorithm may cycle back to the GMRES stage and carry out more steps if necessary. In the table we show the approximate number of vector operations required so the log of the residual norm is less than the tolerance given in brackets. After the method name there also follows a

number in brackets. This number indicates the number of steps, of the GMRES or Arnoldi algorithm, that is taken before either restarting, in the case of GMRES, or switching to stage 2, for all other methods. Starke and Varga also considered solving this system with CGS (see Section (4.3.6)) and CGNR (see Section 4.3.3) but they report that these methods showed no signs of convergence. The starting vectors used in these examples were random; therefore the results only show that our method is comparable to both Arnoldi/Faber and Hybrid GMRES.

As well as these examples with random vectors, Starke and Varga [76] consider the solution of this linear system with a more structured $\mathbf{b}$ and $\mathbf{x}_0$. They give two such examples and these will allow a better comparison between the methods. We begin by considering $\mathbf{b} = (-1, 1, \ldots, -1, 1)^T$ and $\mathbf{x}_0 = \mathbf{0}$, with $\mu = 2$. In Figure (5.7) we show the eigenvalue estimates obtained after 16 steps of the Arnoldi algorithm. For these starting vectors, the convergence curves for 4, 8, 12 and 16 eigenvalue estimates are shown in Figure (5.8). In Table 5.4 we give the parameters used by these methods.

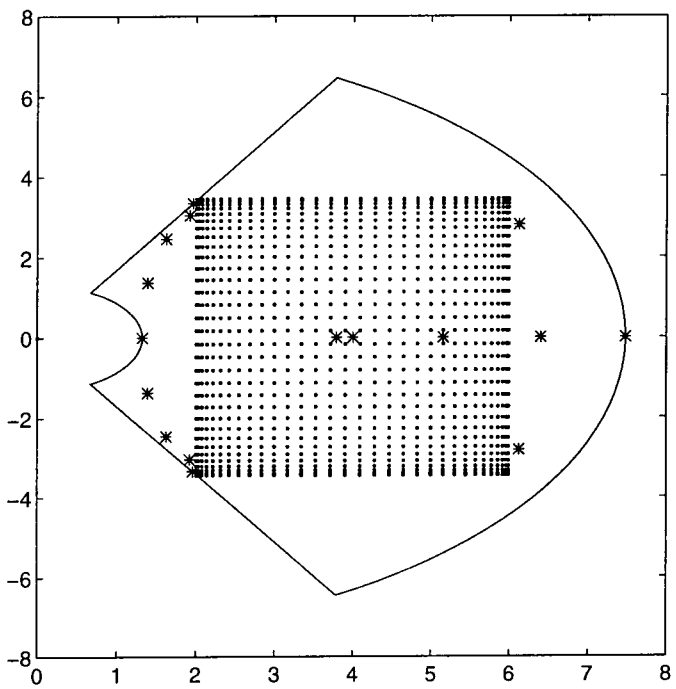

Figure 5.7: A plot of 16 eigenvalue estimates (stars), the actual eigenvalues (dots) and the annular sector for example 1, with $\mathbf{x}_0 = \mathbf{0}, \mathbf{b} = (-1, 1, \ldots, -1, 1)^T$, and $\mu = 2$.

Table 5.4: The parameters for example 1, with $\mathbf{x}_0 = \mathbf{0}, \mathbf{b} = (-1, 1, \ldots, -1, 1)^T$, $\mu = 2$.

eigenvalue estimates	4	8	12	16
R_{max}	5.043308028	7.191601849	7.394821043	7.480913849
R_{min}	2.679906795	1.393162948	1.349902097	1.320964635
μ	0.861156275	0.983294864	1.019946059	1.040038219
a	0.487544645	0.306566305	0.290541970	0.281892977
b	0.681063116	0.447513112	0.431339831	0.422442742
ρ	0.510474705	0.587113047	0.600067798	0.607065941

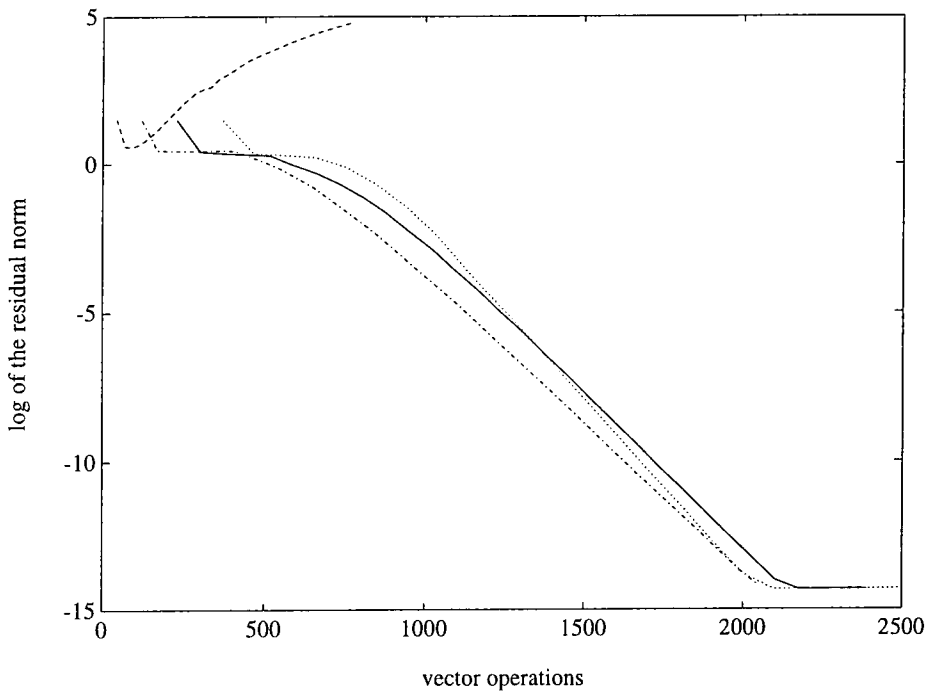

Figure 5.8: Convergence curves for example 1 with $\mathbf{x}_0 = \mathbf{0}, \mathbf{b} = (-1, 1, \ldots, -1, 1)^T$, $\mu = 2$, and $4, 8, 12$ and 16 eigenvalue estimates.

In Table 5.5 we summarise Starke and Varga's results for this example and also include the results for our method. In this case our method converges to 10^{-13} slightly faster than the Arnoldi/Faber method, and both methods converge faster than Hybrid GMRES.

Table 5.5: The approximate number of vector operations to converge to the specified tolerance, for example 1, with structured starting vectors and $\mu = 2$.

Method	$\mathbf{b} = (-1, 1, \ldots, -1, 1)^T$ $\mathbf{x}_0 = \mathbf{0}$	Method	$\mathbf{b} = (1, 1, \ldots, 1)^T$ $\mathbf{x}_0 = \mathbf{0}$
Arnoldi/Faber(16)	$2400(10^{-13})$	Arnoldi/Faber(32)	$5200(10^{-11})$
GMRES(16)	$5000(10^{-7})$	GMRES(32)	$9800(10^{-11})$
Hybrid GMRES(16)	$4800(10^{-8})$	Hybrid GMRES(40)	$7200(10^{-11})$
Our Method(16)	$2000(10^{-13})$	Our Method(32)	$3900(10^{-13})$

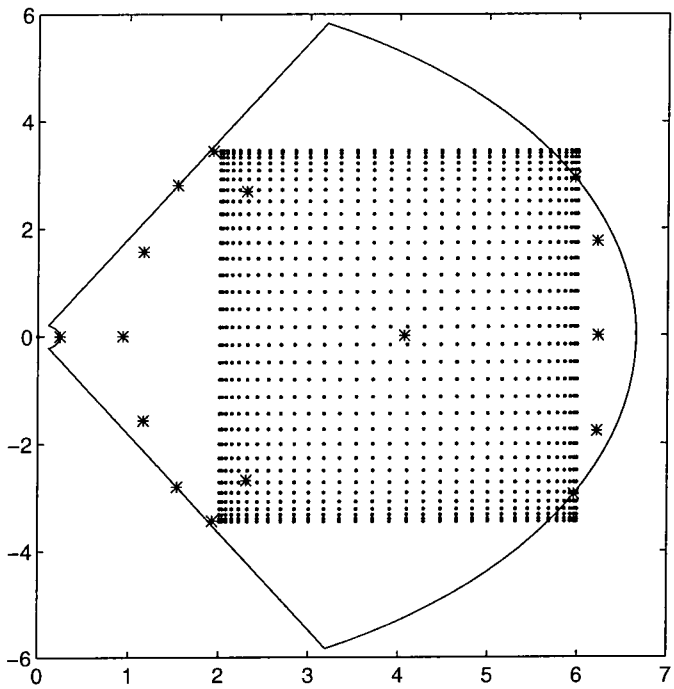

Figure 5.9: A plot of 16 eigenvalue estimates (stars), the actual eigenvalues (dots) and the annular sector for example 1, with $\mathbf{x}_0 = \mathbf{0}, \mathbf{b} = (1, 1, \ldots, 1)^T$, and $\mu = 2$.

Finally we consider solving (5.5) when $\mathbf{x}_0 = 0$, $\mathbf{b} = (1, 1, \ldots, 1)^T$ and $\mu = 2$. In Figure (5.9) we show the eigenvalue estimates obtained by the Arnoldi stage of our algorithm after 16 steps. The annular sector, placed around these 16 estimates, encloses more of the actual eigenvalues than the polygonal region of Starke and Varga [76], but our method still diverges. Taking 24 eigenvalue estimates gives convergence. When solving a linear system

by our method it is therefore important that enough steps of the Arnoldi algorithm are taken. If however, the method seemed to be diverging it would be possible to go back to the first stage and find some more estimates.

Table 5.6: The parameters for example 1, with $\mathbf{x}_0 = \mathbf{0}, \mathbf{b} = (1, 1, \ldots, 1)^T$, and $\mu = 2$.

eigenvalue estimates	16	24	32	40
R_{max}	6.648490985	6.890615844	7.068740047	7.143223527
R_{min}	0.241695673	0.459154090	0.478197598	0.663579562
μ	1.070128097	1.165656109	1.251215932	1.254768505
a	0.150293406	0.170184712	0.154893034	0.176054293
b	0.227984931	0.270714185	0.257530180	0.293355872
ρ	0.622281793	0.651516262	0.677754714	0.678324943

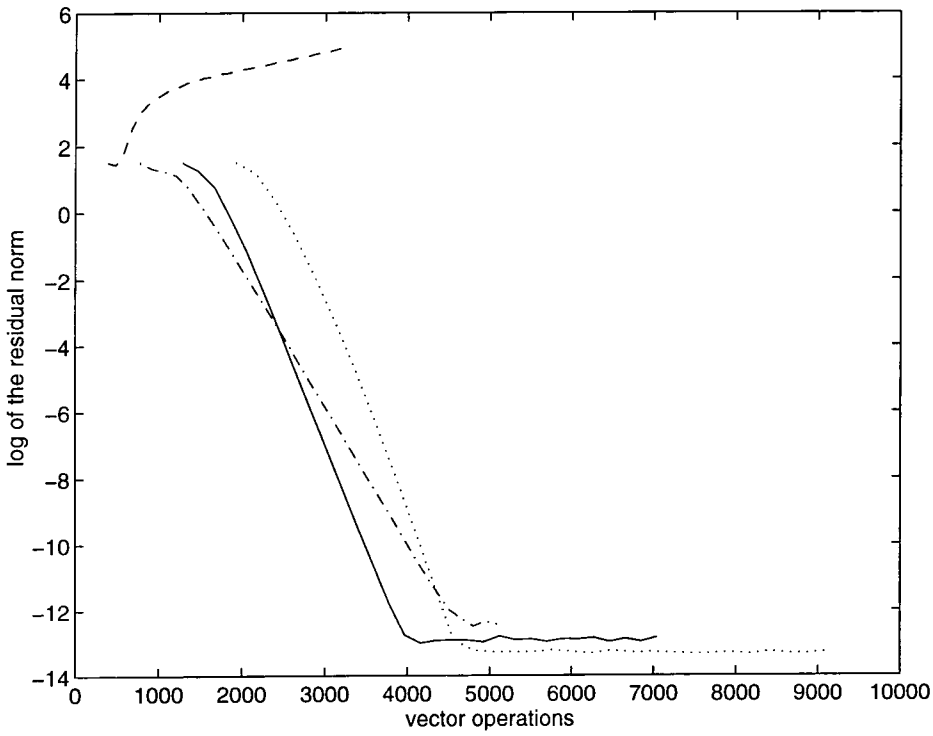

Figure 5.10: Convergence curves for example 1 with $\mathbf{x}_0 = \mathbf{0}, \mathbf{b} = (1, 1, \ldots, 1)^T$, $\mu = 2$, and $16, 24, 32$ and 40 eigenvalue estimates.

In Figure (5.10) we show a plot of the log of the residual norm against the number of vector operations using our method with $16, 24, 32$ and 40 eigenvalue estimates. In Table 5.6

we show the parameters used in these methods and in Table 5.5 we show a comparison between our results and those of Starke and Varga [76]. We see from Table 5.5 that our method, with 32 eigenvalue estimates, is superior to Arnoldi/Faber with the same number of estimates. It is interesting to note that Arnoldi/Faber(40) converges to 10^{-11} faster than Arnoldi/Faber(32), whereas our method with 32 eigenvalue estimates converges to 10^{-13} faster than our method with 40 eigenvalue estimates. In fact considering a tolerance of 10^{-11}, our method with 32 estimates is the fastest to converge, then Arnoldi/Faber(40), then our method with 40 estimates, and then Arnoldi/Faber(32). All these methods have superior convergence to Hybrid GMRES and GMRES. In fact Hybrid GMRES(32) actually diverges.

Example 2

The Grcar matrix example. In this example we consider the matrix

$$A = \begin{pmatrix} \frac{9}{10} & 1 & 1 & 1 & & & \\ -1 & \frac{9}{10} & 1 & 1 & 1 & & \\ & -1 & \frac{9}{10} & 1 & 1 & 1 & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots \end{pmatrix} \in \mathbb{R}^{1024\times 1024}.$$

This is Example 6.3 in Starke and Varga's paper, it is a shifted version of a matrix in a paper by Trefethen [78], which originated in a paper by J. Grcar [45]. Starke and Varga [76] chose it to illustrate that their method would even work if some of the spectrum of the matrix was situated in the left–half plane. Starke and Varga also point out that it is surprising their method would even work at all because some of the spectrum of the matrix is not included in their polygonal region.

We use this example for two reasons, firstly, we can once again compare our results with those of Starke and Varga and secondly this is an example whose convex hull of the spectrum contains the origin and therefore the Arnoldi/Chebyshev algorithm [30] (see Section 4.4) will fail to converge. For this example, we solve the linear system with random vectors for the initial approximation and righthand side.

In Figure (5.11) we plot the log of the residual norm against the number of vector operations. In the graph we show three convergence curves, the solid line represents our method taking 24 eigenvalue estimates, the dashed line 48 eigenvalue estimates and the

dash-dot line 32 eigenvalue estimates. With 32 eigenvalue estimates, our method has a residual norm less than 10^{-14} after about 8800 vector operations. We also observe that it converges to this tolerance faster than the methods with both 24 and 48 eigenvalue estimates.

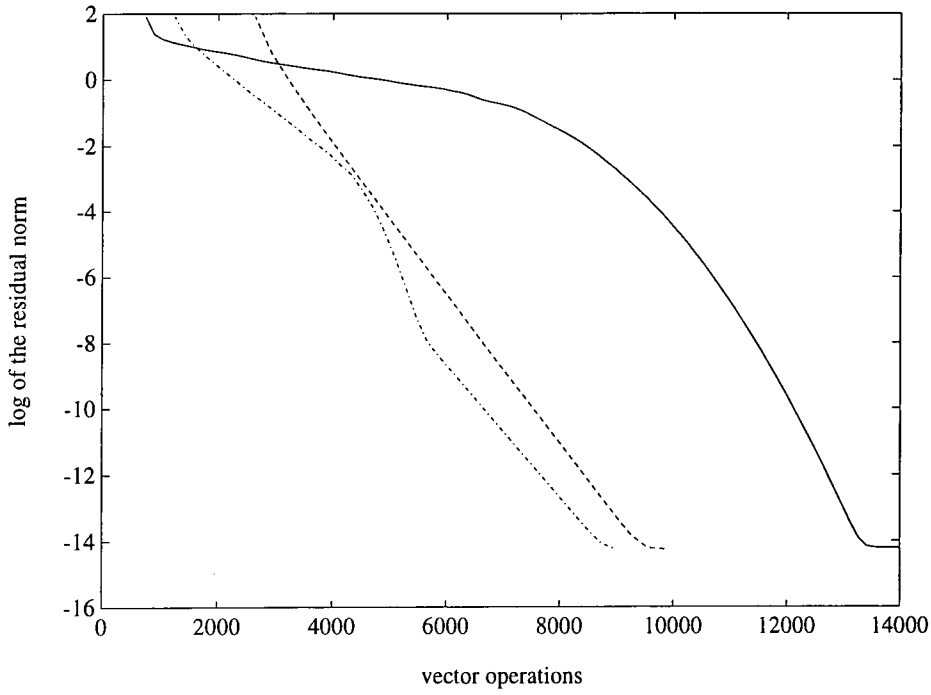

Figure 5.11: Example 2, convergence curves with random starting vectors, and 24, 32, and 48 eigenvalue estimates.

