A Generalization of the Wiener Rational Basis Functions on Infinite Intervals

by

Akil Narayan

B.Sc., Northwestern University; Evanston, IL, 2003

M.A., Brown University; Providence, RI, 2005

A dissertation submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy
in The Division of Applied Mathematics at Brown University

PROVIDENCE, RHODE ISLAND

May 2009

This dissertation by Akil Narayan is accepted in its present form
by The Division of Applied Mathematics as satisfying the
dissertation requirement for the degree of Doctor of Philosophy.

Date_____ _____

Jan S. Hesthaven, Ph.D., Advisor

Recommended to the Graduate Council

Date_____ _____

Johnny Guzmán, Ph.D., Reader

Date_____ _____

Chi-Wang Shu, Ph.D., Reader

Approved by the Graduate Council

Date_____ _____

Sheila Bonde, Dean of the Graduate School

Vitae

Akil Narayan was born on June 19, 1982 in Lancaster, Ohio. He graduated from Northwetsern University in 2003 *summa cum laude* and with honors with degrees in Electrical Engineering and Applied Mathematics.

Acknowledgements

My first thanks go to my advisor, Jan Hesthaven. His guidance has been integral to any success of mine. Thank you for your extreme patience and understanding during these years. Your sharp acumen has steered me through many a murky intellectual trouble, and your easygoing attitude more welcoming than you can imagine.

I cannot thank Chi-Wang Shu and Johnny Guzmán enough for their support and for agreeing to be readers for my thesis.

To my colleagues, with whom I have grown during my time at Brown University, I cannot express my thanks enough. I cannot underestimate the importance of the innumerable debates, both academic and social; the sushi parties and Vermonster outings certainly didn't hurt, either. Any to all my remaining friends, who don't qualify as colleages, my gratefulness to you for all non-academic conversations and events can not be overstated.

My mother and father have always supported me, through my wise decisions as well as my mistakes. They have given me everything and have asked for nothing in return. I can only hope that anything I have accomplished is reward for their unspoken sacrifices. I seem to count few individuals as blessed as I with such a gracious and caring extended family, who have always welcomed me into their homes for a weekend, a fortnight, or even many months. A heartfelt thanks to all of you.

Abstract of " A Generalization of the Wiener Rational Basis Functions on Infinite Intervals " by Akil Narayan, Ph.D., Brown University, May 2009

This thesis concerns the formulation and derivation of a generalization of a collection of basis functions originally devised by Norbert Wiener for function approximation over the entire real line. The generalized basis set may be parameterized by the polynomial rate of decay of the basis functions at infinity.

In order to explore the possible utility of the generalized basis set, we first investigate the applicability of the fast Fourier transform algorithm to Jacobi polynomial expansions. We show that such applicability is robust (efficient and accurate) for certain classes of Jacobi polynomials. In addition, we explore the extent to which Jacobi-Gauss-type nodal sets serve as Lebesgue-optimal interpolation sets. We extend our results to two dimensional triangular simplices to obtain the best-known Lebesgue constants to date to the author's knowledge.

Wiener's generalized basis over the infinite interval is a direct mapping of a generalized Fourier series over the finite interval. Using the properties of Jacobi polynomials and the generalized Fourier series, we are able to show that the generalized Wiener basis set is L^2 orthonormal for any choice of the decay parameter. In addition, we show various other useful properties including fast Fourier transform applicability, efficient decay parameter modification, and sparsity and spectral properties of the stiffness matrix.

We conclude our investigation with a few examples pertaining to function approximation and solutions to partial differential equations. Although we do not claim to have developed a panacea for spectral expansions on infinite intervals, we present the generalized Wiener basis set as a strong competitor to existing methods.

Contents

List of Tables

List of Figures

Chapter One

Introduction

The expansion or approximation of a function via a sum of standarized functions has been a well-utilized tool in mathematics. The term 'Fourier expansion', or 'generalized Fourier expansion' is frequently used to imply replacing a function $f(x)$ with a linear sum of some other functions $\phi_n(x)$:

$$f(x) \simeq f_N(x) = \sum_{n=0}^{N} \hat{f}_n \phi_n(x). \tag{1.1}$$

The accuracy of such an expansion depends on a variety of factors, one of which is the parameter N, which is usually called the order of the expansion. The functions $\phi_n(x)$ are basis functions, chosen in some fashion, and are usually complete in some function space. The modal coefficient $\hat{f}_n$ are f-dependent and can be interpreted as the 'quantity' of the function $\phi_n(x)$ that is present in $f(x)$.

The success of the expansion (1.1) is largely dependent on the choice of $\phi_n(x)$. When x is a variable on a finite interval, very popular (and effective) options are trigonometric polynomials or Jacobi polynomials. If the function f is defined for x on an infinite interval (e.g. either $[0, \infty)$ or $\mathbb{R}$), the available methods become more scant and less effective. Hermite and Laguerre expansions are frequently used on infinite intervals, but the main disadvantage for these methods is that the quality of the expansion is generally poor if the function decays slower than exponentially; this slow convergence makes it necessary to include many more degrees of freedom for a given error tolerance, thus making approximation via these methods expensive.

A remedy for such a problem was originally employed by Christov [27] for solving a differential equation on an infinite interval. Christov borrowed functions first proposed by Wiener [69], which are more or less a simple mapping of the Fourier series, and only decay algebraically at infinity. Boyd later developed a slightly different method for algebraically-decaying functions [15].

In this work we will generalize the algebraically-decaying functions of Wiener. In addition, we shall explore many of their properties including recurrence formulae, orthogonality, completeness, interpolation, numerical quadrature, utility of the Fast Fourier Transform (FFT), and properties of derivatives. Furthermore, we will provide a comparison between these methods and the existing Hermite and Laguerre spectral methods, and other methods for function approximation on an infinite interval.

Chapter 2 concentrates on compiling the requisite knowledge from spectral methods in general and in particular, the theory of Jacobi orthogonal polynomials. Chapter 3 concerns a novel numerical approach for the robust application of Fast Fourier Transform (FFT) to general classes of Jacobi polynomials expansions. This result is requisite for application of the FFT to the generalization of Wiener's basis over infinite intervals. Chapter 4 introduces some new and interesting observations regarding one- and two-dimensional interpolation problems and the close connection to Jacobi-Gauss quadrature rules. Chapter 5 contains the derivation of generalized Fourier series bases for spectral expansion on finite intervals interval. This basis set is integral for the work in Chapter 6 focusing on the derivation of Wiener's generalized rational functions. In Chapter 7 we review existing spectral expansions over the infinite interval so that in Chapter 8 we can show numerical examples of expansions using the generalized Wiener basis. Finally, Section 9 summarizes the main points and presents an outlook for future work and developments.

1.1 Existing Spectral Methods

Here we will give a short survey of results that have already been presented in the literature. All these methods will be reintroduced and more properly explored in

Chapter 7 with an emphasis on comparing them qualitatively to the properties of the generalized Wiener basis set; Chapter 8 will then present quantitative comparisons. When dealing with spectral methods on unbounded domains, one essentially has a choice between two approaches: domain truncation and basis functions over the entire infinite interval.

1.1.1 Domain truncation

This method is very simple: there exist many excellent ways to spectrally expand a smooth function on finite intervals (see e.g. [36] or [43]). One may simply stretch the finite domain to $[-L, L]$ and physically truncate the solution for $|x| > L$. For example, we can think of using the canoncial Fourier basis

$$\phi_k(x) = e^{ikx\pi/L}, \qquad k \in \mathbb{Z}, \quad x \in [-L, L],$$

for as large a value of L as we might like. If L is properly chosen, this method can yield satisfactory results. While such methods are mathematically unsatisfying for solving problems on infinite domains, in practice they perform no worse than genuine spectral methods if care is taken regarding the choice of L [29].

However, it is often difficult to choose the parameter L, and for evolution equations where the nature of the solution may change dramatically over the course of the computation, one must regularly perform some ad-hoc methods for resampling, and determine a robust way to augment L on the fly. However, this method does simplify the implementation in the sense that finite-domain solvers can essentially be recycled. Naturally, one must be willing to accept the error introduced by effectively assuming that the approximation, which should exist over the entire real line $x \in \mathbb{R}$, is zero

outside $|x| \leq L$.

1.1.2 Orthogonal polynomials

On the (semi)infinite interval there are some canonical spectral expansion bases utilizing orthogonal polynomials. For the semi-infinite interval $\mathbb{R}^+$ the Laguerre polynomials L_n form an appropriate basis. We outline some of the properties of the Laguerre polynomial basis in appendix A.3. The Laguerre basis has indeed been used to solve differential equations on semi-infinte intervals in a variety of applications (see e.g. [58] and [6]).

However, there are some significant challenges that plague the Laguerre spectral method. Firstly, if one employs Laguerre polynomials, then it is necessary to pose Galerkin problems in a space weighted by an exponential e^{-x}. In addition, it has been noted [36] that the Laguerre system of orthogonal polynomials does not exhibit resolution properties up to par compared with other systems of orthogonal polynomials. In addition, the Laguerre polynomials cannot be used for pointwise estimation of functions far from the origin. If one uses the Laguerre *functions*, which decay exponentially, then one is automatically excluded from faithfully expanding functions that decay slower than exponentially.

On the infinite interval $\mathbb{R}$, the Hermite polynomials are the natural family of orthogonal polynomials to choose. Appendix A.2 details their properties. However, the Hermite polynomials suffer from the very same disadvatages as the Lageurre polynomials/functions: ill-behaved polynomials and quadrature formulae, subpar resolution, and poor pointwise properties far from the origin.

As with domain truncation, there is a scaling parameter L that one must choose to optimize the spectral expansion. There have been quite a few attempts to optimize the value of this parameter [61], [14]. However, no spectral method on the (semi)infinite interval can escape this need for scaling.

The basic theory for these families of orthogonal polynomials can be found in e.g. [60] and [34].

1.1.3 Rational approximation

An alternative to expanding a function in Hermite or Laguerre polynomials/functions is to use a mapping to translate the Jacobi orthogonal polynomials over the interval $[-1, 1]$ to the half-line $[0, \infty)$ to real line $\mathbb{R}$. Many such coordinate transformations have been studied in the literature [43]. In particular, the 'rational Chebyshev functions' on the semi-infinite interval [11], [39] and on the infinite interval [39], [12], [14] have been widely used. There are, of course, infinitely many ways to define a map from a finite interval to the infinite one. Two of these maps from the orthogonal polynomial domain $r \in [-1, 1]$ take the form

$$x = \tfrac{r+1}{r-1}, \qquad x \in [0, \infty)$$

$$x = \tfrac{r}{\sqrt{1-r^2}}, \qquad x \in \mathbb{R}$$

Orthogonal polynomials on $r \in [-1, 1]$ mapped with one of the transformations above result in rational-type functions in x. These functions have met with great success due to their relative superiority in accuracy compared to Laguerre or Hermite methods. In addition, these methods are direct maps of the finite interval to the

infinite interval, so that much of the implementation on finite intervals can be reused. If the chosen method of expansion in the r variable are Chebyshev-type polynomials, then FFT algorithms for collocation methods are also available.

However, as we shall explore in this work, these mapped basis sets do not exhibit some of the very attractive properties for solving differential equations that the generalized Wiener basis does posess.

1.1.4 The Wiener basis

In this work we concentrate our efforts on generalizing a basis set over the real line that is similar to the rational-type mappings mentioned in the previous section. The difference with the functions we shall explore is that they are not maps of orthogonal polynomials, but instead of generalized Fourier functions. We delay the details until later sections, and for now give a survey of the historical appearance of these functions in the literature.

One of the first appearances of a special case of the generalized Wiener basis set we shall explore in this work is in a book by Wiener [69]. Wiener introduces the 'normal and orthogonal' functions

$$l_n(x) = \frac{(ix-1)^n}{\sqrt{\pi}(ix+1)^{n+1}}, \tag{1.2}$$

for $n \in \mathbb{N}$, where in the context of that discussion, $x \in \mathbb{R}$ is the parameter in the continuous Fourier transform. It is further shown via Cauchy's residue theorem that the functions are orthonormal, and that the functions $\{l_n(x)\}_{n=0}^{\infty}$ are the Fourier transforms of the Laguerre functions, which we shall elaborate on below.

Later, Higgins [45] shows that (1.2) are complete (for $n \in \mathbb{Z}$) and orthonormal. He further fleshes out the relation between the Laguerre functions and the set $l_n(x)$, and he gives the characterization that $\{l_n(x)\}_{n=0}^{\infty}$ form a complete orthonormal system over the collection of L^2 functions whose Fourier transforms vanish on $\mathbb{R}^-$. A similar characterization is made for the set $\{l_n(x)\}_{n=-\infty}^{-1}$.

To be precise in the connection with Laguerre functions, let us define

$$\hat{l}_n(x,\alpha;a) = \left[\frac{a^{1+\alpha}n!}{\Gamma(1+n+\alpha)}\right]^{1/2} e^{-ax/2}x^{\alpha/2}L_n^{(\alpha)}(ax),$$

where $L_n^{(\alpha)}(x)$ are the generalized Laguerre polynomials, which are orthogonal under the weight $w(x) = e^{-x}x^{\alpha}$. We assume that $L_n^{(\alpha)}(x)$ is zero for $x < 0$. By letting $\mathcal{F}(\cdot)$ represent the (continuous) Fourier transform, we have

$$\mathcal{F}(\hat{l}_n(x,\alpha;a))(y) \quad = \quad \left[\tfrac{a^{1+\alpha}n!}{2\pi\Gamma(1+n+\alpha)}\right]^{1/2} \sum_{m=0}^{n}(-1)^m \begin{pmatrix} n+\alpha \\ n-m \end{pmatrix} \tfrac{a^m}{m!} \tfrac{\Gamma(a+m+\alpha/2)}{\left(iy+\frac{a}{2}\right)^{1+m+\alpha/2}}.$$

If we let $\alpha = 0$ and $a = 2$, we recover (1.2). Actually, Higgins shows completeness for (1.2) by appealing to the completeness of the Laguerre functions in conjunction with the isometric properties of the continuous Fourier transform.

The first work which exclusively deals with the basis functions (1.2) appears to come from Christov [27] in 1982. Christov reiterates the contributions from Wiener and Higgins, and explores the possibility of using the basis functions (1.2) for solving partial differential equations via the Galerkin method. In order to do this Christov derives, via quite simple algebra, a three-term recurrence relation for the derivative:

$$\frac{\mathrm{d}}{\mathrm{d}x}l_n(x) = \frac{i}{2}\left[nl_{n-1}(x) - (2n+1)l_n(x) + (n+1)l_{n+1}(x)\right] \tag{1.3}$$

This explicitly shows the sparse, banded structure of the Galerkin stiffness matrix, and also proves its skew-symmetry. He goes on to show that the matrix for the second derivative operator is penta-diagonal, and derives a three-term recurrence relation for the quantity $xl'_n(x)$ and a two-term recurrence relation for the product $l_n(x)l_m(x)$.

It is also in Christov's work that the first sign of breaking up the complex-valued basis set into real and imaginary parts on the half-infinite real line is seen. Indeed, the real and imaginary parts of the basis set (1.2) are odd and even functions, respectively, and so Christov associates them with the sine and cosine functions in the orthonormal system $\left\{e^{in\theta}\right\}_{n\in\mathbb{Z}}$. He considers the real parts of $l_n(x)$ as an orthonormal system on $\mathbb{R}^+$, and the imaginary parts of $l_n(x)$ as another orthonormal system on $\mathbb{R}^+$. The difference between these two is that one of them (the 'sine' basis) has all of its functions vanish at the origin, whereas the other does not.

Boyd [15] takes all of Christov's observations and builds on them. He is the first to make the connection between (1.2) and the Chebyshev polynomials of the first and second kinds (he actually emphasizes the relation between (1.2) and the 'Chebyshev rational functions' [14] instead of the Chebyshev polynomials). Boyd analyzes the relationship between the Chebyshev polynomials, the Chebyshev rational functions, the functions (1.2), and the functions

$$\lambda_n(x) = \frac{(ix-1)^n}{(ix+1)^n},$$

which Higgins introduced as orthonormal in $L^2(\mathbb{R},\mathbb{C};(1+x^2)^{-1})$, which is a straightforward conclusion from the form of (1.2). The missing link which connects the Chebyshev rational functions on the infinite domain to those on the finite domain

(Chebyshev polynomials) is the 'algebraic transformation' [14]

$$r = \frac{x}{\sqrt{1+x^2}}. \tag{1.4}$$

In further sections, we shall work with relations similar to the above, but they will be markedly different.

One particularly important computational aspect of the functions (1.2) which has not yet been revealed is its intimate relation with Fourier series and the FFT. Weidemann [67] makes this connection explicit. He introduces the transformation

$$e^{i\theta} = \frac{x+i}{x-i}, \tag{1.5}$$

as a method for transforming a Fourier series

$$f(\theta) = \sum_{|n|\leq N} \hat{a}_n e^{in\theta}$$

into the expression

$$g(x) = \sum_{|n|\leq N} \hat{a}_n \left(\frac{x+i}{x-i}\right)^n.$$

The relation between Fourier series and (1.2) is now clear. We have the orthogonality relation

$$\frac{1}{2\pi}\int_{-\pi}^{\pi} e^{im\theta}e^{-in\theta}\mathrm{d}\theta = \delta_{m,n},$$

and by changing variables from θ to x using (1.5), we obtain the relation

$$\frac{1}{\pi}\int_{\mathbb{R}} \left(\frac{x+i}{x-i}\right)^m \left(\frac{x+i}{x-i}\right)^{-n} \frac{1}{x^2+1}\mathrm{d}x = \delta_{m,n},$$

which can be recast into a more familiar form:

$$\int_{\mathbb{R}} \left[\frac{1}{\sqrt{\pi}}\frac{(x+i)^m}{(x-i)^{m+1}}\right]\left[\frac{1}{\sqrt{\pi}}\frac{(x+i)^{-n-1}}{(x-i)^{-n}}\right]\mathrm{d}x = \delta_{m,n},$$

and by taking into account the conjugate bilinear inner product for complex-valued functions, this is exactly

$$\int_{\mathbb{R}} l_m(x)\overline{l_n(x)}\mathrm{d}x = \delta_{m,n}.$$

Thus, as Weidemann states, the basis (1.2) is nothing more than a Fourier series in disguise. From this follow completeness and orthonormality. More importantly, for solving differential equations, the FFT can also be employed to calculate banded Galerkin operators, such as the differentiation operator. How do we apply the FFT? Firstly, we assume we have a function which can be expanded in terms of (1.2):

$$\xi(x) = \sum_{n=-N-1}^{N} \hat{a}_n \frac{1}{\sqrt{\pi}}\frac{(x+i)^n}{(x-i)^{n+1}}.$$

We have chosen the particular limits on n because of the conjugation properties of the function (1.2): $\overline{l_n(x)} = l_{-n-1}(x)$. The ultimate goal is to take nodal evaluations of $\xi(x)$ to nodal evaluations of $\xi'(x)$ for some nodal set x_j. We shall specify the nodal set below, and we shall even make precise what "$\xi'(x)$" means.

By some manipulation, we have

$$\Xi(x) := (x-i)\sqrt{\pi}\xi(x) = \sum_{n=-N-1}^{N} \hat{a}_n e^{in\theta}, \tag{1.6}$$

which we can now view as a function of θ. The right hand side is an expansion in orthogonal functions, and the transform from nodal values $\Xi(x_j)$ to the $\hat{a}_n$ can be

accomplished for a particular choice of the θ_j. We thus let

$$\theta_j = -\pi + \frac{(j-1)}{2N+3} 2\pi,$$

for $j = 1, 2, \ldots, 2N+3$. Then we have

$$x_j = \cot\left(\frac{\theta_j}{2}\right),$$

and we can form an isometry between the $\Xi(x_j) = (x_j - i)\sqrt{\pi}\xi(x_j)$ and the modal coefficients $\{\hat{a}_j\}_{j=-N-1}^{N+1}$, where we set $\hat{a}_{N+1} \equiv 0$. This isometry is connected via the FFT as can be seen by (1.6). Once we've calculated the $\hat{a}_j$, then we can use a sparse tridiagonal stiffness matrix formed from (1.3) to calculate the modal coefficients $\hat{b}_n$, which represent

$$\tilde{\xi}'(x) = \mathcal{P}_{2N+2}\xi'(x) = \sum_{n=-N-1}^{N} \hat{b}_n l_n(x),$$

where $\mathcal{P}$ denotes the L^2 projection operator (see Section (2.1.2). Note that we need the L^2 projection operator $\mathcal{P}_{2N+2}$ in the above equation: unlike the Fourier case, equation (1.3) makes it clear that the differentiation operators and projection operators do not commute for the basis (1.2). Thus, our Galerkin calculation will "chop off" some contributions from higher-order basis functions outside of the space.

Finally, to transport the $\hat{b}_n$ back to the nodal values $\tilde{\xi}'(x_j)$, we again need to observe that

$$\tilde{\Xi}'(x) := \tilde{\xi}'(x)\sqrt{\pi}(x - i) = \sum_{n=-N-1}^{N} \hat{b}_n e^{ij\theta}.$$

Thus, the $\tilde{\Xi}'(x_j)$ can again be calculated via a $(2N+3)$-point FFT, and finally a further $O(N)$ operations takes the $\tilde{\Xi}'(x_j)$ to the $\tilde{\xi}'(x_j)$, which was the desired quantity.

Weidemann also proves that the eigenvalues of the Galerkin differentiation matrix scale like $O(N)$, just like those of the Fourier-Gakerin differentiation matrix. I.e. the map (1.5) does not affect the spectral properties of differentiation (unlike the map $r = \cos\theta$, which transforms the $O(N)$ collocation eigenvalues of the Fourier basis into the $O(N^2)$ collocation eigenvalues of the Chebyshev polynomials). This is an important property for numerical stability if one wishes to solve partial differential equations.

All of the observations made in this section regarding Wiener's original orthogonal basis (1.2) will be addressed in our journey to generalize those functions.

Chapter Two

Preliminary Tools

We shall first introduce much of the notation and background material necessary for future discussions. First we introduce notation. Following that, we shall review some previous results on Sturm-Liouville theory and orthogonal polynomials. Finally, we introduce the topics which will serve to lead into our discussion of the orthogonal rational functions over the infinite interval.

2.1 Notation

In this section we lay out the notation necessary for our exploration of the Wiener rational basis. In all future sections we attempt to adhere to the conventions laid out here.

2.1.1 Domains

We take $\mathbb{R}$ and $\mathbb{C}$ to be the real and complex number fields, respectively. We shall consider the following domains:

- $\theta \in [-\pi, \pi]$
- $z \in \mathbb{T} = \{w \in \mathbb{C} : |w| = 1\}$
- $r \in I_r = [-1, 1]$
- $x \in \mathbb{R}$

We also make use of the subdomains:

	x	z	θ	r
x	$x \in [0, \infty]$	$z = -\frac{x-i}{x+i}$	$\theta = 2 \arctan x$	$r = \frac{1-x^2}{1+x^2}$
z	$x = i\frac{1-z}{1+z}$	$z \in \mathbb{T}^+$	$\theta = \arg z$	$r = \frac{1}{2}(z + \bar{z})$
θ	$x = \tan\left(\frac{\theta}{2}\right)$	$z = e^{i\theta}$	$\theta \in [0, \pi]$	$r = \cos\theta$
r	$x = \sqrt{\frac{1-r}{1+r}}$	$z = e^{i \arccos r}$	$\theta = \arccos r$	$r \in [-1, 1]$

Table 2.1: Isomorphic transforms between different domains.

$b \backslash a$	x	θ	r
x	1	$\frac{2}{1+x^2}$	$\frac{-4x}{(x^2+1)^2}$
θ	$\frac{1}{1+\cos\theta}$	1	$-\sin\theta$
r	$\frac{-1}{(1+r)(1-r^2)^{1/2}}$	$-\frac{1}{\sqrt{1-r^2}}$	1

Table 2.2: Jacobian relations for transformations. The table lists $\frac{\mathrm{d}a}{\mathrm{d}b}$ as a function of b.

- $\theta \in I_\theta = [0, \pi]$
- $z \in \mathbb{T}^+ = \{w \in \mathbb{T} : \mathrm{Im}\{w\} \geq 0\}$
- $x \in I_x = \mathbb{R}^+ = [0, \infty]$

In all that follows, the variables θ, z, r, and x will be reserved to be associated with only their respectively domains above. Transformations between these domains are not unique, but in Tables 2.1 and 2.2 we list the main transformations that will be used throughout this work. We will occasionally utilize transformations that differ from those in the Tables, but in such situations the differnece will be clear.

All of the transformations in Table 2.1 are isomorphisms between the respective domains listed. Table 2.2 lists the Jacobian relations between different domains.

There are a few remarks to make about this. The transformation

$$z = -\frac{x - i}{x + i},$$

is a special kind of linear fractional/Möbius map and is sometimes called the Cayley transform (up to complex conjugation and a phase). It is a certain kind of linear fractional map which maps the extended upper half-plane into the unit disk and is used extensively in the theory of complex-valued orthogonal rational functions [20]. This technique of using a linear fractional map to transform the boundary of the unit disk to the real line shall be used extensively later.

The transformation

$$r = \cos\theta,$$

commonly appears in the theory of orthogonal polynomials [60]. It relates the Chebyshev polynomials to the Fourier basis functions and is vital in the application of the fast Fourier Transform algorithm to polynomial approximation methods. In general, it relates families of polynomials on the interval $I_r = [-1, 1]$ to the Szegö polynomials [60] on the upper unit circle.

We shall sometimes avoid utilizing the explicit expressions listed above and instead we use transformations for them. In particular, the independent variables x, r, θ, and z will often be written as functions of one another. Conforming to the

transformations presented in Table 2.1 above, we have e.g.

$$\begin{aligned} x(r) &= \sqrt{\tfrac{1-r}{1+r}} \\ x(\theta) &= \tan\tfrac{\theta}{2} \\ r(\theta) &= \cos\theta \\ r(x) &= \tfrac{1-x^2}{1+x^2} \end{aligned}$$

In later sections, we will augment the relationship between x, r, and θ, and the above relations will change. Because of this, we shall only use the notation e.g. $r(x)$ when it is clear what the relationship between r and x is.

2.1.2 Function spaces

By $L^2(A, B; \mu)$ we mean the space of square-integrable functions under the positive measure μ mapping $A \subset \mathbb{R}$ to $B \subset \mathbb{F}$. In this work, we shall always have μ absolutely continuous with respect to Lebesgue measure so that a density, which we call $w(\cdot)$, exists almost everywhere and the associated L^2 space is $L^2_w(A, B) = L^2(A, B; w)$. If $w \equiv 1$, we shall often omit writing it, and we also omit either B or both A and B if the context is clear.

Being a Hilbert space, we endow $L^2_w(A, B)$ with the conjugate bilinear inner product

$$\langle f, g \rangle_w = \int_A f \bar{g} w \mathrm{d}x,$$

whose associated norm completes the space. If B is a subset of $\mathbb{R}$, this reverts to

the (non-conjugate) bilinear form.

We shall infrequently make reference to the Sobolev space

$$H^s = \left\{ f : f^{(j)} \in L^2(A,B), j = 0, 1, \ldots s \right\},$$

where $f^{(j)}$ denotes the jth derivative, and $f^{(0)} \equiv f$. For s an integer, the norm is defined as

$$\|\rho\|_{H^s}^2 = \sum_{q=0}^{s} \left|\left| \frac{\mathrm{d}^q \rho}{\mathrm{d}r^q} \right|\right|_{L^2}^2 = \sum_{q=0}^{s} \int_A \left(\frac{\mathrm{d}^q \rho}{\mathrm{d}r^q} \right)^2 \mathrm{d}r.$$

It is also possible to define real-valued Sobolev spaces; one may either use the continuous Fourier transform or it can be done via the concept of complex interpolation [64]. The Sobolev space parameter s is intricately tied with spectral expansion convergence rates for orthogonal polynomials [21] and Fourier series [43].

There are specific weight functions w that will play central roles in this work. The following weight functions are often-used:

$$\begin{aligned} w_r^{(\alpha,\beta)} &= (1-r)^\alpha (1+r)^\beta \\ w_\theta^{(\gamma,\delta)} &= w_r^{(\delta,\gamma)} = (1+\cos\theta)^\gamma (1-\cos\theta)^\delta \\ w_x^{(s,t)} &= w_\theta^{(s,t)} = \tfrac{2^{s+t}}{(1+x^2)^s} \left(\tfrac{x^{2t}}{(1+x^2)^t} \right). \end{aligned}$$

We have defined these weights so that e.g. $w_x^{(s,t)}(x(r)) = w_r^{(t,s)}(r)$. The motivation for the order of the indices in the weight functions is to mimic the orientation of the Jacobi weight $w_r^{(\alpha,\beta)}$, i.e., α is the exponent on the right-hand zero $(1-r)$, and

β is the exponent on the left-hand weight $(1+r)$. Thus, (α,β), (γ,δ), and (s,t) correspond to the order of the root at the (right,left) endpoints on I_r, I_θ, and I_x, respectively.

All of the above weights are semi-positive. However, it will be common for us to take non-principal branches when we take square roots. When this is the case, we'll denote this particular square root as $\sqrt[*]{\cdot}$. In particular, for the weights given above, we take the following non-principal square roots:

$$\begin{aligned}
\sqrt[*]{w_r^{(\alpha,\beta)}} &= \sqrt{w_r^{(\alpha,\beta)}} = (1-r)^{\alpha/2}(1+r)^{\beta/2} \\
\sqrt[*]{w_\theta^{(\gamma,\delta)}} &= 2^{\left(\frac{\gamma+\delta}{2}\right)} e^{i\left(\frac{\gamma+\delta}{2}\right)\theta} \sin^\delta\left(\tfrac{\theta}{2}\right)\cos^\gamma\left(\tfrac{\theta}{2}\right) \\
\sqrt[*]{w_x^{(s,t)}} &= \frac{2^{\left(\frac{s+t}{2}\right)} x^t}{(x-i)^{s+t}}.
\end{aligned} \tag{2.1}$$

We will also make use of the principal-branch square roots $\sqrt{\cdot}$, which will have obvious value since all the weight functions are non-negative. The non-principal branch square roots satisfy the property

$$\sqrt[*]{w}\overline{\sqrt[*]{w}} = w.$$

On $L^2(I_r,\mathbb{R};w)$ we define polynomial projection operators $\mathcal{P}_N^w$. For the weight w, there exist a family of orthonormal polynomials $\{p_n(r)\}_{n=0}^\infty$. We shall consider expansions of $\rho \in L_w^2(I_r,\mathbb{R})$ in these polynomials

$$\rho = \sum_{n=0}^\infty \hat{\rho}_n p_n,$$

where equality holds in the L^2_w norm. The $\hat{\rho}_n$ are defined by

$$\hat{\rho}_n = \frac{\langle \rho, p_n \rangle_w}{\|p_n\|^2_w}$$

That such an expansion exists for certain w can be deduced by considering the appropriate self-adjoint singular Sturm-Liouville eigenvalue problems associated with e.g. the Jacobi polynomials, and then applying the spectral theorem. Associated with the polynomials, we define the polynomial subspace $\mathcal{B}_N = \text{span}\{r^n : 0 \leq n \leq N-1\}$. We define the projection operator $\mathcal{P}^w_N$ on $L^2_w(I_r, \mathbb{R})$ as

$$\mathcal{P}^w_N \rho = \sum_{n=0}^{N} \hat{\rho}_n p_n.$$

Thus, $\mathcal{P}^w_N$ is the $L^2_w(I_r, \mathbb{R})$-orthogonal projection onto $\mathcal{B}_N$. And it satisfies

$$\langle \mathcal{P}^w_N \rho - \rho, \phi \rangle_w = 0, \quad \forall \phi \in B_N$$

On $L^2_w([-\pi, \pi], \mathbb{C})$ or $L^2_w(I_\theta, \mathbb{C})$ we define the space

$$\mathcal{T}_N = \text{span}\left\{ e^{in\theta} : |n| \leq N-1 \right\},$$

and for any $\Theta \in L^2_w([-\pi, \pi], \mathbb{C})$ which is Fourier-expandable as

$$\Theta = \sum_{n \in \mathbb{Z}} \hat{\Theta}_n e^{in\theta}$$

the associated projection operator is

$$\mathcal{P}^{\theta,w}_N \Theta = \sum_{|n|<N} \hat{\Theta}_n e^{in\theta},$$

so that again $\mathcal{P}_N^{\theta,w}$ is the $L^2_w([-\pi,\pi],\mathbb{C})$-orthogonal projection onto $\mathcal{T}_N$.