Table 5.7: The parameters for example 2, the Grcar matrix, with random starting vectors.

eigenvalue estimates	24	32	48
R_{max}	2.993374182	3.039322633	3.115282305
R_{min}	1.812674158	1.507586450	1.422573506
μ	1.771079535	1.869182867	1.859730317
a	0.252591979	0.200454514	0.192404532
b	0.591032839	0.500442075	0.475803666
ρ	0.810665346	0.839748298	0.838368199

In Table 5.7 we give the parameters we calculated for these methods, once again $\eta = \pi$. In Table 5.8 we compare our results with those of Starke and Varga [76]. For this example, it appears that the method of Nachtigal et al. [60] is the best. Considering 48 eigenvalue estimates our method is comparable with Starke and Varga's, but both are beaten by Hybrid GMRES. It is interesting that our method with 24 eigenvalue estimates converges whereas Arnoldi/Faber(24) diverges. Our method, however, is slow to converge compared to Hybrid GMRES(24).

Table 5.8: The approximate number of vector operations to converge to the specified tolerance, for example 2, with random starting vectors.

Method	Number of Vector Operations (tolerance)
Our Method(24)	$13500(10^{-14})$
Our Method(32)	$8800(10^{-14})$
Our Method(48)	$9200(10^{-14})$
Arnoldi/Faber(24)	diverges
Arnoldi/Faber(48)	$9200(10^{-14})$
Hybrid GMRES(24)	$9200(10^{-14})$
Hybrid GMRES(48)	$8000(10^{-14})$
GMRES(24)	$15000(10^{-8})$
GMRES(48)	$15000(10^{-4})$

Example 3

The coefficient matrices in the next two examples are shifted matrices from a paper by Reichel and Trefethen [68]. In the first example we consider the matrix

$$A = \begin{pmatrix} 4 & 0 & 1 & .7 & & \\ 2i & 4 & 0 & 1 & .7 & \\ & 2i & 4 & 0 & 1 & \ddots \\ & & 2i & 4 & 0 & \ddots \\ & & & 2i & 4 & \ddots \\ & & & & \ddots & \ddots \end{pmatrix} \in \mathbb{R}^{1024\times 1024},$$

which is Example 3 in a paper by Gutknecht [47]. We choose this particular example to

illustrate that our method works well when the eigenvalue estimates are not symmetric with respect to the positive (or negative) real axis. Although Starke and Varga [76] note that their method could be applied to a complex matrix, they do not give the details. In fact their method for placing a polygonal region around the eigenvalue estimates must change if the estimates are not symmetric with respect to the positive (or negative) real axis. For our method, there is little difference in placing an annular sector around eigenvalue estimates symmetric with respect to the positive (or negative) real axis, and those that are not (see Section 5.2). Therefore, for a complex matrix our method should be easier to implement than Starke and Varga's. Once again we consider solving the linear system with $\mathbf{b}$ and $\mathbf{x}_0$ given by random vectors.

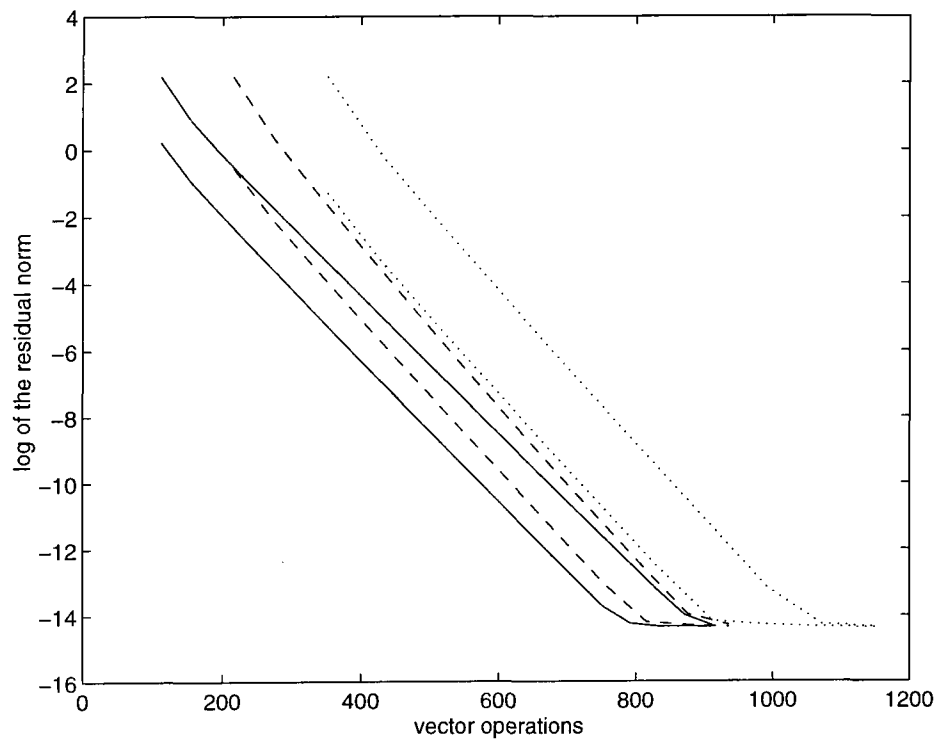

Figure 5.12: Example 3, convergence curves for our method with 8 (solid), 12 (dashed) and 16 (dotted) eigenvalue estimates. The lower curves are the same method, but with a new initial iterate for the second stage in our algorithm.

For the previous two examples the link between the Arnoldi and GMRES algorithms was not taken into account. We ignored the link because producing a new initial iterate, by the GMRES algorithm, did not vastly alter the number of vector operations required

to converge to a specific tolerance. The situation is different for both this and the next example. Figure (5.12) shows a plot of the log of the residual norm against the number of vector operations for our method with 8 (solid), 12 (dashed), and 16 (dotted) eigenvalue estimates. In each case two curves are plotted; one employing the GMRES algorithm to produce a new initial solution; and one using the initial solution from the first stage of our algorithm. The lower curves correspond to the new initial solution being used for the second stage of our algorithm. When 16 steps of the Arnoldi algorithm are taken we see that using the new initial iterate can speed up the convergence by almost 200 vector operations. In Figure (5.13) we show the convergence curves for CGNR, CGS and GMRES(16). For GMRES(16) we only plot the log of the residual norm obtained before each restart. We note that CGS stagnates, whereas, CGNR converges, but slower than our method with 16 eigenvalue estimates. GMRES(16) also converges, but in about twice as many vector operations as our method with 16 eigenvalue estimates. In Table 5.9 we give the parameters required by our method.

In all the other examples we consider we will now exploit the link between the Arnoldi and GMRES algorithms to produce a new initial iterate for the second stage of our algorithm.

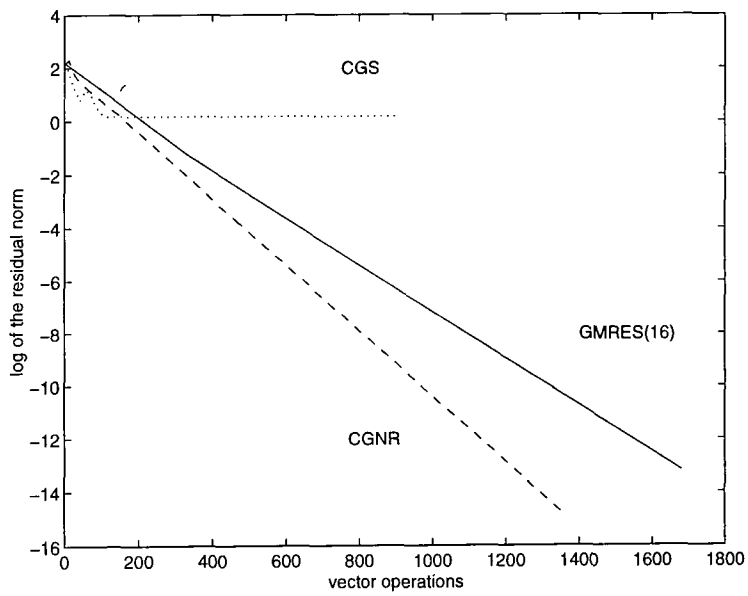

Figure 5.13: Example 3, the convergence curves for CGNR, CGS and GMRES(16).

Table 5.9: The parameters for example 3, with random starting vectors.

eigenvalue estimates	8	12	16
R_{max}	6.318322543	6.372127226	6.360749188
R_{min}	1.065257802	1.008595806	0.984856615
μ	0.436699747	0.479291832	0.497019179
η	3.069432036	3.054225337	3.066771724
a	0.449478135	0.427205882	0.418712433
b	0.522741301	0.504802220	0.498089123
ρ	0.394321512	0.411610653	0.418604483

Example 4

In this example we consider another shifted Reichel and Trefethen [68] matrix which was also considered by Gutknecht [47]. For this example the coefficient matrix is given by

$$A = \begin{pmatrix} 2 & 1 & & & \\ 0 & 2 & 1 & & \\ 1 & 0 & 2 & 1 & \\ & 1 & 0 & 2 & \ddots \\ & & \ddots & \ddots & \ddots \end{pmatrix} \in \mathbb{R}^{1024\times 1024},$$

and we consider random $\mathbf{b}$ and $\mathbf{x}_0$.

Table 5.10: The parameters for example 4, with random starting vectors.

eigenvalue estimates	8	12	16
R_{max}	3.831448773	3.833247733	3.889711330
R_{min}	1.817542827	1.788237848	1.674694109
μ	0.985672968	1.000343712	0.998750492
a	0.428990716	0.421809306	0.409286367
b	0.632892801	0.626389581	0.606359549
ρ	0.564506685	0.570726575	0.573686080

Figures (5.14) and (5.15) show similar plots to those given in Figures (5.12) and (5.13). Once again employing the link between the Arnoldi and GMRES algorithms reduces the

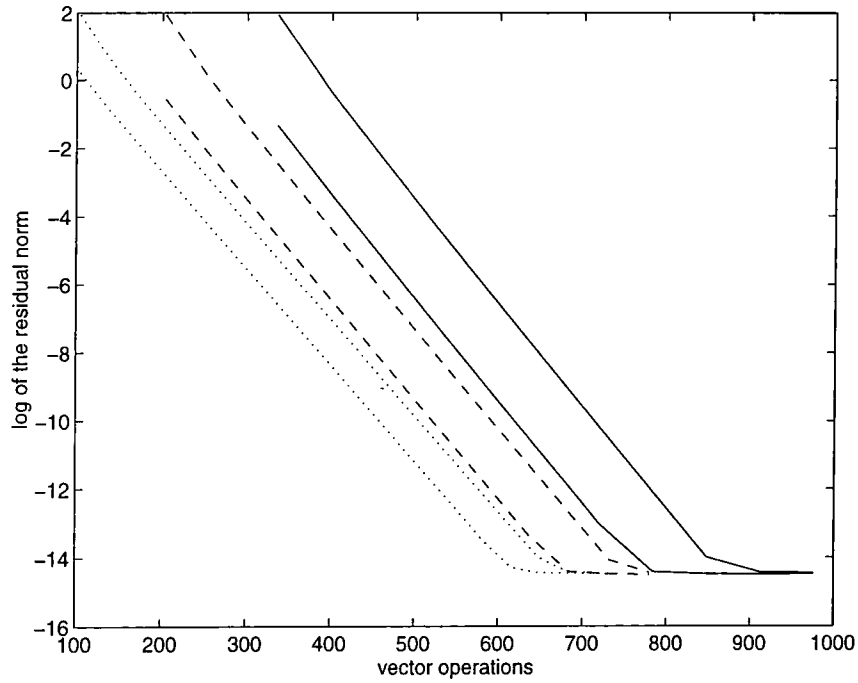

Figure 5.14: Example 4, convergence curves for our method with 8 (dotted), 12 (dashed) and 16 (solid) eigenvalue estimates. The lower curves are the same method, but with a new initial iterate for the second stage in our algorithm.

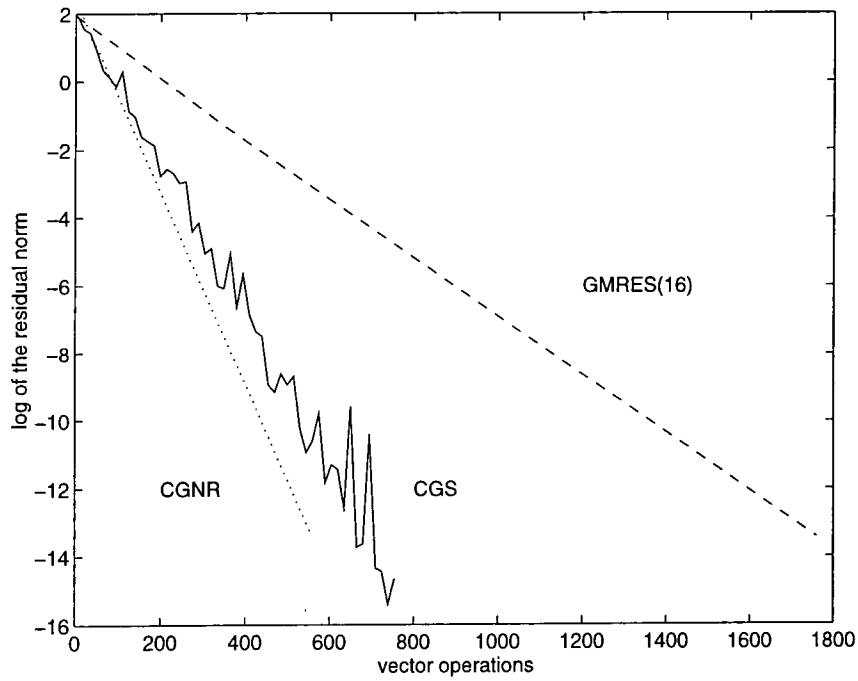

Figure 5.15: Example 4, the convergence curves for CGNR, CGS and GMRES(16).

number of vector operations our method takes to converge to a specific tolerance. For this example, CGNR converges to 10^{-14} in the least number of vector operations. Our method with 16 steps of the Arnoldi/GMRES algorithm is about comparable with CGS, though the convergence curve is much smoother. We find that 8 steps of the Arnoldi/GMRES algorithm is sufficient to obtain a reasonable approximation to the effective spectrum of the matrix. Therefore, because of the linear growth in computer time of the GMRES algorithm, our method with 8 eigenvalue estimates is the best of those exhibited in Figure (5.14). In fact it is almost comparable to CGNR. In Table 5.10 we once again give the required parameters for our method; $\eta = \pi$.

Example 5

In at least two papers in the literature, namely Li [53] and Smolarski and Saylor [74], a matrix is devised that has spectrum contained in a "boomerang"–shaped region in the complex plane. An annular sector is a natural region to enclose this type of domain. Therefore our method may be useful in solving a linear system with a coefficient matrix of this type.

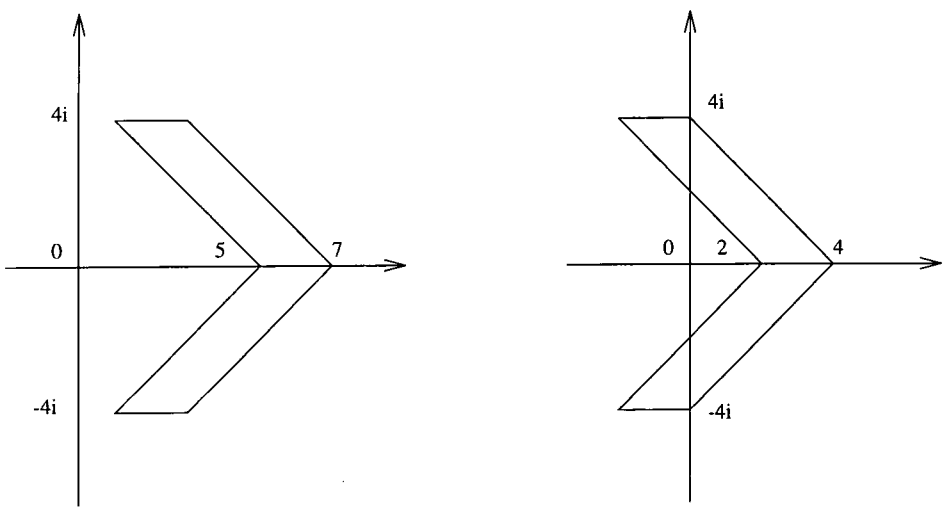

Figure 5.16: The left region contains the eigenvalues of T, whereas the right region contains the eigenvalues of $T - 3I$.

In the left-half of Figure (5.16) we show a plot of the aforementioned "boomerang"–shape. We consider a 1024×1024 matrix, T, with eigenvalues in this region. Following Smolarski and Saylor [74] we place four eigenvalues at the points $(1, 4)$, $(1, -4)$, $(5, 0)$ and

$(6,0)$. The other 1020 are spread uniformly along the lines from $(2,4)$ and $(2,-4)$ to $(6,0)$ and the lines from $(3,4)$ and $(3,-4)$ to $(7,0)$. That is each of these four lines contains 255 eigenvalues.

For example, considering the line from $(2,4)$ to $(6,0)$ the real parts of the eigenvalues are given by

$$x_j = 2 + \frac{j}{64} \qquad \text{for } j = 1, \ldots, 255,$$

with corresponding imaginary part

$$y_j = 6 - x_j.$$

The eigenvalues, with the exception of the two real ones, occur in complex conjugate pairs. We form the real tridiagonal matrix, T, from 2×2 block submatrices, the eigenvalues of which yield one of the conjugate pairs. If the complex conjugate pair is $x + iy$ and $x - iy$ then the 2×2 submatrix is chosen to be

$$\begin{pmatrix} x & y \\ -y & x \end{pmatrix}. \tag{5.10}$$

For the real eigenvalues we consider a 1×1 block containing the eigenvalue itself.

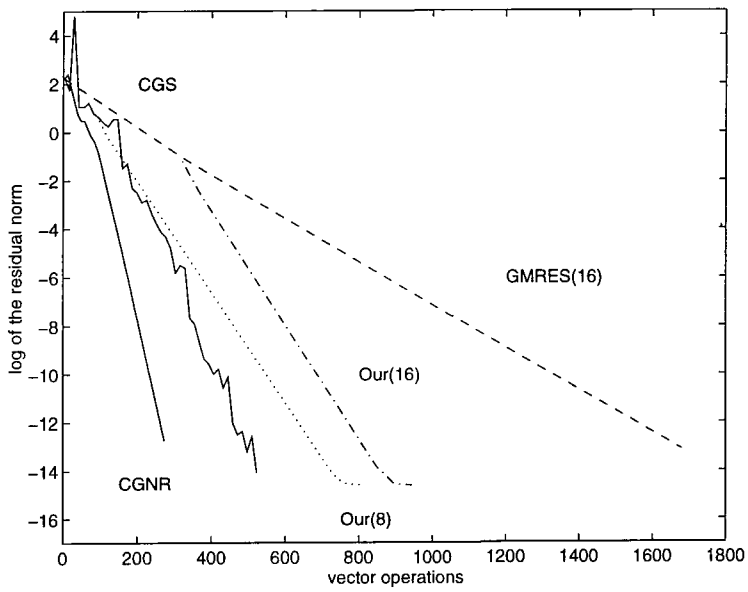

Figure 5.17: Convergence curves for various methods and coefficient matrix T.

In Figure (5.17) we show the convergence curves for various methods starting with the same random $\mathbf{b}$ and $\mathbf{x}_0$. For this matrix, CGNR converges, to a tolerance of 10^{-14}, in the

least number of vector operations. Even with 8 eigenvalue estimates our method is slower to converge to 10^{-14} than CGS.

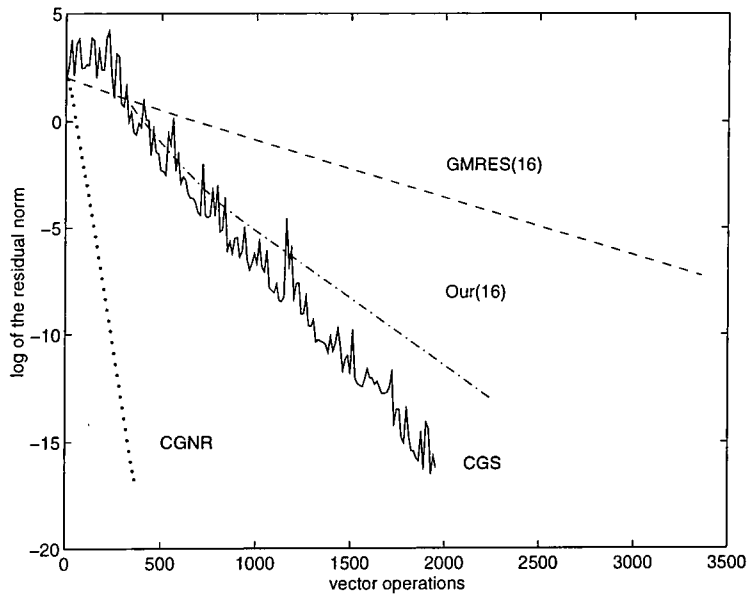

Figure 5.18: Convergence curves for various methods and coefficient matrix $T - 3I$.

Out of interest we perturb this example so that, like example 2, the convex hull of the spectrum contains the origin. That is we consider the matrix $T - 3I$. In the right-half of Figure (5.12) we show a region containing the eigenvalues of $T - 3I$. As we have mentioned this region encloses the origin and therefore the Arnoldi/Chebyshev method will fail to converge. Figure (5.18) shows the convergence curves for various methods, namely CGNR, CGS, our method with 16 eigenvalue estimates, and GMRES(16). With 8 or 12 eigenvalue estimates our method shows no signs of convergence, but with 16 eigenvalue estimates our method converges to 10^{-14} in about 2300 vector operations. As Starke and Varga pointed out for their method with example 2, this highlights the need to produce enough information in the first stage of a hybrid method. In our case this means carrying out enough steps of the Arnoldi algorithm. With this matrix CGS does slightly better than our method with 16 eigenvalue estimates, but once again CGNR outperforms all the other methods. The convergence of CGNR depends on the singular values of the coefficient matrix, in this case T or $T - 3I$, (see Nachtigal et al. [60]). For both of these examples the matrix is well conditioned, in fact $K_2(T) = 1.69$ and $K_2(T - 3I) = 2.24$, to 2 decimal

places, where $K_2(X)$ is the 2-norm condition number of the matrix X. Therefore, CGNR has fast convergence for these matrices. We also note that T^HT and $(T-3I)^H(T-3I)$ are diagonal. If we consider forming another matrix, T_1, from submatrices of the form

$$\begin{pmatrix} x & y/2 \\ -2y & x \end{pmatrix}, \tag{5.11}$$

then T_1 and $T_1 - 3I$ are not as well conditioned. To 2 decimal places $K_2(T_1) = 4.20$ and $K_2(T_1 - 3I) = 7.04$. Therefore, CGNR will not converge as fast as it did with submatrices given by (5.10). In fact, considering $T_1 - 3I$, CGNR has a residual norm less than 10^{-15} after about 1100 vector operations. On the other hand, CGS and our method with 16 eigenvalue estimates have convergence curves very similar to those in Figure (5.18). CGNR is still the best in this case, but not by such a large margin.

Example 6

In examples 2, 4 and 5 we have CGNR converging faster than all the other given methods. We have given two examples where our hybrid method, and others, can converge faster than CGNR. To find another such matrix, we consider the examples given in Nachtigal et al. [59]. This paper gives 6 examples, for which one of CGS, CGNR or GMRES does best and one does worst, that is, in terms of the number of iterations to converge to a specific tolerance. The example they give for which CGS "wins" and CGN "loses" is a diagonal matrix,

$$D = \text{diag}(x_1, \ldots, x_N),$$

where

$$x_j = 1 + \frac{1}{2}(K-1)\left(1 + \cos\frac{(j-1)\pi}{N-1}\right) \qquad 1 \le j \le N.$$

For the example here we choose $N = 1024$ and $K = 20$.

Figure (5.19) shows convergence plots for various methods, where we have solved the linear system with the same random $\mathbf{b}$ and $\mathbf{x}_0$. The matrix, D, is real and diagonal and the Arnoldi algorithm produces eigenvalue estimates that are also real. This causes a problem in our numerical evaluation of the parameters a and b (see Section 3.2), because when the half-angle is zero (or alternatively $\theta = \pi$) we do not have starting values for our modified Newton iteration. As we saw in Section 2.2.4, when $\theta = \pi$ we have

$a = b = \sqrt[4]{R}$, $\rho = (1 - R)/4$ and the Faber polynomials are the suitably scaled Chebyshev polynomials for the interval $[R, 1]$ (see Section 2.4.1). Therefore we should use these parameters for our method. We note that with these parameters our method is actually the Arnoldi/Chebyshev algorithm of Elman et al. [30]. Alternatively we could choose a small angle for the half-angle of the sector, for example $\mu = 0.14$ radians, and find a, b and ρ for this annular sector and then use these parameters in our method. In Table 5.11 we give both these sets of parameters found from eigenvalue estimates produced after 16 steps of the Arnoldi/GMRES phase of our algorithm. In Figure (5.19) we show both these plots, the lower one (dash-dot) is the Arnoldi/Chebyshev method and the upper one (dashed) is our method with $\mu = 0.14$.

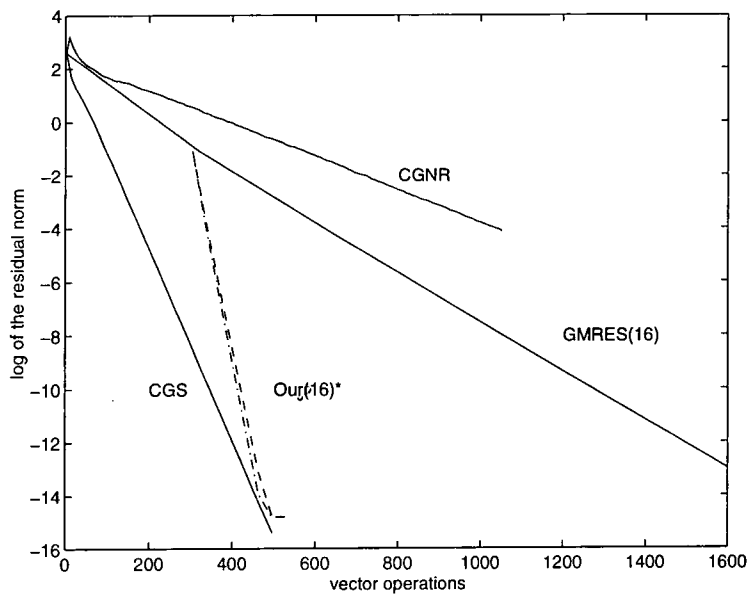

Figure 5.19: Convergence curves for various methods and coefficient matrix D. The star on Our(16) reminds us that two curves are pictured here.

Although CGS has a residual norm less than 10^{-14} in fewer vector operations than both of our methods (that is, Arnoldi/Chebyshev with $\mu = 0.0$ and our method with $\mu = 0.14$), the slopes of our methods are steeper than those of CGS and it would appear that our method would beat CGS if higher precision was required. As was stated above, for this matrix, CGNR is the slowest to converge, not just in terms of the number of iterations, but also in terms of the number of vector operations.

Table 5.11: The parameters for example 6, with random starting vectors.

eigenvalue estimates	Arnoldi/Chebyshev	Our method(16)
R_{max}	19.957514299	19.957514299
R_{min}	1.056885688	1.056885688
μ	0.0	0.14
a	0.479711788	0.415066055
b	0.479711788	0.434455942
ρ	0.236760805	0.299907255

Example 7

Dr. G. Opfer [61] mentioned to my Supervisor, Dr. J.P. Coleman, that he thought it would be useful to solve linear systems of equations with coefficient matrices whose eigenvalues (or eigenvalue estimates) are situated close to the n–th roots of unity. He suggested applying the transformation $Z = z^n$ so that the eigenvalues (or their estimates) are situated about the positive real axis. His idea was then to use the Faber polynomials, $F_m(Z)$, for an annular sector placed around these estimates, as the residual polynomials. As a simple example we consider the matrix

$$A = \begin{pmatrix} 1 & 1 & & & & \\ & -1 & 1 & & & \\ & & 1 & 1 & & \\ & & & -1 & 1 & \\ & & & & \ddots & \ddots \end{pmatrix} \in \mathbb{R}^{200\times 200},$$

which has its eigenvalues at the points 1 and -1. We note that the eigenvalues are situated on both sides of the origin and therefore the Arnoldi/Faber method of Starke and Varga, as it is described, cannot be used to solve this example. We begin by trying to solve

$$A\mathbf{x} = \mathbf{b}, \tag{5.12}$$

with a random $\mathbf{x}_0$ and $\mathbf{b}$.

In Figure (5.20) we show convergence curves for various methods. The fastest to converge is CGNR, followed by CGS. It is interesting to see that our method for solving

equation 5.12, with 16, 32 or 48 eigenvalue estimates, diverges initially and then converges. We also note that in all cases our method stagnates. If the residual norm rises to 10^7, say, then our method, using double precision, will stagnate when the residual norm is about 10^{-8}. We have not produced a satisfactory reason for this. In Table 5.12 we give the parameters one would require to run our methods for this example.

Table 5.12: The parameters for example 7, with random starting vectors.

eigenvalue estimates	16	32	48
R_{max}	1.309407302	1.352849882	1.378345068
R_{min}	0.323257614	0.255873833	0.102660078
μ	1.967132252	2.086166027	2.223948399
η	1.595575242	1.566125230	1.568494460
a	0.110696486	0.073014090	0.023615741
b	0.296529065	0.217430031	0.080850177
ρ	0.886040399	0.891161542	0.917063783

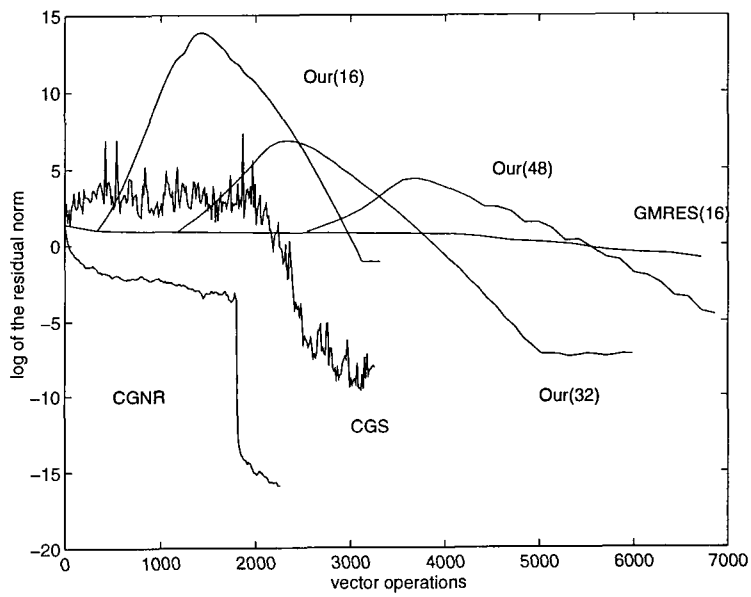

Figure 5.20: Convergence curves for solving (5.12) by various methods

We now consider multiplying the linear system (5.12) by A, that is, exploit the idea of

G. Opfer, and try to solve

$$A^2\mathbf{x} = \mathbf{b}_1, \tag{5.13}$$

with $\mathbf{b}_1 = A\mathbf{b}$. If we find a solution to this linear system then we will have a solution to the original linear system (5.12). The eigenvalues of A^2 are simply the eigenvalues of A squared. Therefore the idea is to square the approximate eigenvalues we obtained above, then place an annular sector around these squared estimates and use the Faber polynomials for this region as the residual polynomials.

To compare results we need to calculate how many vector operations we use to implement the above method. The number of vector operations required to carry out m steps of the Arnoldi algorithm and produce m eigenvalue estimates is the same as before, namely $m(l+3+m)$ vector operations. The number of vector operations to implement

$$\mathbf{x}_m = \mathbf{x}_0 + q_{m-1}(A^2)\mathbf{r}_0,$$

in a Horner-type iteration, will be different because two matrix multiplications are required where previously we only required one. As in Section 5.3.4 the Horner iteration is as follows,

$$\mathbf{r}_{\text{old}} = A\mathbf{r}_{\text{new}},$$

$$\mathbf{w}_0 = \alpha_{m-1,m}\mathbf{r}_{\text{old}},$$

$$\mathbf{w}_j = A^2\mathbf{w}_{j-1} + \alpha_{m-1-j,m}\mathbf{r}_{\text{old}}, \qquad j = 1, \ldots, m-1,$$

$$\mathbf{x}_{\text{new}} = \mathbf{x}_{\text{old}} + \mathbf{w}_{m-1},$$

$$\mathbf{r}_{\text{new}} = \mathbf{b} - A\mathbf{x}_{\text{new}}.$$

This requires a total of $m(2l+1)$ vector operations.