2.2 The Sturm-Liouville eigenvalue problem, Jacobi polynomials

For an in-depth survey of the theory in this field, we point the reader to [70], [60], [34], and [1]. We begin by presenting the canonical second-order equation

$$-\tfrac{\mathrm{d}}{\mathrm{d}x}\left[p(x)y'\right] + q(x)y - \lambda w(x)y = 0, \quad x \in I \tag{2.2}$$

If we are to be precise, we also require some associated boundary conditions. The functions $p(x)$, $q(x)$, and $w(x)$ must obey certain regularity conditions.

Under these conditions, in conjunction with the spectral theorem, then equation (2.2) admits a discrete spectrum $\{\lambda_n\}$ whose associated eigenfunctions $\{y_n\}$ form a complete basis for $L^2(I,\mathbb{R};w)$ and are orthogonal under the weight $w(x)$. For polynomial solutions to a Sturm-Liouville problem one may consider the Jacobi differential equation

$$(1-r^2)\rho'' + \left[\beta - \alpha - (\alpha+\beta+2)r\right]\rho' + n(n+\alpha+\beta+1)\rho = 0, \quad r \in [-1,1], \tag{2.3}$$

for $\alpha, \beta > -1$ and any $n \in \mathbb{N}_0$. The associated polynomial eigenfunctions, which depend on the values of α and β are called Jacobi polynomials, and are standard classical orthogonal polynomials. The monic Jacobi polynomial of order n, as a solution to (2.3), is written as $P_n^{(\alpha,\beta)}(r)$. For any $\alpha, \beta > -1$, the Jacobi polynomials

of class (α, β) are orthogonal under a weighted L^2 inner product:

$$\int_{-1}^{1} P_m^{(\alpha,\beta)} P_n^{(\alpha,\beta)} (1-r)^\alpha (1+r)^\beta \mathrm{d}r = h_n^{(\alpha,\beta)} \delta_{m,n}, \tag{2.4}$$

where $\delta_{m,n}$ is the Kronecker delta function, and $h_n^{(\alpha,\beta)}$ is a normalization constant given in Appendix A.1. These orthogonal polynomials form the foundation for a large portion of this work. We shall usually deal with the orthonormal Jacobi polynomials $\tilde{P}_n^{(\alpha,\beta)}(r)$ which satisfy the relation

$$\int_{-1}^{1} \tilde{P}_n^{(\alpha,\beta)}(r) \tilde{P}_m^{(\alpha,\beta)}(r) w_r^{(\alpha,\beta)} \mathrm{d}r = \delta_{m,n}.$$

It will be common for use to use a tilde $\tilde{()}$ above a function to denote that it is normalized with respect to the L^2 norm under which the functions are orthogonal. The orthonormal Jacobi polynomials $\tilde{P}_n^{(\alpha,\beta)}$ will be integral in the derivation of the Wiener rational function basis on the real line. All of the various properties of Jacobi polynomials that we will use at some point are collected in Appendix A.1. We require a minor generalization of Jacobi polynomials: we perform a change of dependent variable in (2.3) to obtain the following:

Lemma 2.1 *(Jacobi Functions)*

The Jacobi functions defined as

$$P_n^{(\alpha,\beta,a,b)}(r) = (1-r)^a (1+r)^b P_n^{(\alpha,\beta)}(r)$$

satisfy the following properties:

1. $\left\{ P_n^{(\alpha,\beta,a,b)}(r) \right\}_{n \in \mathbb{N}_0}$ *are orthogonal and complete in* $L^2(I_r, \mathbb{R}; w)$ *under the weight* $w(r) = (1-r)^{\alpha-2a}(1+r)^{\beta-2b}$.

2. The $P_n^{(\alpha,\beta,a,b)}(r)$ are eigenfunctions $\rho_n(r)$ of the Sturm-Liouville problem

$$-\frac{\mathrm{d}}{\mathrm{d}r}\left[p(r)\rho'(r)\right] + q(r)\rho(r) - \lambda w(r)\rho(r) = 0, \tag{2.5}$$

with the parameters

$$\left.\begin{aligned} p(r) &= (1-r)^{\alpha+1-2a}(1+r)^{\beta+1-2b} \\ q(r) &= \left[a(\alpha-a)(1-r)^{-2} + b(\beta-b)(1+r)^{-2}\right](1-r)^{\alpha+1-2a}(1+r)^{\beta+1-2b} \\ w(r) &= (1-r)^{\alpha-2a}(1+r)^{\beta-2b} \\ \lambda_n &= n(n+\alpha+\beta+1) - 2ab + a(\beta+1) + b(\alpha+1) \end{aligned}\right\} \tag{2.6}$$

The proof of Lemma 2.1 is simple but tedious. It relies on the ability to compute an appropriate integrating factor to transform equation (2.3) into (2.5). Note that if one takes $a = b = 0$ in Lemma (2.1), the resulting problem and eigensolutions revert back to the Jacobi polynomials, as expected. We shall often write $P_n^{(\alpha,\beta,\alpha/2,\beta/2)}$=$p_n^{(\alpha,\beta)}$, i.e., $p_n^{(\alpha,\beta)}$ are the Jacobi polynomials $P_n^{(\alpha,\beta)}$ multiplied by the square root of their associated weight function, and thus these functions are orthogonal under the unweighted L^2 inner product. We also define tilde'd versions as the normalized functions: $\tilde{p}_n^{(\alpha,\beta)} = \sqrt{w_r^{(\alpha,\beta)}}\tilde{P}_n^{(\alpha,\beta)}$. Due to the previous lemma, these Jacobi functions $\tilde{p}_n^{(\alpha,\beta)}$ are orthonormal under unweighted Lebesgue measure.

The above discussion has centered mainly around the class of Jacobi orthogonal polynomials, but there are other classes of orthogonal polynomials that are relevant to our future tasks. In particular, the very popular Legendre polynomials and

	Domain	w
Jacobi	$r \in [-1, 1]$	$(1-r)^\alpha(1+r)^\beta$, $\alpha, \beta > -1$
Laguerre	$x \in [0, \infty)$	$x^\alpha e^{-x}$, $\alpha > -1$
Hermite	$x \in \mathbb{R}$	$\lvert x\rvert^{2\alpha} e^{-x^2}$, $\alpha > -\frac{1}{2}$

Table 2.3: Classical orthogonal polynomials: their integral weights and domains. See appendix A for the recurrence coefficients.

Chebyshev polynomials (of the first, second, third, and fourth kinds) are subsets of the Jacobi orthogonal polynomials. In addition, we will make use of the Laguerre and Hermite polynomials. Because these two latter families are only for comparison purposes, we will not expound on their properties as much as with the Jacobi case, but we do include basic information about these polynomial families in Table 2.3, and more in-depth information can be found in Chapter 7 and Appendix A.

In all future sections we make extensive use of the notation that a tilde'd function is the normalized version (with respect to some norm), and a lowercase letter (Roman or Greek) is the uppercase function multiplied by the (perhaps non-principal) square root of the associated weight.

2.2.1 Polynomial Quadrature

In this section we review some basic theoretical and computational results for orthogonal polynomials. Although the results here are applicable to any collection of general orthogonal polynomials, much of this work will only apply these results to the Jacobi polynomial case.

We adopt the following notation for quadrature: a quadrature rule $Q_N(x, \omega)$ is de-

fined as the linear operator defined by N nodal points $\{x_j\}_{j=1}^N$ and weights $\{\omega_j\}_{j=1}^N$, which acts on a function f:

$$Q_N(x,\omega)\,[f] = \sum_{j=1}^{N} \omega_j f(x_j). \tag{2.7}$$

Quadrature rules are used as approximations of integrals against certain weight functions. A very special type of quadrature rule is one that is Gaussian:

Definition 2.1 *(Gaussian quadrature)*

Given a collection of functions $\{\phi_n(x)\}_{n=1}^{\infty}$ defined over some interval $x \in I$ and a weight function $w(x)$. The N-point quadrature rule $Q_N(x_n,\omega_n)[\cdot]$ is called Gaussian if

$$\int_I \phi_n(x)w(x)\mathrm{d}x = \sum_{m=1}^{N} \phi_n(x_m)\omega_m := Q_N(x_i,\omega_i)[\phi_n], \qquad n = 1,2,\ldots,2N.$$

The following two results are classical well-known results in the theory of orthogonal polynomials:

Lemma 2.2 *All families of polynomials orthogonal with respect to some weight $w(x)$ admit a three-term recurrence relation of the form*

$$p_{n+1}(x) = (x-a_n)p_n(x) - b_n p_{n-1}(x), n > 1 \tag{2.8}$$

for constants a_n and b_n.

The nodal locations of the N-point Gauss quadrature rule for an orthogonal polynomial family P_n are given by the roots of the polynomial P_N. This, along with

the existence of a recurrence relation for orthogonal polynomials allows us to easily compute the Gaussian quadrature rule [35]:

Theorem 2.1 *The Gaussian quadrature nodes and weights $(r_n, \omega_n)_{n=1}^N$ for the polynomials $P_n^{(\alpha,\beta)}$ are given as eigenvalues and components of eigenvectors of a symmetric tridiagonal matrix. Define*

$$J = \begin{bmatrix} a_0 & \sqrt{b_1} & 0 & 0 & \cdots & & 0 \\ \sqrt{b_1} & a_1 & \sqrt{b_2} & 0 & \cdots & & 0 \\ 0 & \ddots & \ddots & \ddots & & & \\ & & & & & & \sqrt{b_{N-1}} \\ 0 & & \cdots & & 0 & \sqrt{b_{N-1}} & a_{N-1} \end{bmatrix}.$$

Let $\{\lambda_n\}_{n=1}^N$ be the eigenvalues of J and $\{v_n\}_{n=1}^N$ be the orthonormalized eigenvectors of J. Then $r_n = \lambda_n$ and $\omega_n = b_0 v_{n,1}^2$ for $n = 1, \ldots, N$.

For a very special case of the Jacobi polynomials, we can find an explicit formula for the Gauss quadrature nodes and weights:

Lemma 2.3 *The Gaussian quadrature $\{(r_n, \omega_n)\}_{n=1}^N$ for the Chebyshev polynomials $\left(i.e.\ the\ polynomials\ \tilde{P}_n^{(-1/2,-1/2)}(r)\right)$ are cosine maps of equidistant nodes. Define*

$$\theta_n = \frac{(2n+1)\pi}{2N}.$$

Then

$$r_n = -\cos\theta_n$$

$$\omega_n = \tfrac{\pi}{N}.$$

Theorem 2.1 allows us to easily compute Gaussian quadrature formulae for any collection of orthogonal polynomials provided that we know the recurrence relation. This shows how powerful the recurrence constants a_n and b_n are: they allow us not only to evaluate the polynomials via (2.8), but they also allow us to perform accurate (Gaussian) quadrature. There are, of course, other kinds of quadrature rules. In particular, Clenshaw-Curtis quadrature [28] is a popular alternative, and is in many cases competitive with Gaussian quadrature [63]. One of the much-touted advantages of Clenshaw-Curtis quadrature is that it can be implemented via the fast Fourier transform (FFT). However, we will show in Chapter 3 that Gaussian-like quadrature can also be implemented with the FFT for relatively general values of the parameters (α, β).

It is already well-known that for $(\alpha, \beta) = \left(-\frac{1}{2}, -\frac{1}{2}\right)$, the Gaussian quadrature rule for Jacobi polynomials (i.e. the Chebyshev polynomials) can be implemented via the FFT. This is evident from the result of Lemma 2.3, since the Chebyshev-Gauss quadrature is just a Fourier quadrature in θ-space. We will present a similar FFT result for a greater variety of parameter families (α, β).

We'll denote $\left\{\left(r_n^{(\alpha,\beta)}, \omega_n^{(\alpha,\beta)}\right)\right\}_{n=1}^{N}$ the N-point nodes and weights for the Jacobi polynomial (α, β) Gaussian quadrature rule. One of the major reasons that we require a quadrature rule is our desire to determine the spectral coefficients of the expansion

$$\mathcal{P}_N^{w_r^{(\alpha,\beta)}} f(r) = \textstyle\sum_{n=0}^{N-1} \hat{f}_n \tilde{P}_n^{(\alpha,\beta)}(r),$$

where

$$\hat{f}_n = \left\langle f, \tilde{P}_n^{(\alpha,\beta)} \right\rangle = \int_{-1}^{1} f(r)\tilde{P}_n^{(\alpha,\beta)}(r)w_r^{(\alpha,\beta)}(r)\mathrm{d}r \simeq Q_N\left(r_n^{(\alpha,\beta)}, \omega_n^{(\alpha,\beta)}\right)\left[f\tilde{P}_n^{(\alpha,\beta)}\right].$$

Thus, quadrature rules give us an accurate way to compute the modal coefficients for a spectral expansion given the nodal values $f\left(r_n^{(\alpha,\beta)}\right)$. The error made in using the quadrature rule Q_N to approximate the modal coefficient integral in lieu of computing the actual value is often called the *aliasing error*.

Other Gauss-type quadrature rules also exist for polynomials, including Gauss-Radau, Gauss-Lobatto, and Gauss-Kronrod rules. We introduce the Radau and Lobatto rules below, both of which will later use.

Definition 2.2 *(Gauss-Radau Quadrature)*

Given a collection of functions $\{\phi_n(x)\}_{n=1}^{\infty}$ *defined over some interval* $x \in I$*, a fixed point* $x_1 \in I$*, and a weight function* $w(x)$*. The* N*-point quadrature rule* $Q_N(x_n, \omega_n)[\cdot]$ *is called Gauss-Radau if:*

$$\int_I \phi_n(x)w(x)\mathrm{d}x = \phi_n(x_1)\omega_1 + \sum_{m=2}^{N} \phi_n(x_m)\omega_m := Q_N(x_i, \omega_i)[\phi_n], \qquad n = 1, 2, \ldots, 2N-1.$$

Gauss-Radau quadrature is the integration rule that results when we attempt a Gauss-like rule but fix a single quadrature node x_1. Due to this constraint, we pay a price in accuracy: we can only correctly integrate $2N-1$ polynomials compared with the $2N$ polynomials in pure Gauss quadrature.

Our use of Gauss-Radau quadrature will be limited to Jacobi polynomials, and although we are free to choose the fixed node r_1 to be any point in $[-1, 1]$, we will

always choose it to be $r_N = +1$.

Definition 2.3 *(Gauss-Lobatto Quadrature)*

Given a collection of functions $\{\phi_n(x)\}_{n=1}^{\infty}$ *defined over some finite interval* $x \in I$*, the two endpoints* x_1 *and* x_N *of* I*, and a weight function* $w(x)$*. The* N*-point quadrature rule* $Q_N(x_n, \omega_n)[\cdot]$ *is called Gauss-Lobatto if for all* $n = 1, 2, \ldots, 2N-2$*:*

$$\int_I \phi_n(x) w(x) \mathrm{d}x = \phi_n(x_1)\omega_1 + \phi_n(x_N)\omega_N + \sum_{m=2}^{N-1} \phi_n(x_m)\omega_m := Q_N(x_i, \omega_i)[\phi_n]$$

Again our usage of Gauss-Lobatto rules will be restricted to the Jacobi polynomial case where we will always take $r_1 = -1$ and $r_N = +1$. As noted in [32], the unknown nodes and weights for Gauss-Radau/Lobatto quadrature, like Gauss quadrature, can be computed as eigenvalues and eigenvector components of a symmetric tridiagonal matrix, i.e, an analogue of Theorem 2.1 exists for Gauss-Radau/Lobatto quadrature rules, but we shall not present it. The reader is referred to [32] and [33] for the algorithm, which also contains algorithms for Gauss-Kronrod quadrature. Some theory is presented in [53].

For notation, we will use $\left\{r_n^{(\alpha,\beta)}, \omega_{r,n}^{(\alpha,\beta)}\right\}_{n=1}^{N}$ to denote the Gauss-quadrature nodes and weights for the Jacobi polynomials of order (α, β). Similarly, $\left\{r_n^{(\alpha,\beta);\mathrm{GR}}, \omega_{r,n}^{(\alpha,\beta);\mathrm{GR}}\right\}_{n=1}^{N}$ denotes the Gauss-Radau quadrature nodes and weights for the Jacobi polynomials of order (α, β), and $\left\{r_n^{(\alpha,\beta);\mathrm{GL}}, \omega_{r,n}^{(\alpha,\beta);\mathrm{GL}}\right\}_{n=1}^{N}$ denotes the Lobatto rule. Regrettably, the subscript r on the quadrature weight ω is necessary to distinguish this weight from others we shall define later. In addition, we confess our sloppiness in notation: $\left\{r_n^{(\alpha,\beta)}\right\}_{n=1}^{N}$ and $\left\{r_n^{(\alpha,\beta)}\right\}_{n=1}^{N+1}$ refer to two completely different sets of nodes; i.e. $r_1^{(\alpha,\beta)}$ from the first set has no relation to $r_1^{(\alpha,\beta)}$ from the second set. We have omitted denoting the dependence on N to keep notation as understandable as possible for our

purposes.

2.3 Basic Approximation Theory

We transcribe some theorems relating to polynomial approximation theory here. To faciliate notation, we assume that for any space $L^2_w(A,B)$, there exists some collection of polynomial basis functions $\{p_n\}_{n\in\mathbb{N}\cup\{0\}}$. For example, if $w\equiv 1$, $A=[-1,1]$, and $B=\mathbb{R}$, then p_n may be the Legendre polynomials, where we assume that the index of the basis functions is directly related to the degree of the polynomial function. We start off by presenting the following result [21], [43]:

Lemma 2.4 *Suppose $u\in H^p_{w_r^{(\alpha,\beta)}}(I_r,\mathbb{R})$ with $|\alpha|<1$ and $|\beta|<1$. There exists a constant C for all $0\le q\le p$*

$$\|u-\mathcal{P}_N^{w_r^{(\alpha,\beta)}}u\|_{H^q_{w_r^{(\alpha,\beta)}}(I_r,\mathbb{R})}\le CN^{f(q)-p}||u||_{H^p_w(I_r,\mathbb{R})},$$

where $f(q)$ is defined by

$$f(q)=\begin{cases}\frac{3}{2}q, & 0\le q\le 1\\ \\ 2q-\frac{1}{2}, & q\ge 1\end{cases}$$

The upshot of the Lemma is that if u is not infinitely smooth, then one can only expect algebraic convergence in N. However, if u very smooth, then one can expect spectral, or exponential, convergence of the polynomial projection to the function in some Sobolev norm. Of course, if one requires convergence in a high-degree Sobolev norm (large q), one must pay some algebraic price ($f(q)$ increases as q increases).

Similar results exist for Fourier expansions. We will not present any novel approximation theory results, but we present this to give a flavor of what we expect from any new basis set that we derive: convergence rate commensurate with function regularity.

Chapter Three

Jacobi Polynomials And The Fast Fourier Transform

3.1 Introduction

This relatively self-contained chapter addresses the question of the use of the fast Fourier transform for modal-nodal transformations of Jacobi polynomials extended beyond the simple Chebyshev case. The ability to do this will enable us to use the FFT for the same purposes for the generalized Wiener rational functions, to be derived in Chapter 6. Along the way, we will also develop a novel method for quadrature that we show to be more robust than traditional Gaussian quadrature methods for Jacobi polynomials when the class parameters (α, β) are very large.

As introduced in Section 2.2, the classical Jacobi polynomials $P_n^{(\alpha,\beta)}$ are a family of orthogonal polynomials [60] that have been used extensively in many applications for their ability to approximate general classes of functions. They are a class of polynomials that encompass the Chebyshev, Legendre, and Gegenbauer/ultraspherical polynomials. In addition, they have a very close connection to the associated Legendre functions, which are widely used in spherical harmonic expansions.

Jacobi polynomial expansions have been used in a variety of applications, some of which are the resolution of the Gibbs' phenomenon [38], electrocardiogram data compression [62], and the solution to differential equations [23]. Due to the range of applications for Jacobi polynomials, it is desirable to perform spectral expansion computations as quickly and accurately as possible. In this chapter, we show that for a variety of (but not all) classes of Jacobi polynomials, one can exploit the Fast Fourier Transform (FFT) to perform spectral transformations; this implies an asymptotic reduction of the direct-method unoptimized $O(N^2)$ cost to an $O(N \log N)$ cost, which is significant if N is large. In addition, we show that computational savings is apparent even for small N. Our method is applicable for discrete polynomial trans-

forms for an expansion in $\left\{P_n^{(\alpha,\beta)}\right\}_{n=0}^{N-1}$ if both 2α and 2β are odd integers. This notably does not include the Legendre case where $(\alpha, \beta) = (0, 0)$.

The idea of using an FFT to compute Jacobi spectral transformations is not new. That the FFT can be used to compute Chebyshev modes via nodal evaulations at the Chebyshev quadrature points is well-known [43]. In addition, the use of the FFT for other Jacobi polynomial transformations has also been studied. For Legendre polynomial transforms, Alpert [2] employed a submatrix decomposition of the modal connection matrix to travel between Chebyshev modes and Legendre modes. Orszag [54] used the WKB approximation to propose an algorithm for a broad class of eigenfunction transforms (including Legendre polynomial transforms). Driscoll [30], [31] provides an $O(N \log N)$ algorithm for general discrete polynomial transforms as long as the three-term recurrence for the polynomial family is known. The method relies in part on a Toeplitz-product factorization of the interpolating Vandermonde matrix, for which an asymptotic $O(N \log N)$ algorithm for matrix-vector multiplication exists [55].

Our method is based upon two main observations: firstly that the transformation from function evaluations located at the Chebyshev quadrature nodes to Chebyshev polynomial expansion coefficients can be implemented with the FFT. Secondly, we show that the exact connection coefficient relation between Chebyshev polynomials and several other Jacobi polynomials classes can be computed and implemented robustly in an inexpensive $O(N)$ cost. The connection coefficient relation is simple enough that it does not require the evaluation of any transcendental or higher-order functions, requires very little preprocessing, and only $O(N)$ storage is required. This should be compared to the previously mentioned methods which are either approximate in the computation, require substantial preprocessing, more storage,

and/or are significantly more complicated to implement. The disadvantage of our method is that it can only be applied for certain classes of Jacobi polynomials that, as already mentioned, excludes the Legendre case.

In section 3.2 we give an overview of the Jacobi connection problem and review the relevant properties needed for our discussion. Section 3.3 is devoted to a theoretical description of the method and includes most of the major results. Section 3.4 is a special application of the results from section 3.3 to Chebyshev-like systems where the FFT may be exploited. Finally, section 3.5 gives some numerical examples.

3.2 The Jacobi Polynomial Connection Problem

In the remainder of this chapter, we will denote the norm and inner product on $I_r = [-1, 1]$ with respect to $w_r^{(\alpha,\beta)}$ as e.g. $\|\cdot\|_{(\alpha,\beta)}$, i.e.,

$$\begin{aligned}
\langle \cdot, \cdot \rangle_{(\alpha,\beta)} &:= \langle \cdot, \cdot \rangle_{w_r^{(\alpha,\beta)}} \\
\| \cdot \|_{(\alpha,\beta)} &:= \| \cdot \|_{w_r^{(\alpha,\beta)}} \\
L^2_{(\alpha,\beta)} &:= L^2_{w_r^{(\alpha,\beta)}} \\
\mathcal{P}_N^{(\alpha,\beta)} &:= \mathcal{P}_N^{w_r^{(\alpha,\beta)}}
\end{aligned}$$

All orthogonal polynomial families satisfy a three-term recurrence relation from which the Kernel polynomials and the Christoffel-Darboux identity can be derived. These last two properties yield the following *promotions* and *demotions* of the Jacobi

polynomial class parameters (α, β):

$$(1-r)\tilde{P}_n^{(\alpha,\beta)} = \mu_{n,0}^{(\alpha,\beta)}\tilde{P}_n^{(\alpha-1,\beta)} - \mu_{n,1}^{(\alpha,\beta)}\tilde{P}_{n+1}^{(\alpha-1,\beta)}, \tag{3.1}$$

$$(1+r)\tilde{P}_n^{(\alpha,\beta)} = \mu_{n,0}^{(\beta,\alpha)}\tilde{P}_n^{(\alpha,\beta-1)} + \mu_{n,1}^{(\beta,\alpha)}\tilde{P}_{n+1}^{(\alpha,\beta-1)}, \tag{3.2}$$

$$\tilde{P}_n^{(\alpha,\beta)} = \nu_{n,0}^{(\alpha,\beta)}\tilde{P}_n^{(\alpha+1,\beta)} - \nu_{n,-1}^{(\alpha,\beta)}\tilde{P}_{n-1}^{(\alpha+1,\beta)}, \tag{3.3}$$

$$\tilde{P}_n^{(\alpha,\beta)} = \nu_{n,0}^{(\beta,\alpha)}\tilde{P}_n^{(\alpha,\beta+1)} + \nu_{n,-1}^{(\beta,\alpha)}\tilde{P}_{n-1}^{(\alpha,\beta+1)}, \tag{3.4}$$

where $\mu_{n,0/1}^{(\alpha,\beta)}$ and $\nu_{n,0/-1}^{(\alpha,\beta)}$ are constants for which we derive explicit formulae in appendix A.1. Formulae (3.1)-(3.4) are the main ingredients for our results.

The spectral expansion of a function $f(r)$ in Jacobi polynomials is defined by the modal coefficients $\hat{f}_n^{(\alpha,\beta)}$. For any $f \in L^2_{(\alpha,\beta)}$ we define the modal coefficients

$$\hat{f}_n^{(\alpha,\beta)} = \left\langle f, \tilde{P}_n^{(\alpha,\beta)} \right\rangle_{(\alpha,\beta)}.$$

We also have a Parseval-type relation:

$$\|f\|^2_{(\alpha,\beta)} = \sum_{n=0}^{\infty} \left[\hat{f}_n^{(\alpha,\beta)}\right]^2.$$

Naturally, $\hat{f}_n^{(a,b)}$ is well-defined if $a \geq \alpha$ and $b \geq \beta$ because of the inclusion $L^2_{(\alpha,\beta)} \subseteq L^2_{(a,b)}$. We recall the definition of the projection operator

$$\mathcal{P}_N^{(\alpha,\beta)} f = \sum_{n=0}^{N-1} \hat{f}_n^{(\alpha,\beta)} \tilde{P}_n^{(\alpha,\beta)}.$$

By completeness and orthogonality, this operator satisfies the relations

$$\left| f - \mathcal{P}_n^{(\alpha,\beta)} f \right|_{(\alpha,\beta)} \longrightarrow 0, \qquad n \to \infty$$

$$\left\langle f - \mathcal{P}_n^{(\alpha,\beta)} f, \phi \right\rangle_{(\alpha,\beta)} = 0, \qquad \phi \in \mathcal{B}_n$$

The problem of rewriting an expansion of one class of Jacobi polynomials into an expansion in a different class can be cast as the problem of determining the connection coefficients $\lambda_{n,m}^{(\alpha,\beta,\gamma,\delta)}$ satisfying

$$\tilde{P}_n^{(\alpha,\beta)} = \sum_{m=0}^{n} \lambda_{n,m}^{(\alpha,\beta,\gamma,\delta)} \tilde{P}_m^{(\gamma,\delta)}. \tag{3.5}$$

We clearly have by orthogonality that

$$\lambda_{n,m}^{(\alpha,\beta,\gamma,\delta)} = \left\langle \tilde{P}_n^{(\alpha,\beta)}, \tilde{P}_m^{(\gamma,\delta)} \right\rangle_{(\gamma,\delta)} \tag{3.6}$$

These connection coefficients are explicitly known for a very large class of problems. Askey [5] compiles earlier formulae originated by Gegenbauer to give explicit formulae for the $\lambda_{m,n}^{(\alpha,\beta,\gamma,\delta)}$ for the different cases (a) $\alpha = \gamma$, (b) $\beta = \delta$, and (c) $\alpha = \beta$ and $\gamma = \delta$. These relations can in general be bootstrapped to determine the general coefficients for any $(\alpha, \beta, \gamma, \delta)$. In terms of the hypergeometric function ${}_3F_2$, an explicit expression for the connection coefficients can also been derived from the Rodrigues relation (A.6). See e.g. [4].

An alternative expression has been derived by Maroni and da Rocha [51] utilizing a recurrence relation satisfied by the connection coefficients in conjunction with significant symbolic computational arithmetic. The disadvantage of both of these latter methods is that the computation of these connection coefficients requires a relatively sophisticated algorithm. Askey's method does not actually give the con-

nection coefficients explicitly if α is different from γ and β is different from δ, and the coefficients that are given require the computation of falling factorials and possibly Gamma functions; of course one may evaluate the function ${}_3F_2$, but that is likewise relatively complicated. Maroni's results give explicit formulae for the λ coefficients, but require the computation of several products and sums of Gamma functions for each coefficient.

Even if the coefficients λ can be computed quickly and accurately, in order to transfer modal coefficients of class (α, β) into those of class (γ, δ), the sum

$$\hat{f}_n^{(\gamma,\delta)} = \sum_{m=n}^{\infty} \lambda_{m,n}^{(\gamma,\delta,\alpha,\beta)} \hat{f}_m^{(\alpha,\beta)} \tag{3.7}$$

must be computed in some fashion if the *exact* modes $\hat{f}_n^{(\gamma,\delta)}$ are desired. If we wish to simply approximate the modes, we may assume that $\hat{f}_m^{(\alpha,\beta)} = 0$ for $m > N$ for some large N. In this case, we still must perform the operation (3.7) (with the upper limit truncated to N) for each $n = 0, 1, 2, \ldots, N-1$, which takes $O(N^2)$ operations as written.

The main contribution of this chapter to the problem of finding connection coefficients is given in Section 3.3. We will assume certain restrictions on the values of α, β, γ,and δ and show that the infinite sum (3.7) is actually a finite sum (independent of the truncation order N) allowing us in theory to *exactly* compute the modes $\hat{f}_n^{(\gamma,\delta)}$ from a finite collection of modes $\hat{f}_n^{(\alpha,\beta)}$, and the computation of each mode can be done in an operation count that is n-independent. In practice, we must still compute the connection coefficients $\lambda_{m,n}^{(\gamma,\delta,\alpha,\beta)}$. While e.g. Askey's or Maroni's formulae may in principle be used to accomplish this, we also present a simple algorithm for robustly computing the connection coefficients assuming restrictions (to be given in the next section) on the quartet $(\alpha, \beta, \gamma, \delta)$.