In Figure (5.21) we show plots for solving the linear system (5.13) by our method. In all cases the methods do not diverge for as long as they did previously, and therefore the methods stagnate at a smaller tolerance than they did in Figure (5.20). The number of vector operations, required to converge to a specific tolerance, is less when solving the linear system (5.13) than solving (5.12). For example, our method to solve (5.12) with 16 steps of the Arnoldi/GMRES algorithm converges to 10^{-1} in about 3000 vector

operations, whereas, when solving (5.13) our method converges to 10^{-8} in about 1900 vector operations. In fact our method with 16 eigenvalue estimates for solving (5.13) is comparable with CGNR for solving (5.12). We also note that when solving (5.13) our method with 32 eigenvalue estimates is comparable to CGS for solving (5.12). These results show that when the eigenvalues (or their estimates) are tightly clustered around the roots of unity the idea of G. Opfer may be a useful one. In Table 5.13 we give the parameters we used in this example. That is, the parameters for the annular sectors placed around the squared eigenvalue estimates.

Table 5.13: Example 7, with random starting vectors. The parameters for an annular sector placed around the squared eigenvalue estimates.

eigenvalue estimates	16	32	48
R_{max}	1.714547483	1.830202803	1.899835127
R_{min}	0.104495485	0.065471418	0.010539092
μ	0.842229681	1.040081594	1.310907878
a	0.231168909	0.154764191	0.049199872
b	0.316014282	0.231410936	0.084431146
ρ	0.547503339	0.612776474	0.696272420

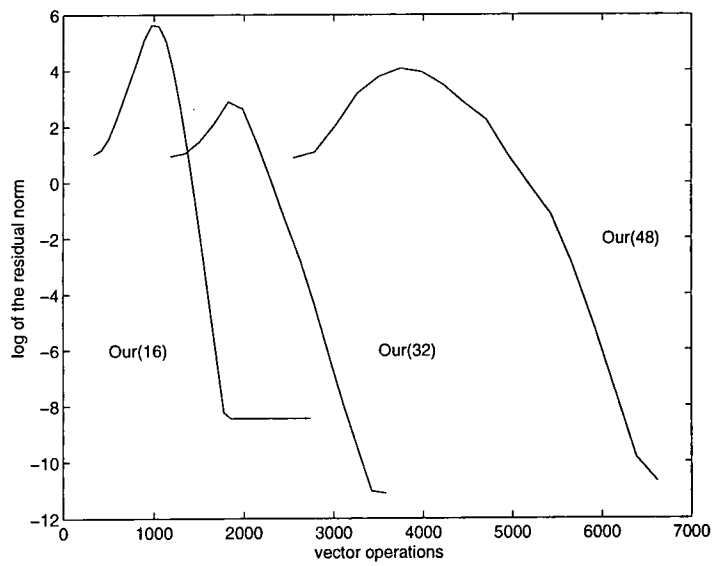

Figure 5.21: Convergence curve for solving (5.13) by our method.

For clarity, we would like to point out that the results given for methods other than our own were carried out by us and not taken from the literature. This is with the exception of examples 1 and 2, where the results given are those from a paper by Starke and Varga [76].

5.5 Conclusions

In this chapter we have given a new hybrid iterative method based on the Faber polynomials for an annular sector. The method is closely related to that of Starke and Varga [76]. If the spectrum, or rather the ϵ-pseudospectrum (see Trefethen [78]), of the matrix can be well approximated by an annular sector then the method should work well. Our method has two major differences over the method of Starke and Varga. Firstly, the annular sector is an easy region to scale and rotate so that it encloses any eigenvalue estimates situated anywhere in the complex plane. On the other hand the method of Starke and Varga [76] is set up only for eigenvalue estimates situated symmetrically about the real axis. Starke and Varga do state that their method can be applied to a general non-Hermitian matrix. However, for a general non-Hermitian matrix, their method for placing a polygonal region around the eigenvalue estimates and their method for finding the conformal mapping must change. Therefore, their method would seem more complicated to implement than ours. Secondly, the Arnoldi/Faber method requires a numerical conformal mapping package to generate the Faber polynomials, whereas, the Faber polynomials for an annular sector are known analytically (see Chapter 2). Of course, to generate the Faber polynomials, we need to determine the numbers a, b and ρ numerically. Therefore, we can either generate the Faber polynomials for a polygonal region using a numerical conformal mapping package, or use the Faber polynomials for an annular sector known exactly in terms of the numbers a, b and ρ, which we only know numerically.

In the examples of this chapter we have shown how our method compares with many of the methods we described in Chapter 4. In particular, CGS, CGNR, GMRES, Hybrid GMRES and Arnoldi/Faber. Our method seems to perform quite well for the examples we have given. As with all methods, there exist matrices for which our method will

perform badly. In agreement with the method and examples of Starke and Varga [76], our examples show that our method is sensitive to the amount of information obtained during the Arnoldi/GMRES stage. That is, if too few steps of the Arnoldi/GMRES algorithm are taken then the method may converge slowly, or even diverge. If too many steps of the Arnoldi/GMRES algorithm are taken then more work is done in stage 1 than is required.

Chapter 6

Conclusions and future work

In this thesis we have considered the Faber polynomials for annular sectors. The conformal map from the complement of the unit disc to the complement of the annular sector was given in Chapter 2. In this chapter we also derived the Faber polynomials for an annular sector and gave the Faber series for $1/z$. In Chapter 3 we considered numerical aspects of the Faber polynomials for an annular sector. The numerical approximation of the transfinite diameter and the parameters a and b were considered. We also gave three norms of Faber polynomials and exhibited some numerical values for various annular sectors. At the end of the chapter we improved upon a bound for the Faber projection given in a paper by Ellacott [22]. In Chapter 5 we gave a new hybrid method for the iterative solution of linear systems of equations. The method is similar to the Arnoldi/Faber method developed by Starke and Varga [76], but with a few major differences. For Example 1, with a structured righthand side, $\mathbf{b}$, and initial solution $\mathbf{x}_0$ the residual norm converged faster with our method than it did with Starke and Varga's. In most of the given examples the method seemed promising. As with Starke and Varga's method, however, our method is sensitive to the amount of information obtained in the first stage of the hybrid method, namely the Arnoldi/GMRES algorithm.

Chapters 2 and 3 are mainly complete. Most of the future work will lie in Chapter 5 and extensions of ideas contained in it. At present, no convergence estimate exists for the method and it would be nice to obtain one. Of course the method needs further testing, especially on real world examples. As for possible extensions to the work contained in Chapter 5, we have arrived at three or four different ideas for residual polynomials which could be implemented in an iterative method.

• The first idea combines the residual polynomials from the Hybrid GMRES method of Nachtigal et al. [60] and our method from Chapter 5. A hybrid iterative method, employing this idea, would carry out a few steps of the Arnoldi/GMRES algorithm and obtain some eigenvalue estimates, a good starting guess, and the GMRES polynomial, $g_n(\lambda)$. Then one would find the Faber polynomials, $F_n(\lambda)$, for an annular sector placed around the eigenvalue estimates and iterate with a residual polynomial given by

$$R_{2n}(\lambda) = \sigma g_n(\lambda) F_n(\lambda),$$

where σ is a constant chosen so that $R_{2n}(0) = 1$. The strength of this idea is that Hybrid

GMRES seems to perform poorly when $\mathbf{b}$ and $\mathbf{x}_0$ are structured, whereas, our method appears to do better. Hybrid GMRES, however, does not rely on any approximate eigenvalue information which can be misleading (see Nachtigal et al. [60]). Hopefully this idea would lead to a method with the advantages of both Hybrid GMRES and our method.

• The second idea is very similar to the first. Instead of GMRES as the first step of the method, why not try the BCG algorithm or, to avoid some of the breakdowns, the QMR algorithm? As in Equation (4.22), after $n+1$ steps of the BCG algorithm, $AV_n = V_{n+1}\widehat{H}_n$. Freund et al. [37] therefore suggest using the eigenvalues of $\widehat{H}_n$ as estimates of the eigenvalues of A. For this idea we may consider a similar hybrid iterative method to that proposed in the first idea given above. This time with a residual polynomial given by

$$R_{2n}(\lambda) = \sigma\psi_n^{BCG}(\lambda)F_n(\lambda),$$

where once again σ is chosen so that $R_{2n}(0) = 1$ and ψ_n^{BCG} is the residual polynomial of degree n for the BCG algorithm. We note that this idea could be motivated along similar lines to BiCGSTAB and BiCGSTAB2. By this we mean that the BCG polynomial is multiplied by another polynomial of degree n. This time, however, no local minimisation problem is solved (see Section 4.3.7).

• The third idea comes from the work of Eiermann [17] on semi-iterative methods (see Li [53] for an overview). As in Section 4.4.7, we once again note that, in our case, we must consider the Faber series for $1/z$ not $1/(z-1)$. The idea here is to use the truncated generalised Faber series for $1/z$ as the residual polynomial (see SIM1 in Eiermann [17]). An alternative would be to use the truncated Faber series for $1/z$ as the residual polynomial (see SIM2 in Eiermann [17]). The latter Faber series is readily available and given by Equation (2.41) in Section 2.7.

The three given ideas may or may not be of use and we leave their consideration to future researchers.

Finally, we consider an idea for future research in a completely different area of Numerical Analysis. That is, the approximation, in the complex plane, of a function which satisfies an ordinary differential equation (ODE). The idea would build on the work of Coleman [9, 10]. In this work, Coleman subdivides the complex plane into circular sectors,

and solves the differential equation on these subdivisions of the plane by using the Lanczos τ-method and the Faber polynomials for a circular sector. If better approximations are required, than those given by this method, then one could further subdivide the circular sector into a smaller circular sector and an annular sector (see Figure 6.1). The idea would

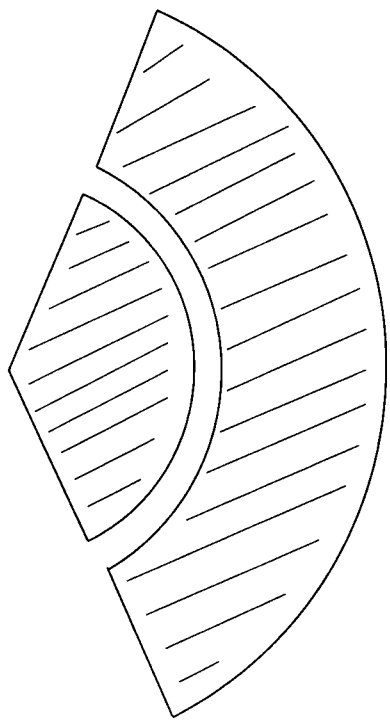

Figure 6.1: The large circular sector is subdivided into a smaller one and an annular sector.

proceed by solving the ODE on the smaller circular sector, using the ideas of Coleman [9, 10]. Then it would solve the ODE on the annular sector, using the Lanczos τ-method and the Faber polynomials for this annular sector.

The problem with this idea is in obtaining a sensible starting point for the solution of the ODE on the annular sector. Many ideas could be thought of to solve this problem, for example,

- Use the original initial solution for the ODE;

- Consider a point on the common boundary between the circular and annular sectors. Find an approximation to the value of the function at this point, using the approximation to the function obtained on the small circular sector. Use this estimate of the value of the

function as the starting point for the approximation on the annular sector.

Other possibilities may exist, but we leave these and the implementation of this idea as matters for future research.

Appendix A

An alternative proof of Theorem 7

In this appendix we give another proof of Theorem 7 in section 2.2.4. The result is as $a \to 0$ and $b \to 0$ then $R \to 0$ and $\theta \to \lambda\pi$ where $a = \lambda b$.

Considering $a = \lambda b$ then (2.7) becomes

$$\pi - \theta = \int_{\lambda^2 b^2}^{b^2} \left[\frac{(b^2 - x)(b^{-2} - x)}{(x - \lambda^2 b^2)(\lambda^{-2} b^{-2} - x)} \right]^{\frac{1}{2}} \frac{dx}{x}.$$

In this we make the substitution $x = tb^2$, hence,

$$\pi - \theta = \int_{\lambda^2}^{1} \left[\frac{(1 - t)(1 - b^4 t)}{(t - \lambda^2)(\lambda^{-2} - b^4 t)} \right]^{\frac{1}{2}} \frac{dt}{t}.$$

When $b \to 0$ the integral becomes elementary, and the expression may be written as

$$\pi - \theta = \lambda \int_{\lambda^2}^{1} \frac{(1 - t)^{\frac{1}{2}}\, dt}{t(t - \lambda^2)^{\frac{1}{2}}}.$$

In this we make the substitution $y^2 = (1 - t)/(t - \lambda^2)$ and therefore $dt/\,dy = 2y(\lambda^2 - 1)/(y^2 + 1)^2$, so

$$\begin{aligned}
\pi - \theta &= \lambda \int_{\infty}^{0} \frac{2(\lambda^2 - 1)y^2}{(y^2 + 1)(1 + \lambda^2 y^2)}\, dy \\
&= 2\lambda \int_{\infty}^{0} \left(\frac{1}{y^2 + 1} - \frac{1}{1 + \lambda^2 y^2} \right) dy \\
&= 2\left[\lambda \arctan(y) - \arctan(\lambda y) \right]_{\infty}^{0} \\
&= 2\left[0 - \left(\frac{\lambda\pi}{2} - \frac{\pi}{2} \right) \right] \\
&= \pi - \lambda\pi
\end{aligned}$$

Therefore $\lambda = \theta/\pi$ as was proven in section 2.1.4.

When we substitute $a = \lambda b$ equation (2.10) becomes

$$\log R = -2 \int_{b^2}^{1} \left[\frac{(x - b^2)(b^{-2} - x)}{(x - \lambda^2 b^2)(b^{-2}\lambda^{-2} - x)} \right]^{\frac{1}{2}} \frac{dx}{x}. \tag{A.1}$$

In this expression we again substitute $x = tb^2$ and obtain

$$\begin{aligned}
\log R &= -2 \int_{1}^{b^{-2}} \left[\frac{(t - 1)(1 - tb^4)}{(t - \lambda^2)(\lambda^{-2} - b^4 t)} \right]^{\frac{1}{2}} \frac{dt}{t} \\
&= 2 \int_{1}^{b^2} \left[\frac{(1 - y)(y - b^4)}{(1 - y\lambda^2)(\lambda^{-2} y - b^4)} \right]^{\frac{1}{2}} \frac{dy}{y},
\end{aligned}$$

when we let $t = 1/y$. In the limit as $b \to 0$ this then becomes

$$\log R = 2\lambda \int_1^0 \frac{(1-y)^{\frac{1}{2}}\, dy}{y(1-\lambda^2 y)^{\frac{1}{2}}}.$$

A substitution of $t^2 = (1-y)/(1-\lambda^2 y)$ gives us

$$\log R \;=\; 4\lambda \int_0^1 \frac{(\lambda^2-1)t^2}{(1-t^2)(1-t^2\lambda^2)}\, dt.$$

By using partial fractions we obtain

$$\begin{aligned} \log R &= 2\lambda \int_0^1 \left(-\frac{1}{1-t} - \frac{1}{t+1} + \frac{1}{1-\lambda t} + \frac{1}{1+\lambda t} \right) dt \\ &= 2\left[\lambda \log \frac{1-t}{1+t} + \log \frac{1+\lambda t}{1-\lambda t} \right]_0^1 \\ &= \log \frac{0^{2\lambda}(1+\lambda)^2}{(1-\lambda)^2} \end{aligned} \tag{A.2}$$

From this we obtain $R = 0$, the only exception being when the expression in (A.2) is undefined that is when $\lambda = 1$. In this case (A.1) gives $R = b^4$, for all b, so here $b = 0$ and we have $R = 0$. So the result holds even for the case when $\lambda = 1$. This proves the above result and finishes off this appendix.