3.3 Jacobi-Jacobi transformations

The purpose of this section is to form a relationship between the expansions $\mathcal{P}_N^{(\alpha,\beta)} f$ and $\mathcal{P}_N^{(\gamma,\delta)} f$ for $\alpha \neq \gamma$ and/or $\beta \neq \delta$. More than that, we will determine a relationship that will allow us to travel between the two expansions robustly and with very little effort. Of course, the speed of this transformation comes with a natural price: this can only be done when $|\delta - \beta|$ and $|\gamma - \alpha|$ are integers.

We begin by proving a lemma that is an inductive application of equations (3.1) and (3.2):

Lemma 3.1 *For any $A, B \in \mathbb{N}$ and $\alpha, \beta > -1$, the following promotion relations hold:*

$$(1-r)^A \tilde{P}_n^{(\alpha+A,\beta)} = \sum_{m=0}^{A} M_{m,n}^{(\alpha,A,\beta)} \tilde{P}_{n+m}^{(\alpha,\beta)} \tag{3.8}$$

$$(1+r)^B \tilde{P}_n^{(\alpha,\beta+B)} = \sum_{m=0}^{B} (-1)^m M_{m,n}^{(\beta,B,\alpha)} \tilde{P}_{n+m}^{(\alpha,\beta)}, \tag{3.9}$$

where the $M_{m,n}^{(\alpha,\beta)}$ are constants. Similarly, we can demote the class parameters $\alpha + A$ and $\beta + B$ down to α and β as well:

$$\tilde{P}_n^{(\alpha,\beta)} = \sum_{m=0}^{A} N_{m,n}^{(\alpha,A,\beta)} \tilde{P}_{n-m}^{(\alpha+A,\beta)} \tag{3.10}$$

$$\tilde{P}_n^{(\alpha,\beta)} = \sum_{m=0}^{B} (-1)^m N_{m,n}^{(\beta,B,\alpha)} \tilde{P}_{n-m}^{(\alpha,\beta+B)} \tag{3.11}$$

Proof The proofs of (3.8) and (3.9) are accomplished via the demotion relations (3.1) and (3.2): repeated application of (3.1) to the left-hand side of (3.8), and

repeated application of (3.2) to the left-hand side of (3.9) yields the desired result.

The relations (3.10) and (3.11) are proven in exactly the same fashion using the promotion relations (3.3) and (3.4). □

Note that we did not give the formulae for the constants $M_{m,n}^{(\alpha,A,\beta)}$ and $N_{m,n}^{(\alpha,A,\beta)}$ in Lemma 3.1. It would be possible to derive such formulae in terms of the $\mu_{n,i}^{(\alpha,\beta)}$ and $\nu_{n,i}^{(\alpha,\beta)}$, but it is of little value. The main use of Lemma 3.1 is in the proof of the following theorem:

Theorem 3.1 *For $A, B \in \mathbb{N}_0$ and $\alpha, \beta > -1$, let $f \in L^2_{(\alpha,\beta)}$. Then*

$$\hat{f}_n^{(\alpha+A,\beta+B)} = \sum_{m=0}^{A+B} \lambda_{n+m,n}^{(\alpha+A,\beta+B,\alpha,\beta)} \hat{f}_{n+m}^{(\alpha,\beta)}, \qquad n \geq 0 \tag{3.12}$$

where the $\lambda_{n+m,n}^{(\alpha+A,\beta+B,\alpha,\beta)}$ are the connection coefficients (3.6).

Proof For ease of exposition, we drop the superscripts on $\lambda_{n+m,n}^{(\alpha+A,\beta+B,\alpha,\beta)}$ in this proof. We use Lemma 3.1 twice on $(1-r)^A(1+r)^B\tilde{P}_n^{(\alpha+A,\beta+B)}$ to show that there exist constants $\Lambda_{n+m,n}$ such that

$$\omega^{(A,B)}\tilde{P}_n^{(\alpha+A,\beta+B)} = \sum_{m=0}^{A+B} \Lambda_{n+m,n}\tilde{P}_{n+m}^{(\alpha,\beta)}.$$

Following this we note that $\hat{f}_n^{(\alpha,\beta)}$ is well-defined because $f \in L^2_{(\alpha,\beta)}$, and $\hat{f}_n^{(\alpha+A,\beta+B)}$

is well-defined because of the inclusion of $L^2_{(\alpha,\beta)}$ in $L^2_{(\alpha+A,\beta+B)}$. Finally, we have

$$\begin{aligned}
\hat{f}_n^{(\alpha+A,\beta+B)} &= \left\langle f, \tilde{P}_n^{(\alpha+A,\beta+B)} \right\rangle_{(\alpha+A,\beta+B)} \\
&= \left\langle f, (1-r)^A (1+r)^B \tilde{P}_n^{(\alpha+A,\beta+B)} \right\rangle_{(\alpha,\beta)} \\
&= \left\langle f, \textstyle\sum_{m=0}^{A+B} \Lambda_{n+m,n} \tilde{P}_{n+m}^{(\alpha,\beta)} \right\rangle_{(\alpha,\beta)} \\
&= \textstyle\sum_{m=0}^{A+B} \Lambda_{n+m,n} \left\langle f, \tilde{P}_{n+m}^{(\alpha,\beta)} \right\rangle_{(\alpha,\beta)} \\
&= \textstyle\sum_{m=0}^{A+B} \Lambda_{n+m,n} \hat{f}_{n+m}^{(\alpha,\beta)}.
\end{aligned}$$

Furthermore, it is easy to show by orthogonality that $\Lambda_{n+m,n} \equiv \lambda_{n+m,n}$, where $\lambda_{n+m,n}$ are the Jacobi connection coefficients (3.6). □

Remark 1 Theorem 3.1 may also be proven by utilizing equations (3.3)-(3.4) and by noting that the projection operator $\mathcal{P}_N^{(\alpha,\beta)}$ is the identity operator on the space of $(N-1)^{\text{st}}$ degree polynomials $\mathcal{B}_N$ for any pair (α, β). In this case it is trivial to see that $\Lambda_{n+m,n} \equiv \lambda_{n+m,n}$.

Theorem 3.1 gives us a special version of the connection coefficient relation (3.7). In the language of section 3.2, we have obtained a result when $\gamma = \alpha + A$ and $\delta = \beta + B$. It says that we can express the modes of higher-class Jacobi expansions in terms of a *finite* linear combinations of lower-class Jacobi expansions. This relation is exact. Roughly speaking, in order to convert a degree-N expansion of class (α, β) to one of class $(\alpha + A, \beta + B)$, we require about $O(N(A+B))$ computations, as long as A and B are both natural numbers.

The above result is likely known by many authors in this field due to the voluminous literature on Jacobi polynomials. In particular, the connection coefficient relations presented by e.g. Askey [5] and Maroni [51] can no doubt be simplified with the assumptions of Theorem 3.1 to produce the result. However we have not seen this observation made in the literature, and so we present the above theorem.

One deficiency in the theorem above is that we do not present explicit values of the connection coefficients $\lambda_{m,n}^{(\alpha+A,\beta+B,\alpha,\beta)}$. While this is a notable loss (especially since in principle such formulae, albeit cumbersome, are present in existing literature), it is of little consequence as we present a rather straightforward algorithm in the next section for computing these connection coefficients. This algorithm is at worst slightly higher in operation count than if we were to derive explicit formulae. Furthermore, the numerical results in section 8 show that the one-time overhead cost we incur by computing these coefficients is insignificant compared to the savings we gain from being able to use the FFT for certain spectral mode calculations.

3.3.1 Computing the transformation

We collect the modes $\left\{\hat{f}_n^{(\alpha,\beta)}\right\}_{n=0}^{N+A+B-1}$ into the vector $\hat{\boldsymbol{f}}^{(\boldsymbol{\alpha},\boldsymbol{\beta})}$. Theorem 3.1 makes it clear that there exists an $N\times(N+A+B)$ connection coefficient matrix $\hat{C}$, dependent on α, β, A, and B such that

$$\hat{\boldsymbol{f}}^{(\boldsymbol{\alpha}+\boldsymbol{A},\boldsymbol{\beta}+\boldsymbol{B})} = \hat{C}\hat{\boldsymbol{f}}^{(\boldsymbol{\alpha},\boldsymbol{\beta})}. \tag{3.13}$$

(We assume that $A, B \in \mathbb{N}_0$.) This relation transports $N + A + B$ modes from an expansion in the polynomials $\tilde{P}^{(\alpha,\beta)}$ to N modes from an expansion in $\tilde{P}^{(\alpha+A,\beta+B)}$. The entries of the matrix $\hat{C}$ are the connection coefficients $\hat{C}_{m,n} = \lambda_{n,m}^{(\alpha+A,\beta+B,\alpha,\beta)}$, but

we only have relatively complicated expressions for the explicit entries of the matrix $\hat{C}$ (see e.g. [5], [51]). In order to more easily construct the matrix $\hat{C}$ we first note that it is a sparse matrix, banded upper-triangular, with about $(A+B)(N+A+B)$ non-zero entries. To construct $\hat{C}$, we essentially write out the proof of Lemma 3.1 via induction. To this end, let us define the following set of matrices: let $U^{(\alpha,\beta)}$ and $V^{(\alpha,\beta)}$ be bidiagonal matrices with entries defined by

$$\left.\begin{aligned} U^{(\alpha,\beta)}_{n,n} &= \mu^{(\alpha,\beta)}_{n,0} \\ U^{(\alpha,\beta)}_{n,n+1} &= -\mu^{(\alpha,\beta)}_{n,1} \\ V^{(\alpha,\beta)}_{n,n} &= \mu^{(\beta,\alpha)}_{n,0} \\ V^{(\alpha,\beta)}_{n,n+1} &= -\mu^{(\beta,\alpha)}_{n,1} \end{aligned}\right\} \quad n = 1, 2, \ldots, N+A+B-1$$

$$U^{(\alpha,\beta)}_{N+A+B,N+A+B} = \mu^{(\alpha,\beta)}_{n,0}$$

$$V^{(\alpha,\beta)}_{N+A+B,N+A+B} = \mu^{(\beta,\alpha)}_{n,0}$$

The matrix $U^{(\alpha,\beta)}$ transforms the modes of an $(\alpha-1,\beta)$ expansion to those of an (α,β) expansion whenever $\alpha > 0$. Similarly, $V^{(\alpha,\beta)}$ transforms the modes of an $(\alpha,\beta-1)$ expansion into those of an (α,β) expansion whenever $\beta > 0$. To define the entries of U and V we have used the demotion relations (3.1) and (3.2). Note, however, that the last mode $N+A+B$ will be incorrect because we require information about mode $N+A+B+1$ (which we don't have) to determine it. Thus, the last output mode will be corrupted. However, given that we are merely computing connection coefficients, this 'corruption' can simply be characterized as a type of filtering of the terminal

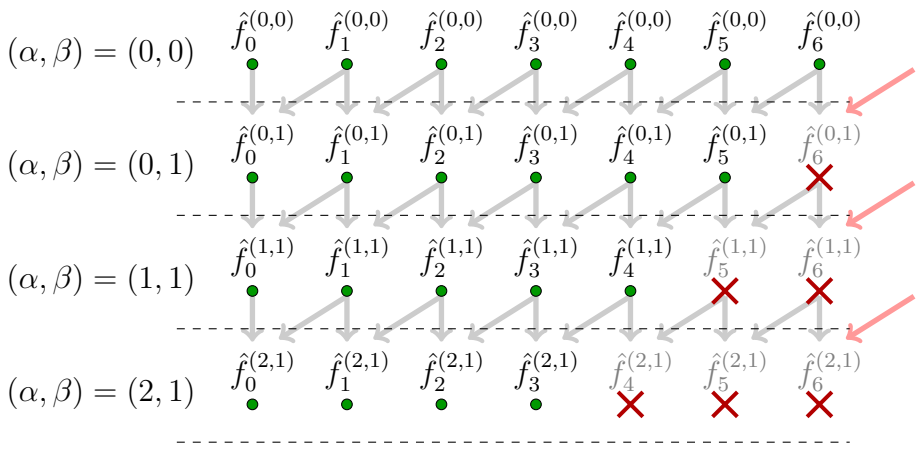

Figure 3.1: Graphical representation of acting on modal coefficients with the matrices U and V. We start with 7 modal coefficients for $(\alpha,\beta) = (0,0)$ (top). Subsequent promotions of α and β require performing the actions specified by the matrices U and V. However, the lack of information about modes $\hat{f}_7^{(0,0)}$, $\hat{f}_8^{(0,0)}$, and $\hat{f}_9^{(0,0)}$ implies that some of the promoted modes are incorrect (denoted by red cross marks).

modes to reproduce the original finite-degree polynomial instead of the true set of modes $\hat{f}_n^{(\alpha+A,\beta+B)}$ for which information is embedded in the unavailable high-degree modes $\hat{f}_n^{(\alpha,\beta)}$. A graphical depiction of computing one set of modal coefficients from another is given in Figure 3.1. This figure also shows how some of the higher-class modal coefficients are lacking information due to knowledge of only finitely many lower-class nodes.

We define a square $(N+A+B)\times(N+A+B)$ matrix C as

$$C = \prod_{b=1}^{B} V^{(\alpha+A,\beta+b)} \prod_{a=1}^{A} U^{(\alpha+a,\beta)} \tag{3.14}$$

The matrix C has some nice properties. As the product of banded bidiagonal matrices, we know that C has non-zero entries on at most $(A+B+1)$ diagonals, and is upper-triangular.

Proposition 3.1 *The matrix C defined by equation (3.14) is invertible and positive-*

definite.

Proof C is upper-triangular, and thus the diagonal entries are the eigenvalues. By utilizing equation (3.14) and expressions (A.11) in the appendix, we have, for all $n = 1, 2, \ldots, N + A + B$

$$C_{n,n} \;=\; \prod_{b=1}^{B} \mu_{n,0}^{(\beta+b,\alpha+A)} \prod_{a=1}^{A} \mu_{n,0}^{(\alpha+a,\beta)} \;>\; 0 \tag{3.15}$$

□

See also [4]. Proposition 3.1 tells us that the matrix C is invertible, which guarantees that we can travel back and forth between the modes $\hat{f}_n^{(\alpha+A,\beta+B)}$ and $\hat{f}_n^{(\alpha,\beta)}$. Clearly this is the case; we did not need Proposition 3.1 to be aware of this in light of the well-posedness of the general connection coefficient problem. In practice, inverting the conventional connection coefficient problem may be difficult because of either the need to generate an additional set of connection coefficients, or because of the need to invert a full linear system. In our case, the inversion can be accomplished via back-substitution because C is banded upper-triangular. The back substitution has a sequential computational cost $O\left(N\left(A+B\right)\right)$, similar to the cost of the formation or application of the matrix. Thus, promoting the parameters (α, β) by (A, B) requires a (parallelizable) $O(N)$ multiplication by C, the demotion by (A, B) requires a sequential $O(N)$ application of C^{-1}.

The positive-definiteness of C is not necessary for our results, but it hearkens to similar questions in the literature about the positivity of the Jacobi connection coefficients λ, e.g. [4]. Note however that all the entries in the matrix C are not always positive; the above proposition only deals with the main diagonal.

With the explicit expression (3.15) for the eigenvalues $C_{n,n}$, we can show that the $N \times N$ matrix C satisfies

$$\lambda_{\min}(C) \sim 2^{-(A+B)/2}, \qquad N \to \infty,$$

where $\lambda_{\min}(C)$ denotes the minimum eigenvalue of C. Since the maximum eigenvalue $C_{1,1}$ is $O(1)$, this provides a bound for the ratio between maximum and minimum eigenvalues for C. We note however that experimentally the ratio of extremal singular values of C (the condition number) seems to be much worse, and by objective standards one would judge the conditioning of C very poorly. However, we show in section 3.5.2 that in practice using the connection matrix is a stable and robust methods available for computing modal expansions.

We tangentially note that it is not entirely necessary to form the matrix C; one can merely store the coefficients $\mu_{n,i}^{(\alpha,\beta)}$ required for the formation of C and view the application (inversion) of C as $A+B$ sequential applications (inversions) of invertible bidiagonal matrices.

Returning to the goal presented at the beginning of this section, we define the non-square matrix $\hat{C}$ as the first N rows of C, and this is exactly the matrix we were looking for. Thus, the formation of $\hat{C}$ (or C) requires $(A+B)$ multiplications between (sparse) bidiagonal matrices.

However, in contrast to the relation presented in (3.13), we prefer an invertible transformation. The reason is that we will eventually use this transform as an ingredient in a modal/nodal transformation. The matrix $\hat{C}$ is not invertible (it's not even square), but of course C is invertible, so we shall frequently use that matrix. (As mentioned before, one should be aware that the terminal $A+B$ $(\alpha+A, \beta+B)$-

modes are not the true modes, but some 'filtered' version of them.)

3.3.2 Properties of the transformation

In this section we assume that we are given a collection of modes $\left\{\hat{f}_n^{(\alpha,\beta)}\right\}_{n=0}^{\infty}$ defining a function $f = \sum_{n=0}^{\infty} \hat{f}_n^{(\alpha,\beta)} \tilde{P}_n^{(\alpha,\beta)}$, and that we wish to obtain some finite number N of the $(\alpha + A, \beta + B)$-modes. We can use the results from the previous section to compute an $N \times N$ matrix C, and an $N \times (N + A + B)$ matrix $\hat{C}$. Define the length-N vectors $\hat{\boldsymbol{f}}_{\boldsymbol{n}}^{(\mathbf{1})}$ and $\hat{\boldsymbol{f}}_{\boldsymbol{n}}^{(\mathbf{2})}$ as

$$\hat{\boldsymbol{f}}_{\boldsymbol{n}}^{(\mathbf{1})} = C \hat{\boldsymbol{f}}_{\boldsymbol{n}}^{(\boldsymbol{\alpha},\boldsymbol{\beta})}$$

$$\hat{\boldsymbol{f}}_{\boldsymbol{n}}^{(\mathbf{2})} = \hat{C} \hat{\boldsymbol{f}}_{\boldsymbol{n}}^{(\boldsymbol{\alpha},\boldsymbol{\beta})}.$$

The vector $\hat{\boldsymbol{f}}_{\boldsymbol{n}}^{(\boldsymbol{\alpha},\boldsymbol{\beta})}$ is of length N in the first expression, and of length $N + A + B$ in the second expression. Note that due to the definition of C (a submatrix of $\hat{C}$), we have that $\hat{f}_n^{(1)} = \hat{f}_n^{(2)}$ for $0 \leq n \leq N - A - B - 1$. Define the corresponding polynomials

$$f^{(1)} = \sum_{n=0}^{N-1} \hat{f}_n^{(1)} \tilde{P}_n^{(\alpha+A,\beta+B)}$$

$$f^{(2)} = \sum_{n=0}^{N-1} \hat{f}_n^{(2)} \tilde{P}_n^{(\alpha+A,\beta+B)}.$$

Application (inversion) of the matrix C is equivalent to performing the connection coefficient problem (3.7) with $\gamma = \alpha + A$ and $\delta = \beta + B$ (respectively, $\gamma = \alpha - A$, $\delta = \beta - B$). This is easy to see if one accepts the statement of Remark 1 (namely that Theorem 3.1 is just a connection coefficient problem). Therefore, it is clear that

$$f^{(1)} = \mathcal{P}_N^{(\alpha,\beta)} f.$$

Function $f^{(2)}$, the application of $\hat{C}$ is a little more subtle, but the result is that

$$f^{(2)} = \mathcal{P}_N^{(\alpha+A,\beta+B)} \mathcal{P}_{N+A+B}^{(\alpha,\beta)} f = \mathcal{P}_N^{(\alpha+A,\beta+B)} f.$$

The latter equality follows from Theorem 3.1, which proves the finite termination of the modal connection expression (3.7). Therefore the two functions $f^{(1)}$ and $f^{(2)}$ are not equal; they differ in the last $A + B$ modes. Both functions can essentially be computed in the same computational time (although it does take about $\frac{1}{2}(A + B)(A + B + 1)$ more operations to compute the N modes $\hat{f}_n^{(2)}$) but they are different projections. Note that the difference between these two functions is indeed important. For example in the resolution of the Gibbs' phenomenon [38] the function required is exactly the projection $\mathcal{P}^{(\alpha+A,\beta+B)} f$; the lower-order projection $\mathcal{P}^{(\alpha,\beta)}$ will not accomplish the task.

3.4 Exploiting the FFT

Up until this point we have not discussed any implementation issues. In this section we explore the use of the FFT for performing spectral transformations based on Jacobi polynomials. Section 3.4.1 introduces some necessary notation for future discussion and section 3.4.2 provides the methodology for using the FFT to compute Jacobi spectral expansions.

3.4.1 Quadrature and interpolation

All the results in this subsection are well-known and we point to the given references for details. In particular, we refer the reader back to Section 2.2.1 which details the definitions and properties of Gauss-type polynomial quadrature. In this chapter, we will adopt the notation $Q_N^{(\alpha,\beta)}$ to mean the N-point Gauss polynomial quadrature for the Jacobi polynomial family of class (α, β).

In many computations we cannot exactly compute the modal coefficients $\hat{f}_n^{(\alpha,\beta)}$ because we cannot compute the integral exactly, or we can only evaluate f but do not know an analytic form for it. Instead, we can use quadrature rules to approximate the modal coefficients:

$$\hat{f}_n^{(\alpha,\beta)} \simeq \tilde{f}_n^{(\alpha,\beta)} := Q_N^{(\alpha,\beta)} \left[f \tilde{P}_n^{(\alpha,\beta)} \right].$$

We can then form the following approximation to $\mathcal{P}_N^{(\alpha,\beta)} f$:

$$\mathcal{I}_N^{(\alpha,\beta)} := \sum_{n=0}^{N-1} \tilde{f}_n^{(\alpha,\beta)} \tilde{P}_n^{(\alpha,\beta)}.$$

Due to the properties of quadrature rules, and the exactness of the Gauss quadrature rule, $\mathcal{I}_N^{(\alpha,\beta)} f = \mathcal{P}_N^{(\alpha,\beta)} f$ for any $f \in \mathcal{B}_N$. Because of the Christoffel-Darboux identity, the expansion $\mathcal{I}_N^{(\alpha,\beta)} f$ is the unique $(N-1)^{st}$ degree polynomial interpolant of f at the nodes $\left\{ r_n^{(\alpha,\beta)} \right\}_{n=1}^{N}$. For $f \notin \mathcal{B}_N$, the difference between the interpolation $\mathcal{I}_N^{(\alpha,\beta)} f$ and the projection $\mathcal{P}_N^{(\alpha,\beta)} f$ is called the aliasing error [43], and arises due to the error in the quadrature rule.

Let $A, B \in \mathbb{N}_0$. We define the following discrete approximations to the N modal

coefficients $\left\{\hat{f}_n^{(\alpha+A,\beta+B)}\right\}_{n=0}^{N-1}$:

$$\tilde{f}_{n;\mathrm{GQ}}^{(\alpha+A,\beta+B)} = Q_N^{(\alpha+A,\beta+B)}\left[f\tilde{P}_n^{(\alpha+A,\beta+B)}\right] \tag{3.16}$$

$$\tilde{f}_{n;C}^{(\alpha+A,\beta+B)} = \sum_{m=0}^{A+B} \lambda_{n+m,n}^{(\alpha+A,\beta+B,\alpha,\beta)} \tilde{f}_{n+m;\mathrm{GQ}}^{(\alpha,\beta)} \tag{3.17}$$

$$\tilde{f}_{n;\mathrm{DQ}}^{(\alpha+A,\beta+B)} = Q_N^{(\alpha,\beta)}\left[f\tilde{P}_n^{(\alpha+A,\beta+B)}\omega^{(A,B)}\right] \tag{3.18}$$

The modes $\tilde{f}_{n;\mathrm{GQ}}$ are obtained using the Gaussian quadrature native to the expansion class $(\alpha+A,\beta+B)$. The modes $\tilde{f}_{n;C}$ are obtained by performing the matrix multiplication in (3.13); i.e., the matrix C (or $\hat{C}$) is used to obtain the modes by promoting the lower-class discerte modes from class (α,β) to class $(\alpha+A,\beta+B)$. The last class of modes $\tilde{f}_{n;\mathrm{DQ}}$ are obtained by using a Gaussian quadrature under the weight $w^{(\alpha,\beta)}$ to simulate quadrature under the weight $w^{(\alpha+A,\beta+B)}$. We call this last expansion one via 'demoted quadrature', which motivates the subscript DQ.

With these three modal definitions, we can define the three expansions $f_i = \sum_{n=0}^{N-1} \tilde{f}_{n;i}^{(\alpha+A,\beta+B)} \tilde{P}_n^{(\alpha}$ for i =GQ,C,DQ. Since GQ represents a strict Gaussian quadrature, we know that $f_{\mathrm{GQ}} = \mathcal{I}_N^{(\alpha+A,\beta+B)} f$. By the results of Section 3.3.2 we also know $f_C = f_{\mathrm{GQ}}^{(\alpha,\beta)} = \mathcal{I}_N^{(\alpha,\beta)} f$, which we formally state:

Corollary 3.1 *The following relation holds:*

$$f_C^{(\alpha+A,\beta+B)} = \mathcal{I}_N^{(\alpha,\beta)} f.$$

The first two expansions GQ and C are exact for any $f \in \mathcal{B}_N$. The last expansion, DQ, is only exact if $f \in \mathcal{B}_{N-A-B}$. However, it is important to notice that all of the

three expansions are different for $f \notin \mathcal{B}_N$ because the aliasing error for each method is different.

3.4.2 Chebyshev Interpolants

We consider the following problem: Given a function $f(r)$ for $r \in [-1, 1]$, a Jacobi class $(\alpha, \beta) = \left(-\frac{1}{2} + A, -\frac{1}{2} + B\right)$, and a maximum modal number N, use an N-point quadrature rule to compute an approximation to the first N modes. A standard practice is to compute the Gauss-Jacobi interpolant $f_{n;\mathrm{GQ}}^{(\alpha,\beta)}$ (3.16) using the Gaussian quadrature native to class (α, β), or to apply the associated Vandermonde interpolating matrix. The total operation count for either of these equivalent methods is $O\left(N^2\right)$.

However, given the results in Section 3.3, we can instead compute the modes for f in class $\left(-\frac{1}{2}, -\frac{1}{2}\right)$ and then translate them to class (α, β) using the matrix C (or $\hat{C}$). Computing the modes for f in class $\left(-\frac{1}{2}, -\frac{1}{2}\right)$ (i.e. the Chebyshev modes) can be accomplished with a fast Fourier transform [43]. Therefore, for general classes $\left(-\frac{1}{2} + A, -\frac{1}{2} + B\right)$ with any natural numbers A, B we can compute the modes using an FFT coupled with a sparse matrix-multiply. This results in the function $f_C^{(\alpha,\beta)}$.

In the determination of the expansion coefficients $\tilde{f}_{n;\mathrm{GQ}}$ and $\tilde{f}_{n;C}$, we can characterize overhead and online computational times:

Overhead Time (GQ)

- Computation of the quadrature rule. This requires evaluation of the values $\tilde{P}_{n-1}^{(\alpha,\beta)}\left(r_m^{(-1/2,-1/2)}\right)$ for $m, n = 1, \ldots, N$. The total cost for this is about $O\left(N^2\right)$

Online Time (GQ)

- Performing the quadrature rule evaluation in (3.16). This is basically a matrix-vector multiplication requiring $O\left(N^2\right)$ computations.

Overhead Time (C)

- Precomputations for the FFT. This involves a small prime factorization of N and storage of function evaluations.
- Computing the shift to transform standard Fourier modes, the output of the FFT, to the Chebyshev modes that we desire. This is $O(N)$.
- Calculating the entries in the matrix C. This is an $O\left(N\left(A+B\right)\right)$ operation.

Online Time (C)

- Performing the FFT to recover the Fourier modes and transforming them to the Chebyshev modes. $O(N \log N)$
- Multiplication by the sparse matrix C. This requires $O\left(N\left(A+B\right)\right)$ operations.

To summarize, the GQ method has $O\left(N^2\right)$ complexity for both the overhead and online times. The C method is $O\left(N\left(\log N + A + B\right)\right)$ total. The GQ method produces the modes $\tilde{f}^{(\alpha,\beta)}_{n;\mathrm{GQ}}$, and the C method produces the modes $\tilde{f}^{(\alpha,\beta)}_{n;C}$.

We will use these characterizations of the online and overhead times for our results in the next section.

3.5 Numerical examples

This section is devoted to presenting quantitative evidence showing that the machinery we have developed is more robust than straightforward Gaussian quadrature, and more efficient than direct quadrature methods for computing Jacobi spectral coefficients if 2α and 2β are odd integers.

3.5.1 Efficiency

In this section we will test the theoretical methods developed in Section 3.3 applied to Chebyshev-Jacobi polynomial expansions $\left(-\frac{1}{2}+A,-\frac{1}{2}+B\right)$ as described in Section 3.4.2. We take the test function

$$f(r)=\exp\left(-5\left(r-\frac{\pi}{6}\right)^2\right)+\sin(r),$$

which is analytic but does not exhibit any symmetry on the interval $r\in[-1,1]$. Using the algorithms presented in Section 3.4, we can compute two spectral expansions of f: $f_{\mathrm{GQ}}^{(\alpha,\beta)}$ and $f_{C}^{(\alpha,\beta)}$. We split the computer work required into the 'overhead', and 'online' divisions, which are defined in Section 3.4.2. By performing the algorithms delineated in Section 3.4.2, we can measure the time required to perform spectral expansions using the GQ and C methods.

In Figure 3.2 we show the online times for the GQ and C methods for A and B taking values 5 and 10. We see that the online time for the C method with the FFT is relatively insensitive to N, growing only linearly with N. However, the GQ (direct quadrature) method grows with N^2 and is always more expensive than the C method. The time spent for the C method is split almost evenly between the FFT

and application of the matrix C. Application of C becomes the dominant factor when A and B become large.

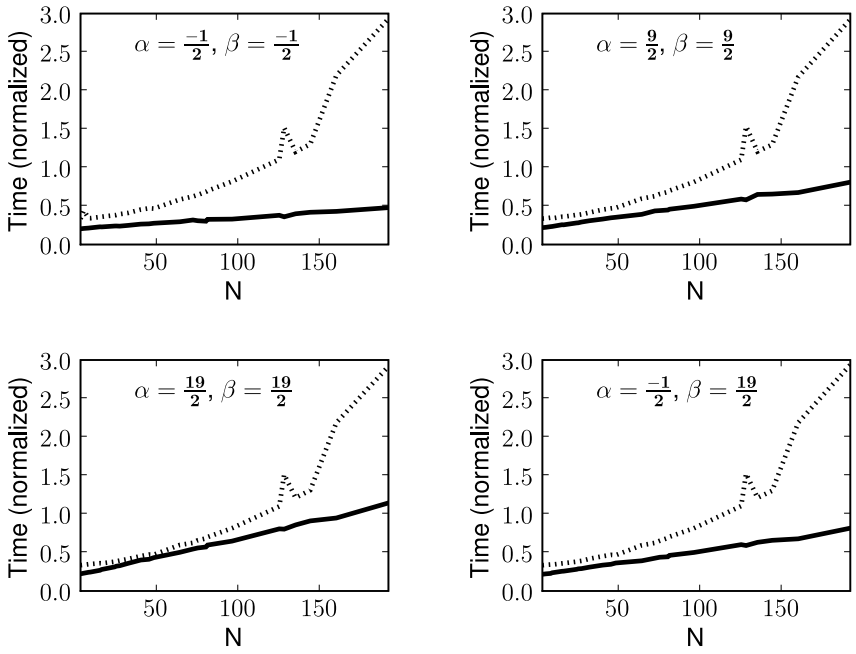

Figure 3.2: Plots of the online times for determining N modal coefficients using N nodal evaluations. FFT+C-matrix calculations are plotted (solid line) vs. direct quadrature calculations (dashed line) for various expansion classes (α, β). The kink around $N = 128$ represents the limitations of the hardware used to perform computations and does not reflect the algorithm complexity.

It is worth noting that the timings in Figure 3.2 should be taken with a grain of salt: computational timings for the FFT vs quadrature for small N are extremely sensitive to the method of coding, the particular libraries used, the compiler, and the computer hardware. However, it is clear that for large N the FFT-based approach is undoubtedly faster than a direct application of quadrature, and that this speed advantage remains despite the cost of multiplication by C.

In Figure 3.3 we show the same results as in Figure 3.2, but we focus on how the parameters A and B affect the online time required for the C method. As expected, the slope of the linear relation between computational time and N is directly related

to $A + B$. In the cases $(\alpha, \beta) = \left(\frac{9}{2}, \frac{9}{2}\right)$ and $(\alpha, \beta) = \left(-\frac{1}{2}, \frac{19}{2}\right)$, we have $A + B = 10$, and the online time required is almost identical.

Another topic to consider is the overhead time required to compute the connection matrix for the C method or the quadrature rule for the GQ method. In Figure 3.4 we show these results. We see that even for small N the overhead time required for the C method is less than that required for the quadrature method. Note that we have even given the GQ method a bit of an advantage: in Section 3.4.2, we defined the overhead time for the GQ method, and we did *not* include the time required to form the Gaussian quadrature rule nodes and weights $r_n^{(\alpha,\beta)}$ and $\omega_n^{(\alpha,\beta)}$, respectively. This requires an additional $O\left(N^2\right)$ operations (via the Golub-Welsch algorithm); we did not include it because in practice one may not compute f_{GQ}, but instead the demoted quadrature expansion f_{DQ} from (3.18) since the Chebyshev-Gauss quadrature rule has an analytic form and is very easy to compute [43]. However, the DQ quadrature becomes more and more inaccurate as A and B are increased if N is fixed. The 'quadrature' overhead time plotted in Figure 3.4 is actually that for the DQ method, and not the GQ method.

3.5.2 Accuracy

We present in this section examples to show that an algorithm that judiciously uses the connection coefficients is just as or more accurate than a direct Gaussian quadrature method, and accomplishes the task in much less time. We use the test function

$$f(r) = \sin\left(q\pi r + \frac{\pi}{4}\right) = \frac{1}{\sqrt{2}}\left[\sin(q\pi r) + \cos(q\pi r)\right]. \tag{3.19}$$

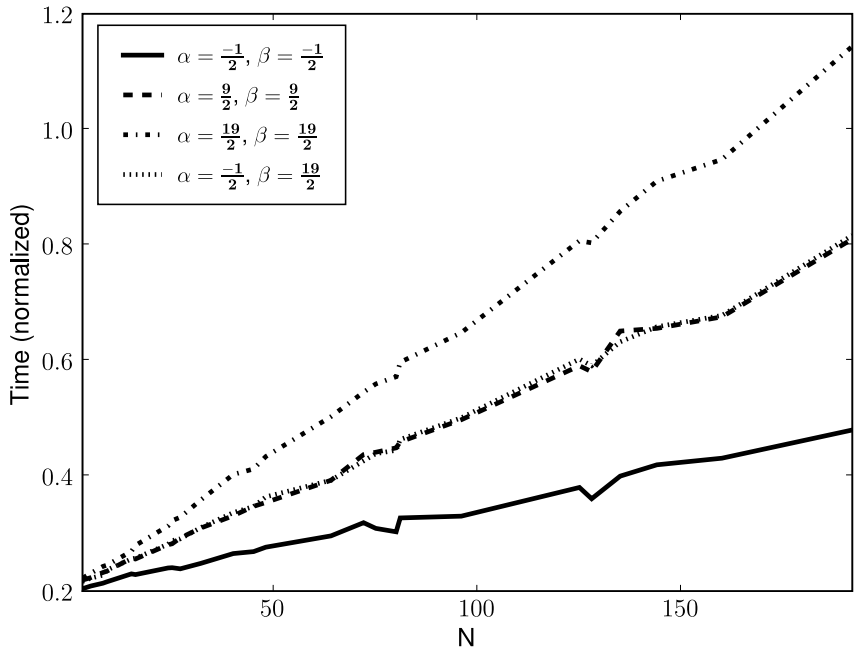

Figure 3.3: Plot of the online computational time required for the C method vs. N. The results plotted are the same as in Figure 3.2.

If we wish to expand a function $f(r)$ in symmetric Jacobi (i.e. ultraspherical/Gegenbauer) polynomials, we obtain

$$f(r) = \sum_{n=0}^{\infty} \hat{f}_n^{(\alpha,\alpha)} \tilde{P}_n^{(\alpha,\alpha)}(r).$$

In this case, we can find an exact formula for the spectral coefficients. From e.g. Lemma 2.5 in [37] we have:

$$\begin{aligned}
\hat{f}_n^{(\alpha,\alpha)} &:= \left\langle f, \tilde{P}_n^{(\alpha,\alpha)} \right\rangle_{(\alpha,\alpha)} \\
&= \frac{1}{\pi^\alpha q^{\alpha+1/2}} \sqrt{\frac{\Gamma(n+2\alpha+1)(n+\alpha+1/2)}{n!}} J_{n+\alpha+1/2}(q\pi) \times \\
&\qquad \begin{cases} (-1)^{(n-1)/2}, & n \text{ odd} \\ (-1)^{n/2}, & n \text{ even}, \end{cases}
\end{aligned}$$

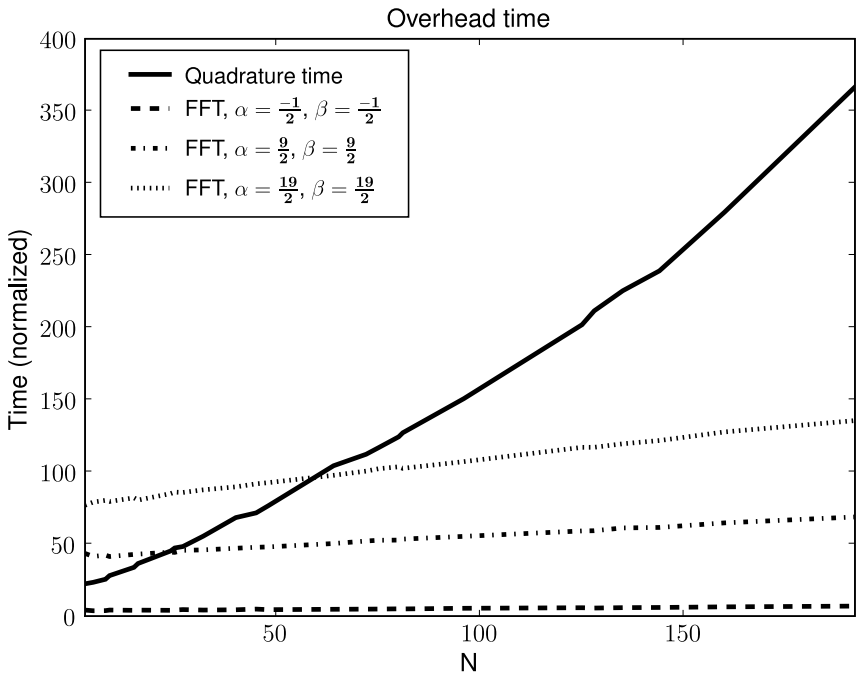

Figure 3.4: Plot of the overhead times vs. number of nodal/modal degrees of freedom N. The overheads are calculated for the f_{GQ} (direct quadrature) and f_C (FFT) methods.

where $J_\nu(x)$ is the Bessel function of the first kind, and Γ is the Gamma function. In the case of $\alpha = -\frac{1}{2}$ with $n = 0$, we must divide the above by $\sqrt{2}$. One of the standard ways of numerically obtaining approximations to the modes $\hat{f}_n^{(\alpha,\alpha)}$ is to use the Gaussian quadrature rule $Q_N^{(\alpha,\alpha)}$ native to the Jacobi polynomial class (α, α). The argument we make here is that it is better to determine coefficients to a different Jacobi polynomial class and then use the connection coefficient relations we have derived to obtain the desired modes. Normally we would incur an additional $O\left(N^2\right)$ cost for this procedure, but if we only apply the connection between classes with integral-valued separation, we can accomplish this task in only $O(N)$ cost, which is the result from Theorem 3.1. We thus identify three methods for obtaining approximations to the modes $\left\{\hat{f}_n^{(\alpha,\alpha)}\right\}_{n=0}^{N-1}$, which are derived from the quadrature methods (3.16–3.18):

- Native Gauss quadrature (cf. equation (3.16))
  - Determine the quadrature rule $Q_N^{(\alpha,\alpha)}$
  - Apply $Q_N^{(\alpha,\alpha)}$ to the nodal evaluations $f\left(r_n^{(\alpha,\alpha)}\right)$ to obtain approximations to the modes. The function recovered is $\mathcal{I}_N^{(\alpha,\alpha)} f = f_{\text{GQ}}^{(\alpha,\alpha)}$
- Demoted Gauss quadrature (cf. equation (3.18))
  - Define $A := \lceil \alpha \rceil$ and $\underline{\alpha} := \alpha - A \in (-1, 0]$. Determine the quadrature rule $Q_{N+2A}^{(\underline{\alpha},\underline{\alpha})}$
  - Apply $Q_{N+2A}^{(\underline{\alpha},\underline{\alpha})}$ to the nodal evaluations $f\left(r_n^{(\underline{\alpha},\underline{\alpha})}\right) w^{(A,A)}\left(r_n^{(\underline{\alpha},\underline{\alpha})}\right)$ to obtain approximations to the modes N modes $\hat{f}_n^{(\alpha,\alpha)}$. The function recovered is $f_{\text{DQ}}^{(\alpha,\alpha)}$
- Demoted Gauss quadrature + FFT (useful only when $\alpha - \lceil \alpha \rceil = -\frac{1}{2}$) (cf. equation (3.17))
  - Define $A := \lceil \alpha \rceil$ and $\underline{\alpha} := \alpha - A$. Determine the quadrature rule $Q_{N+2A}^{(\underline{\alpha},\underline{\alpha})}$
  - Apply $Q_{N+2A}^{(\underline{\alpha},\underline{\alpha})}$ to the nodal evaluations $f\left(r_n^{(\underline{\alpha},\underline{\alpha})}\right)$ to obtain approximations to the modes $N + 2A$ modes $\hat{f}_n^{(\underline{\alpha},\underline{\alpha})}$. This can be done with the FFT if $\underline{\alpha} = -\frac{1}{2}$. The function recovered here is $\mathcal{I}_{N+2A}^{(\underline{\alpha},\underline{\alpha})} f$.
  - Apply the connection matrix $\hat{C}$ to recover $\hat{f}_n^{(\alpha,\alpha)}$, which are the exact modes for the function $\mathcal{P}_N^{(\alpha,\alpha)} \mathcal{I}_{N+2A}^{(\underline{\alpha},\underline{\alpha})} f$.

Note that in the above algorithms we have chosen an N-point quadrature rule for the native Gauss quadrature and demoted quadrature, and an $(N + 2A)$-point rule for the demoted quadratures; we need the extra nodes for the demoted quadrature rule in order to claim exactness for $f \in \mathcal{B}_N$. While this initially seems like a disadvantage, our results show that the demoted quadrature rule is more accurate than the native

Gauss quadrature, especially for large α. The reasoning comes from the observation in e.g. [43] that the Gauss quadrature nodal points tend to approach the equidistant nodal set as α is increased. It is well-known that the equidistant nodal set for interpolation produces a much worse quality interpolant than a nodal set where the nodes cluster near the boundaries.

Take specifically the case when $\alpha = -\frac{1}{2} + A$ and $\beta = -\frac{1}{2} + B$. We know that $f_{\text{GQ}} = \mathcal{I}_N^{(\alpha,\beta)} f$, and from Corollary 3.1 that $f_C = \mathcal{I}_N^{(-1/2,-1/2)} f$. Again, the nodes $\left\{ r_n^{(\alpha,\beta)} \right\}_{n=1}^{N}$ approach the equidistant nodal set as α and β are increased. The equidistant nodal set for polynomial interpolation has an exponentially growing Lebesgue constant [52], whose magnitude bounds the maximum pointwise error of an interpolant. Because large (α, β) will yield an interpolant f_{GQ} on near-equidistant nodes, we have good reason to suspect that f_{GQ} will be a bad pointwise estimator for the function f. By contrast, it is also known that the Chebyshev-Gauss nodes have a nearly-optimal Lebesgue constant [40], of order $\log(N)$. Therefore, the function f_C is an orders-of-magnitude better pointwise interpolant than f_{GQ} for large α, β, especially near the endpoints of the interval. This is why we 'demote' α to $\underline{\alpha}$ in order to obtain a better nodal set for interpolation (even when $\underline{\alpha} \neq -1/2$).

We recall that the last $A + B$ modes of the f_C expansion are incomplete in information. Therefore, if one discards these last few modes (effectively using $\hat{C}$ instead of C) we have an approximation to $\mathcal{P}_N^{(-1/2+A,-1/2+B)} f$. However, it is also known that for large A, B that projection also exhibits very ill-behaved oscillations; this is termed the generalized Runge phenomenon [18]. It is interesting to note that the only difference between a 'well-behaved' Chebyshev projection $\mathcal{P}_N^{(-1/2,-1/2)} f$, and a projection $\mathcal{P}_N^{(\alpha,\alpha)} f$ exhibiting the ill-desired generalized Runge phenomenon for large α, is simply the information embedded in the $2\alpha+1$ Chebyshev modes after $\hat{f}_N^{(-1/2,-1/2)}$.

Returing to the case when $\underline{\alpha} = -\frac{1}{2}$, we can use the FFT for the demoted quadrature rule (since then $Q_N^{(\underline{\alpha},\underline{\alpha})}$ is the Chebyshev quadrature rule), which mitigates the marginal increase in computational cost of using $N + 2A$ modes. Thus, we have strong reason to believe that using the demoted quadrature rule has accuracy advantages over the native Gauss rule, and if $\underline{\alpha} = -\frac{1}{2}$, the small increase in cost is largely offset by the use of the FFT.

To quantitatively validate these claims, we assume that we obtain a function g using one of the above methods, and we introduce an error measure ε_{rel} defined as

$$\varepsilon_{\text{rel}} = \frac{\left\| \mathcal{P}_N^{(\alpha,\alpha)} f - g \right\|_{(\alpha,\alpha)}}{\left\| \mathcal{P}_N^{(\alpha,\alpha)} f \right\|_{(\alpha,\alpha)}}.$$

In Figure 3.5 we show the scaled error ε_{rel} as a function of α for the function f in (3.19) with $q = 80$ for the native Gauss quadrature and demoted quadrature methods. We have used 500 quadrature points (for both methods) and $N = 300$ modes were computed. For large values of α, the the Gauss rule produces ill-conditioned results, whereas the demoted quadrature rule can provide very accurate modal information for $\underline{\alpha}$ that can be promoted robustly to modal information for α using $\hat{C}$.

In addition, we tabulate ε_{rel} and computational times in Table 3.1 for various values of α for which $\underline{\alpha} = -\frac{1}{2}$. Using demoted quadrature in conjunction with the FFT is both more accurate and much faster than using a native Gauss quadrature. Although the error for the native Gauss rule does not seem significant for $\alpha \lesssim 20$, a modification of the parameter q in equation (3.19) can greatly exacerbate the accuracy problem for Gauss quadrature and further motivates the desire to use the demoted quadrature rule, even for small α.

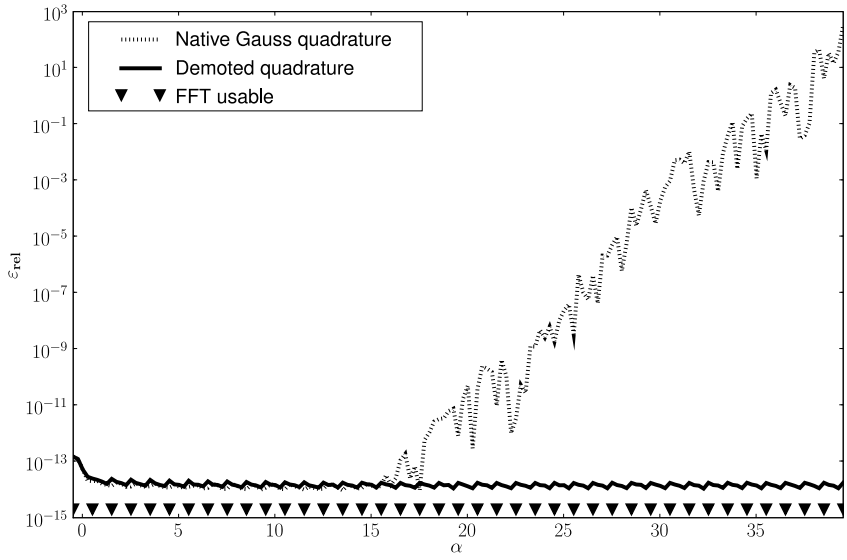

Figure 3.5: The relative error ε_{rel} for the native Gauss and demoted quadrature rule methods. 500 interpolation points are used in both cases, and $N = 300$ modes are computed.

Finally, we note that the FFT can be more accurate than direct methods for Chebyshev-Gauss quadrature if accurate methods are employed to compute the overhead sine/cosine table factors [57]. We have not made an attempt to verify the accuracy of our FFT code against direct methods. However, the argument that the FFT is more accurate than direct methods (if implemented properly), along with the observation that demoted quadrature yields more accurate results than native Gauss quadrature, makes a strong case for both the stability and efficiency of our proposed method.

3.6 Summary

We have presented an inexpensive algorithm for transforming modal coefficients of a Jacobi polynomial expansion of class (α, β) to the modal coefficients of class $(\alpha + A, \beta + B)$ for $A, B \in \mathbb{N}_0$. Such an exact transformation is possible due to the

ε_{rel} error

	Native Gauss	Demoted	Demoted + FFT*
$\alpha = -0.5$	1.495 E-13	1.495 E-13	3.228 E-14
$\alpha = 9.5$	1.170 E-14	1.869 E-14	4.422 E-14
$\alpha = 19.5$	8.142 E-13	1.739 E-14	4.475 E-14
$\alpha = 29.5$	1.197 E-04	1.729 E-14	4.509 E-14
$\alpha = 39.5$	3.347 E+02	1.759 E-14	4.556 E-14

Computational Time (s)

	Native Gauss	Demoted	Demoted + FFT*
$\alpha = -0.5$	0.326	0.323	0.001
$\alpha = 9.5$	1.156	0.324	0.009
$\alpha = 19.5$	1.187	0.327	0.018
$\alpha = 29.5$	1.209	0.325	0.025
$\alpha = 39.5$	1.225	0.327	0.032

Table 3.1: ε_{rel} and computational time tabulations for various values of α. ($*$ FFT only available when α is an odd multiple of $-1/2$).

sparsity of the connection coefficients relating different Jacobi polynomial classes. This transformation can also be efficiently implemented, transforming N modes in $O\left(N\left(A+B\right)\right)$ computations. In the case where $\alpha = \beta = -\frac{1}{2}$, one can apply the FFT to compute modal expansions of class $\left(-\frac{1}{2}+A, -\frac{1}{2}+B\right)$ in $O(N \log N)$ time. For large N and large A and B the FFT algorithm proves to be a more efficient method for computing modal coefficients than direct quadrature. In addition, we observe that using 'demoted quadrature' produces much more accurate results than using 'native Gauss quadrature', which we attribute to the decreased pointwise accuracy of the Gauss interpolant versus the demoted quadrature interpolant.

Although our numerical examples have concentrated on the Jacobi-Gauss quadrature, the method is equally applicable to the Radau and Lobatto quadratures as well, since Chebyshev-Gauss-Radau and Chebyshev-Gauss-Lobatto quadrature rules are FFT-implementable as well. In addition, fast algorithms for Legendre polynomial expansions also exist (e.g. [2], [30], [31]), which means that this algorithm also yields

asymptotically $O(N \log N)$ spectral transformations for polynomial expansions in any class of the form (A, B) as well when combined with one of the fast Legendre algorithms. We also note that Boyd provides the possibility of using the fast multipole method (FMM) for evaluating modal coefficients on Gaussian quadrature nodes for *any* values of (α, β) [16]. This enables the possibility of an $O(N \log N)$ algorithm for the Gauss quadrature method as well; but as mentioned in [16], while the FMM has the same asymptotic complexity as the FFT, it unfortunately has a much larger constant of proportionality than the FFT, so the FFT is vastly superior if one can use it.

There is still room for improvement for our method: we have used packaged FFT routines, but for our purpose of calculating Chebyshev modes, the method is more efficient by a factor of 2 if instead we use fast cosine transform routines. In addition, we have assumed for given A and B that exactly $(A+B+1)\left(N - \frac{(A+B)}{2}\right)$ elements of C are nonzero (that is, the first $A + B + 1$ superdiagonals). However, if $\alpha = \beta$ and $A = B$, then the odd superdiagonals are actually zero (this symmetry in the connection coefficients for Gegenbauer polynomials is known [5]) . This means that an online application of C for Gegenbauer/Ultraspheric expansions can be optimized by a factor of roughly 2, which we have not done in this paper. Because application of C is usually on the same order of magnitude as application of the FFT, this can result in a significant speedup of the method.

Finally, the inverse transform from modes of order $\left(-\frac{1}{2} + A, -\frac{1}{2} + B\right)$ to Chebyshev-Gauss(-Radau/Lobatto) nodes is also possible via the FFT: one must back-substitute to solve the system

$$\tilde{\boldsymbol{f}}^{(\boldsymbol{\alpha}+\boldsymbol{A},\boldsymbol{\beta}+\boldsymbol{B})} = C\,\tilde{\boldsymbol{f}}^{(\boldsymbol{\alpha},\boldsymbol{\beta})},$$

for the modes $\tilde{f}^{(\alpha,\beta)}$, which is a sequential operation with nearly the same number

of operations as the forward application of C. (Recall by proposition 3.1 that C is invertible.) Then if $\alpha = \beta = -\frac{1}{2}$, one can use the FFT to recover the nodal evaluations.

Both the forward problem of determining modal coefficients for general classes of Jacobi polynomial expansions, and the backward problem of computing nodal evaluations given Jacobi modes are critical ingredients in our ability to use the FFT for the generalized Wiener basis set over the infinite interval, as we discuss later.

Chapter Four

The Lebesgue Constant For The Jacobi-Gauss Nodes

4.1 Introduction

This chapter, like the previous one, is largely self-contained with respect to the material. We build on the notion of the 'demoted' quadrature rules explored in the previous chapter by seeking Lebesgue-optimal interpolation sets in the framework of Jacobi-Gauss quadrature. These results are not directly applicable to our ultimate goal of the generalized Wiener basis set, but we will uncover some very interesting and curious properties.

The one-dimensional interpolation of a set of data points $(r_k, y_k)_{k=1}^K$ on a finite interval has long been accomplished via polynomial approximation. For certain smoothness conditions on the true function from which the data points were obtained, and for certain choices of the nodal locations r_k, the mean-square error between the polynomial interpolant and the true function converges to zero with a certain rate dependent on the number of data points interpolated K [8]. However, if one wishes to consider the pointwise error, it can be shown that for *any* choice of the nodes r_k there exists a continuous function for which the error diverges to infinity. This rather discouraging fact has prompted many to begin searching for an optimal set of nodes r_k, where 'optimal' can mean a variety of things.

There are many criteria one may use to judge a nodal set $\{r_k\}_{k=1}^N$. One of the more complete discussion is given by Boyd [17], where he considers (among other criteria) Lebesgue-optimality, VDM-optimality, and Cauchy optimality. While all three of these metrics have merit, we shall concentrate here on Lebesgue-optimality; that is, minimizing the Lebesgue constant. The Lebesgue constant is intimately connected with pointwise error, as it is the norm of the interpolation operator.

The question of finding nodal sets with minimal Lebesgue constants for a general

function is still an open question. A linear programming algorithm [3] has existed for many years to find optimal nodal sets in one dimension, but it is not very computationally efficient. It is based on the strong characterization of the Lebesgue-optimal nodal set provided by the Bernstein-Erdös conjecture, which was proven independently and simultaneously in [47] and [9]. There have been many comparisons between easily-computable nodal sets and optimal nodal sets [42], [17], [24], [25].

In previous chapters we've considered the classical Jacobi polynomials $P_n^{(\alpha,\beta)}$. In [42] the Gauss quadrature nodes associated with this orthogonal polynomial family was compared to other Lebesgue-favorable sets and was found to be rather competitive. In this chapter we will explore this observation more carefully in the pursuit of determining quasi-optimal interpolation nodes in one and two dimensions.

Section 4.2 gives an overview of interpolation theory and basic results from the literature regarding Lebesgue optimality. In Section 4.3 we use a brute-force numerical method to compare Lebesgue-optimal nodal sets with Jacobi-Gauss quadrature nodal sets. Finally, Section 4.4 gives an interesting extension to two dimensions based heavily on an existing nodal construction on triangle simplices given by Warburton [66].

4.2 Mathematical Preliminaries

4.2.1 Polynomial Interpolation

We take $I_r = [-1, 1]$ as the reference interval in the rest of the chapter. Any other finite interval may be considered via an affine map. We consider a set of points $\{r_k\}_{k=1}^N$ as the nodes for polynomial interpolation and define the nodal basis functions

$$l_{k,N}(r) = \prod_{\substack{j=1,\\ j\neq k}}^{N} \frac{(r - r_j)}{(r_k - r_j)},$$

which are called the Lagrange interpolating polynomials and are the unique polynomials of degree $N-1$ with the property $l_{k,N}(r_j) = \delta_{k,j}$ for $j, k = 1, 2, \ldots, N$. In an effort to keep notational complexities to a minimum we omit the dependence of r_k on N. Then given a function $f : I_r \to \mathbb{R}$, the interpolant of this function is defined as

$$\mathcal{I}_N [r_k] f = \sum_{k=1}^{N} f(r_k) l_{k,N}(r).$$

To determine the quality of an interpolant, we appeal to a classic result of Lebesgue. To facilitate this, we introduce the Lebesgue constant

$$\Lambda_N (r_k) = \max_{r \in I_r} \sum_{k=1}^{N} |l_{k,N}(r)|$$

Lebesgue's result bounds the interpolant error

$$\|f - \mathcal{I}_N[r_k] f\|_\infty \leq [1 + \Lambda(r_k)] E_N^*, \tag{4.1}$$

where E_N^* is defined as the minimum pointwise error over all polynomials of degree $N-1$ or less, i.e.,

$$E_N^* = \min_{p \in \mathcal{B}_N} \|f - p\|_\infty.$$

The polynomial p that realizes the minimum is the best approximating polynomial of order $N-1$. Equation (4.1) effectively says that the overall error is a balance between the Lebesgue constant magnitude and the (interpolation node independent) approximation error E_N^*. It is known that the Lebesgue constant grows in N for any choice of nodes. Because the Lebesgue constant is a function only of the interpolation nodes, a judicious choice of nodes can therefore lead to good interpolant quality. Searching for a Lebesgue constant minimizing set of nodes is the central theme of this chapter.

We are thus led to the question of finding the nodal set for a given N that minimizes the Lebesgue constant Λ_N, starting in one dimension. The minimizing nodal set is not unique as specified. However, if we insist that two of the nodes are located at the endpoints $r_1 = -1$ and $r_N = +1$, then the nodal set that minimizes the Lebesgue constant is unique. We shall mostly consider nodal sets containing the endpoints in this chapter. We define the optimal set of nodes r_n^* as the unique nodal locations that minimize the Lebesgue constant. It has been shown that the optimal value of the Lebesgue constant has the asymptotic expression [65]

$$\Lambda_N^* \sim \frac{2}{\pi} \log N + \frac{2}{\pi} \left(\gamma + \log \frac{4}{\pi} \right) + O\left(\frac{\log \log N}{\log N} \right)^2 \tag{4.2}$$

Examples of existing nodal sets with asymptotic expressions for their associated Lebesgue constants are

- Equidistant nodes: define

$$r_n^{\text{eq}} = 1 - 2\left(\frac{n-1}{N-1}\right), \quad n = 1, 2, \ldots, N$$

The Lebesgue constant grows exponentially:

$$\Lambda_N^{\text{eq}} := \Lambda_N\left(r_n^{\text{eq}}\right) \sim \frac{2^N}{N \log N} \tag{4.3}$$

- The 'optimum' boundary nodes. No closed-form expression is known for these nodes, but for $N \leq 20$ the positions and values of the Lebesgue constant are given in [24] and are consistent with the estimate in equation (4.2).

- The 'extended' Chebyshev nodes (e.g. [40]):

$$r_n^{\text{ec}} = \frac{\cos\left(\frac{\pi}{2N}(2n-1)\right)}{\cos\left(\frac{\pi}{2N}\right)}, \quad n = 1, 2, \ldots, N. \tag{4.4}$$

The Lebesgue constant is almost optimal:

$$\Lambda_N^{\text{ec}} := \Lambda_N\left(r_n^{\text{ec}}\right) \sim \frac{2}{\pi}\log N + \frac{2}{\pi}\left(\gamma + \log\frac{8}{\pi} - \frac{2}{3}\right) + \frac{16}{81\pi^3 \log N} + O\left(\frac{1}{\log\log N}\right)$$

- Jacobi-Gauss-Lobatto quadrature nodes. The nodal set is given by $\left\{r_n^{(\alpha,\alpha);\text{GL}}\right\}$, where α is an unknown parameter. Hesthaven [42] gives numerical values of α for $N < 25$ that minimize Λ_N. The resulting values of Λ_N are the smallest we have found in the literature that do not rely on the linear programming technique introduced in [3] with results in [24]. Our results will be an extension of Hesthaven's work.

4.2.2 Jacobi-Gauss Interpolation

This section is aimed at combining the expositions of Section 4.2.1 with Jacobi-Gauss quadrature nodes introduced in Section 2.2.1. We can use the Jacobi-Gauss nodes $r_k^{(\alpha,\alpha)}$ or the Jacobi-Gauss-Lobatto nodes $r_k^{(\alpha,\alpha);\mathrm{GL}}$ to perform interpolation. To simplify the notation, we define the following operators:

$$\mathcal{I}_N^{(\alpha)} := \mathcal{I}_N \left[r_k^{(\alpha,\alpha))} \right]$$

$$\mathcal{I}_N^{(\alpha);\mathrm{GL}} := \mathcal{I}_N \left[r_k^{(\alpha,\alpha);\mathrm{GL}} \right].$$

As mentioned in the previous chapter, it is well-established that the nodes $r_n^{(\alpha,\alpha);\mathrm{GQ\,/\,GL}}$ tend toward equidistant nodes [43] as $\alpha \to \infty$. Therefore, we expect that $\Lambda_N \left(r_n^{(\alpha,\alpha);\mathrm{GL}} \right)$ will grow to the value given in (4.3) as α is increased. Naturally, since we wish to minimize Λ_N, it makes sense to determine the optimum value of α:

$$\alpha^*(N) := \underset{\alpha\in(-1,\infty)}{\operatorname{argmin}} \Lambda_N \left(r_n^{(\alpha,\alpha);\mathrm{GL}} \right). \tag{4.5}$$

In [42] the values of $\alpha^*(N)$ are given for $N < 24$. The work presented in this paper is an extension of the work in [42] in the sense that we will compute $\alpha^*(N)$ for larger values of N and will give a conjecture for the analytic closed-form expression for α^*.