Appendix B

The first few Faber Polynomials for an Annular Sector

From Section 2.4, we know that

$$F_n(z) = \rho^n \Phi_n\left(\frac{z}{\rho}\right)$$

where

$$\Phi_n(z) = z^n + \phi_{n-1}(z).$$

The polynomials

$$\phi_{n-1}(z) = \sum_{j=0}^{n-1} p_j z^{n-j-1} \text{ for } 1 \leq n \leq 15,$$

are given by

```
Degree of Faber Polynomial:  1
p(0):   U

Degree of Faber Polynomial:  2
p(0):   2*U
p(1):   (S^2+2*S*U-U^2-4)/2

Degree of Faber Polynomial:  3
p(0):   3*U
p(1):   (3*(S^2+2*S*U+U^2-4))/4
p(2):   (2*S^3+S^2*U-2*S*U^2-8*S+U^3+2*U)/2

Degree of Faber Polynomial:  4
p(0):   4*U
p(1):   S^2+2*S*U+3*U^2-4
p(2):   (4*S^3+5*S^2*U+2*S*U^2-16*S+U^3-8*U)/3
p(3):   (11*S^4-4*S^3*U-6*S^2*U^2-48*S^2+12*S*U^3
        +32*S*U-5*U^4-16*U^2+16)/8

Degree of Faber Polynomial:  5
p(0):   5*U
p(1):   (5*(S^2+2*S*U+5*U^2-4))/4
p(2):   (5*(S^3+2*S^2*U+2*S*U^2-4*S+U^3-5*U))/3
p(3):   (5*(9*S^4+8*S^3*U+4*S^2*U^2-42*S^2+4*S*U^3
        -20*S*U-U^4-10*U^2+24))/24
p(4):   (12*S^5-13*S^4*U+8*S^3*U^2-56*S^3+14*S^2*U^3
        +60*S^2*U-20*S*U^4-72*S*U^2+32*S+7*U^5+28*U^3+8*U)/8

Degree of Faber Polynomial:  6
p(0):   6*U
p(1):   (3*(S^2+2*S*U+7*U^2-4))/2
p(2):   (4*S^3+11*S^2*U+14*S*U^2-16*S+11*U^3-32*U)/2
p(3):   (39*S^4+76*S^3*U+74*S^2*U^2-192*S^2+44*S*U^3
        -256*S*U+7*U^4-128*U^2+144)/16
p(4):   (46*S^5+31*S^4*U+34*S^3*U^2-248*S^3+22*S^2*U^3
        -120*S^2*U-20*S*U^4-136*S*U^2+256*S+7*U^5+24*U^3
        +104*U)/20
p(5):   (21*S^6-34*S^5*U+65*S^4*U^2-100*S^4-20*S^3*U^3
        +112*S^3*U-65*S^2*U^4-312*S^2*U^2+72*S^2+70*S*U^5
        +304*S*U^3+144*S*U-21*U^6-100*U^4-72*U^2-32)/16
```

```
Degree of Faber Polynomial:  7

p(0):   7*U
p(1):   (7*(S^2+2*S*U+9*U^2-4))/4
p(2):   (7*(2*S^3+7*S^2*U+10*S*U^2-8*S+11*U^3-22*U))/6
p(3):   (7*(7*S^4+20*S^3*U+26*S^2*U^2-36*S^2+20*S*U^3
        -72*S*U+7*U^4-52*U^2+32))/16
p(4):   (7*(112*S^5+217*S^4*U+228*S^3*U^2-656*S^3
        +134*S^2*U^3-960*S^2*U+20*S*U^4-672*S*U^2
        +832*S+9*U^5-112*U^3+608*U))/240
p(5):   (7*(1045*S^6+694*S^5*U+1379*S^4*U^2-6440*S^4-44*S^3*U^3
        -4352*S^3*U-757*S^2*U^4-5760*S^2*U^2+9760*S^2+790*S*U^5
        +3296*S*U^3+7744*S*U-227*U^6-1144*U^4-224*U^2-2880))/2880
p(6):   (14*S^7-19*S^6*U+102*S^5*U^2-64*S^5-145*S^4*U^3
        -50*S^4*U+10*S^3*U^4-320*S^3*U^2+64*S^3+147*S^2*U^5
        +780*S^2*U^3+528*S^2*U-126*S*U^6-640*S*U^4
        -576*S*U^2-128*S+33*U^7+182*U^5+208*U^3+16*U)/16

Degree of Faber Polynomial:  8
p(0):   8*U
p(1):   2*(S^2+2*S*U+11*U^2-4)
p(2):   (2*(4*S^3+17*S^2*U+26*S*U^2-16*S+37*U^3-56*U))/3
p(3):   (45*S^4+164*S^3*U+262*S^2*U^2-240*S^2+244*S*U^3
        -608*S*U+125*U^4-592*U^2+240)/12
p(4):   (44*S^5+129*S^4*U+176*S^3*U^2-272*S^3+138*S^2*U^3
        -600*S^2*U+60*S*U^4-544*S*U^2+384*S+13*U^5
        -184*U^3+416*U)/10
p(5):   (1475*S^6+2978*S^5*U+3403*S^4*U^2-10060*S^4+1652*S^3*U^3
        -16624*S^3*U+241*S^2*U^4-13320*S^2*U^2+18080*S^2
        +410*S*U^5-2288*S*U^3+20288*S*U-79*U^6-908*U^4
        +4832*U^2-5760)/360
p(6):   (6552*S^7+6541*S^6*U+12214*S^5*U^2-46464*S^5-9853*S^4*U^3
        -54080*S^4*U+316*S^3*U^4-43040*S^3*U^2+97536*S^3
        +11123*S^2*U^5+57408*S^2*U^3+120352*S^2*U-9002*S*U^6
        -47776*S*U^4-33344*S*U^2-66048*S+2269*U^7+12992*U^5
        +17056*U^3-14592*U)/2520
p(7):   (51*S^8+184*S^7*U+532*S^6*U^2-256*S^6-2296*S^5*U^3
        -3072*S^5*U+2450*S^4*U^4+1920*S^4*U^2+960*S^4+392*S^3*U^5
        +7680*S^3*U^3+8960*S^3*U-2604*S^2*U^6-15360*S^2*U^4
        -17280*S^2*U^2-3072*S^2+1848*S*U^7+10752*S*U^5+14080*S*U^3
        +2048*S*U-429*U^8-2688*U^6-4160*U^4-1024*U^2+256)/128

Degree of Faber Polynomial:  9
p(0):   9*U
p(1):   (9*(S^2+2*S*U+13*U^2-4))/4
p(2):   3*(S^3+5*S^2*U+8*S*U^2-4*S+14*U^3-17*U)
p(3):   (3*(6*S^4+26*S^3*U+49*S^2*U^2-33*S^2+52*S*U^3
        -98*S*U+35*U^4-121*U^2+36))/4
p(4):   (3*(38*S^5+143*S^4*U+242*S^3*U^2-244*S^3+236*S^2*U^3
        -690*S^2*U+140*S*U^4-788*S*U^2+368*S+41*U^5
        -378*U^3+532*U))/20
p(5):   (3*(160*S^6+486*S^5*U+691*S^4*U^2-1155*S^4+544*S^3*U^3
        -2788*S^3*U+262*S^2*U^4-2850*S^2*U^2+2260*S^2+90*S*U^5
        -1316*S*U^3+3496*S*U+7*U^6-291*U^4+1444*U^2-800))/80
p(6):   (2751*S^7+6103*S^6*U+7054*S^5*U^2-21612*S^5+2288*S^4*U^3
        -41135*S^4*U+1171*S^3*U^4-32516*S^3*U^2+50928*S^3
        +1463*S^2*U^5-4506*S^2*U^3+72256*S^2*U-896*S*U^6
        -7240*S*U^4+22240*S*U^2-33984*S+226*U^7+1169*U^5+5776*U^3
        -16416*U)/560
p(7):   (11627*S^8+20412*S^7*U+18094*S^6*U^2-97314*S^6
        -39552*S^5*U^3-180092*S^5*U+41712*S^4*U^4-5806*S^4*U^2
        +275856*S^4+9228*S^3*U^5+134392*S^3*U^3+417280*S^3*U
        -44534*S^2*U^6-270814*S^2*U^4-276736*S^2*U^2-300608*S^2
        +30072*S*U^7+181060*S*U^5+259904*S*U^3-73856*S*U-6739*U^8
        -43666*U^6-71504*U^4-28736*U^2+40320)/4480
p(8):   (24*S^9+611*S^8*U-736*S^7*U^2-400*S^7-1988*S^6*U^3
        -5768*S^6*U+6160*S^5*U^4+13104*S^5*U^2+3072*S^5
        -4886*S^4*U^5-4760*S^4*U^3+10320*S^4*U-2016*S^3*U^6
        -22960*S^3*U^4-38400*S^3*U^2-7680*S^3+5676*S^2*U^7
        +37128*S^2*U^5+57120*S^2*U^3+12160*S^2*U-3432*S*U^8
        -22512*S*U^6-38400*S*U^4-12800*S*U^2+1024*S+715*U^9
        +5016*U^7+9744*U^5+4480*U^3+128*U)/128

Degree of Faber Polynomial:  10
p(0):   10*U
p(1):   (5*(S^2+2*S*U+15*U^2-4))/2
p(2):   (5*(4*S^3+23*S^2*U+38*S*U^2-16*S+79*U^3-80*U))/6
p(3):   (5*(17*S^4+84*S^3*U+182*S^2*U^2-96*S^2+212*S*U^3
        -320*S*U+177*U^4-480*U^2+112))/16
p(4):   (43*S^5+193*S^4*U+387*S^3*U^2-284*S^3+446*S^2*U^3
        -960*S^2*U+320*S*U^4-1308*S*U^2+448*S+123*U^5
        -808*U^3+812*U)/6
p(5):   (595*S^6+2314*S^5*U+4079*S^4*U^2-4460*S^4+4156*S^3*U^3
        -13712*S^3*U+2693*S^2*U^4-17640*S^2*U^2+9220*S^2
        +1090*S*U^5-11344*S*U^3+18184*S*U+193*U^6-3244*U^4
        +10396*U^2-3600)/72
p(6):   (441*S^7+1403*S^6*U+2036*S^5*U^2-3652*S^5+1585*S^4*U^3
        -9560*S^4*U+869*S^3*U^4-10392*S^3*U^2+9168*S^3+357*S^2*U^5
        -5536*S^2*U^3+17116*S^2*U+14*S*U^6-2052*S*U^4+10392*S*U^2
        -6464*S+15*U^7-168*U^5+2524*U^3-4736*U)/56
p(7):   (46543*S^8+116424*S^7*U+120004*S^6*U^2-418656*S^6
        +23624*S^5*U^3-903040*S^5*U+59314*S^4*U^4-607584*S^4*U^2
        +1233624*S^4+15800*S^3*U^5-131072*S^3*U^3+1995488*S^3*U
        -38892*S^2*U^6-311072*S^2*U^4+598032*S^2*U^2-1265152*S^2
        +25592*S*U^7+150912*S*U^5+452064*S*U^3-944128*S*U
        -5529*U^8-37408*U^6-45608*U^4-209920*U^2+201600)/8064
p(8):   (99134*S^9+258177*S^8*U-46062*S^7*U^2-996504*S^7
        -300212*S^6*U^3-2225376*S^6*U+916530*S^5*U^4
        +1666776*S^5*U^2+3610464*S^5-665814*S^4*U^5-733008*S^4*U^3
        +4862448*S^4*U-327410*S^3*U^6-3520104*S^3*U^4
        -5466336*S^3*U^2-5220224*S^3+796356*S^2*U^7+5350848*S^2*U^5
        +8842080*S^2*U^3-352128*S^2*U-460752*S*U^8-3117528*S*U^6
        -5595840*S*U^4-2338944*S*U^2+1511424*S+92933*U^9
        +671856*U^7+1367664*U^5+653696*U^3+302976*U)/36288
p(9):   (127*S^10+1654*S^9*U-5499*S^8*U^2-2492*S^8+3768*S^7*U^3
        -10720*S^7*U+14070*S^6*U^4+52976*S^6*U^2+15344*S^6
        -31500*S^5*U^5-87584*S^5*U^3-4704*S^5*U+18186*S^4*U^6
        +13720*S^4*U^4-89040*S^4*U^2-30400*S^4+13464*S^3*U^7
        +132832*S^3*U^5+275520*S^3*U^3+83200*S^3*U-24453*S^2*U^8
        -176400*S^2*U^6-341040*S^2*U^4-144000*S^2*U^2+3200*S^2
        +12870*S*U^9+93984*S*U^7+196896*S*U^5+108800*S*U^3
        +6400*S*U-2431*U^10-18876*U^8-44016*U^6-30400*U^4-3200*U^2
        -512)/256

Degree of Faber Polynomial:  11
p(0):   11*U
p(1):   (11*(S^2+2*S*U+17*U^2-4))/4
p(2):   (11*(2*S^3+13*S^2*U+22*S*U^2-8*S+53*U^3-46*U))/6
p(3):   (11*(27*S^4+148*S^3*U+362*S^2*U^2-156*S^2+452*S*U^3
        -568*S*U+451*U^4-1004*U^2+192))/48
p(4):   (11*(64*S^5+329*S^4*U+756*S^3*U^2-432*S^3+998*S^2*U^3
        -1680*S^2*U+820*S*U^4-2624*S*U^2+704*S+393*U^5-1984*U^3
        +1536*U))/80
p(5):   (11*(571*S^6+2650*S^5*U+5549*S^4*U^2-4400*S^4+6796*S^3*U^3
        -16160*S^3*U+5333*S^2*U^4-24912*S^2*U^2+9472*S^2
        +2650*S*U^5-19840*S*U^3+22528*S*U+643*U^6-7264*U^4
        +16384*U^2-4032))/576
p(6):   (11*(2646*S^7+10643*S^6*U+19298*S^5*U^2-22692*S^5
```

```
        +20317*S^4*U^3-74590*S^4*U+14162*S^3*U^4-103828*S^3*U^2
        +59568*S^3+6601*S^2*U^5-77496*S^2*U^3+139616*S^2*U
        +1694*S*U^6-33104*S*U^4+116624*S*U^2-44544*S+239*U^7
        -5810*U^5+36992*U^3-43656*U))/2520
p(7):   (11*(4109*S^8+13720*S^7*U+19756*S^6*U^2-38528*S^6
        +15432*S^5*U^3-107696*S^5*U+10054*S^4*U^4-119512*S^4*U^2
        +118272*S^4+3720*S^3*U^5-72448*S^3*U^3+247808*S^3*U
        -196*S^2*U^6-34864*S^2*U^4+178944*S^2*U^2-125216*S^2
        +728*S*U^7-1872*S*U^5+74240*S*U^3-143936*S*U-123*U^8
        -1400*U^6+7936*U^4-44448*U^2+22400))/4480
p(8):   (11*(224672*S^9+616755*S^8*U+490248*S^7*U^2-2294232*S^7
        +151780*S^6*U^3-5271768*S^6*U+556872*S^5*U^4
        -2675184*S^5*U^2+8119392*S^5-262758*S^4*U^5
        -1948224*S^4*U^3+13577472*S^4*U-163976*S^3*U^6
        -2185080*S^3*U^4+4042560*S^3*U^2-11138432*S^3
        +362196*S^2*U^7+2423976*S^2*U^5+6434688*S^2*U^3
        -9714432*S^2*U-200616*S*U^8-1404864*S*U^6-2330784*S*U^4
        -4501248*S*U^2+3958272*S+39227*U^9+291696*U^7+638592*U^5
        -47488*U^3+1484928*U))/362880
p(9):   (11*(2121147*S^10+5829694*S^9*U-6999537*S^8*U^2
        -24818652*S^8+5268744*S^7*U^3-45883632*S^7*U
        +20664110*S^6*U^4+71485128*S^6*U^2+105034752*S^6
        -42494460*S^5*U^5-126671232*S^5*U^3+77767680*S^5*U
        +22252926*S^4*U^6+11912160*S^4*U^4-118152576*S^4*U^2
        -177481152*S^4+19091432*S^3*U^7+185368272*S^3*U^5
        +404934144*S^3*U^3+42148352*S^3*U-31488489*S^2*U^8
        -233078568*S^2*U^6-471449856*S^2*U^4-229334784*S^2*U^2
        +76563072*S^2+15933390*S*U^9+119678112*S*U^7
        +261591552*S*U^5+151814400*S*U^3+34903296*S*U
        -2921357*U^10-23309988*U^8-56512896*U^6-41162816*U^4
        -2602368*U^2-7257600))/7257600
p(10):  (342*S^11+943*S^10*U-8270*S^9*U^2-5416*S^9+19515*S^8*U^3
        +3222*S^8*U-6180*S^7*U^4+52128*S^7*U^2+27456*S^7
        -45906*S^6*U^5-201096*S^6*U^3-77056*S^6*U+77028*S^5*U^6
        +257040*S^5*U^4+49728*S^5*U^2-43904*S^5-30690*S^4*U^7
        +8820*S^4*U^5+344960*S^4*U^3+182560*S^4*U-38610*S^3*U^8
        -370272*S^3*U^6-902720*S^3*U^4-492800*S^3*U^2-3840*S^3
        +52195*S^2*U^9+412632*S^2*U^7+956928*S^2*U^5
        +624960*S^2*U^3+55680*S^2*U-24310*S*U^10-195624*S*U^8
        -485184*S*U^6-380800*S*U^4-57600*S*U^2-3072*S+4199*U^11
        +35750*U^9+97152*U^7+90272*U^5+19840*U^3+256*U)/256

Degree of Faber Polynomial:  12
p(0):   12*U
p(1):   3*(S^2+2*S*U+19*U^2-4)
p(2):   4*S^3+29*S^2*U+50*S*U^2-16*S+137*U^3-104*U
p(3):   (57*S^4+340*S^3*U+926*S^2*U^2-336*S^2+1220*S*U^3
        -1312*S*U+1417*U^4-2672*U^2+432)/8
p(4):   (212*S^5+1217*S^4*U+3128*S^3*U^2-1456*S^3+4634*S^2*U^3
        -6360*S^2*U+4220*S*U^4-11072*S*U^2+2432*S+2429*U^5
        -9912*U^3+6208*U)/20
p(5):   (1115*S^6+5938*S^5*U+14283*S^4*U^2-8780*S^4+20212*S^3*U^3
        -37104*S^3*U+18361*S^2*U^4-66120*S^2*U^2+19520*S^2
        +10730*S*U^5-61808*S*U^3+53888*S*U+3281*U^6
        -27788*U^4+47552*U^2-8960)/80
p(6):   (5376*S^7+25727*S^6*U+55682*S^5*U^2-47328*S^5
        +71353*S^4*U^3-185200*S^4*U+60140*S^3*U^4-311008*S^3*U^2
        +128640*S^3+33985*S^2*U^5-286752*S^2*U^3+361280*S^2*U
        +11858*S*U^6-150272*S*U^4+375296*S*U^2-101376*S+1991*U^7
        -35504*U^5+150848*U^3-124416*U)/336
p(7):   (422359*S^8+1752408*S^7*U+3231140*S^6*U^2-4081728*S^6
        +3503016*S^5*U^3-14138752*S^5*U+2606250*S^4*U^4
        -20748992*S^4*U^2+12980352*S^4+1294056*S^3*U^5
        -17181952*S^3*U^3+34007552*S^3*U+389732*S^2*U^6
        -8902592*S^2*U^4+34254592*S^2*U^2-14350336*S^2
        +103320*S*U^7-2345344*S*U^5+17101312*S*U^3-21803008*S*U
        +3319*U^8-338240*U^6+3262592*U^4-9005056*U^2
        +2822400)/26880
p(8):   (42896*S^9+148471*S^8*U+207232*S^7*U^2-449936*S^7
        +172140*S^6*U^3-1309744*S^6*U+127928*S^5*U^4
        -1456224*S^5*U^2+1630656*S^5+26594*S^4*U^5
        -1041232*S^4*U^3+3649248*S^4*U+1264*S^3*U^6