4.3 Numerical Results

From this point on most of our conclusions are based on heuristic evidence rather than mathematical rigor. We can brute-force the computation in (4.5) and determine

α^* as a function of N. Of course, it is not clear that such a unique value of α^* exists for each N, but our results are rather convincing and intruiging.

4.3.1 Gauss-Lobatto nodes

First we exhibit the behavior of the Jacobi-Gauss-Lobatto Lebesgue constant $\Lambda_N^{(\alpha);\mathrm{GL}} := \Lambda_N\left(r_n^{(\alpha,\alpha);\mathrm{GL}}\right)$ vs. both N and α.

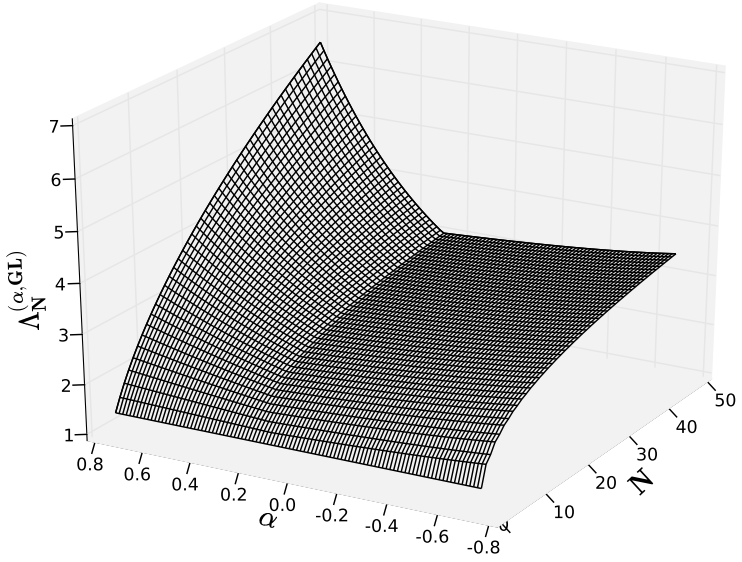

Figure 4.1: Mesh plot of the Lebesgue constant $\Lambda_N^{(\alpha);\mathrm{GL}}$ vs α and N.

The results of Figure 4.1 give us hope that $\alpha^*(N)$ does indeed exist and is unique (see Figure 4.2 for confirmation of this). We see that for values of $\alpha \gtrsim 0.5$, the Lebesgue constant seems to grow much faster than for smaller values of α. Note that $\alpha = 0.5$ are Chebyshev (of the second kind) nodes and it makes sense for those nodes to have small Lebesgue constant. The apparent loss of smoothness of the meshplot shown in Figure 4.1 is not a visual artifact or a numerical one. We are unable to justify this phenomenon regarding why the location of the minimum of the Lebesgue constant

seems to coincide with the only noticable locations where $\Lambda_N^{(\alpha);\mathrm{GL}}$ is not smooth with respect to α and N.

To make a clearer case for the existence of $\alpha^*(N)$ we show N cross-sections in Figure 4.2. It is clear that a minimum value of $\alpha^*(N)$ exists; the first few values of α^* are given empirically in [42].

Another interesting phenomenon to notice is given in Figure 4.3 where we take cross sections of Figure 4.1 for constant values of α. For certain values of α, the Jacobi-Gauss-Lobatto points exhibit a sublinear (log-like) growth, whereas for $\alpha \gtrsim 1.5$, we see that a superlinear growth is observed. There appears to be some critical value of α around 1.5 that separates superlinear and sublinear growth rates (in N) for the Lebesgue constant.

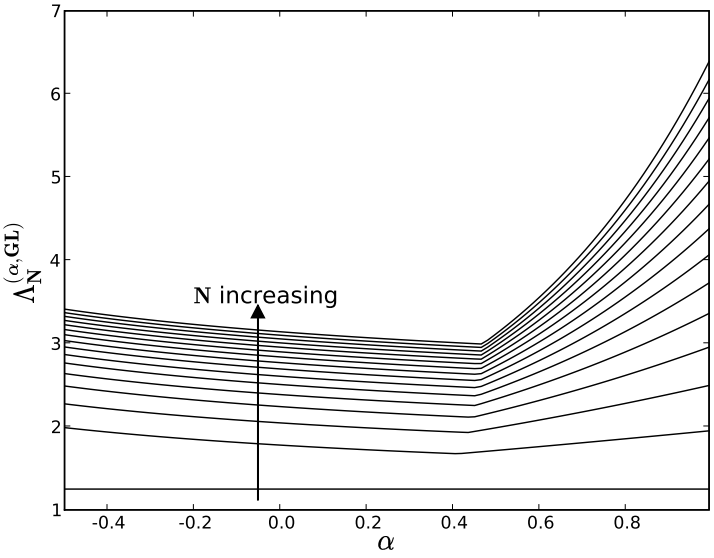

Figure 4.2: Cross sections of figure 4.1 for fixed N. For each value of N, there exists an α that minimizes the Lebesgue constant $\Lambda_N^{(\alpha,\mathrm{GL})}$.

We now move one step beyond the plots in Figures 4.1 through 4.3 and use brute-force computation to determine $\alpha^*(N)$ for $4 \leq N \leq 100$. (For $N = 2$ and 3, all

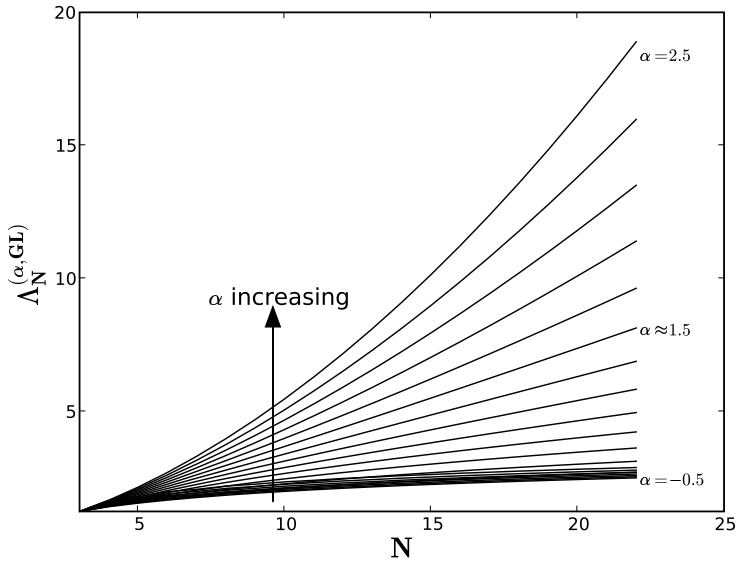

Figure 4.3: Cross sections of figure 4.1 for fixed α. There is a critical value of $\alpha \approx 1.5$ that divides super- and sublinear growth of the Lebesgue constant $\Lambda_N^{(\alpha,\mathrm{GL})}$ in N.

values of α give the same nodes.) We employ a basic sampling procedure for each N to hone in on α^*, and we compute α^* to a tolerance of 10^{-8}. The results for $N \leq 50$ are shown in Figure 4.4. The values of α^* seem to follow a smooth functional form, at least for large N. Although only $N \leq 50$ is shown in the figure, we have carried out the computation to around $N \sim 150$. Using this data, we have been unable to determine the exact form of α^* related to N, but we have determined a very close fit as a $\log^{(4)}$ distribution. The value of the Jacobi parameter α that minimizes the Lebesgue constant $\Lambda_N^{(\alpha);\mathrm{GL}}$ seems to have an asymptotic form, approximated very well by

$$\hat{\alpha}^*(N) \simeq \; c_1 + c_2 \log\left(\log\log\log N + 1\right), \qquad N \geq 5, \tag{4.6}$$

with

$$c_1 = 0.45125396 \approx \tfrac{\sqrt{2}}{\pi},$$

$$c_2 = 0.06240026 \approx \tfrac{\pi}{50}.$$

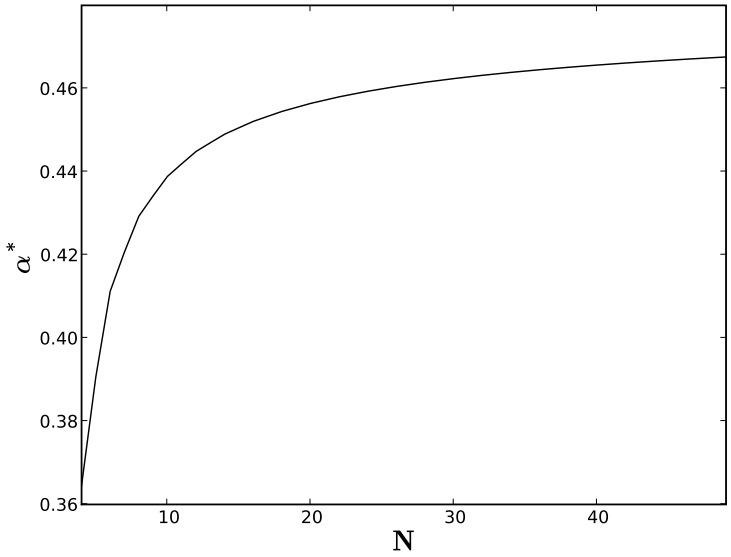

Figure 4.4: Plot of the numerically-determined $\alpha^*(N)$ for $4 \leq N \leq 50$.

The fit we have used in the conjecture is of a $\log^{(4)}$-type. We cannot offer an explanation for this particular functional fit: this choice of function simply seems to fit the data relatively well, but even if this particular form may appear contrived. The right-hand-side approximations for c_1 and c_2 yield a function $\hat{\alpha}^*$ that matches the numerical values for α^* to two decimal places for $5 \leq N \leq 8$, three decimal places for $9 \leq N \leq 18$, four decimal places for $19 \leq N \leq 100$, and six decimal places for $N \gtrsim 100$. We do not need $\alpha^*(N)$ for $N = 2, 3$ since any value of α produces the same (optimal) nodes. For $N = 4$ the formula (4.5) is not defined, but one may instead use the numerically-computed value of α^* that we tabulate in Appendix E.

Since we are taking a perturbation of α^* with $\hat{\alpha}^*$, one may wonder how close the Jacobi-Gauss-Lobatto nodes for $\hat{\alpha}^*(N)$ compare to the Lebesgue constant for the optimal nodes, and also how they compare to the (much more easily computed) extended Chebyshev set. We pictorially compare some Lebesgue constants in Figure 4.5. The Jacobi-Gauss-Lobatto nodes have a slightly smaller value for the Lebesgue

constant than the Extended Chebyshev nodes. Thus, it appears that our conjecture (4.6) for the analytic form for α^* is good enough to give results better than the Extended Chebyshev set. Of course, the price we must pay is that the Jacobi-Gauss-Lobatto nodal set $\left\{r_n^{(\alpha,\alpha);\mathrm{GL}}\right\}$ requires $O\left(N^2\right)$ operations to compute as the eigenvalues of a symmetric tridiagonal matrix [35], whereas the Extended Chebyshev nodal set has an analytic expression (4.4) and is computable in a simple $O(N)$ operation.

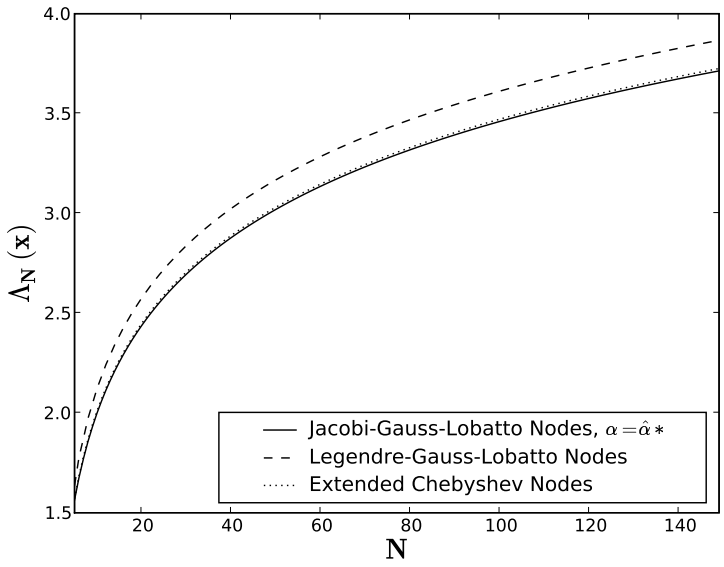

Figure 4.5: Plot of the Lebesgue constants for the Legendre-Gauss-Lobatto, Extended Chebyshev, and quasi-optimized (i.e. $\alpha = \hat{\alpha}^*$) Jacobi-Gauss-Lobatto nodal sets.

To give an idea of the slight improvement in optimization of the Lebesgue constant our algorithm gives, we present in Table 4.1 the Lebesgue constants for 1.) the equidistant nodes, 2.) the Legendre-Gauss-Lobatto nodes, 3.) the Extended Chebyshev nodes, 4.) the Jacobi-Gauss-Lobatto nodes for $\hat{\alpha}^*$, 5.) the Jacobi-Gauss-Lobatto nodes for the numerically determined α^*, and 6.) the optimum nodal set. The table confirms that $\hat{\alpha}^*$ is a relatively accurate representation for the true minimizing value α^*. Indeed, it is very expensive to compute α^* numerically so using the conjecture

in (4.5) provides an extremely accurate solution using an inconsequential amount of resources.

Of course, computing the Extended Chebyshev set is still much cheaper computationally as one only needs to implement (4.4) rather than the classic $O(N^2)$-complexity algorithm for polynomial Gaussian quadrature. However, using modern computers to determine a Gaussian quadrature for $N \lesssim 1000$ is doable in at worst a couple of seconds. Thus, the conjectured value for $\hat{\alpha}^*(N)$ given in equation (4.5) provides an extremely good estimate for the optimal nodal choice.

N	EQ	LGL	EC	JGL – $\hat{\alpha}^*$	JGL – α^*	Opt
5	2.207824e+00	1.635882	1.570167	1.563689	1.559490	0.559490
10	1.784861e+01	2.120964	2.008327	1.997567	1.997267	1.991685
15	2.832112e+02	2.386187	2.265518	2.254701	2.254542	2.247596
20	5.889585e+03	2.575773	2.448199	2.437628	2.437456	2.429842
25	1.378520e+05	2.719979	2.589974	2.579342	2.579177	2.572002
30	3.447739e+06	2.838589	2.705850	2.695218	2.695049	2.687580
35	9.001189e+07	2.937756	2.803843	2.793118	2.792952	2.785829
40	2.421997e+09	3.024122	2.888741	2.877965	2.877797	2.870755
45	6.664660e+10	3.099711	2.963635	2.952769	2.952604	2.945618
50	1.865926e+12	3.167630	3.030636	3.019709	3.019545	3.012275
60	1.520352e+15	3.284675	3.146589	3.135523	3.135362	3.128219
70	1.286310e+18	3.383511	3.244637	3.233445	3.233288	3.226401
80	1.118136e+21	3.469049	3.329575	3.318272	3.318118	3.311261
90	9.919497e+23	3.544445	3.404501	3.393098	3.392947	3.386236
500	3.497634e+146	4.638877	4.495628	4.482892	4.482812	4.478201

Table 4.1: Lebesgue constants for various nodal sets. EQ=equidistant; LGL=Legendre-Gauss-Lobatto; EC=Extended Chebyshev; JGL–$\hat{\alpha}^*$=Jacobi-Gauss-Lobatto with $\alpha = \hat{\alpha}^*(N)$; JGL–$\alpha^*$=Jacobi-Gauss-Lobatto with $\alpha = \alpha^*(N)$ (numerically calculated); Opt=optimal nodal set, calculated using a version of the linear programming algorithm given in [3].

4.3.2 Gauss interior nodes

In this subsection we present a brief reshash of the results of the previous section applied to purely interior nodal sets: those *without* nodes at $r = \pm 1$. For polynomial quadrature rules, these correspond to the classical Gauss quadrature rules, which can again be computed by the Golub-Welsch algorithm [35] as eignevalues of a symmetric tridiagonal matrix. Again we suppose that there exists some optimal value of α, called α^*, which minimizes the Lebesgue constant over Jacobi-Gauss quadrature rules:

$$\alpha^*(N) := \underset{\alpha \in (-1,\infty)}{\operatorname{argmin}} \Lambda_N \left(r_n^{(\alpha,\alpha)} \right).$$

One should suspect similar results and phenomena that were present in the Gauss-Lobatto case, and indeed we do see similar results. Because of this, we only present the Gauss quadrature versions of selected figures from the previous section. Figure 4.6 shows the meshplot of the Lebesgue constant against α and N. We see that the optimal value α^* now happens around $\alpha = -0.5$, which corresponds to the Gaussian quadrature rule for the Chebyshev polynomials.

Again, we can show cross-section plots similar to those in Figures 4.2 and 4.3, but the results for the interior node case mimic the boundary node case so well that we omit these. We can again brute-force the numerical determination of α^* and attempt to fit the data values to an analytic form. Having done this, we propose that the value of the Jacobi parameter α that minimizes the Lebesgue constant $\Lambda_N^{(\alpha,\mathrm{GQ})}$ has an asymptotic form given by

$$\alpha^*(N) = c_1 + \frac{c_2}{\log\log\left(N + \frac{\pi}{2}\right)}, \qquad N \geq 5 \tag{4.7}$$

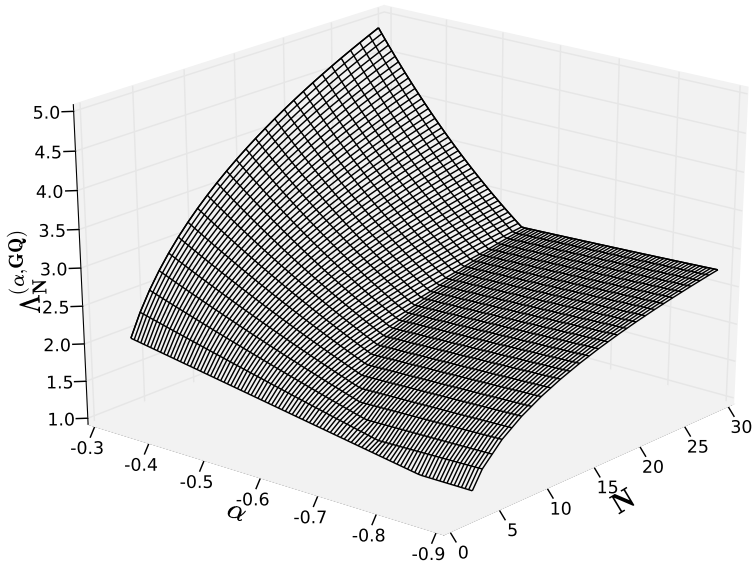

Figure 4.6: Mesh plot of the Lebesgue constant $\Lambda_N^{(\alpha);\mathrm{GQ}}$ vs α and N.

where the constants are given by

$$c_1 = -0.44221386$$

$$c_2 = -0.14768772$$

Again we can offer no insight into our choice of fitting function. Conjectures (4.6) and (4.7) are merely rules-of-thumb; they provide a excellent approximation to $\alpha^*(N)$ and do furnish nodal sets with Lebesgue constants smaller than any canonical nodal set found we have found in the literature.

We show Lebesgue constants for 1.) Legendre-Gauss, 2.) Chebyshev-Gauss, 3.) Jacobi-Gauss-$\hat{\alpha}^*$, and 4.) Jacobi-Gauss-α^* nodal sets in Table 4.2. Observations similar to those from the previous section on Jacobi-Gauss-Lobatto interpolation can be made.

N	LGQ	CQ	JGQ – $\hat{\alpha}^*$	JGQ – α^*	Opt
5	3.322200e+00	1.988854	1.602223	1.577828	1.559490
10	5.192700e+00	2.428828	2.031200	2.012739	1.991685
15	6.649519e+00	2.686711	2.281165	2.266091	2.247596
20	7.885033e+00	2.869767	2.458836	2.446979	2.429842
25	8.977085e+00	3.011781	2.596948	2.587271	2.572002
30	9.966399e+00	3.127822	2.710028	2.702202	2.687580
35	1.087745e+01	3.225937	2.805819	2.799364	2.785829
40	1.172630e+01	3.310928	2.888942	2.883660	2.870755
45	1.252419e+01	3.385894	2.962378	2.958007	2.945618
50	1.327931e+01	3.452952	3.028166	3.024581	3.012275
60	1.468474e+01	3.568986	3.142230	3.139817	3.128219
70	1.597805e+01	3.667081	3.238897	3.237306	3.226401
80	1.718241e+01	3.752044	3.322805	3.321795	3.311261
90	1.831399e+01	3.826974	3.396951	3.396346	3.386236
500	4.516141e+01	4.909286	4.485702	4.483677	4.478201

Table 4.2: Lebesgue constants for various interior nodal sets. LGQ=Legendre-Gauss quadrature nodes; CQ=Chebyshev-Gauss quadrature nodes; JGQ–$\hat{\alpha}^*$=Jacobi-Gauss with $\alpha = \hat{\alpha}^*(N)$; JGQ–$\alpha^*$=Jacobi-Gauss with $\alpha = \alpha^*(N)$ (numerically calculated); Opt=optimal nodal set, values copied from Table 4.1.

4.4 Two-Dimensional Extensions

We learned in section 4.3 that we can obtain near-optimal nodal interpolation sets by using Jacobi-Gauss-Lobatto nodal sets with particular values of the Jacobi class parameter α. Having computed these optimal values of α that minimize boundary nodal sets, one might wonder whether we can use this knowledge to synthesize nodal sets in two dimensions that minimize the Lebesgue constant on triangular simplices. To facilitate this, we adopt Warburton's explicit construction procedure for the nodes [66].

The basic idea behind Warburton's 'warp & blend' algorithm is as follows. In one dimension, we observe that the Legendre-Gauss-Lobatto nodes form a reasonable nodal set for interpolation (cf. Table 4.1). If we so desire, we may define a 'warp'

function for N nodes as the equispaced interpolant of the difference between the Legendre nodal set and the equispaced nodal set:

$$w(r) := \sum_{n=1}^{N} \left(r_n^{(0);\mathrm{GL}} - r_n^{\mathrm{eq}} \right) l_{n,N}^{\mathrm{eq}}(x), \tag{4.8}$$

where $l_{n,N}^{\mathrm{eq}}(x)$ is the $(N-1)^{\mathrm{st}}$ degree Lagrange interpolant for the equispaced nodal set. With this definition, we have $r_n^{\mathrm{eq}} + w\left(r_n^{\mathrm{eq}}\right) \equiv r_n^{\mathrm{eq}} + r_n^{(0,0);\mathrm{GL}} - r_n^{\mathrm{eq}} = r_n^{(0,0);\mathrm{GL}}$, i.e., the warp function $w(r)$ is a continuous interpolant of the discrete differential between the equidistant nodal set and the Gauss-Lobatto nodal set. We can use this for generating interpolating nodal sets in more than one dimension. The effect of this one-dimensional warping is shown in Figure 4.7.

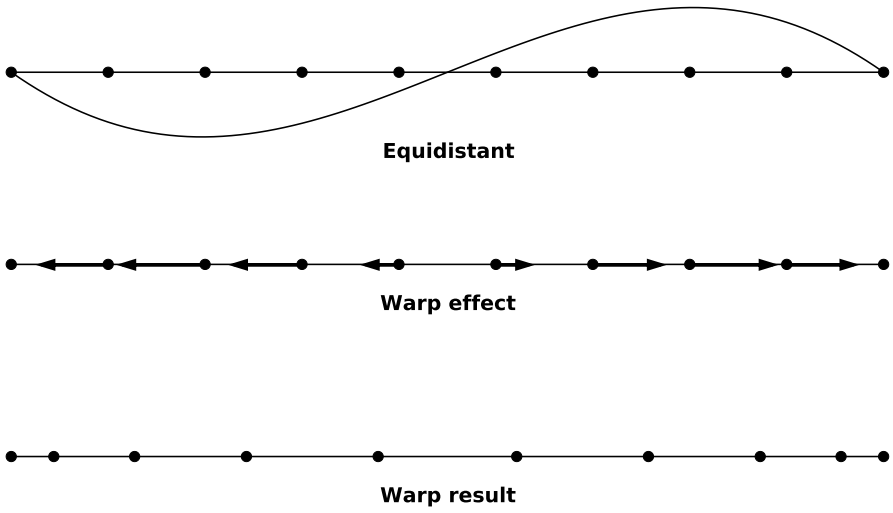

Figure 4.7: Effect of one-dimensional warping as given by equation (4.8). The resulting nodal set is the α-optimized Jacobi-Gauss-Lobatto nodal set for $N = 10$ ($\alpha = 0.43889$).

To produce a symmetric nodal set on simplices in two dimensions the algorithm is to proceed as follows:

1. Generate the equidistant nodal set on a simplex, featuring an N-point equidistant stencil on each edge.

2. For each edge, apply a warping that acts parallel to that edge.

   (a) Apply the one-dimensional N-point warping function (4.8) to each node based on the barycentric location of the node parallel to the edge.

   (b) Multiplicatively 'blend' the warping based on the barycentric distance perpendicular to the edge. This ensures that the nodes stay in the interior of the triangle. (See [66])

The blending step consists of determining an optimization parameter γ that pushes the nodes towards the vertices of the simplex. γ must be determined numerically, and was introduced as an *ad hoc* strategy to combat the observed tendency of the nodal sets to not cluster at the vertices, which is a qualitative phenomenon at odds with other existing quasi-optimal multidimensional nodal sets. (Unfortunately, in [66] this optimization parameter is called α, which we have already employed to mean the Jacobi class parameter.) The entire procedure is graphically illustrated in Figure 4.8. The parameter γ for $N = 3, \ldots, 15$ is tabulated in Table E.2 in Appendix E.

Warburton's method uses a one-dimensional transformation (from equidistant to Legendre-Gauss nodes) as the building block for a two-dimensional extension. Given that we have characterized near-optimal one-dimensional nodal sets, we can try to extend Warburton's method using this. The idea is simple: instead of using the warp function from equidistant nodes to Legendre-Gauss-Lobatto nodes, we will instead replace that function by the associated warp function for the α^* Jacobi Gauss-Lobatto set. The procedure is graphically illustrated in Figure 4.9.

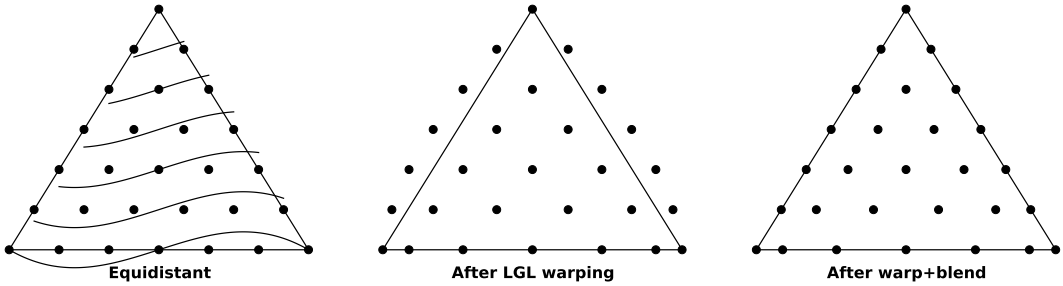

Figure 4.8: Legendre-based warp and blend illustration on the bottom edge for 6^{th} order polynomial interpolation. The value for the blending parameter γ is the optimal value cited in [66]. Warping and blending must be done on the equidistant set for the other two edges as well to recover a symmetric nodal set.

This Jacobi-based warping based on the numerically-optimal one-dimensional Jacobi Gauss-Lobatto nodal sets produces the best (smallest) symmetric two-dimensional Lebesgue constants on simplices (as of the date of this writing, and to the author's knowledge) for $N \leq 10$. For $N > 10$, the method produces Lebesgue suboptimal nodes compared to implicit constructions, but has the best-known Lebesgue constants for explicit constructions. We summarize these results in Table 4.3, for which we term the nodes determined from our Jacobi-Gauss-Lobatto scheme the 'pyramid' nodes.

The optimal values of the Lebesgue constant that we have derived are smaller than Warburton's nodes only by a small amount. Because there is only a modest improvement, one must balance the simplicity of each method with the payoffs. In conception, the pyramid construction scheme based on the one-dimensional optimal Jacobi Gauss-Lobatto set is no more complicated than the Legendre Gauss-Lobatto scheme; in theoretical computational effort it is more expensive because one needs to compute a series of Gauss-Lobatto nodal sets; in practical computational effort, there is no noticable slowdown because determination of these one dimensional nodal sets is cheap if $N < 100$.

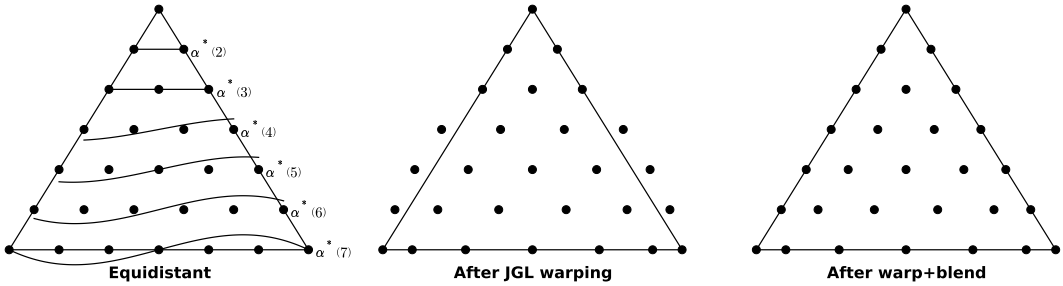

Figure 4.9: Jacobi-based warp and blend illustration on the bottom edge for 6^{th} order polynomial interpolation. The value for the blending parameter γ is numerically determined using the same procedure as in [66]. Warping and blending must be done on the equidistant set for the other two edges as well to recover a symmetric nodal set.

4.5 Summary

We have used the Jacobi-Gauss type quadrature nodal sets to seek quasi-optimal interpolation nodes on finite intervals in one dimension and simplices in two dimensions. All of our results are based exclusively on numerical evidence. We have shown empirically that the one-dimensional parameter α has a value for each N that minimizes the Lebesgue constants $\Lambda_N\left(r_n^{(\alpha,\alpha);\text{GL}}\right)$ and $\Lambda_N\left(r_n^{(\alpha,\alpha)}\right)$. Based on this observation we have used a minimization search to determine the value of this parameter α for each N. These values are tabulated in Appendix E. The Jacobi-Gauss type nodal sets are explicitly constructable nodal sets that produce the smallest Lebesgue constant in one dimension on finite intervals, save the genuinely optimal set.

We extend our one-dimensional characterization to two dimensions on a triangular simplex by employing Warburton's warp and blend algorithm to explicitly generate nodal sets; for $N < 10$ these sets have the smallest Lebesgue Constant we have found in the literature; for $N \geq 10$ the only nodal sets to best them are based on

N	Equidistant	Chen,[24]	Heinrichs,[41]	Warburton,[66]	Pyramid
3	2.27	**2.11**	—	**2.11**	2.13
4	3.47	2.69	—	2.66	**2.61**
5	5.45	3.30	—	3.12	**3.10**
6	8.75	3.79	3.87	3.70	**3.64**
7	14.34	4.39	—	4.27	**4.23**
8	24.01	5.09	—	4.96	**4.89**
9	40.92	5.92	5.89	5.74	**5.67**
10	70.89	7.09	—	6.67	**6.56**
11	124.53	**7.27**	—	7.90	7.79
12	221.41	8.67	**7.59**	9.36	9.23
13	397.70	**9.29**	—	11.47	11.29
14	720.69	**8.99**	—	13.97	13.76
15	1315.89	10.31	**9.25**	17.65	17.29

Table 4.3: Tabulated Lebesgue constants for various two-dimensional symmetric interpolation sets on simplices. Bold indicates the lowest Lebesgue constant. 'Pyramid' indicates the nodes produced from the algorithm introduced in this chapter. The Chen (computed in [56]) and Heinrichs nodal sets are implicit constructions based on nontrivial optimizations.

computationally intensive search and minimization algorithms.