        -528272*S^3*U^4+2963328*S^3*U^2-2295296*S^3+20508*S^2*U^7
        +20784*S^2*U^5+1664448*S^2*U^3-3259648*S^2*U-9720*S*U^8
        -87168*S*U^6+189888*S*U^4-1709056*S*U^2+905216*S+1887*U^9
        +13392*U^7+63072*U^5-262400*U^3+505984*U)/3360
p(9):   (4951359*S^10+13797398*S^9*U+7533921*S^8*U^2-56455164*S^8
        +9241848*S^7*U^3-125955744*S^7*U+13460470*S^6*U^4
        -50342544*S^6*U^2+230662944*S^6-19791660*S^5*U^5
        -107264544*S^5*U^3+363572160*S^5*U+9808242*S^4*U^6
        -10409160*S^4*U^4+153008928*S^4*U^2-389395584*S^4
        +9628504*S^3*U^7+90109344*S^3*U^5+285979008*S^3*U^3
        -357095936*S^3*U-14533893*S^2*U^8-110586576*S^2*U^6
        -218638752*S^2*U^4-292147968*S^2*U^2+218001024*S^2
        +7082310*S*U^9+54648864*S*U^7+125832384*S*U^5
        +44705280*S*U^3+143891712*S*U-1261699*U^10-10336476*U^8
        -25959072*U^6-22129792*U^4+19131264*U^2-21772800)/604800
p(10):  (27310220*S^11+51600727*S^10*U-119251786*S^9*U^2
        -348089600*S^9+293635543*S^8*U^3-302062632*S^8*U
        -64435816*S^7*U^4+745973808*S^7*U^2+1586798400*S^7
        -672382610*S^6*U^5-3075995872*S^6*U^3-73628448*S^6*U
        +1046203660*S^5*U^6+3607711248*S^5*U^4+957926400*S^5*U^2
        -2903936000*S^5-367369354*S^4*U^7+341917680*S^4*U^5
        +5474419104*S^4*U^3+1425471488*S^4*U-542942708*S^3*U^8
        -5214432688*S^3*U^6-13143377856*S^3*U^4-7999996928*S^3*U^2
        +1618188800*S^3+682849211*S^2*U^9+5531018592*S^2*U^7
        +13323576864*S^2*U^5+9148368896*S^2*U^3+1633199232*S^2*U
        -306966770*S*U^10-2534632848*S*U^8-6517108608*S*U^6
        -5368924160*S*U^4-740285184*S*U^2-419604480*S
        +51583283*U^11+450256312*U^9+1265403744*U^7
        +1228335104*U^5+283073152*U^3-59397120*U)/6652800

p(11):  (2399*S^12-4828*S^11*U-20746*S^10*U^2-32560*S^10
        +137060*S^9*U^3+123040*S^9*U-240735*S^8*U^4-92880*S^8*U^2
        +137232*S^8-1848*S^7*U^5-900480*S^7*U^3-713088*S^7*U
        +557172*S^6*U^6+2745120*S^6*U^4+1907136*S^6*U^2
        -143360*S^6-723096*S^5*U^7-2753856*S^5*U^5-862848*S^5*U^3
        +917504*S^5*U+173745*S^4*U^8-635040*S^4*U^6
        -5070240*S^4*U^4-4372480*S^4*U^2-138880*S^4
        +408980*S^3*U^9+3991680*S^3*U^7+11136384*S^3*U^5
        +8888320*S^3*U^3+1093120*S^3*U-442442*S^2*U^10
        -3809520*S^2*U^8-10285632*S^2*U^6-9175040*S^2*U^4
        -1800960*S^2*U^2-55296*S^2+184756*S*U^11+1624480*S*U^9
        +4650624*S*U^7+4759552*S*U^5+1308160*S*U^3+36864*S*U
        -29393*U^12-272272*U^10-844272*U^8-989184*U^6-353920*U^4
        -18432*U^2+2048)/1024

Degree of Faber Polynomial:  13
p(0):   13*U
p(1):   (13*(S^2+2*S*U+21*U^2-4))/4
p(2):   (13*(S^3+8*S^2*U+14*S*U^2-4*S+43*U^3-29*U))/3
p(3):   (13*(5*S^4+32*S^3*U+96*S^2*U^2-30*S^2+132*S*U^3
        -124*S*U+175*U^4-286*U^2+40))/8
p(4):   (13*(116*S^5+731*S^4*U+2064*S^3*U^2-808*S^3+3382*S^2*U^3
        -3900*S^2*U+3340*S*U^4-7416*S*U^2+1376*S+2247*U^5
        -7676*U^3+4024*U))/120
p(5):   (13*(1925*S^6+11474*S^5*U+30949*S^4*U^2-15430*S^4
        +49316*S^3*U^3-73192*S^3*U+50503*S^2*U^4-147060*S^2*U^2
```

```
        +35240*S^2+33530*S*U^5-156104*S*U^3+109904*S*U+12383*U^6
        -83414*U^4+114056*U^2-17280))/1440
p(6):   (13*(917*S^7+5036*S^6*U+12564*S^5*U^2-8244*S^5
        +18707*S^4*U^3-37105*S^4*U+18341*S^3*U^4-72300*S^3*U^2
        +23056*S^3+12138*S^2*U^5-78182*S^2*U^3+74992*S^2*U
        +5138*S*U^6-48616*S*U^4+92096*S*U^2-19008*S+1079*U^7
        -14273*U^5+45056*U^3-28032*U))/560
p(7):   (13*(28301*S^8+139104*S^7*U+308768*S^6*U^2-280140*S^6
        +410392*S^5*U^3-1151912*S^5*U+365930*S^4*U^4
        -2058132*S^4*U^2+917952*S^4+224368*S^3*U^5
        -2097328*S^3*U^3+2878336*S^3*U+92568*S^2*U^6
        -1313140*S^2*U^4+3599040*S^2*U^2-1057280*S^2+24472*S*U^7
        -471720*S*U^5+2208768*S*U^3-1977344*S*U+2769*U^8
        -78764*U^6+556544*U^4-1029632*U^2+225792))/16128
p(8):   (13*(1170400*S^9+4972173*S^8*U+9246312*S^7*U^2
        -12584640*S^7+10374764*S^6*U^3-45254880*S^6*U
        +8120280*S^5*U^4-69028032*S^5*U^2+46922880*S^5
        +4057230*S^4*U^5-62322720*S^4*U^3+132486336*S^4*U
        +1446872*S^3*U^6-35860800*S^3*U^4+150753408*S^3*U^2
        -68523520*S^3+535788*S^2*U^7-11214240*S^2*U^5
        +94652160*S^2*U^3-129325056*S^2*U-25464*S*U^8
        -3124800*S*U^6+28219392*S*U^4-86942208*S*U^2+29122560*S
        +18445*U^9-110688*U^7+4194624*U^5-20028416*U^3
        +21680640*U))/725760
p(9):   (13*(664251*S^10+2323142*S^9*U+3136639*S^8*U^2
        -7693656*S^8+2988232*S^7*U^3-22621376*S^7*U
        +2135910*S^6*U^4-25543456*S^6*U^2+31996416*S^6
        -37660*S^5*U^5-21877056*S^5*U^3+73741440*S^5*U
        +517478*S^4*U^6-8894800*S^4*U^4+67578432*S^4*U^2
        -55850496*S^4+402376*S^3*U^7+857536*S^3*U^5
        +44838912*S^3*U^3-89851904*S^3*U-499617*S^2*U^8
        -4412064*S^2*U^6+2236032*S^2*U^4-63572992*S^2*U^2
        +34745856*S^2+237510*S*U^9+1828416*S*U^7+6185856*S*U^5
        -13957120*S*U^3+32857088*S*U-41061*U^10-347544*U^8
        -785088*U^6-2559488*U^4+7661056*U^2-3763200))/537600
p(10):  (13*(15317575*S^11+39080432*S^10*U+16934694*S^9*U^2
        -190582780*S^9+59260133*S^8*U^3-375484107*S^8*U
        +3135634*S^7*U^4-193198872*S^7*U^2+867700680*S^7
        -85355220*S^6*U^5-564902372*S^6*U^3+1212610632*S^6*U
        +127054520*S^5*U^6+402561648*S^5*U^4+1010385360*S^5*U^2
        -1710146080*S^5-38640074*S^4*U^7+60471030*S^4*U^5
        +1099627584*S^4*U^3-1644612032*S^4*U-67874433*S^3*U^8
        -655832648*S^3*U^6-1628181336*S^3*U^4-2081035968*S^3*U^2
        +1314023680*S^3+79853356*S^2*U^9+662208492*S^2*U^7
        +1664249544*S^2*U^5+961874176*S^2*U^3+1232189952*S^2*U
        -34692790*S*U^10-293714388*S*U^8-781278528*S*U^6
        -703743520*S*U^4+187424256*S*U^2-322606080*S+5676573*U^11
        +50761997*U^9+147398064*U^7+148146624*U^5+60538112*U^3
        -96520320*U))/19958400
p(11):  (13*(191348773*S^12+101232512*S^11*U
        -430211968*S^10*U^2-2511726250*S^10
        +3012851196*S^9*U^3+617501692*S^9*U
        -4812817871*S^8*U^4-2359161570*S^8*U^2
        +11715590040*S^8-480461264*S^7*U^5
        -20476975856*S^7*U^3-8241189792*S^7*U
        +11203528744*S^6*U^6+56635278236*S^6*U^4
        +44012926896*S^6*U^2-22288816000*S^6
        -13553545304*S^5*U^7-52771516920*S^5*U^5
        -13932353280*S^5*U^3+8097670912*S^5*U
        +2657296279*S^4*U^8-15731018372*S^4*U^6
        -1087043717776*S^4*U^4-101936289152*S^4*U^2
        +154946598 40*S^4+7898683056*S^3*U^9
        +77797107920*S^3*U^7+223454533344*S^3*U^5
        +186985108992*S^3*U^3+34435831808*S^3*U
        -8063957704*S^2*U^10-71033788914*S^2*U^8
        -198180305904*S^2*U^6-184934729344*S^2*U^4
        -35991524352*S^2*U^2-8214319872*S^2
        +3260740604*S*U^11+29358286364*S*U^9+86740367808*S*U^7
        +92464474880*S*U^5+26508139520*S*U^3-1372939776*S*U
        -505685453*U^12-4793003930*U^10-15310125336*U^8
        -18623554688*U^6-6916994816*U^4-471773952*U^2
        +479001600))/479001600
p(12):  (2900*S^13-16169*S^12*U+28968*S^11*U^2-32872*S^11
        +105666*S^10*U^3+230692*S^10*U-505780*S^9*U^4
        -771320*S^9*U^2+94240*S^9+665973*S^8*U^5+317460*S^8*U^3
        -857160*S^8*U+225456*S^7*U^6+3598320*S^7*U^4
        +4487040*S^7*U^2+72960*S^7-1592052*S^6*U^7-8680056*S^6*U^5
        -8699040*S^6*U^3-139776*S^6*U+1631916*S^5*U^8
        +6826512*S^5*U^6+2439360*S^5*U^4-5306112*S^5*U^2
        -379904*S^5-146575*S^4*U^9+3100680*S^4*U^7
        +17755920*S^4*U^5+20536320*S^4*U^3+3539200*S^4*U
        -1030744*S^3*U^10-10450440*S^3*U^8-32807040*S^3*U^6
        -34433280*S^3*U^4-8529920*S^3*U^2-236544*S^3
        +932178*S^2*U^11+8691540*S^2*U^9+26753760*S^2*U^7
        +30352896*S^2*U^5+10017280*S^2*U^3+440832*S^2*U
        -352716*S*U^12-3364504*S*U^10-10913760*S*U^8
        -13810944*S*U^6-5770240*S*U^4-451584*S*U^2+12288*S
        +52003*U^13+520676*U^11+1813240*U^9+2568192*U^7
        +1309952*U^5+154112*U^3+1024*U)/1024

Degree of Faber Polynomial:  14
p(0):   14*U
p(1):   (7*(S^2+2*S*U+23*U^2-4))/2
p(2):   (7*(4*S^3+35*S^2*U+62*S*U^2-16*S+211*U^3-128*U))/6
p(3):   (7*(63*S^4+428*S^3*U+1402*S^2*U^2-384*S^2+1996*S*U^3
        -1664*S*U+2975*U^4-4288*U^2+528))/48
p(4):   (7*(42*S^5+287*S^4*U+878*S^3*U^2-296*S^3+1574*S^2*U^3
        -1560*S^2*U+1660*S*U^4-3192*S*U^2+512*S+1279*U^5-3752*U^3
        +1688*U))/20
p(5):   (7*(2185*S^6+14326*S^5*U+42581*S^4*U^2-17780*S^4
        +75004*S^3*U^3-93008*S^3*U+85067*S^2*U^4-207000*S^2*U^2
        +41560*S^2+62590*S*U^5-243856*S*U^3+143536*S*U+27127*U^6
        -151156*U^4+171304*U^2-21600))/720
p(6):   (19782*S^7+121721*S^6*U+341498*S^5*U^2-180984*S^5
        +574411*S^4*U^3-914920*S^4*U+637802*S^3*U^4
        -2014672*S^3*U^2+518496*S^3+480151*S^2*U^5
        -2478192*S^2*U^3+1905992*S^2*U+235718*S*U^6-1772600*S*U^4
        +2680400*S*U^2-444288*S+59957*U^7-621992*U^5+1547912*U^3
        -765312*U)/720
p(7):   (40397*S^8+227752*S^7*U+585244*S^6*U^2-407904*S^6
        +907288*S^5*U^3-1929152*S^5*U+942622*S^4*U^4
        -4014816*S^4*U^2+1371216*S^4+680088*S^3*U^5
        -4798848*S^3*U^3+4978240*S^3*U+337596*S^2*U^6
        -3541024*S^2*U^4+7334624*S^2*U^2-1637888*S^2+106792*S*U^7
        -1548480*S*U^5+5353024*S*U^3-3613696*S*U+16061*U^8
        -317856*U^6+1660496*U^4-2274816*U^2+376320)/1280
p(8):   (329588*S^9+1654869*S^8*U+3746340*S^7*U^2-3621264*S^7
        +5142340*S^6*U^3-15467184*S^6*U+4791300*S^5*U^4
        -28987152*S^5*U^2+13862976*S^5+3101982*S^4*U^5
        -31855728*S^4*U^3+46998720*S^4*U+1430524*S^3*U^6
        -22282800*S^3*U^4+66338112*S^3*U^2-20973824*S^3
        +470052*S^2*U^7-9777936*S^2*U^5+50346624*S^2*U^3
        -48599040*S^2*U+80616*S*U^8-2667312*S*U^6+20107008*S*U^4
        -40356096*S*U^2+9474048*S+9125*U^9-301008*U^7+3527232*U^5
        -11998720*U^3+8815104*U)/10368
p(9):   (28574721*S^10+122920682*S^9*U+230171619*S^8*U^2
        -337929156*S^8+269437032*S^7*U^3-1243218336*S^7*U
        +215232250*S^6*U^4-1964194416*S^6*U^2+1441388736*S^6
```

```
        +105521100*S^5*U^5-1910834016*S^5*U^3+4281774720*S^5*U
        +49503318*S^4*U^6-1145256600*S^4*U^4+5358117312*S^4*U^2
        -2608402176*S^4+17835016*S^3*U^7-412478304*S^3*U^5
        +3868684032*S^3*U^3-5638083584*S^3*U-4081707*S^2*U^8
        -171349104*S^2*U^6+1397865792*S^2*U^4-4873032192*S^2*U^2
        +1734529536*S^2+3204570*S*U^9+5721696*S*U^7
        +396556416*S*U^5-1828930560*S*U^3+2149475328*S*U
        -481801*U^10-5485764*U^8+20668992*U^6-292511488*U^4
        +688283136*U^2-203212800)/1036800
p(10):  (8485960*S^11+29101933*S^10*U+39323546*S^9*U^2
        -107075360*S^9+43923857*S^8*U^3-309747888*S^8*U
        +22520936*S^7*U^4-372483968*S^7*U^2+499322560*S^7
        -3979830*S^6*U^5-358469568*S^6*U^3+1161777728*S^6*U
        +21310140*S^5*U^6-73802688*S^5*U^4+1232152320*S^5*U^2
        -1034248960*S^5-4142886*S^4*U^7-30804640*S^4*U^5
        +822869696*S^4*U^3-1823707648*S^4*U-10051712*S^3*U^8
        -108326272*S^3*U^6-23572544*S^3*U^4-1588285952*S^3*U^2
        +885340160*S^3+11222409*S^2*U^9+93834048*S^2*U^7
        +296342976*S^2*U^5-335763456*S^2*U^3+1107337728*S^2*U
        -4712070*S*U^10-40990752*S*U^8-106810752*S*U^6
        -199914240*S*U^4+425636864*S*U^2-228352000*S+751317*U^11
        +6875088*U^9+20894016*U^7+14496256*U^5+65397248*U^3
        -102809600*U)/422400
p(11):  (3303782251*S^12+7024441844*S^11*U+4895281286*S^10*U^2
        -43863967840*S^10+17832459396*S^9*U^3-72026192576*S^9*U
        -18087411659*S^8*U^4-79043410272*S^8*U^2+216863671200*S^8
        -3348264344*S^7*U^5-130606732544*S^7*U^3
        +274202708736*S^7*U+46646307988*S^6*U^6
        +229332872000*S^6*U^4+440924221824*S^6*U^2
        -484647961600*S^6-52576455128*S^5*U^7
        -211305462912*S^5*U^5+53732002560*S^5*U^3
        -538866814976*S^5*U+8020375429*S^4*U^8
        -77235137984*S^4*U^6-455626805568*S^4*U^4
        -814250525696*S^4*U^2+480316418560*S^4
        +31560647844*S^3*U^9+314019099392*S^3*U^7