We can envision this an extension of this method to three dimensions on tetrahedral simplices. In addition, an investigation must be performed in determining the conditioning of an interpolating Vandermonde matrix for a polynomial basis in two and three dimensions to give a more complete picture of the advantage of using these nodal sets over other choices.

Chapter Five

Generalized Fourier Series

5.1 Introduction

We now address topics that are directly related to the task of deriving the generalized Wiener rational function basis over infinite intervals. A significant intermediate task to be accomplished is the generalization of the Fourier series on a finite interval; this is the contents of this chapter.

We will break down the canonical Fourier basis using the Jacobi polynomials in order to understand the individual parts of trigonometic polynomials. Having done so we will be in a position to use our knowledge of Jacobi polynomials to create more general trigonometric basis sets.

The strong relation to orthogonal polynomials will allow us to derive many useful and interesting properties of the generalized Fourier series; in addition, we will learn how to employ the FFT, just as it can be employed for the canonical Fourier system.

In Section 5.2 we review some salient properties of Jacobi polynomials and use them to derive the generalized Fourier basis. Section 5.3 explores the orthogonal polynomial decomposition in more detail by presenting an alternative to the periodic Fourier series. Section 5.4 is an application of the Jacobi connection problem, Section 3.2, to the Fourier connection problem. Finally, Section 5.5 furnishes some recurrence relations and properties of derivatives. All of these properties will have significant contributions to the derivation of the generalization of Wiener's rational functions.

5.2 Generalizing the Fourier basis

All of our notation from Section 2.1 regarding functions over the variables r, θ, z, and x will now come directly to the fore. In addition, we will now make use of some of the properties of Jacobi polynomials, introduced in Section 2.2 and detailed at length in Section A.1.

Finally, we excerpt two classical notational conventions from Appendix A.1 that we will use briefly. The Jacobi polynomials that result from the cases $\alpha = \beta = -\frac{1}{2}$ and $\alpha = \beta = +\frac{1}{2}$ are the Chebyshev polynomials of the first and second kinds, respectively. Recalling the relation $r = \cos\theta$, these polynomials are typically denoted $T_n(r)$ and $U_n(r)$ and have a very special and concise representation as trigonometric polynomials:

$$\sqrt{\tfrac{\pi}{2}}\tilde{P}_n^{(-1/2,-1/2)}(r) \;=\; T_n(r) \;=\; \cos(n\theta) \;=\; \cos[n\arccos(r)]$$

$$\sqrt{\tfrac{\pi}{2}}\tilde{P}_n^{(1/2,1/2)}(r) \;=\; U_n(r) \;=\; \tfrac{\sin[(n+1)\theta]}{\sin\theta} \;=\; \tfrac{\sin[(n+1)\arccos(r)]}{\sin[\arccos(r)]}.$$

We will now generalize the canonical Fourier basis given by

$$\Psi_k(\theta) = e^{ik\theta}.$$

Our methodology is based upon the following dissection of the Fourier basis for

$k \neq 0$:

$$\begin{aligned} e^{ik\theta} &= \cos(k\theta) + i\sin(k\theta) \\ &= \cos(|k|\theta) + i\,\mathrm{sgn}(k)\sin(|k|\theta) \\ &= T_{|k|}(\cos\theta) + i\,\mathrm{sgn}(k)\sin(\theta)U_{|k|-1}(\cos\theta) \\ &= \sqrt{\tfrac{\pi}{2}}\left[\tilde{P}_{|k|}^{(-1/2,-1/2)}(\cos\theta) + i\,\mathrm{sgn}(k)\sin(\theta)\tilde{P}_{|k|-1}^{(1/2,1/2)}(\cos\theta)\right]. \end{aligned}$$

We have broken down the Fourier basis into two components: the first component is even with respect to θ as it is simply a polynomial in $\cos\theta$. The second term is odd in θ because it is a polynomial in $\cos\theta$ (an even function) multiplied by the odd function $\sin\theta$. This breakdown suggests that we can construct more general kinds of Fourier-type functions by merely augmenting the type of polynomials that we use.

However, we cannot switch around polynomials with impunity; we still want to retain orthogonality (at least with respect to some weight function). Due to the even-odd decomposition just performed, it is clear that the even and odd functions will be orthogonal with respect to each other. However, we must retain orthogonality of the even functions and orthogonality of the odd functions under the same inner product. To do this we use Lemma 2.1.

For $\alpha, \beta > -1$, we have the polynomials $\tilde{P}_n^{(\alpha,\beta)}$ that are orthogonal in $L^2\left([-1,1], \mathbb{R}; w_r^{(\alpha,\beta)}\right)$. By setting $a = b = \frac{1}{2}$ in Lemma 2.1, we also have that the Jacobi functions $\tilde{P}_n^{(\alpha+1,\beta+1,1/2,1/2)} = (1-r^2)^{1/2}\tilde{P}_n^{(\alpha+1,\beta+1)}$ are also orthogonal under the same weight. If we set $\alpha = \beta = -\frac{1}{2}$, add these two functions together with the appropriate scaling factors, then we exactly recover the Fourier basis by reversing the dissection steps

above. Of course, we are free to choose any values of (α, β) that we desire in order to derive generalized trigonometric Fourier functions. In fact, this technique has already been used by Szegö [60] to determine orthogonal polynomials on the unit circle. Because the statement in [60] is merely a passing comment and is a rather different result from what we desire, we give the following theorem:

Theorem 5.1 (cf. Szegö, [60]) *For any $\gamma, \delta > -\frac{1}{2}$, the functions*

$$\Psi_k^{(\gamma,\delta)}(\theta) = \begin{cases} \frac{1}{\sqrt{2}} \tilde{P}_0^{(\delta-1/2,\gamma-1/2)}(\cos\theta), & k = 0 \\ \\ \frac{1}{2}\left[\tilde{P}_{|k|}^{(\delta-1/2,\gamma-1/2)}(\cos\theta) + i\,\mathrm{sgn}(k)\sin(\theta)\tilde{P}_{|k|-1}^{(\delta+1/2,\gamma+1/2)}(\cos\theta)\right], & k \neq 0 \end{cases}$$

are complete and orthonormal in $L^2\left([-\pi,\pi], \mathbb{C}; w_\theta^{(\gamma,\delta)}\right)$.

Proof For orthonormality, it suffices to show

1. $\left\langle \tilde{P}_{|k|}^{(\delta-1/2,\gamma-1/2)}(\cos\theta), \tilde{P}_{|l|}^{(\delta-1/2,\gamma-1/2)}(\cos\theta) \right\rangle_{w_\theta^{(\gamma,0)}} = 2\delta_{|k|,|l|}$
2. $\left\langle \sin\theta \tilde{P}_{|k|-1}^{(\delta+1/2,\gamma+1/2)}(\cos\theta), \sin\theta \tilde{P}_{|l|-1}^{(\delta+1/2,\gamma+1/2)}(\cos\theta) \right\rangle_{w_\theta^{(\gamma,0)}} = 2\delta_{|k|,|l|}$, for $k, l \neq 0$.
3. $\left\langle \tilde{P}_k^{(\delta-1/2,\gamma-1/2)}(\cos\theta), \sin\theta \tilde{P}_{|l|-1}^{(\delta+1/2,\gamma+1/2)}(\cos\theta) \right\rangle_{w_\theta^{(\gamma,0)}} = 0$, for $l \neq 0$.

The first property is a direct result of orthonormality of the normalized Jacobi polynomials $\tilde{P}$ and the observation that on $[0, \pi]$,

$$\langle f(\cos\theta), g(\cos\theta)\rangle_{w_\theta^{(\gamma,\delta)}} = \langle f(r), g(r)\rangle_{w_r^{(\delta-1/2,\gamma-1/2)}}.$$

The second property is a result of the same observations as the first property along with the result of Lemma 2.1. The third property results from the fact that an odd function integrated over a symmetric interval is 0.

Orthonormality then follows from an explicit calculation of $\left\langle \Psi_k^{(\gamma,\delta)}, \Psi_l^{(\gamma,\delta)} \right\rangle_{w_\theta^{(\gamma,\delta)}}$ using the above three properties.

For completeness, we note that any function $f \in L^2$ can be decomposed into an even f_e and an odd f_o part. That f_e is representable is clear from the fact that $\tilde{P}_n^{(\delta-1/2,\gamma-1/2)}(\cos\theta)$ is complete over $\theta \in [0,\pi]$ (i.e. Jacobi polynomial completeness), which by symmetry means completeness over all L^2-even functions f_e. Similary, the collection of functions $\sin\theta \tilde{P}_n^{(\delta-1/2,\gamma-1/2)}$ is complete over all L^2-odd functions f_o by the completeness result of Lemma 2.1. Linearity and orthogonality of the even and odd parts then gives the result. □

In the definition of the functions $\Psi_k^{(\delta,\gamma)}$ it is desirable to use the L^2-normalized versions of the Jacobi polynomials $\tilde{P}$, rather than the monic polynomials P. If the monic polynomials instead are used, then the norm of the Szegö-Fourier functions $\Psi_k^{(\gamma)}$ depends on the rather unpleasant-looking sum $h_{|k|}^{(\delta-1/2,\gamma-1/2)} + h_{|k|-1}^{(\delta+1/2,\gamma+1/2)}$. To illustrate how the generalized Fourier functions $\Psi_k^{(\gamma,\delta)}$ behave, we plot $|\Psi_k^{(\gamma,0)}|$ for $\gamma = 0, 1, 2,$ and 3 in Figure 5.1.

We can also distribute the weight function onto the basis functions, which gives us orthogonality in the unweighted L^2-norm:

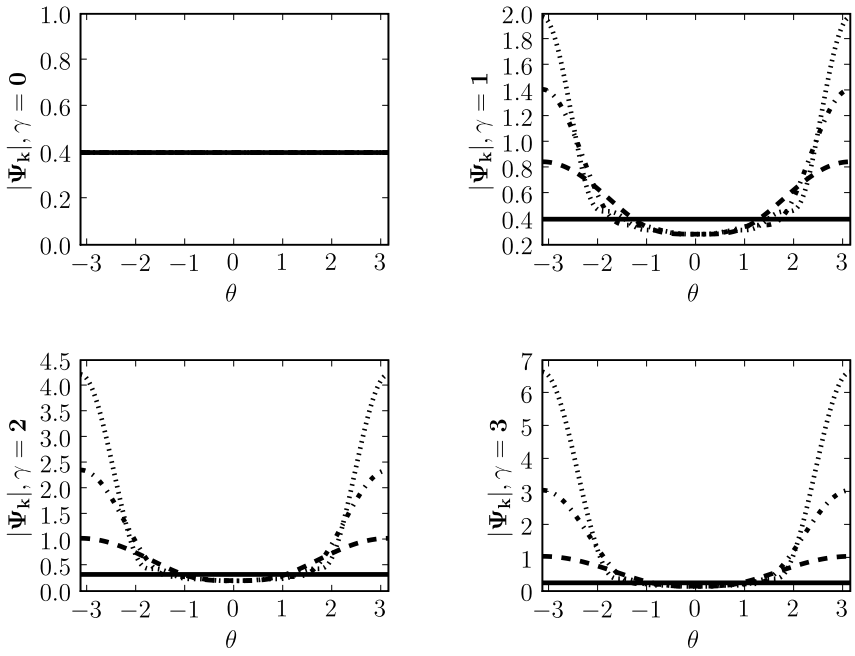

Figure 5.1: Plots of the magnitudes $\left|\Psi_k^{(\gamma,0)}\right|$ for $\gamma = 0, 1, 2, 3$. In each plot, we plot magnitudes for $k = 0$ (solid $-$), $k = 1$ (dashed - -), $k = 2$ (dot-dash $-\cdot$) and $k = 3$ (dotted, $\cdot\cdot$).

Corollary 5.1 *For any $\gamma > -\frac{1}{2}$, the functions*

$$\psi_k^{(\gamma)}(\theta) = \begin{cases} \frac{\sqrt[*]{w_\theta^{(\gamma,0)}}}{\sqrt{2}} \tilde{P}_0^{(-1/2,\gamma-1/2)}(\cos\theta), & k = 0 \\ \\ \frac{\sqrt[*]{w_\theta^{(\gamma,0)}}}{2} \left[\tilde{P}_{|k|}^{(-1/2,\gamma-1/2)}(\cos\theta) + i\,\mathrm{sgn}(k)\sin(\theta)\tilde{P}_{|k|-1}^{(1/2,\gamma+1/2)}(\cos\theta)\right], & k \neq 0 \end{cases}$$

are complete and orthonormal in $L^2([-\pi,\pi],\mathbb{C})$.

Due to the properties of $\sqrt[*]{w_\theta^{(\gamma,0)}}$ given in (2.1), the functions $\psi_k^{(\gamma)}(\theta)$ decay like $\left(\cos\frac{\theta}{2}\right)^\gamma$ at $\theta = \pm\pi$. This is exmplified in Figure 5.2 where we plot the real and imaginary parts of the functions for $\gamma = 2$. The even/odd behavior in θ for real/imaginary components depicted in the figure depends on the even/odd parity of γ. (There is no such characterization possible when $\gamma \notin \mathbb{N}_0$.) Clearly for $\gamma = 0$ we have

$\Psi_k^{(0)} = \psi_k^{(0)} = \frac{1}{\sqrt{2\pi}} e^{ik\theta}$, the canonical Fourier basis.

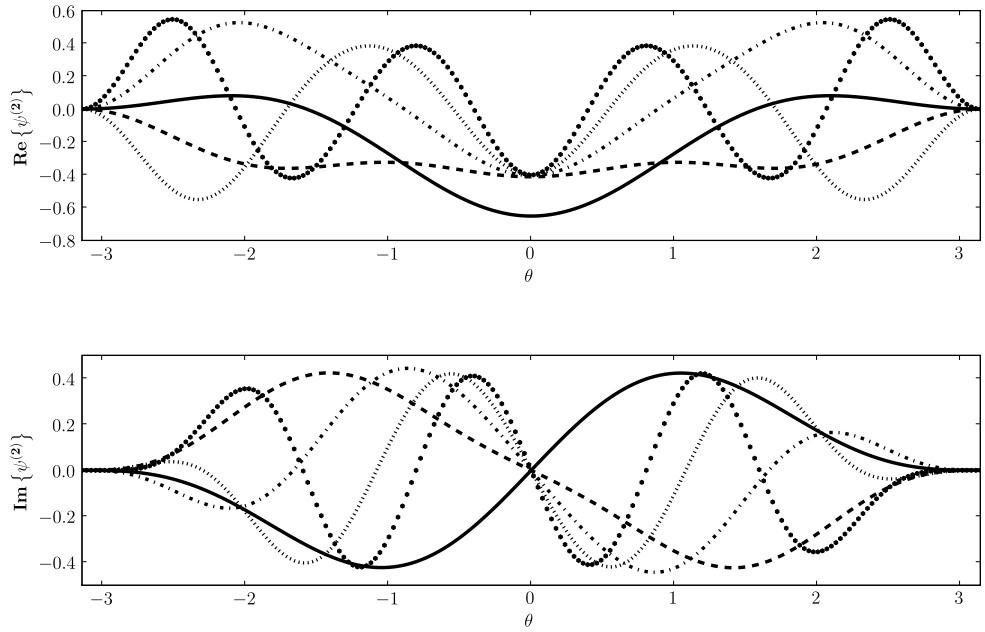

Figure 5.2: Plots of the weighted Szegö-Fourier functions $\psi_k^{(2)}(\theta)$ for $k = 0, 1, 2, 3$,and 4 (respectively: solid, dashed, dot-dash, dotted, circle marks). Real part (top) and imaginary part (bottom).

Remark 2 Szegö [60] essentially gives the result of Theorem 5.1. It is even possible to generalize Szegö's result: he derived polynomials on the unit disk orthogonal with respect to $w_\theta^{(\gamma,\delta)}$. By using Lemma 2.1 with a, b different from $\frac{1}{2}$, we can in fact derive non-polynomial basis sets that are orthogonal under a great variety of weights. These functions naturally may not be periodic on $\theta \in [-\pi, \pi]$ if the weight function is not periodic.

5.3 An Alternative to Periodicity

As an aside, let us use the even/odd decomposition detailed in the previous section to create other function sets that are not genuinely periodic Fourier functions. Using

the logic employed to construct the $\Psi_k^{(\gamma,\delta)}$, we propose the following basis set:

$$\Upsilon_k^{(\gamma,\delta)}(\theta) = \begin{cases} \frac{1}{\sqrt{2}}\left[\cos\left(\frac{\theta}{2}\right)\tilde{P}_{|k|}^{(\delta-1/2,\gamma+1/2)}(r) + i\,\mathrm{sgn}(k)\sin\left(\frac{\theta}{2}\right)\tilde{P}_{|k|-1}^{(\delta+1/2,\gamma-1/2)}(r)\right], & |k|>0 \\ \cos\left(\frac{\theta}{2}\right)\tilde{P}_0^{(\delta-1/2,\gamma+1/2)}, & k=0 \end{cases}$$

These (nonperiodic!) functions are orthogonal under the weight $w_\theta^{(\gamma,\delta)}$ in L^2 over $\theta\in[-\pi,\pi]$ for all $\gamma,\delta>-\frac{1}{2}$. When $\gamma,\delta=0$ then $\Upsilon_k^{(0,0)}$ is a weighted combination of the Chebyshev polynomials of the third and fourth kinds. Note, however, that these functions are 'antiperiodic' – that is, the even derivatives of the even part vanishes at $\theta=\pm\pi$ and the odd derivatives of the odd part vanishes at $\theta=\pm\pi$. Contrast this with analyic periodic functions (e.g. $\Psi_k^{(\gamma,\delta)}$) where the odd derivatives of the even parts vanish and the even derivatives of the odd part vanishes at $\theta=\pm\pi$. If one attempts to expand an analytic function that is not genuinely 'antiperiodic' with $\Upsilon_k^{(\gamma,\delta)}$, then spectral convergence will not be obtained. This is exactly the same problem as attempting to expand a nonperiodic function using a Fourier series.

That these functions are not periodic can be made clear by utilizing the representations [34]

$$\tilde{P}_n^{(-1/2,1/2)}(\cos\theta) = \frac{\cos\left(n+\frac{1}{2}\right)\theta}{\sqrt{\pi}\cos\frac{\theta}{2}},$$

$$\tilde{P}_n^{(1/2,-1/2)}(\cos\theta) = \frac{\sin\left(n+\frac{1}{2}\right)\theta}{\sqrt{\pi}\sin\frac{\theta}{2}}.$$

Combining these with the definition of $\Upsilon_k^{(0,0)}$ yields

$$\Upsilon_k^{(0,0)} = \begin{cases} \frac{1}{\sqrt{2\pi}}\left[\cos\left(|k|+\frac{1}{2}\right)\theta + i\,\mathrm{sgn}(k)\sin\left(|k|-\frac{1}{2}\right)\theta\right], & k\neq 0, \\ \frac{\cos\left(\frac{\theta}{2}\right)}{\sqrt{\pi}}, & k=0. \end{cases}$$

Note that if we had chosen a different linear combination of the polynomials then $\Upsilon_k^{(0,0)}$ would have been proportional to $\exp\left[i\left(k+\frac{1}{2}\right)\theta\right]$, which are in fact nonperiodic functions. For $\gamma, \delta \neq 0$ these functions are variants of the above nonperiodic Fourier series.

The basis $\Upsilon_k^{(\gamma,\delta)}$ may seem to be mainly of academic interest, but it does have potential for application. Consider the function

$$f(\theta) = \arctan\left(\frac{(\theta-\pi)^7}{500}\right) + \arctan\left(\frac{(\theta+\pi)^7}{500}\right), \tag{5.1}$$

plotted in the figure below. This function is constructed so that the even part of its first few derivatives are zero at the points $\theta = \pm\pi$, thus guaranteeing accuracy of a spectral expansion in the $\Upsilon_k^{(\gamma,\delta)}$. We consider three projective spectral expansions of this function: one in the $\Upsilon^{(0,0)}$, the second in the Fourier basis $\tilde{\Psi}^{(0,0)}$, and finally in Chebyshev polynomials $\tilde{P}^{(-1/2,-1/2)}$.

$$\begin{aligned} f_\Xi(\theta) &= \textstyle\sum_{k=-N}^{N} \hat{f}_{\Xi,k} \Upsilon_k^{(0,0)}(\theta) \\ f_F(\theta) &= \textstyle\sum_{k=-N}^{N} \hat{f}_{F,k} \tilde{\Psi}_k^{(0,0)}(\theta) \\ f_P(\theta) &= \textstyle\sum_{n=0}^{2N} \hat{f}_{P,n} \tilde{P}_n^{(-1/2,-1/2)}\left(\frac{\theta}{\pi}\right). \end{aligned}$$

In Figure 5.4 we show the results of this expansion. The Fourier basis does the worst, not surprisingly: the function $f(\theta)$ is not even continuously periodic, so we expect Gibbs' oscillations near the endpoints. The Chebyshev expansion f_P does a good job since it does not require any unusual regularity at endpoints. The expansion f_Ξ *does* require additional regularity at the endpoints, but this function $f(\theta)$ satisfies

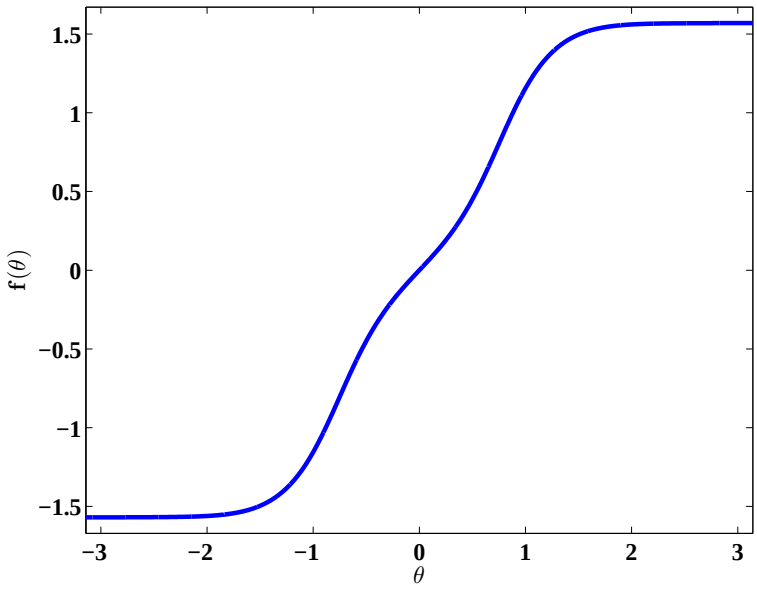

Figure 5.3: Plot of the function $f(\theta)$ defined in equation (5.1).

this regularity requirement to a moderate degree. We see in the figure below that the expansion f_{Ξ} produces a superior approximation for this particular choice of f.

We are not necessarily advocating use of the functions $\Upsilon_k^{(\gamma,\delta)}$ for practical purposes; however, we have shown that judicious (and unorthodox) linear combinations of Jacobi polynomials can yield basis sets with interesting properties. However, we shall concentrate the rest of our discussion on the particular combination $\Psi_k^{(\gamma,\delta)}$, defined by Theorem 5.1.

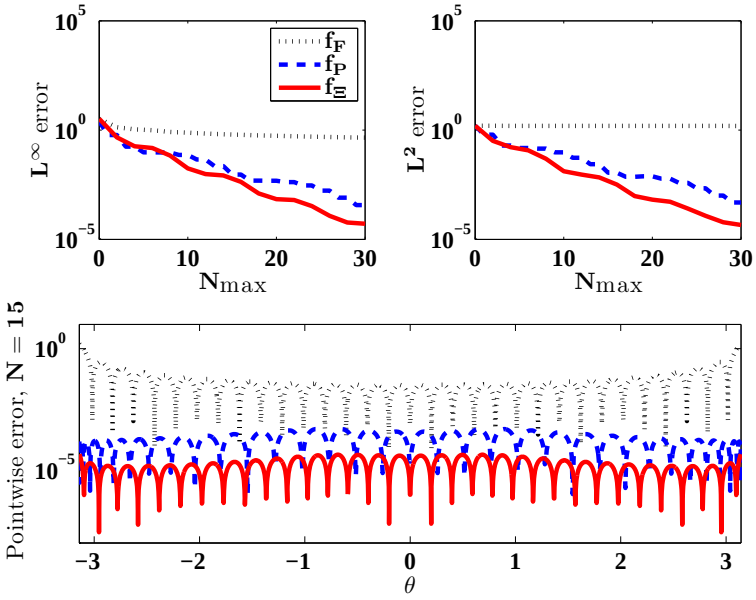

Figure 5.4: Plot of the L^∞ (upper left), L^2 (upper right), and pointwise errors for the three spectral expansions considered. In the top two sub-figures, the x-axis corresponds to the total number of basis functions used in the reconstruction.

5.4 The Connection Problem

We have already seen the connection problem in the context of Jacobi polynomials (see section 3.3). In that section we were concerned with solving the problem

$$\sum_{n=0}^{\infty} \hat{f}_n^{(\alpha,\beta)} \tilde{P}_n^{(\alpha,\beta)} = \sum_{n=0}^{\infty} \hat{f}_n^{(\varepsilon,\zeta)} \tilde{P}_n^{(\varepsilon,\zeta)},$$

where we know one set of modal coefficients and are trying to determine the other set. We consider now the same problem in a different basis set; that is, we wish to solve the problem

$$\sum_{k\in\mathbb{Z}} \hat{f}_k^{(\gamma,\delta)} \Psi_k^{(\gamma,\delta)} = \sum_{k\in\mathbb{Z}} \hat{f}_k^{(\eta,\iota)} \Psi_k^{(\eta,\iota)}, \tag{5.2}$$

where we assume that we know $\hat{f}_k^{(\gamma,\delta)}$ and we want to know $\hat{f}_k^{(\eta,\iota)}$. Note that it is necessary for us to assume that both $f \in L^2_{w_\theta^{(\gamma,\delta)}}$ and $f \in L^2_{w_\theta^{(\eta,\iota)}}$ for the modal

coefficients to be well-defined.

With truncated expansions, performing this operation quickly will allow us to efficiently use the FFT for nodal/modal transformations. Additionally, this change of basis will allow us to perform change-of-decay transformations for the generalized Wiener basis introduced in Chapter 6. The solution to this problem is again determined by the connection coefficients defined as

$$\hat{f}_k^{(\eta,\iota)} = \sum_{l\in\mathbb{Z}} \hat{f}_l^{(\gamma,\delta)} \lambda_{l,k}^{\Psi,(\eta,\iota,\gamma,\delta)} \tag{5.3}$$

$$\lambda_{l,k}^{\Psi,(\eta,\iota,\gamma,\delta)} = \left\langle \Psi_l^{(\gamma,\delta)}, \Psi_k^{(\eta,\iota)} \right\rangle_{w_\theta^{(\eta,\iota)}} \tag{5.4}$$

The Fourier functions $\Psi_k^{(\gamma,\delta)}$ are composed of Jacobi polynomials, so naturally these connection coefficients can be derived from the Jacobi polynomial connection coefficients $\lambda_{n,m}^{P,(\alpha,\beta,\varepsilon,\zeta)}$.

5.4.1 Performing the Connection

Performing this connection quickly without resorting to complicated expressions in the literature for Jacobi polynomials [51], [5] hinges on the use of Theorem 3.1. Recall from Theorem 3.1 that the expression analogous to (5.3) for Jacobi polynomials collapses to a finite relation if the class parameters are integrally separated (e.g. $|\eta - \gamma| \in \mathbb{N}_0$ and $|\iota - \delta| \in \mathbb{N}_0$). Moreover, the number of calculations required to compute each mode is directly proportional to the integral separation. Because Jacobi polynomials are building blocks for the generalized Fourier functions, the aforementioned properties hold true for Fourier functions as well.

Proposition 5.1 *For any $\gamma, \delta > -\frac{1}{2}$ and any $G, D \in \mathbb{N}$, the connection problem*

$$f(\theta) = \sum_{k=-\infty}^{\infty} \hat{f}_k^{(\gamma,\delta)} \Psi_k^{(\gamma,\delta)}(\theta) \quad \longrightarrow \quad f(\theta) = \sum_{k=-\infty}^{\infty} \hat{f}_k^{(\gamma+G,\delta+D)} \Psi_k^{(\gamma+G,\delta+D)}(\theta),$$

can be solved exactly via the relation

$$\hat{f}_k^{(\gamma+G,\delta+D)} = \sum_{l=|k|}^{|k|+G+D} \lambda_{k,l}^{\Psi} \hat{f}_l^{(\gamma,\delta)} + \sum_{l=-|k|-G-D}^{-|k|} \lambda_{k,l}^{\Psi} \hat{f}_l^{(\gamma,\delta)}. \tag{5.5}$$

Note that in general $\lambda_{k,l} \neq 0$ for all k, l. Because the $\left\{\Psi_k^{(\gamma,\delta)}\right\}_{|k|\leq K} \in \mathcal{T}_K$ for all (γ, δ), we have

$$\lambda_{k,l}^{\Psi} = 0, \qquad |k| > |l|. \tag{5.6}$$

In addition, the result of Proposition 5.1 tells us that

$$\lambda_{k,l}^{\Psi} = 0, \qquad |k| + G + D < |l|. \tag{5.7}$$

We stress again that this result is nontrivial. This also therefore yields the connection

$$\Psi_m^{(\gamma,\delta)}(\theta) \;=\; \begin{cases} \sum_{k=-m}^{m} \lambda_{k,m}^{\Psi} \Psi_k^{(\gamma+G,\delta+D)}(\theta), & |m| \leq G + D \\ \\ \sum_{|k|=m-G}^{m} \lambda_{k,m}^{\Psi} \Psi_k^{(\gamma+G,\delta+D)}(\theta), & |m| > G + D, \end{cases} \tag{5.8}$$

i.e. $\Psi_m^{(\gamma,\delta)}$ is a linear combination of at most $2(G+D)+1$ functions $\Psi_k^{(\gamma+G,\delta+D)}$. This result will later assist us in utilizing the FFT for the generalized Wiener functions.