        +931439092992*S^3*U^5+717297647616*S^3*U^3
        +606044604416*S^3*U-30514717882*S^2*U^10
        -274852460256*S^2*U^8-791199686784*S^2*U^6
        -787550989312*S^2*U^4+10616552448*S^2*U^2
        -203423514624*S^2+11961211988*S*U^11
        +110207662400*S*U^9+335753465088*S*U^7
        +371406786560*S*U^5+136828553216*S*U^3
        -103689879552*S*U-1809575813*U^12-17539114208*U^10
        -57665502048*U^8-72833518592*U^6-26722096640*U^4
        -14775681024*U^2+13412044800)/273715200
p(12):  (2668026942*S^13-2073966101*S^12*U+8235191234*S^11*U^2
        -34820571000*S^11+30927123674*S^10*U^3+37650345848*S^10*U
        -130136089920*S^9*U^4-218624252168*S^9*U^2
        +161363744400*S^9+157256133349*S^8*U^5
        +50951250792*S^8*U^3-128918608968*S^8*U
        +69857234212*S^7*U^6+957013770448*S^7*U^4
        +1288540499952*S^7*U^2-316517739840*S^7
        -383400637748*S^6*U^7-2137244279632*S^6*U^5
        -2208149418336*S^6*U^3-265232710912*S^6*U
        +367496627134*S^5*U^8+1555099451760*S^5*U^6
        +441188260080*S^5*U^4-1631133584576*S^5*U^2
        +298975987200*S^5-16645452395*S^4*U^9
        +835344653008*S^4*U^7+4547720968752*S^4*U^5
        +5433598900864*S^4*U^3+1239932845312*S^4*U
        -239268990246*S^3*U^10-2456778632152*S^3*U^8
        -7916606208240*S^3*U^6-8656062024384*S^3*U^4
        -2184854356480*S^3*U^2-286512457728*S^3
        +206146107674*S^2*U^11+1963656383832*S^2*U^9
        +6222304577760*S^2*U^7+7334265609728*S^2*U^5
        +2522819080704*S^2*U^3+203144785922*S^2*U
        -75738488956*S*U^12-738497562088*S*U^10
        -2463832505472*S*U^8-3229970713408*S*U^6
        -1400831266816*S*U^4-118972569600*S*U^2+38477168640*S
        +10904201947*U^13+111525191752*U^11+398874834072*U^9
        +583809766528*U^7+308562330880*U^5+36457383936*U^3
        +5111976960*U)/444787200
p(13):  (4349*S^14-44362*S^13*U+210197*S^12*U^2-28548*S^12
        -225004*S^11*U^3+428496*S^11*U-981695*S^10*U^4
        -3040488*S^10*U^2-96360*S^10+3413410*S^9*U^5
        +7155280*S^9*U^3-175120*S^9*U-3330327*S^8*U^6
        -889020*S^8*U^4+10957320*S^8*U^2+926880*S^8
        -2433288*S^7*U^7-26766432*S^7*U^5-43771200*S^7*U^3
        -7576320*S^7*U+8656791*S^6*U^8+51636816*S^6*U^6
        +66620400*S^6*U^4+5631360*S^6*U^2-1016064*S^6
        -7082790*S^5*U^9-31366368*S^5*U^7-8049888*S^5*U^5
        +50830080*S^5*U^3+15579648*S^5*U-464321*S^4*U^10
        -23191740*S^4*U^8-118660080*S^4*U^6-168436800*S^4*U^4
        -54754560*S^4*U^2-1756160*S^4+5022004*S^3*U^11
        +53390480*S^3*U^9+186310080*S^3*U^7+242000640*S^3*U^5
        +95585280*S^3*U^3+5820416*S^3*U-3909269*S^2*U^12
        -39265512*S^2*U^10-135649800*S^2*U^8-187044480*S^2*U^6
        -89752320*S^2*U^4-9332736*S^2*U^2+50176*S^2
        +1352078*S*U^13+13907088*S*U^11+50393200*S*U^9
        +76074240*S*U^7+43577856*S*U^5+6623232*S*U^3+100352*S*U
        -185725*U^14-1998724*U^12-7720856*U^10-12835680*U^8
        -8617728*U^6-1756160*U^4-50176*U^2-4096)/2048

Degree of Faber Polynomial:  15
p(0):   15*U
p(1):   (15*(S^2+2*S*U+25*U^2-4))/4
p(2):   (5*(2*S^3+19*S^2*U+34*S*U^2-8*S+127*U^3-70*U))/2
p(3):   (5*(33*S^4+236*S^3*U+838*S^2*U^2-204*S^2+1228*S*U^3
        -920*S*U+2033*U^4-2620*U^2+288))/16
p(4):   (272*S^5+1997*S^4*U+6548*S^3*U^2-1936*S^3+12734*S^2*U^3
        -11040*S^2*U+14180*S*U^4-24032*S*U^2+3392*S+12317*U^5
        -31632*U^3+12448*U)/16
p(5):   (1635*S^6+11642*S^5*U+37637*S^4*U^2-13480*S^4
        +72268*S^3*U^3-76736*S^3*U+89629*S^2*U^4-186720*S^2*U^2
        +32160*S^2+71770*S*U^5-240032*S*U^3+121152*S*U+35739*U^6
        -169432*U^4+163488*U^2-17600)/64
p(6):   (11634*S^7+78847*S^6*U+244354*S^5*U^2-108048*S^5
        +455405*S^4*U^3-602990*S^4*U+561766*S^3*U^4
        -1472528*S^3*U^2+316032*S^3+471233*S^2*U^5
        -2018364*S^2*U^3+1289584*S^2*U+260806*S*U^6-1621888*S*U^4
        +2030848*S*U^2-279936*S+78115*U^7-662662*U^5+1353136*U^3
        -551664*U)/336
p(7):   (32359*S^8+204552*S^7*U+592772*S^6*U^2-332328*S^6
        +1040952*S^5*U^3-1766800*S^5*U+1227882*S^4*U^4
        -4164392*S^4*U^2+1142112*S^4+1011000*S^3*U^5
        -5667616*S^3*U^3+4686464*S^3*U+578564*S^2*U^6
        -4792376*S^2*U^4+7886656*S^2*U^2-1408576*S^2+214536*S*U^7
        -2443984*S*U^5+6646912*S*U^3-3562624*S*U+39463*U^8
        -600824*U^6+2447456*U^4-2632000*U^2+345600)/768
p(8):   (245392*S^9+1413731*S^8*U+3722504*S^7*U^2-2747152*S^7
        +5968404*S^6*U^3-13513088*S^6*U+6488920*S^5*U^4
        -29603712*S^5*U^2+10765632*S^5+4986898*S^4*U^5
        -38113904*S^4*U^3+42324384*S^4*U+2746040*S^3*U^6
        -31441552*S^3*U^4+70328832*S^3*U^2-16813312*S^3
        +1055508*S^2*U^7-16736736*S^2*U^5+62988480*S^2*U^3
        -45780224*S^2*U+250584*S*U^8-5417184*S*U^6+30582720*S*U^4
        -45197312*S*U^2+8000512*S+28899*U^9-804912*U^7
        +6544032*U^5-16394752*U^3+8933888*U)/5376
p(9):   (8412327*S^10+42876854*S^9*U+98720355*S^8*U^2
```

```
        -101481012*S^8+139606920*S^7*U^3-445876224*S^7*U
        +134271550*S^6*U^4-869567040*S^6*U^2+443564640*S^6
        +90446532*S^5*U^5-1014488160*S^5*U^3+1593155904*S^5*U
        +45838158*S^4*U^6-763778280*S^4*U^4+2458973088*S^4*U^2
        -829569024*S^4+16797448*S^3*U^7-381585984*S^3*U^5
        +2133787008*S^3*U^3-2204412416*S^3*U+3356379*S^2*U^8
        -133546656*S^2*U^6+1066581408*S^2*U^4-2320566528*S^2*U^2
        +580760064*S^2+866550*S*U^9-23206752*S*U^7
        +313830720*S*U^5-1129896960*S*U^3+888210432*S*U-4465*U^10
        -2682372*U^8+37334112*U^6-221577472*U^4+359944704*U^2
        -73156608)/193536
p(10):  (190269170*S^11+819323947*S^10*U+1555389314*S^9*U^2
        -2448976520*S^9+1900427323*S^8*U^3-9112802262*S^8*U
        +1484821124*S^7*U^4-14986789392*S^7*U^2+11724049920*S^7
        +749963590*S^6*U^5-15401444392*S^6*U^3
        +36125444352*S^6*U+477321940*S^5*U^6-9179089152*S^5*U^4
        +49128126720*S^5*U^2-25243531520*S^5+82086686*S^4*U^7
        -4119075780*S^4*U^5+38408772864*S^4*U^3-60149740672*S^4*U
        -46130678*S^3*U^8-1961340208*S^3*U^6+15670754304*S^3*U^4
        -61173018368*S^3*U^2+22986475520*S^3+76024751*S^2*U^9
        +378650712*S^2*U^7+6743656704*S^2*U^5-28720382464*S^2*U^3
        +37027425792*S^2*U-29242070*S*U^10-291669048*S*U^8
        +78350592*S*U^6-8459118080*S*U^4+20016070656*S*U^2
        -6286233600*S+4602503*U^11+41512042*U^9+191372544*U^7
        -669824896*U^5+3795308032*U^3-3772800000*U)/5322240
p(11):  (117480979*S^12+386620396*S^11*U+559277222*S^10*U^2
        -1595147180*S^10+671783260*S^9*U^3-4493951384*S^9*U
        +135211821*S^8*U^4-6093655356*S^8*U^2+8198460160*S^8
        +86286040*S^7*U^5-5649893664*S^7*U^3+19309295104*S^7*U
        +496125012*S^6*U^6+53501672*S^6*U^4+23841616384*S^6*U^2
        -19599923200*S^6-465761512*S^5*U^7-2728684176*S^5*U^5
        +13098728960*S^5*U^3-37783304704*S^5*U+53617245*S^4*U^8
        -1044908056*S^4*U^6+204537344*S^4*U^4-37353573632*S^4*U^2
        +21508129280*S^4+294559804*S^3*U^9+2930739936*S^3*U^7
        +10250914304*S^3*U^5-5886476288*S^3*U^3+32635787264*S^3*U
        -270258714*S^2*U^10-2491566396*S^2*U^8-7198809600*S^2*U^6
        -11024495104*S^2*U^4+15204142080*s^2*U^2-9243621376*S^2
        +102787212*S*U^11+968456808*S*U^9+3056693760*S*U^7
        +3054819840*S*U^5+5079062528*S*U^3-7217410048*S*U
        -15178365*U^12-150367404*U^10-507868416*U^8-690374912*U^6
        +84238848*U^4-1421694976*U^2+662323200)/4730880
p(12):  (23471267232*S^13+40127913193*S^12*U+67357500380*S^11*U^2
        -327431575200*S^11+106304376770*S^10*U^3
        -470047833952*S^10*U-274663260372*S^9*U^4
        -1048120022912*S^9*U^2+1735414485120*S^9
        +321647175511*S^8*U^5-233862228288*S^8*U^3
        +2319603715392*S^8*U+177551043640*S^7*U^6
        +2080858845760*S^7*U^4+5214569051136*S^7*U^2
        -4361135819520*S^7-794221222436*S^6*U^7
        -4544765264704*S^6*U^5-3912659275008*S^6*U^3
        -6400166785792*S^6*U+712619483992*S^5*U^8
        +3037895333760*S^5*U^6+838561608960*S^5*U^4
        -7770073372160*S^5*U^2+5446824176640*S^5
        -227082713*S^4*U^9+1905284357632*S^4*U^7
        +100365531651184*S^4*U^5+11166162853888*S^4*U^3
        +8822630497792*S^4*U-478555871508*S^3*U^10
        -4978697175712*S^3*U^8-16452310683648*S^3*U^6
        -18956299669248*S^3*U^4-2333354991616*S^3*U^2
        -3426875301888*S^3+393767575682*S^2*U^11
        +3830151330720*S^2*U^9+12484484337408*S^2*U^7
        +15256845798656*S^2*U^5+6012640478208*S^2*U^3
        -2130228117504*S^2*U-140601872164*S*U^12
        -1400639908096*S*U^10-4802731260288*S*U^8
        -6520896729088*S*U^6-2886893403136*S*U^4
        -7580879585528*S*U^2+556521062400*S+19779555193*U^13
        +206543651392*U^11+758182311744*U^9+1146228811264*U^7
        +630245658112*U^5+38354411520*U^3
        +140905267200*U)/1660538880
p(13):  (277643075437*S^14-432683201762*S^13*U
        +3183464957863*S^12*U^2-3522429860768*S^12
        -2654408019924*S^11*U^3+3627868053312*S^11*U
        -1433329737564 3*S^10*U^4-46864367284192*S^10*U^2
        +16100985035200*S^10+45139385236386*S^9*U^5
        +967111920404 48*S^9*U^3+110363218845 44*S^9*U
        -400791589561 37*S^8*U^6+124303046400*S^8*U^4
        +177071711966400*S^8*U^2-35034436478080*S^8
        -349382115457 52*S^7*U^7-365855415277184*S^7*U^5
        -614727441757696*S^7*U^3-149906588015104*S^7*U
        +108206174301631*S^6*U^8+658001719505984*S^6*U^6
        +870738942921088*S^6*U^4+374236910128 64*S^6*U^2
        +554202965888 00*S^6-828285390398 70*S^5*U^9
        -366559575970560*S^5*U^7-41357020877568*S^5*U^5
        +754215228776448*S^5*U^3+287841932534784*S^5*U
        -947140336728 3*S^4*U^10-313009787362208*S^4*U^8
        -1573374054129280*S^4*U^6-2297092797837312*S^4*U^4
        -767114331565568*S^4*U^2-779007191572 48*S^4
        +607868167840 44*S^3*U^11+655521024957760*S^3*U^9
        +2342841398977024*S^3*U^7+3150577891978752*S^3*U^5
        +1296378915321856*S^3*U^3+540291979018 24*S^3*U
        -453824758852 01*S^2*U^12-465085960461408*S^2*U^10
        -1649222812603200*S^2*U^8-2350983048646400*S^2*U^6
        -1170703018987008*S^2*U^4-1270923155865 60*S^2*U^2
        +151463938129 92*S^2+15276352951678*S*U^13
        +1603727281994 24*S*U^11+5960789394876 16*S*U^9
        +9283424689960 96*S*U^7+5504863702476 80*S*U^5
        +8515622477824 0*S*U^3+4817739350016*S*U
        -2052233585867*U^14-225278418718 08*U^12
        -8915683315052 8*U^10-152617754171264*U^8
        -1058506349245 44*U^6-220681054658 56*U^4
        -4361959342 08*U^2-697426329600)/46495088640
p(14):  (134*S^15-25499*S^14*U+310534*S^13*U^2+42352*S^13
        -1024933*S^12*U^3-107770*S^12*U+605150*S^11*U^4
        -2941536*S^11*U^2-480384*S^11+4121117*S^10*U^5
        +15550964*S^10*U^3+4135824*S^10*U-10684674*S^9*U^6
        -27650480*S^9*U^4-316800*S^9*U^2+1450240*S^9
        +7443579*S^8*U^7-4162158*S^8*U^5-57031920*S^8*U^3
        -19844880*S^8*U+9565842*S^7*U^8+93213120*S^7*U^6
        +184124160*S^7*U^4+62008320*S^7*U^2+445440*S^7
        -22567545*S^6*U^9-145835976*S^6*U^7-226616544*S^6*U^5
        -37181760*S^6*U^3+20186880*S^6*U+14736722*S^5*U^10
        +66038544*S^5*U^8-8338176*S^5*U^6-221760000*S^5*U^4
        -119669760*S^5*U^2-5160960*S^5+3556553*S^4*U^11
        +76303370*S^4*U^9+379573920*S^4*U^7+631683360*S^4*U^5
        +312480000*S^4*U^3+25385472*S^4*U-11933558*S^3*U^12
        -133568864*S^3*U^10-513152640*S^3*U^8-793605120*S^3*U^6
        -440294400*S^3*U^4-57802752*S^3*U^2-229376*S^3
        +8164471*S^2*U^13+87927060*S^2*U^11+336782160*S^2*U^9
        +547039680*S^2*U^7+349896960*S^2*U^5+64963584*S^2*U^3
        +1576960*S^2*U-2600150*S*U^14-28687568*S*U^12
        -114820992*S*U^10-201527040*S*U^8-148270080*S*U^6
        -36126720*S*U^4-1605632*S*U^2-32768*S+334305*U^15
        +3848222*U^13+16325712*U^11+31174000*U^9+26146560*U^7
        +7967232*U^5+544768*U^3+2048*U)/2048
```

Bibliography

[1] W.E. Arnoldi, *The principle of minimized iterations in the solution of the matrix eigenvalue problem*, Quart. Appl. Math. **9**, 17-29.

[2] K.E. Atkinson, *An Introduction to Numerical Analysis*, John Wiley and Sons, Inc., New York, 1989.

[3] R. Barrett, M. Berry, T. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhoui, R. Pozo, C. Rommie and H. van der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*.

[4] J. Bartolomeo and M. He, *The Faber polynomials for a regular m-star*, Math. Comp. **62** (1994), 277-287.

[5] R.L. Burden and J.D. Faires, *Numerical Analysis*, Prindle, Weber and Schmidt, Boston, 1985.

[6] E.W. Cheney, *Introduction to Approximation Theory*, McGraw-Hill, New York, 1966.

[7] E.W. Cheney and K.H. Price, *Minimal Projections*, Approximation Theory, pp. 261-289, (A.Talbot, eds.), Academic Press, London, 1970.

[8] C.K. Chui, J. Stöckler and J.D. Ward, *A Faber series approach to cardinal interpolation*, Math. Comp. **58** (1992), 255-273.

[9] J.P. Coleman, *Complex polynomial approximation by the Lanczos τ-method: Dawson's integral*, J. Comput. Appl. Math. **20** (1987), 137-151.

[10] J.P. Coleman, *Polynomial approximations in the complex plane*, J. Comput. Appl. Math. **18** (1987), 193-211.

[11] J.P. Coleman, *Private communication.*

[12] J.P. Coleman and N.J. Myers, *The Faber polynomials for annular sectors*, accepted for publication in Math. Comp. to be published Jan 1995.

[13] J.P. Coleman and R.A. Smith, *The Faber polynomials for circular sectors*, Math. Comp. **49** (1987), 231-241.

[14] J.P. Coleman and R.A. Smith, *Supplement to the Faber polynomials for circular sectors*, Math. Comp. **49** (1987), S1-S4

[15] J.H. Curtiss, *Faber polynomials and Faber series*, Amer. Math. Monthly **78** (1971), 577-596.

[16] T.A. Driscoll, *Schwarz-Christoffel toolbox user's guide, version 1.0*, Cornell University report.

[17] M. Eiermann, *On semiiterative methods generated by Faber polynomials*, Numer. Math. **56** (1989), 139-156.

[18] M. Eiermann, X. Li and R.S. Varga, *On hybrid semiiterative methods*, SIAM J. Numer. Anal., **26** (1989), 152-168.

[19] M. Eiermann and W. Niethammer, *On the construction of semiiterative methods*, SIAM J. Numer. Anal., **20** (1983), 1153-1160.

[20] M. Eiermann, W. Niethammer and R.S. Varga, *A study of semi-iterative methods for non-symmetric systems of linear equations*, Numer. Math. **47** (1985), 505-533.

[21] M. Eiermann and R.S. Varga, *Zeros and local extreme points of Faber polynomials associated with hypocycloidal domains*, ETNA **1** (1993), 49-71

[22] S.W. Ellacott, *Computation of Faber series with application to numerical polynomial approximation in the complex plane*, Math. Comp. **40** (1983), 575-587.

[23] S.W. Ellacott, *On Faber polynomials and Chebyshev polynomials*, Approximation Theory IV (C.K. Chui, L.L. Schumaker and J.D. Ward, eds), Academic Press, New York, 1983.

[24] S.W. Ellacott, *On the Faber transform and efficient numerical rational approximation*, SIAM J. Numer. Anal. **20** (1983), 989-1000.

[25] S.W. Ellacott and M.H. Gutknecht, *The polynomial Carathéodory-Fejér approximation method for Jordan regions*, IMA J. Numer. Anal. **3** (1983), 207-220.

[26] S.W. Ellacott and E.B. Saff, *Computing with the Faber transform*, Rational Approximation and Interpolation, pp. 412-418, (P.R. Graves-Morris, E. Saff and R.S. Varga, eds.), Lecture Notes in Mathematics 1105, Springer Verlag, Berlin, 1984.

[27] S.W. Ellacott and E.B. Saff, *On Clenshaw's method and a generalisation to Faber series*, Numer. Math. **52** (1988), 499-509.

[28] G.H. Elliott, *The construction of Chebyshev approximations in the complex plane*, Ph.D. Thesis, University of London, 1978.

[29] H.C. Elman, *Iterative Methods for Linear Systems*, Lancaster Summer School notes, 1992.

[30] H.C. Elman, Y. Saad and P.E. Saylor, *A hybrid Chebyshev Krylov subspace algorithm for solving nonsymmetric systems of linear equations.*, SIAM J. Sci. Stat. Comput., **7** (1986), 840-855.

[31] H.C. Elman and R. Streit, *Polynomial iteration for nonsymmetric indefinite linear systems*, in Numerical Analysis, Lecture Notes in Mathematics 1230, J.P. Hennert, ed., Springer Verlag, Berlin, 1986.

[32] V. Faber and T. Manteuffel, *Necessary and sufficient conditions for the existence of a conjugate gradient method*, SIAM J. Numer. Anal., **21** (1984), 352-362.

[33] V. Faber and T. Manteuffel, *Orthogonal error methods*, SIAM J. Numer. Anal., **24**, 170-187.

[34] L. Fox and I.B. Parker, *Chebyshev Polynomials in Numerical Analysis*, Oxford University Press, London, 1968.

[35] R.W. Freund, *Krylov Subspace Methods for Complex Non-Hermitian Linear Systems*, RIACS Technical Report, May 1991.

[36] R.W. Freund and N.M. Nachtigal, *QMR: a quasi-minimal residual method for nonHermitain linear systems*, Numer. Math., **60** (1991), 315-339.

[37] R.W. Freund, G.H. Golub and N.M. Nachtigal, *Iterative Solution of Linear Systems*, Acta Numerica, pp. 57-100, Cambridge Uni. Press, 1991.

[38] D. Gaier, *Lectures on Complex Approximation*, Birkhäuser, Boston, 1987.

[39] K. Gatermann, C. Hoffmann and G. Opfer, *Explicit Faber polynomials on circular sectors*, Math. Comp. **58** (1992), 241-253.

[40] K. Gatermann, C. Hoffmann and G. Opfer, *Supplement to explicit Faber polynomials on circular sectors*, Math. Comp. **58** (1992), S1-S6.

[41] K.O. Geddes, *Near minimax polynomial approximation in an elliptical region*, Siam J. Numer. Anal. **15** (1978), 1225-1233.

[42] K.O. Geddes, *Chebyshev nodes for interpolation on a class of ellipses*, Theory of Approximation with Applications (Proc. Conf. Univ. Calgary Alta.), Academic Press, New York, 1975.

[43] K.O. Geddes and J.C. Mason, *Polynomial approximation by projections on the unit circle*, Siam J. Numer. Anal. **12** (1975), 111-120.

[44] G. Golub and C. Van Loan, *Matrix Computations*, second edition, The John Hopkins University Press, Baltimore, 1989.

[45] J.F. Grcar, *Operator Coefficient Methods for Linear Equations*, Sandia Report: SAND 89-8691, 1989.

[46] U. Grothkopf and G. Opfer, *Complex Chebyshev polynomials on circular sectors with degree six or less*, Math. Comp. **39** (1982), 599-615.

[47] M.H. Gutknecht, *Variants of BICGSTAB for matrices with complex spectrum*, SIAM J. Sci. Comput. **14** (1993), 1020-1033.

[48] P. Henrici, *Applied and Computational Complex Analysis*, vol 1, Wiley, New York, 1974.

[49] P. Henrici, *Applied and Computational Complex Analysis*, vol 3, Wiley, New York, 1986.

[50] M.R. Hestenes and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, J. Res. Natl. Bur. Stand., **49** (1952), 409-436.

[51] E. Kreyszig, *Advanced Engineering Mathematics*, John Wiley and Sons, Inc., New York, 1983.

[52] T. Kövari and C. Pommerenke, *On Faber polynomials and Faber expansions*, Math. Zeitschr. **99** (1967), 193-206.

[53] X. Li, *An adaptive method for solving nonsymmetric linear systems involving applications of SCPACK*, J. Comp and Appl. Maths., **44** (1992), 351-370.

[54] J.C. Mason, *Orthogonal polynomial approximation methods in numerical analysis*, Approximation Theory, pp. 7-34, (A. Talbot, eds.), Academic Press, London, 1970.

[55] T.A. Manteuffel, *The Tchebychev iteration for nonsymmetric linear systems*, Numer. Math. **28** (1977), 307-327.

[56] T.A. Manteuffel, *Adaptive procedure for estimating parameters for the nonsymmetric Tchebychev iteration*, Numer. Math. **31** (1978), 183-208.

[57] A.I. Markushevich, *Theory of Functions of a Complex Variable*, Chelsea, New York, 1977.

[58] N.J. Myers, *An iterative method for the solution of linear systems using the Faber polynomials for annular sectors*, in volume 1 of the preliminary proceedings of the Colorado Conference on Iterative Methods, Breckenridge, April 1994.

[59] N.M. Nachtigal, S.C. Reddy, L.N. Trefethen, *How fast are nonsymmetric matrix iterations?*, SIAM J. Matrix Anal. Appl. **13** (1992), 778-795.

[60] N.M. Nachtigal, L. Reichel and L.N. Trefethen, *A hybrid GMRES algorithm for nonsymmetric linear systems*, Siam J. Matrix Anal. Appl., **13**, #3 (1992), 796-825.

[61] G. Opfer, *Private communication* , given at the Dundee Conference on Numerical Analysis, 1993.

[62] C.C. Paige and M.A. Saunders, *Solution of sparse indefinite systems of linear equations*, SIAM J. Numer. Anal. **12** (1975), 617-629.

[63] N. Papamichael, M.J. Soares and N.S. Stylianopoulos, *A numerical method for the computation of Faber polynomials for starlike domains*, IMA J. Numer. Anal. **13** (1993), 181-193.

[64] C. Pommerenke, *Über die Faberschen Polynome schlichter Funktionen*, Math. Zeitschr. **85** (1964), 197-208.

[65] C. Pommerenke, *Konforme Abbidldung und Fekete-Punkte*, Math. Z. **89** (1965), 422-438.

[66] A.P. Prudnikov, Yu.A. Brychkov and O.I. Marichev, *Integrals and Series Vol I: Elementary Functions*, Gordon and Breach, New York, 1988.

[67] J. Radon, *Über die Randwertaufgaben beim logarithmischen Potential*, Sitz.-Ber. Akad. Wiss. Wien. Math.-naturw. **128** (1919), 1123-1167.

[68] L. Reichel and L.N. Trefethen, *Eigenvalues and pseudo-eigenvalues of Toeplitz matrices*, Lin. Alg. Applics. 162–164: 153–185, 1992.

[69] Y. Saad, *Least-squares polynomials in the complex plane and their use for solving nonsymmetric linear systems*, SIAM J. Numer. Anal., **24** (1987), 155-169.

[70] P.E. Saylor and D.C. Smolarski, *Implementation of an adaptive algorithm for Richardson's method*, Linear Algebra Appl., **154-156** (1991), 615-646.

[71] J.R. Shewchuk, *An introduction to the conjugate gradient method without the agonizing pain*, Pittsburgh University Report CMU-CS-94-125, 1994.

[72] V.I. Smirnov and N.A. Lebedev, *Functions of a Complex Variable, Constructive Theory*, Iliffe, London, 1968.

[73] G.D. Smith, *Numerical solution of partial differenial equations: Finite difference methods*, Oxford Applied Maths and Computing Science Series, second edition, 1978.

[74] D.C. Smolarski and P.E. Saylor, *An optimum iterative method for solving any linear system with a square matrix.*, BIT **28** (1988), 163-178.

[75] P. Sonneveld, *CGS, a fast Lanczos-type solver for nonsymmetric linear systems.*, SIAM J. Sci. Stat. Comput. **10** (1989), 36-52.

[76] G. Starke and R.S. Varga, *A hybrid Arnoldi-Faber iterative method for nonsymmetric systems of linear equations*, Numer. Math. **64** (1993).

[77] L.N. Trefethen, *Numerical computation of the Schwarz-Christoffel transformation*, SIAM J. Sci. Stat. Comput. **1** (1980), 82-102.

[78] L.N. Trefethen, *Pseudospectra of matrices*, In D.F. Griffiths and G.A. Watson, *Numerical Analysis 1991*, Longman, 234-266, 1992.

[79] H.A. Van der Vorst, *Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems.*, SIAM J. Sci. Stat. Comput. **13** (1992), 631-644.

[80] J.L. Walsh, *Interpolation and Approximation by Rational Functions in the Complex Domain*, 5th ed., Amer. Math. Soc. Colloq. Publ., Vol 20, 1969.