We now illustrate how to calculate the Szegö-Fourier connection coefficients λ^{Ψ} in Proposition 5.1 from the Jacobi coefficients λ^{P}. In the following, we make use of the

notation:

$$n := |k| - 1, \qquad \alpha = \delta - \tfrac{1}{2}, \qquad \beta = \gamma - \tfrac{1}{2}.$$

We have

$$\left.\begin{aligned} \tilde{P}_{n+1}^{(\alpha,\beta)} &= \Psi_k^{(\gamma,\delta)} + \Psi_{-k}^{(\gamma,\delta)} \\ \tilde{P}_n^{(\alpha+1,\beta+1)} &= \Psi_{|k|}^{(\gamma,\delta)} - \Psi_{-|k|}^{(\gamma,\delta)} \end{aligned}\right\} n \geq 0,$$

$$\tilde{P}_0^{(\alpha,\beta)} = \sqrt{2}\Psi_0^{(\gamma,\delta)}.$$

Therefore, from the modes $\hat{f}_k^{(\gamma,\delta)}$ we can derive two sets of Jacobi modes:

$$\hat{e}_n^{(\alpha,\beta)} = \hat{f}_n^{(\gamma,\delta)} + \hat{f}_{-n}^{(\gamma,\delta)}, \qquad n \geq 1,$$

$$\hat{o}_n^{(\alpha+1,\beta+1)} = \hat{f}_{n+1}^{(\gamma,\delta)} - \hat{f}_{-n-1}^{(\gamma,\delta)}, \qquad n \geq 0,$$

$$\hat{e}_0^{(\alpha,\beta)} = \sqrt{2}\hat{f}_0^{(\gamma)}.$$

The Jacobi modes $\hat{e}_n$ are modes of an expansion in polynomials $\tilde{P}_n^{(\alpha,\beta)}$ and the modes $\hat{o}_n$ are for an expansion in $\tilde{P}_n^{(\alpha+1,\beta+1)}$. With these modes in hand, we can use the Jacobi connection coefficients to promote the coefficients:

$$\hat{e}_n^{(\alpha+D,\beta+G)} = \sum_{m=0}^{G} \lambda_{n,n+m}^{P} \hat{e}_{n+m}^{(\alpha,\beta)}, \quad \text{where} \quad \lambda^P = \lambda^{P,(\alpha,\beta,D,G)}, \qquad n \geq 0$$

$$\hat{o}_n^{(\alpha+D+1,\beta+G+1)} = \sum_{m=0}^{G} \lambda_{n,n+m}^{P} \hat{o}_{n+m}^{(\alpha+1,\beta+1)}, \quad \text{where} \quad \lambda^P = \lambda^{P,(\alpha+1,\beta+1,D,G)}, \qquad n \geq 0.$$

Finally, we redistribute the modes back into Szegö-Fourier form to yield what we desire:

$$\begin{aligned}
\hat{f}_n^{(\gamma+G,\delta+D)} &= \tfrac{1}{2}\left[\hat{e}_n^{(\alpha+D,\beta+G)} + \hat{o}_{n-1}^{(\alpha+D+1,\beta+D+1)}\right], \qquad n \geq 1 \\
\hat{f}_{-n}^{(\gamma+G,\delta+D)} &= \tfrac{1}{2}\left[\hat{e}_n^{(\alpha+D,\beta+G)} - \hat{o}_{n-1}^{(\alpha+D+1,\beta+G+1)}\right], \qquad n \geq 1 \\
\hat{f}_0^{(\gamma+G,\delta+D)} &= \tfrac{\hat{e}_0^{(\alpha+D,\beta+G)}}{\sqrt{2}}.
\end{aligned}$$

With this, we may explicitly write the connections as:

$$\begin{aligned}
\hat{f}_k^{(\gamma+G,\delta+D)} &= \textstyle\sum_{m=0}^{G+D} \tfrac{1}{2}\left[\lambda^{P;(\alpha,\beta)}_{|k|,|k|+m} + \operatorname{sgn}(k)\lambda^{P;(\alpha+1,\beta+1)}_{|k|,|k|+m}\right]\hat{f}^{(\gamma,\delta)}_{|k|+m} + \\
&\qquad \textstyle\sum_{m=0}^{G+D} \tfrac{1}{2}\left[\lambda^{P;(\alpha,\beta)}_{|k|,|k|+m} - \operatorname{sgn}(k)\lambda^{P;(\alpha+1,\beta+1)}_{|k|,|k|+m}\right]\hat{f}^{(\gamma,\delta)}_{-|k|-m}, \quad |k| \geq 1 \\
\hat{f}_0^{(\gamma+G,\delta+D)} &= \tfrac{1}{\sqrt{2}}\textstyle\sum_{m=0}^{G+D} \lambda^{P;(\alpha,\beta)}_{0,m}\hat{f}^{(\gamma,\delta)}_{m} + \tfrac{1}{\sqrt{2}}\sum_{m=0}^{G+D} \lambda^{P;(\alpha,\beta)}_{0,m}\hat{f}^{(\gamma,\delta)}_{-m}.
\end{aligned}$$

Therefore we have an explicit expression for the Szegö-Fourier connection coefficients in (5.3):

$$\lambda^{\Psi}_{k,\pm(|k|+m)} = \begin{cases} \tfrac{1}{2}\left[\lambda^{P;(\alpha,\beta)}_{|k|,|k|+m} \pm \operatorname{sgn}(k)\lambda^{P;(\alpha+1,\beta+1)}_{|k|,|k|+m}\right], & |k| \geq 1 \\ \\ \tfrac{1}{\sqrt{2}}\lambda^{P;(\alpha,\beta)}_{0,m}, & k = 0 \end{cases} \tag{5.9}$$

Of course, owing to the constraints (5.6-5.7), the above equation restricts $0 \leq m \leq G + D$. The reverse connection problem (converting $\hat{f}_k^{\Psi,(\gamma+G,\delta+D)}$ modes to $\hat{f}_k^{(\gamma,\delta)}$ modes) can be quickly solved by reversing the above procedure and using the fact that the Jacobi connection problem with integer separation is banded upper-triangular and thus is $O(N)$ calculable sequentially via back-substitution.

Note that we have determined how to quickly and exactly accomplish the connection problems for the unweighted functions

$$\sum_k \hat{f}_k^{\Psi,(\gamma,\delta)} \Psi_k^{(\gamma,\delta)}(\theta) \quad \longleftrightarrow \quad \sum_k \hat{f}_k^{\Psi,(\gamma+G,\delta+D)} \Psi_k^{(\gamma+G,\delta+D)}(\theta),$$

in $O(N)$ time where N is the total number of modes when $G, D \in \mathbb{N}$. These connections can be performed by utilizing the connection coefficients in (5.9) along with the sparse connection result of Proposition 5.1. For $G, D \notin \mathbb{Z}$, there is no sparse connection result for the modes, and so while the connection coefficients $\lambda_{k,l}^{\Psi}$ can still be calculated based on known connection coefficients for Jacobi polynomials, the coefficients do not terminate finitely, and it is more expensive to change δ or γ.

5.4.2 Using the FFT

Performing the connection problem quickly as outlined in the previous section allows us to use the FFT for modal/nodal transformations in the $\Psi_k^{(\gamma,\delta)}$. The procedure is as follows:

Let $\gamma, \delta \in \mathbb{N}_0$ be given, and suppose that we have $2N+1$ nodal evaluations f_k of a function f at the canonical Fourier quadrature nodes.

1. Use the FFT to compute $2N+1$ modal coefficients $\hat{f}_k^{(0,0)}$, i.e., the canoncial Fourier modes.
2. Perform the $\Psi^{(0,0)} \to \Psi^{(\gamma,\delta)}$ connection problem as outlined in section 5.4.1 to obtain the $2N+1$ modes $\hat{f}_k^{(\gamma,\delta)}$.

Step 1 takes $O(N \log N)$ calculations, and step 2 takes $O(N(\gamma+\delta))$ calculations.

In addition, the upper and lower $G + D$ modes are wrong, but every other mode is exact, modulo the FFT aliasing error. That this procedure is faster than using direct quadrature may not be clear; however, we postpone giving numerical results until Section 6.6 where we use this algorithm as one component of a method for using the FFT with the generalized Wiener basis functions.

5.5 Other Properties

The generalized Fourier functions also inherit some other properties from Jacobi polynomials that we explain here.

5.5.1 Recurrence Relations

Due to the strong dependence of the Szegö-Fourier functions on the Jacobi polynomials, they inherit six-term recurrence relations from the three-term recurrences for orthogonal polynomials.

$$D_n \Psi_{n+1}^{(\gamma,\delta)} = \left[A_n e^{i\theta} - B_n\right] \Psi_n^{(\gamma,\delta)} + \left[A_{-n} e^{-i\theta} - B_{-n}\right] \Psi_{-n}^{(\gamma,\delta)} + C_n \Psi_{n-1}^{(\gamma,\delta)} + C_{-n} \Psi_{-(n-1)}^{(\gamma,\delta)},$$

$$\Psi_{n+1}^{(\gamma,\delta)} = \left[U_n \cos\theta - V_n\right] \Psi_n^{(\gamma,\delta)} + \left[U_{-n} \cos\theta - V_{-n}\right] \Psi_{-n}^{(\gamma,\delta)} + W_n \Psi_{n-1}^{(\gamma,\delta)} + W_{-n} \Psi_{-(n-1)}^{(\gamma,\delta)},$$

$$\Psi_{n+1}^{(\gamma,\delta)} = \left[\tilde{U}_n i \sin\theta - \tilde{V}_n\right] \Psi_n^{(\gamma,\delta)} + \left[\tilde{U}_{-n} i \sin\theta - \tilde{V}_{-n}\right] \Psi_{-n}^{(\gamma,\delta)} + \tilde{W}_n \Psi_{n-1}^{(\gamma,\delta)} + \tilde{W}_{-n} \Psi_{-(n-1)}^{(\gamma,\delta)}.$$

We give formulae for all the constants $A, B, C, D, U, V, W, \tilde{U}, \tilde{V}, \tilde{W}$ in Appendix C. Note that since the $\Psi_k^{(\gamma,\delta)}$ are *not* polynomials in $z = e^{i\theta}$, there is not a three-term recurrence as there would normally be for orthogonal polynomials on the unit

disk (unless of course $\gamma = \delta = 0$). Although the above formula is a complex-valued six-term recurrence relation, it is no more difficult computationally than the pair of three-term recurrences necessary to generate $\tilde{P}_n^{(\alpha,\beta)}$ and $\tilde{P}_n^{(\alpha+1,\beta+1)}$ because $\Psi_n^{(\gamma,\delta)}$ is the complex conjugate of $\Psi_{-n}^{(\gamma,\delta)}$ and therefore does not need to be generated independently. Additionally, all the recurrence constants are real-valued. Therefore, directly using of any of the above six-term recurrences for generating the $\Psi_k^{(\gamma,\delta)}$ is just as expensive as forming $\Psi_k^{(\gamma,\delta)}$ by the even/odd synthesis of $\tilde{P}_n^{(\alpha,\beta)}$ and $\tilde{P}_n^{(\alpha+1,\beta+1)}$ in Theorem 5.1.

It is reassuring to note that simplifying the recurrence constants in the case $\gamma = \delta = 0$ yields, up to normalization, the trivial recurrence relations for the monomials on the unit disk $\Psi_k^{(0,0)}(z) = \frac{z^k}{\sqrt{2\pi}}$:

$$\begin{aligned}
\Psi_{n+1}^{(0,0)} &= e^{i\theta}\Psi_n^{(0,0)}, \\
\Psi_{n+1}^{(0,0)} &= 2\cos\theta\Psi_n^{(0,0)} - \Psi_{n-1}^{(0,0)}, \\
\Psi_{n+1}^{(0,0)} &= 2i\sin\theta\Psi_n^{(0,0)} + \Psi_{n-1}^{(0,0)}.
\end{aligned}$$

Naturally, a recurrence relation for the unweighted $\Psi_k^{(\gamma)}(\theta)$ translates directly into one for the weighted Fourier functions $\psi_k^{(\gamma,\delta)}$. The weighted functions may be generated by first generating the unweighted functions $\Psi_k^{(\gamma\delta)}$ and then multiplying by the phase-shifted square root $\sqrt[*]{w}$.

5.5.2 Derivatives

There are no properties of the functions $\Psi_k^{(\gamma,\delta)}$ that immediately yield useful expressions for derivatives. However, there is one highly nontrivial (and at present seemingly useless) result that we shall derive which plays a central role in the sparsity of the stiffness matrix for the generalized Wiener basis that is derived in the next chapter.

First we start with an interesting observation:

$$\tfrac{\mathrm{d}}{\mathrm{d}\theta}\tilde{P}_n^{(\alpha,\beta)}(\cos\theta) \;=\; -\sin(\theta)\sqrt{n(n+\alpha+\beta+1)}\tilde{P}_{n-1}^{(\alpha+1,\beta+1)}(\cos\theta)\,, \qquad n>0$$

A more fortuitous way of stating the above is

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\,\mathrm{Re}\left\{\Psi_k^{(\gamma,\delta)}(\theta)\right\} = -\,\mathrm{sgn}(k)\sqrt{|k|(|k|+\gamma+\delta)}\,\mathrm{Im}\left\{\Psi_k^{(\gamma,\delta)}(\theta)\right\},$$

i.e., the imaginary part of $\Psi_k^{(\gamma,\delta)}$ is the derivative of the real part, modulo multiplicative normalization. (This is very obvious if $\gamma=\delta=0$.)

Clearly one can relatively simply write the derivative of the unweighted functions $\Psi_k^{(\gamma,\delta)}$ as combinations of Jacobi polynomials for $|k|\geq 2$:

$$\begin{aligned}\tfrac{\mathrm{d}}{\mathrm{d}\theta}\Psi_k^{(\gamma,\delta)} \;=\; \tfrac{1}{2}\Big[&-\sqrt{|k|(|k|+\gamma+\delta)}\sin(\theta)\tilde{P}_{|k|-1}^{(\alpha+1,\beta+1)}(\cos\theta) + i\,\mathrm{sgn}(k)\cos\theta\tilde{P}_{|k|-1}^{(\alpha+1,\beta+1)}\\ &-i\,\mathrm{sgn}(k)\sin^2(\theta)\sqrt{(|k|-1)(|k|+\gamma+\delta+1)}\tilde{P}_{|k|-2}^{(\alpha+2,\beta+2)}(\cos\theta)\Big]\,.\end{aligned} \tag{5.10}$$

Similar expressions exist for $|k|\leq 1$. However, the 'useless' result we alluded to at the beginning of this section is the following:

Lemma 5.1 *The following expression holds for all* $\gamma, \delta > -\frac{1}{2}$ *when* $|k| \geq 1$*:*

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\Psi_k^{(\gamma,\delta)}(\theta) \;=\; c_1 \operatorname{Im}\left\{\Psi_k^{(\gamma,\delta)}\right\} + c_2 \operatorname{Re}\left\{\Psi_k^{(\gamma+1,\delta)}\right\} + c_3 \operatorname{Re}\left\{\Psi_{k-1}^{(\gamma+1,\delta)}\right\}.$$

Our proof of Lemma 5.1 is not very instructive so we only include it in Appendix D, where the constants c_i can also be found. Again, this lemma does not at first glance appear to have merit for anything useful. However, it will be extremely useful in proving the sparsity of the stiffness matrix for the generalized Wiener rational functions.

We finally note that Lemma 5.1 is very reminiscent of the derivative promotion relation for Jacobi polynomials, equation (A.36). This can be seen by noting that the $\alpha \to \alpha + 1$ and $\beta \to \beta + 1$ Jacobi promotion relations can be used to derive $\gamma \to \gamma + 1$ and $\delta \to \delta + 1$ Fourier promotion relations. Applying these to the three right-hand-side terms in the result of the lemma shows that the derivative of $\Psi_k^{(\gamma,\delta)}$ essentially amounts to promoting γ and δ by 1. This is also essentially the result of equation (A.36) with (γ, δ) replaced by (α, β).

5.6 Conclusions

In this chapter we have used Jacobi polynomials as the building blocks for generalized periodic Fourier series. However, our method can also be used for constructing nonperiodic basis functions that are like trigonometric polynomials. The periodic generalized Fourier series are the foundation for the generalized Wiener rational basis functions in the next chapter.

On their own the generalized Fourier series may be used in situations where it is necessary to perform an L^2-weighted projection on a finite interval or over the unit circle in the complex plane; in such cases the FFT can be used on equidistant samples. Unfortunately, the stiffness matrix for the unweighted functions $\Psi_k^{(\gamma,\delta)}$ when $\gamma, \delta \neq 0$ is not sparse like the diagonal matrix for the canonical basis $\Psi_k^{(0,0)}$. However, the derivatives of these functions are of a form that does admit a sparsity result for the stiffness matrix of the generalized Wiener rational basis. In any case, the weighted norm under which there functions are orthogonal is a useful one for e.g. windowed Fourier transforms. Further investigation regarding the utility of these methods for the windowed Fourier analysis must be carried out.

In view of their use as a tool for the Wiener basis set, we can surmise of some further research that can be done to improve this application. It is unclear that the periodic Fourier series of Section 5.2 is superior to the 'antiperiodic' Fourier series of Section 5.3, or to any other judicious combination of orthogonal polynomials, when viewed from the lens of expansions over infinite intervals. Indeed, it may be the case that there is a different polynomial combination technique that is 'better' to use. Further studies can be done to ascertain this. In addition, there exist possibilities for using other types of linear fractional maps taking $\mathbb{T} \to \mathbb{T}$ that prove beneficial for expansions over infinite intervals; e.g. for shifting the location of $x = 0$ to some other location by mapping techniques rather than by a simple affine shift. Such a possibility might allow for the ability to use the FFT to augment the basis on-the-fly during an expensive computation. In any case, these investigations must be carried out on the finite interval first.

Chapter Six

The Generalized Wiener Rational Basis

6.1 Goals

The approximation of a function by a finite sum of canonical basis elements has long been a hallmark tool in numerical analysis. Over the finite interval much is known about expansion properties and periodic Fourier expansions or polynomial expansions are well-studied. On infinite intervals there are complications due to the unbounded domain on which the approximation is necessary. Nevertheless many expansion sets have been successfully investigated in this case; Hermite functions provide a suitable method for approximation when it can be assumed that the function decays exponentially; for functions that do not decay exponentially, the so-called mapped Chebyshev rational functions can fill the void and open up the possibility for utilizing the Fast Fourier Transform; additionally, a Fourier basis mapped to the real line has been explored and provides an additional method for function approximation over the infinite interval. This last basis set, originally introduced by Wiener, serves as inspiration for the family of basis sets we propose in this chapter.

Our generalized basis is inspired by a collection of orthogonal and complete functions originally published by Wiener [69]. Recall the defintion of the functions from equation (1.2):

$$\phi_n(x) = \frac{(1-ix)^n}{\sqrt{\pi}(1+ix)^{n+1}}, \quad n \in \mathbb{N}_0. \tag{6.1}$$

We note that the functions $\phi_n(x)$ presented above have magnitude that decays like $\frac{1}{x}$ as $|x| \to \infty$. To generalize this, we seek a collection of L^2-orthogonal and L^2-complete basis functions whose domain is the entire real line. In addition, we desire the ability to specify a parameter $s > \frac{1}{2}$ that will denote the polynomial decay at $\pm\infty$ of each of the the basis functions. It has been noted that the functions (6.1) are weighted maps of the canonical Fourier basis $e^{in\theta}$ for $\theta \in [0, 2\pi]$ (see e.g. [15], [67]). Thus, our generalization is based heavily upon the generalized Fourier functions from

Chapter 5; in fact they are direct mappings of this generalized Fourier series.

The outline of this chapter is as follows. In section 6.2 we formulate and derive the basis. Sections 6.3 and 6.4 discuss some elementary periodicity and symmetry relations. Section 6.5 introduces some quadrature methods for the generalized Wiener functions based upon Jacobi-Gauss quadrature rules. The Wiener connection problem is discussed in Section 6.6 and applied in Section 6.7 to usage of the FFT. Finally, Section 6.8 details a sparsity result for the stiffness matrix.

6.2 Mapping to the real line

The goal of this section is the derivation of the long-discussed generalized Wiener rational function basis set. These functions are merely maps of the weighted generalized Fourier functions of Chapter 5. The generalized Fourier functions $\Psi_k^{(\gamma,\delta)}$ are ostensibly parameterized by γ and δ. By virtue of the particular linear fractional transformation we have chosen, $\theta = 0$ gets mapped to $x = 0$ and $\theta = \pm\pi$ gets mapped to $x = \pm\infty$. Therefore the parameter δ, denoting the decay at $\theta = 0$, plays very little role for us.

Having developed the necessary preliminaries on the finite interval, we now jump to the infinite line $x \in \mathbb{R}$ using the mappings introduced in Table 2.1. To facilitate the changes, the following identities characterizing the mapping between θ-space and x-space are useful:

$$\cos\theta = \tfrac{1-x^2}{1+x^2}, \qquad (1-\cos\theta) = \tfrac{2x^2}{x^2+1},$$

$$\sin\theta = \tfrac{2x}{x^2+1}, \qquad (1+\cos\theta) = \tfrac{2}{x^2+1}.$$

Using these identities, we rewrite and relabel the functions $\Psi_k^{(\gamma,\delta)}(\theta)$:

$$\begin{aligned}\Phi_k^{(s,t)}(x) &:= \Psi_k^{(s-1,t)}(\theta(x)) \\ &= \begin{cases} \frac{1}{\sqrt{2}}\tilde{P}_0^{(t-1/2,s-3/2)}\left(\frac{1-x^2}{1+x^2}\right), & k=0 \\ \frac{1}{2}\left[\tilde{P}_{|k|}^{(t-1/2,s-3/2)}\left(\frac{1-x^2}{1+x^2}\right) + \frac{2ix\,\mathrm{sgn}(k)}{x^2+1}\tilde{P}_{|k|-1}^{(t+1/2,s-1/2)}\left(\frac{1-x^2}{1+x^2}\right)\right], & k\neq 0 \end{cases}\end{aligned}$$

The above definition is valid for any $s > \frac{1}{2}$, $t > -\frac{1}{2}$. $(s,t) = (1,0)$ corresponds to a mapping of the canonical Fourier basis. The functions $\Phi_k^{(s,t)}$ are orthogonal over the weight $w_x^{(s,t)}$. By following the route from Corollary 5.1 we can distribute the weight over the basis functions, and in this particular instance we choose the phase-shifted square root given in (2.1):

$$\begin{aligned}\phi_k^{(s,t)} &:= \sqrt[*]{w_x^{(s,t)}}\Phi_k^{(s,t)}(x) \\ &= \begin{cases} \frac{2^{\left(\frac{s+t-1}{2}\right)}x^t}{(x-i)^{s+t}}\tilde{P}_0^{(t-1/2,s-3/2)}\left(\frac{1-x^2}{1+x^2}\right), & k=0 \\ \frac{2^{\left(\frac{s+t}{2}-1\right)}x^t}{(x-i)^{s+t}}\left[\tilde{P}_{|k|}^{(t-1/2,s-3/2)}\left(\frac{1-x^2}{1+x^2}\right) + \frac{2ix\,\mathrm{sgn}(k)}{x^2+1}\tilde{P}_{|k|-1}^{(t+1/2,s-1/2)}\left(\frac{1-x^2}{1+x^2}\right)\right], & k\neq 0. \end{cases}\end{aligned}$$

Because the decay parameter t (tied to the parameter δ) denotes decay at $x = 0$ (which we are not so concerned with), we will frequently assign it to take on the value 0, and when we do this, we will omit mentioning the parameter at all:

$$\phi_k^{(s)}(x) := \phi_k^{(s,0)}(x)$$

$$\Phi_k^{(s)}(x) := \Phi_k^{(s,0)}(x)$$

$$\Psi_k^{(\gamma)}(\theta) := \Psi_k^{(\gamma,0)}(\theta)$$

At present there is no clear reason why we have chosen to use $\sqrt[*]{w_x^{(s,t)}}$ instead of the usual square root $\sqrt{w_x^{(s,t)}}$ to distribute the weight onto $\phi_k^{(s,t)}$. However, the corollary following the coming proposition should provide the motivation.

Proposition 6.1 *For any $s > \frac{1}{2}$, $t > -\frac{1}{2}$, the functions $\Phi_k^{(s,t)}(x)$ are complete and orthonormal in $L^2\left(\mathbb{R}, \mathbb{C}; w_x^{(s,t)}\right)$. The functions $\phi_k^{(s,t)}(x)$ are complete and orthonormal in $L^2(\mathbb{R}, \mathbb{C})$. Furthermore, the decay rate of these functions can be characterized as*

$$\lim_{|x|\to\infty} \left| x^u \phi_k^{(s)}(x) \right| < \infty, \qquad u \leq s$$

Corollary 6.1 *Recalling the definition of Wiener's original basis functions $\phi_n(x)$ in (6.1), the following relation holds:*

$$i\sqrt{2}\phi_n^{(1)}(x) := \phi_n(x), \qquad n \in \mathbb{N}_0.$$

We show plots of the functions $\phi_k^{(4)}$ in Figure 6.1.

The conclusion of the corollary is easily seen if one makes the connection

$$e^{i\theta} = \frac{i - x}{i + x},$$

along with knowledge of the fact that $\Phi_k^{(1)}(x) = \psi_k^{(0)}(\theta) = \frac{1}{\sqrt{2\pi}} e^{ik\theta}$. We have thus shown that the orthogonal functions $\phi_k^{(s)}$ over the real line are a generalization of Wiener's original basis set. Furthermore, $\phi_k^{(s)}$ decays like $\frac{1}{x^s}$ while retaining orthogonality under the same unit weight measure. When s is an integer, the functions are also purely rational: they are the division of one polynomial in x by another. This connection was rather helpful in the nascent stages of the computing when the calculation of a non-polynomial function required significantly more computational

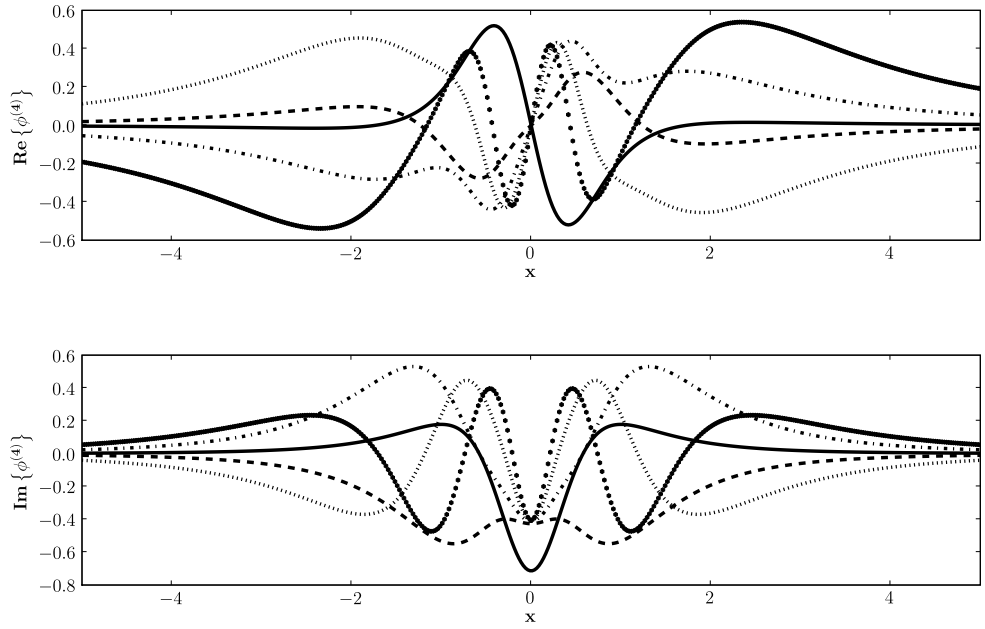

Figure 6.1: Plots of the functions $\phi_k^{(4)}(x)$ for $k = 0, 1, 2, 3, 4$, (respectively solid, dashed, dot-dashed, dotted, circle marks).

investment, but now this property is probably more aesthetic than functional. As a result, our use of the quantity $\sqrt[*]{w_x^{(s,t)}}$ is not entirely necessary; it is equally valid to use the traditional squre root $\sqrt{w_x^{(s,t)}}$. The only sacrifice made is that the analogous written form of Corollary 6.1 becomes less fortuitous and is complicated by x-dependent phase-shift factors. The same observation is true of the weight $\sqrt[*]{w_\theta^{(\gamma,\delta)}}$ used in the definition of the weighted Szegö-Fourier functions $\psi_k^{(\gamma)}(\theta)$.

We have accomplished our goal of deriving basis functions satisfying tunable decay rate while maintaining L^2-orthogonality. However, it is not clear that these are convenient or useful functions at all for a numerical analyst. We will now present some properties of the basis and make the argument that these basis functions indeed are very useful for solving problems in scientific computing.

6.3 Symmetries

The derivation of the basis functions automatically yields various simple properties. Note that due to the mapping, any property of the basis on the real line $x \in \mathbb{R}$ also applies to the respective trigonometric interval $\theta \in [-\pi, \pi]$. We omit the proof of these properties as they are elementary:

1. Index symmetry

$$\begin{aligned} \Phi_k^{(s)}(x) &= \overline{\Phi_{-k}^{(s)}(x)} \\ |\Phi_k^{(s)}(x)| &= |\Phi_{-k}^{(s)}(x)| \\ |\phi_k^{(s)}(x)| &= |\phi_{-k}^{(s)}(x)| \\ \phi_k^{(1)}(x) &= \overline{\phi_{-k-1}^{(1)}(x)} \end{aligned}$$

2. Function symmetry

$$\begin{aligned} \mathrm{Re}\left\{\Phi_k^{(s)}(x)\right\} &= \mathrm{Re}\left\{\Phi_k^{(s)}(-x)\right\} \\ \mathrm{Im}\left\{\Phi_k^{(s)}(x)\right\} &= -\mathrm{Im}\left\{\Phi_k^{(s)}(-x)\right\} \\ |\Phi_k^{(s)}(x)| &= |\Phi_k^{(s)}(-x)| \\ |\phi_k^{(s)}(x)| &= |\phi_k^{(s)}(-x)| \end{aligned}$$

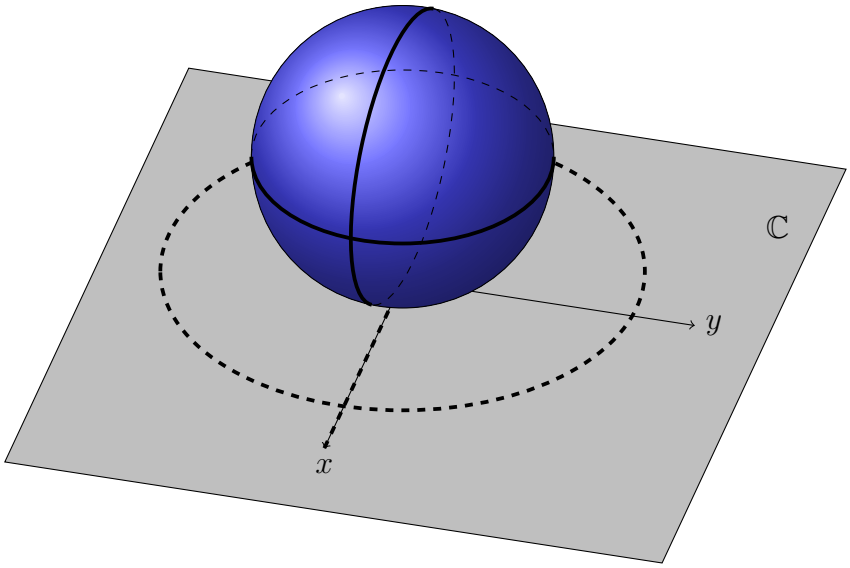

Figure 6.2: Representation of the complex plane $\mathbb{C}$ as the Riemann sphere. The latitudinal equator on the sphere is identified with the unit circle in the complex plane (dashed line), and the longitudinal line on the sphere is identified by the x-axis in the plane. This equivalence can be seen via stereographic projection from the north pole of the sphere.

6.4 Periodicity

Because trigonometric polynomials are periodic over $\theta \in [-\pi, \pi]$, we cannot expect this condition to be violated on the infinite interval $x \in \mathbb{R}$. From the viewpoint of expanding functions over the infinite interval $\mathbb{R}$, the points $x = \pm\infty$ are both unique points. However, because of the mapping, the basis functions view the points $x = \pm\infty$ the same as they view the points $\theta = \pm\pi$: i.e. they are the same point. See Figure 6.2. This may serve as a disadvantage if we wish to e.g. expand functions with different decay rates at $\pm\infty$ because different rates of decay may effectively be non-smooth behavior of the function at a single point, which degrades the convergence rate of the approximation.

In particular it is known that although a Fourier series approximation will converge in

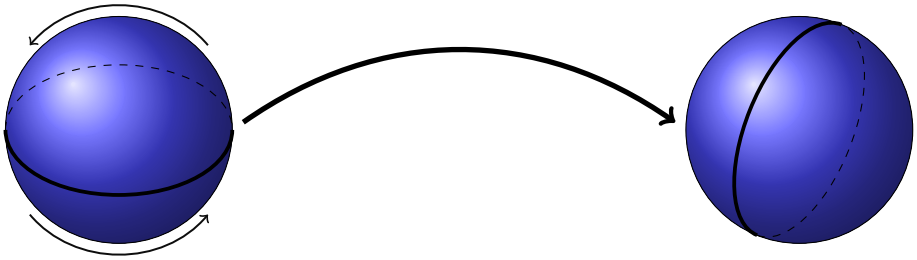

Figure 6.3: The effect of the linear fractional map given in Section 6.2 on the Riemann sphere. Because the map simply rotates the sphere with $\theta = pi$ mapping $x = \infty$, it is no surprise that periodicity originally present at $\theta = \pm\pi$ manifests as the same phenomenon at $x = \pm\infty$.

the L^2 sense for an L^2 function, the rate of convergence is only algebraic. Naturally, this deficiency will follow us to the infinite interval (see Figure 6.3). Indeed, such observations have already been made [22]. In Section 8 we will present examples that explicitly illustrate this lack of fast convergence rate when the function to be expanded is not 'periodic' at $x = \pm\infty$.

To combat this we may naturally wonder how well the 'antiperiodic' functions from Section 5.3 will work when mapped under the same transformation to the real line. Indeed, as remarked in Chapter 5, there may be virtues for expansions on the real line if we choose a nonperiodic Fourier-type construction exemplified by Section 5.3. However, this is still an open area for research.

6.5 Quadrature

We now turn to quadrature rules that will compute integrals over the real line. We recall the following notation: the pair $\left\{r_n^{(\alpha,\beta)}, \omega_n^{(\alpha,\beta)}\right\}_{n=1}^{N}$ denotes the N-point

Gauss-quadrature for the Jacobi polynomial of class (α, β), i.e.,

$$\int_{-1}^{1} f(r) w_r^{(\alpha,\beta)} \mathrm{d}r = \sum_{n=1}^{N} f\left(r_n^{(\alpha,\beta)}\right) \omega_n^{(\alpha,\beta)}, \qquad \forall f \in \mathcal{B}_{2N}.$$

We suppress the dependence of $r_n^{(\alpha,\beta)}$ and $\omega_n^{(\alpha,\beta)}$ on N. We also denote $\left\{r_n^{(\alpha,\beta);\mathrm{GR}}, \omega_n^{(\alpha,\beta);\mathrm{GR}}\right\}_{n=1}^{N}$ as the N-point Gauss-Radau quadrature with the fixed node $r_N^{(\alpha,\beta);\mathrm{GR}} \equiv 1$. Although it is not necessary, we assume for clarity of presentation that the nodes are ordered by n, e.g. $r_{n-1}^{(\alpha,\beta)} < r_n^{(\alpha,\beta)}$.

With the goal in mind that we wish to develop quadrature rules for the infinite line, we will take pains to develop quadrature rules in θ-space that do not have nodes at $\theta = \pm\pi$, which map to $x = \pm\infty$. We use the Jacobi-Gauss quadrature rules as the building blocks for our generalized Fourier quadrature rules.

Suppose we wish to construct an N-point quadrature rule associated with the functions $\Psi_k^{(\gamma,\delta)}(\theta)$. If N is even, then define

$$\theta_n^{(\gamma)} = \begin{cases} -\arccos\left(r_n^{(-1/2,\gamma-1/2)}\right), & 1 \le n \le \frac{N}{2} \\ \\ -\theta_{N+1-n}^{(\gamma)}, & \frac{N}{2}+1 \le n \le N, \end{cases} \tag{6.2}$$

where $r_n^{(\alpha,\beta)}$ comes from an $\frac{N}{2}$ -point quadrature rule, and

$$\Omega_n^{(\gamma)} = \begin{cases} \omega_n^{(-1/2,\gamma-1/2)}, & 1 \le n \le \frac{N}{2} \\ \\ \Omega_{N+1-n}^{(\gamma)}, & \frac{N}{2}+1 \le n \le N, \end{cases}$$

and $\omega_n^{(\alpha,\beta)}$ comes from an $\frac{N}{2}$ -point quadrature rule.

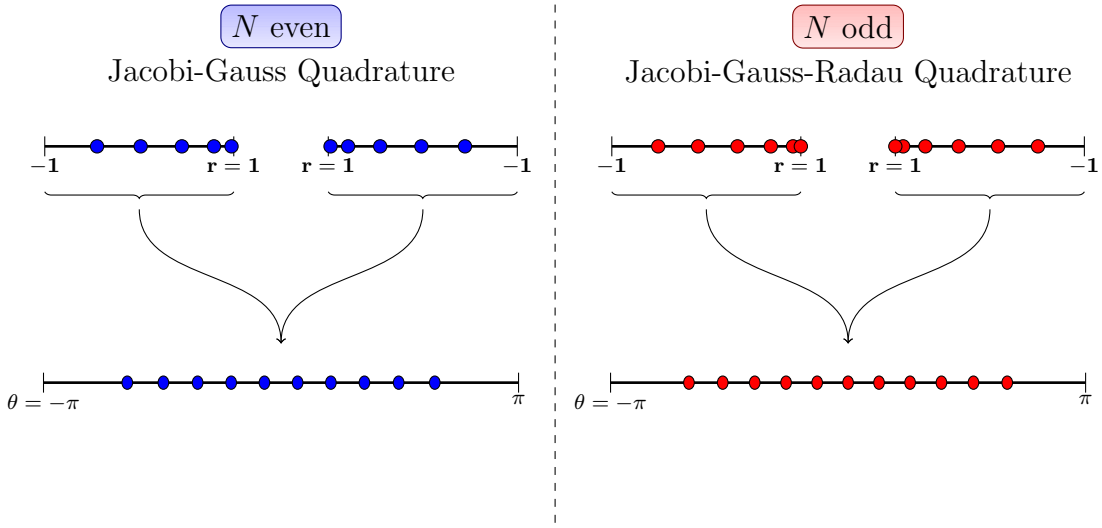

Figure 6.4: Construction of Gauss-type quadrature for generalized Fourier functions. The new quadrature rules are symmetric combinations of Jacobi-Gauss-type quadrature rules. The constructions shown are accurate node locations for $\gamma = 5$.

If N is odd, then define

$$\theta_n^{(\gamma)} = \begin{cases} -\arccos\left(r_n^{(-1/2,\gamma-1/2);\mathrm{GR}}\right), & 1 \le n \le \frac{N+1}{2} \\ \\ \theta_{N+1-n}^{(\gamma)}, & \frac{N+3}{2} \le n \le N, \end{cases} \tag{6.3}$$

where $r_n^{(\alpha,\beta);\mathrm{GR}}$ comes from an $\frac{N+1}{2}$-point quadrature rule, and

$$\Omega_n^{(\gamma)} = \begin{cases} \omega_n^{(-1/2,\gamma-1/2);\mathrm{GR}}, & 1 \le n \le \frac{N-1}{2} \\ \\ 2\omega_n^{(-1/2,\gamma-1/2);\mathrm{GR}}, & n = \frac{N+1}{2} \\ \\ \Omega_{N+1-n}^{(\gamma)}, & \frac{N+3}{2} \le n \le N. \end{cases}$$

For graphical descriptions of the above formulae, see Figure 6.4 We have used Jacobi-Gauss rules for N even and Jacobi-Gauss-Radau rules for N odd. By construction,

when N is odd, $\theta^{(\gamma)}_{\frac{N+1}{2}} = 0$ due to the Gauss-Radau rule requirement that $r^{(\alpha,\beta);R}_{\frac{N+1}{2}} = 1$. The quadrature rules derived above have no nodes at $\theta = \pm\pi$ (since there are no Jacobi-Gauss, or Jacobi-Gauss-Radau nodes at $r = -1$) and are symmetric rules for any γ. Thus they are always exact for any odd function. It is not hard to show the following result:

Proposition 6.2 *For N even, the N-point quadrature rule $\left\{\theta^{(\gamma)}_n, \Omega^{(\gamma)}_n\right\}_{n=1}^{N}$ satisfies*

$$\int_{-\pi}^{\pi} e^{ik\theta} w^{(\gamma,0)}_{\theta} \mathrm{d}\theta = \sum_{n=1}^{N} e^{ik\theta^{(\gamma)}_n} \left(\theta^{(\gamma)}_n\right) \Omega^{(\gamma)}_n, \qquad |k| \leq N-1.$$

When N is odd, the quadrature rule satisfies

$$\textstyle\int_{-\pi}^{\pi} e^{ik\theta} w^{(\gamma,0)}_{\theta} \mathrm{d}\theta = \sum_{n=1}^{N} e^{ik\theta^{(\gamma)}_n} \left(\theta^{(\gamma)}_n\right) \Omega^{(\gamma)}_n, \qquad |k| \leq N.$$

The degeneracy in the quadrature rule for N even is exactly of the same nature as the degeneracy in the canonical equispaced Fourier quadrature rule for an even number of grid points. If $\gamma = 0$ the rule $\left\{\theta^{(0)}_n, \Omega^{(0)}_n\right\}_{n=1}^{N}$ is exactly the same as the equispaced Fourier quadrature rule, symmetric about $\theta = 0$. The quadrature rule $\left\{\theta^{(0)}_n, \Omega^{(0)}_n\right\}_{n=1}^{N}$ can be used to integrate against the weight function $w^{(\gamma,0)}_{\theta}$ when $\gamma \in \mathbb{N}$ since in this case the weight is itself a trigonometric polynomial.

In order to determine a quadrature rule to integrate the weighted functions $\psi^{(\gamma)}_k(\theta)$, we can augment the weights $\Omega^{(\gamma)}_n$ to contain information about the weight function. This can be summed up in the following result:

Corollary 6.2 *The even N-point quadrature rule $\left\{\theta^{(\gamma)}_n, \omega^{(\gamma)}_n\right\}_{n=1}^{N}$, where $\omega^{(\gamma)}_n :=$*

$w_\theta^{(-\gamma,0)}\left(\theta_n^{(\gamma)}\right)\Omega_n^{(\gamma)}$ *satisfies*

$$\int_{-\pi}^{\pi}\psi_k^{(\gamma)}\overline{\psi_l^{(\gamma)}}\mathrm{d}\theta=\sum_{n=1}^{N}\psi_k^{(\gamma)}\left(\theta_n^{(\gamma)}\right)\overline{\psi_l^{(\gamma)}}\left(\theta_n^{(\gamma)}\right)\omega_n^{(\gamma)},\qquad |k|+|l|\leq N-1$$

Multiplying $\Omega_n^{(\gamma)}$ by the inverse of the weight $w_\theta^{(-\gamma,0)}$ is mathematically not a problem since none of the $\theta_n^{(\gamma)}$ are equal to $\pm\pi$, where the weight $w_\theta^{(-\gamma,0)}$ is singular. Note that since the functions $\Phi_k^{(s)}(x)$ are just a mapping of the functions $\Psi_k^{(\gamma)}(\theta)$, then the quadrature rule $\left\{x\left(\theta_n^{(s-1)}\right),\Omega_n^{(s-1)}\right\}_{n=1}^{N}$, which has nodal values over $\mathbb{R}$, can be used to integrate the functions $\Phi_k^{(s)}(x)$ over the real line. Similarly, the rule $\left\{x\left(\theta_n^{(s-1)}\right),\omega_n^{(s-1)}\right\}_{n=1}^{N}$ can be used to integrate Galerkin products of the generalized Wiener functions $\phi_k^{(s)}(x)$ over the real line.

Note that we could equally well develop Guass-like quadrature rules for $\delta\neq 0$, but we do not do so for two reasons: firstly, $\delta\neq 0$ ($t\neq 0$) is not of much concern to us. Secondly there is a technical difficulty: if both γ and δ are different from 0, then we would like to avoid placing nodes at both $\theta=0$ and $\theta=\pm\pi$ so that we can define ω_n; this cannot be done symmetrically if N is odd.

For various γ/s we graphically depict the location of the quadrature nodes for $N=21$ in Figure 6.5 on the unit circle $z\in\mathbb{T}$. Note that as we increase γ the quadrature nodes become more and more concentrated towards $z=1$ ($\theta=0$). On the real line, this manifests itself as higher concentration near $x=0$ which, although rectifiable via affine mapping, is suboptimal if one wishes to resolve functions away from $x=0$. Note that the tendency of Jacobi-Gauss nodes to become more equidistant on $[-1,1]$ as β (i.e. γ or s) is increased [43] also suggests that these generalized quadrature rules for large γ or s will not be as good as the the ones for smaller γ or s since equidistant nodes are bad for finite-interval polynomial interpolation (re-

call the lessons from Chapters 3 and 4). In addition, when $\gamma = 0$, we can use these (equidistant) quadrature nodes to employ the FFT for modal-nodal transformations.

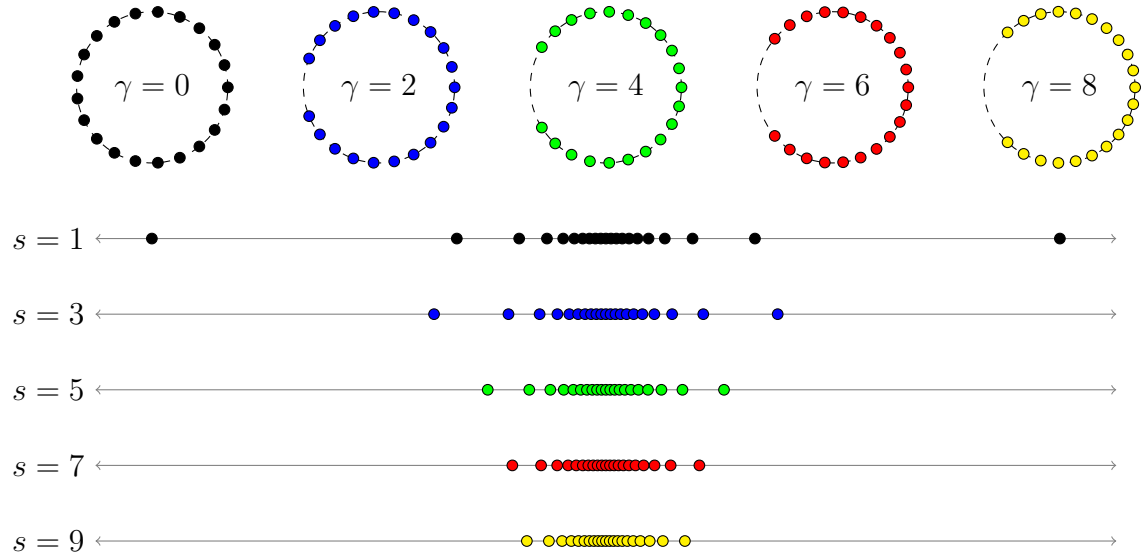

Figure 6.5: (Top) Plots of the Fourier quadrature nodes on the unit circle generated with equation (6.3), $N = 21$. (Bottom) The resulting quadrature nodes on the real line. The scale on the real line is $|x| \leq 15$.

6.6 The Connection Problem

We have already seen the connection problem before; for Jacobi polynomials in Sections 3.3 and generalized Fourier series in Section 5.4. We can use the results from the latter section pertaining to the generalized Fourier connection to accomplish generalized Wiener connection problems.

6.6.1 Unweighted Functions

The connection problem for the unweighted functions $\Phi_k^{(s,t)}(x)$ can be stated as

$$\sum_{k\in\mathbb{Z}} \hat{f}_k^{(s,t)} \Phi_k^{(s,t)} = \sum_{k\in\mathbb{Z}} \hat{f}_k^{(u,v)} \Phi_k^{(u,v)}. \tag{6.4}$$

We note that this is exactly the same statement as for the generalized Fourier functions. (Therefore we state again that this problem only makes sense if both $f \in L^2\left(\mathbb{R},\mathbb{C};w_x^{(s,t)}\right)$ and $f \in L^2\left(\mathbb{R},\mathbb{C};w_x^{(u,v)}\right)$ so that the modal coefficients $\hat{f}_k^{(s,t)}$ and $\hat{f}_k^{(u,v)}$ are well-defined.)

As has been a common theme in this thesis, problem (6.4) can only be accomplished quickly if $|s-u|, |t-v| \in \mathbb{N}_0$. In particular the connection problem for expansions truncated to N modes can be accomplished in $O(N)$ time given this restriction. In fact, this problem is entirely equivalent to problem (5.2) since $\Phi_k^{(s,t)}(x) := \Psi_k^{(s-1,t)}(\theta(x))$. Therefore the connection coefficients are equivalent

$$\lambda_{k,l}^{\Phi,(s,t,u,v)} \equiv \lambda_{k,l}^{\Psi,(s-1,t,u-1,v)},$$

and also the procedure outlined in Section 5.4 is exactly the same procedure required to accomplish problem (6.4). As usual, if $u = s+S$ and $v = t+T$ for some $S, T \in \mathbb{N}_0$, then given N modes $\hat{f}_k^{(s,t)}$, the procedure in Section 5.4 produces N modes $\hat{f}_k^{(u,v)}$ in $O(N)$ time, and all except the terminal $S+T$ modes (positive and negative k) are exact.

6.6.2 Weighted Functions

We can also perform connections for the generalized Wiener rational functions $\phi_k^{(s,t)}(x)$, although the methodology is a little more complicated. The problem statement is

$$\sum_{k\in\mathbb{Z}} \hat{f}_k^{(s,t)} \phi_k^{(s,t)} = \sum_{k\in\mathbb{Z}} \hat{f}_k^{(u,v)} \phi_k^{(u,v)} \tag{6.5}$$

Note that all the modes are well-defined as long as $f \in L^2$. Unfortunately, the connection result for this problem is not easily solvable. It is possible to perform the connection operation in $O(N)$ time if $s, t, u, v \in \mathbb{N}_0$ with a predictable estimation of any corrupted modes. However, the procedure is relatively complicated and does not seem to have great utility compared with another easier method we describe now that makes some sacrifice in the predictability of corrupted modes.

Because we are mainly concerned with expansions in $\phi_k^{(s)}$ (i.e. $t = 0$) we shall concentrate only on this expansion type. For $S \in \mathbb{N}_0$ we are interested in calculcating the modal coefficients $\hat{f}_k^{(s+S)}$ for a given function f. We assume that we have access to the modal coefficients $\hat{g}_k^{\Phi,(s)} := \left\langle g, \Phi_k^{(s)} \right\rangle_{w_x^{(s,0)}}$, where

$$g(x) := f(x) \times \frac{(x-i)^{s+S}}{2^{(s+S)/2}}.$$

If we are given these modal coefficients, then this problem is reduced to that of

Section 6.6.1 because

$$\begin{aligned}
\hat{f}_k^{(s+S)} = \left\langle f, \phi_k^{(s+S)} \right\rangle &= \left\langle f, \tfrac{2^{(s+S)/2}}{(x-i)^{s+S}} \Phi_k^{(s+S)} \right\rangle \\
&= \left\langle f \tfrac{(x-i)^{s+S}}{2^{(s+S)/2}}, \Phi_k^{(s+S)} \tfrac{2^{s+S}}{(x^2+1)^{s+S}} \right\rangle \\
&= \left\langle f \tfrac{(x-i)^{s+S}}{2^{(s+S)/2}}, \Phi_k^{(s+S)} \right\rangle_{w_x^{(s+S)}} \\
&= \hat{g}_k^{\Phi,(s+S)}.
\end{aligned} \tag{6.6}$$

Of course, we have assumed that we know $\hat{g}_k^{\Phi,(s)}$ so that calculating $\hat{g}_k^{\Phi,(s+S)}$ is merely the connection problem of Section 6.6.1. The assumption that we know the modal coefficients $\hat{g}_k^{\Phi,(s)}$ is an unnatural one, and it is not clear that there are indeed situations when we will have knowledge of them. However, one such applicable situation comes in the application of the FFT for modal/nodal transformations.

6.7 Use of the Fast Fourier Transform

Because the generalized Wiener rational functions are direct maps of a Fourier series, we can use the FFT to perform nodal-modal transformations in $O(N \log N)$ time instead of $O(N^2)$ time where N is the number of nodal points. Although it is possible to use the FFT for such transformations for relatively general values of s and t, we shall fix $t = 0$ and only consider varying s. This is in line with our interests of considering a varying decay parameter at $x = \pm\infty$.

We already know from (6.6) that an expansion of a function in the $\phi_k^{(s)}(x)$ is equivalent to an expansion of a different function in the $\Phi_k^{(s)}(x)$, which is equivalent to an

expansion in the $\Psi_k^{(\gamma)}(\theta)$; we already know from Section 5.4.2 that this last expansion can be implemented with an FFT. We have already developed all the necessary tools to use the FFT for calculating expansions in $\phi_k^{(s)}$. We consider a couple of examples in the following subsections.

6.7.1 Modal-Nodal Transformations

Assume we are given a function $f \in L^2(\mathbb{R}, \mathbb{C})$ and N nodal evaluations $f_n = f\left(x\left(\theta_n^{(0)}\right)\right)$, where $\theta_n^{(0)}$ is the canonical Fourier grid introduced in Section 6.5. For any $s \in \mathbb{N}$, the following procedure generates N modal coefficients $\hat{f}_n^{(s)}$:

1. Compute $g_n := \frac{f_n}{\sqrt[*]{w_x^{(s,0)}(x_n)}} = \frac{f_n(x_n - i)^s}{2^{s/2}}$.
2. Use the FFT on $\{g_n\}_{n=1}^N$ to compute the N modal coefficients $\hat{g}_n^{\Phi,(1)}$.
3. Perform the connection problem described in Section 6.6.1 or 5.4.1 to obtain the modes $\hat{g}_n^{\Phi,(s)}$. Then $\hat{f}_n^{(s)} := \hat{g}_n^{\Phi,(s)}$.

Step 1 is just a pointwise multiplication; one should be aware that this generates aliasing error. Step 2 is a standard Fourier series step: take equidistant nodal evaluations and generate Fourier coefficients. Step 3 performs a connection problem on the unweighted functions to generate a promoted set of modes; this step only requires $O(N)$ calculations if s is a natural number. By (6.6) these generated modes are exactly what we sought.

Remark 3 There are two sources of aliasing error in the above calculation: one is the error inherent in the sampling of f at N grid points; this cannot be mitigated without refining the grid. The second source of aliasing arises from the pointwise

multiplication performed in Step 1. However, we know that

$$-i\sqrt{2}\sqrt[*]{w_x^{(1,0)}} = \frac{-2i}{x-i} = 1 + e^{-i\theta}. \tag{6.7}$$

Because of this sparse Fourier representation, the pointwise multiplication can actually be accomplished by taking the FFT of f_n followed by s sequential back-substitutions on a bidiagonal linear system. This eliminates any aliasing error that results from nodal multiplication. Note that there is still the connection problem error where the terminal $s-1$ modes are corrupted due to lack of information from unavailable Fourier modes.

To illustrate that this method is efficient using pointwise multiplication, we show computational speedup ratios indicating how much faster the FFT method is (compared to direct matrix-vector multiplication) in Table 6.1. As s is increased, the connection problem becomes more and more costly, and so it requires more time to complete it, thus decreasing the efficiency of the FFT method.

Because of the ability to transfer nodes to modes (and vice versa) in $N \log N$ time, we can envision applying this method to take derivatives of functions over the infinite interval in $N \log N$ time as well. However, the missing ingredient for this procedure is the ability to perform modal differentiation in $N \log N$ time, or better. This will be addressed in Section 6.8.

6.7.2 Modification of the Decay Parameter

We have determined how to use the FFT to perform nodal-modal transformations with the weighted generalized Wiener rational functions $\phi_k^{(s)}(x)$ when s is an integer.

$N\backslash s$	1	2	3	4	5	6	7	8	9
100	0.7	0.3	0.2	0.2	0.0	0.2	0.2	0.2	0.1
200	2.2	0.7	0.6	0.5	0.5	0.4	0.4	0.4	0.4
320	4.4	1.6	1.4	1.2	1.2	1.0	0.9	0.9	0.8
400	6.1	2.3	2.0	1.9	1.7	1.6	1.5	1.3	1.2
512	8.5	3.6	3.2	2.8	2.6	2.2	2.3	2.0	1.9
640	13.3	5.1	4.7	4.2	4.2	3.6	3.2	3.1	3.0
768	14.5	7.1	6.3	5.2	5.0	4.4	4.0	3.9	3.7
864	18.7	8.9	7.2	6.2	5.9	5.5	5.2	5.1	4.5
1024	22.3	10.8	9.0	8.4	8.1	7.6	6.7	6.5	5.8
1600	39.7	19.3	18.4	17.7	15.9	13.4	12.8	12.5	11.6
2025	50.8	26.8	23.5	21.4	20.4	18.5	19.1	14.5	14.7
2916	103.5	42.3	38.5	35.2	39.2	32.6	28.4	28.4	25.8

Table 6.1: Tabulation of the computational speedup ($T_{FFT}/T_{\text{direct}}$) of using the FFT to compute modal coefficients of the $\phi_k^{(s)}$ basis. In the absence of aliasing error for the FFT quadrature rule, all the modes except the terminal $s-1$ modes (positive k and negative k) are exact to machine precision.

We are now prepared to consider fast modification of the decay parameter s. Let $S_1, S_2 \in \mathbb{N}$. Recall our notation that $x_n := x\left(\theta_n^{(0)}\right)$, the canonical Fourier grid points. Then given the modes $\hat{f}_k^{\phi,(S_1)}$ of an expansion in $\phi_k^{(S_1)}$, we can recover modes $\hat{f}_k^{\phi,(S_2)}$ of an expansion in $\phi_k^{(S_2)}$ using the following procedure:

1. Compute the nodal evaluations $f(x_j)$ from $\hat{f}_k^{\phi,(S_1)}$. $O(N \log N)$
2. Compute the modal coefficients $\hat{f}_k^{\phi,(S_2)}$ from $f(x_j)$. $O(N \log N)$

It should be noted that if $S_2 > S_1$ then the solution to the connection problem does not terminate finitely (i.e. $\phi_k^{(S_1)}$ requires infinitely modes in $\phi_k^{(S_2)}$ for an exact representation). On the other hand, if $S_2 < S_1$, then the top $(S_1 - S_2)$ modes will be aliased due to aliasing error, but otherwise the connection is accurate. Both steps above require performing the algorithm in Section 6.7.1. Thus we require two FFTs and two connection problems.

Remark 4 As in the previous section, the pointwise multiplication substep in each nodal-modal step can be implemented instead in a Galerkin fashion by noting the sparse Fourier representation of the phase-shifted square root, see (6.7). If this is done, then one can actually avoid using the FFT at all. One must demote the $\hat{f}_k^{\phi,(S_1)}$ modes to $\hat{f}_k^{\phi,(1)}$ modes, perform the Galerkin multiplication by the necessary $(x-i)^{S_2-S_1}$ factor, then perform the connection again to promote these modes to $\hat{f}_k^{\phi,(S_2)}$ modes. Thus, the entire operation can be done in asymptotically $O(N)$ time.

6.8 Derivatives and the Stiffness Matrix

This section concerns the computation of derivatives of the generalized Wiener basis functions. For computing derivatives, one can again explicitly compute the derivatives of (a) the Fourier functions and (b) the weight $\sqrt[*]{w_x^{(s,0)}}$ and then use the product rule and chain rule to compute the total derivative. To be explicit:

$$\frac{\mathrm{d}}{\mathrm{d}x}\sqrt[*]{w_x^{(s,0)}} \;=\; \frac{-s2^{(s/2)}}{(x-i)^{s+1}},$$

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\Psi_k^{(\gamma)}(\theta) \;=\; \begin{cases} 0, & k=0 \\ \frac{1}{2}\Big[-\sin(\theta)\sqrt{|k|(|k|+\gamma)}\tilde{P}_{|k|-1}^{(1/2,\gamma+1/2)}(\cos\theta) + \\ \quad i\,\mathrm{sgn}(k)\cos(\theta)\,\tilde{P}_{|k|-1}^{(1/2,\gamma+1/2)}(\cos\theta) - \\ \quad i\,\mathrm{sgn}(k)\sin^2(\theta)\sqrt{(|k|-1)(|k|+\gamma+1)}\tilde{P}_{|k|-2}^{(3/2,\gamma+3/2)}(\cos\theta)\Big], & k\neq 0 \end{cases} \tag{6.8}$$

These can be combined to give

$$\begin{aligned} \tfrac{\mathrm{d}}{\mathrm{d}x}\phi_k^{(s)} &= \tfrac{\mathrm{d}}{\mathrm{d}x}\left(\sqrt[*]{w_x^{(s,0)}}\Psi_k^{(s-1)}(\theta(x))\right) \\ &= \tfrac{-s2^{(s/2)}}{(x-i)^{s+1}}\Phi_k^{(s)}(x) + \sqrt[*]{w_x^{(s,0)}}\tfrac{2}{x^2+1}\tfrac{\mathrm{d}}{\mathrm{d}\theta}\Psi_k^{(s-1)}(\theta(x)). \end{aligned} \tag{6.9}$$

Thus, provided we can compute the easy transformation

$$\theta(x) = 2\arctan(x),$$

then one may use (6.9) along with (6.8) to compute the derivative of $\phi_k^{(s)}$. The cost of this procedure is only slightly more expensive than evaluating the functions themselves.

In many applications to differential equations it is necessary to express the derivative of a basis function as a linear combination of basis functions. We then define entries of the stiffness matrix as

$$S_{k,l}^{\phi} = \left\langle \phi_k^{(s)}, \frac{\mathrm{d}}{\mathrm{d}x}\phi_l^{(s)} \right\rangle.$$

The following result can be proven:

Theorem 6.1 *Let S^{ϕ} denote the $N \times N$ stiffness matrix for the weighted Wiener rational functions $\phi_k^{(s)}$. S^{ϕ} satisfies the following properties for any $s > \frac{1}{2}$:*

1. *S^{ϕ} is skew-Hermitian, i.e. $S_{k,l}^{\phi} = -\overline{S_{l,k}^{\phi}}$*
2. *S^{ϕ} is sparse with entries only on the super-, sub-, and main sinister and dexter diagonals: define*

$$k^{\vee} := \mathrm{sgn}(k)\left(|k|-1\right), \qquad k^{\wedge} := \mathrm{sgn}(k)(|k|+1).$$

Then

$$\frac{\mathrm{d}\phi_k^{(s)}(x)}{\mathrm{d}x} = \sum_{l\in\{\pm k^\vee,\pm k,\pm k^\wedge\}} \tau_{k,l}^{(s)}\phi_l^{(s)}(x),$$

for some k-dependent purely imaginary constants $\tau_{k,l}^{(s)}$. In other words,

$$S_{k,l}^{\phi} = 0, \qquad l \notin \{\pm k^\vee, \pm k, \pm k^\wedge\}.$$

3. The maximum eigenvalue (in magnitude) of S^ϕ satisfies

$$\max|\lambda(S^\phi)| \leq N + 5s.$$

The proof of Theorem 6.1 is quite tedious, so we only mention the main points. The details are given in Appendix D.

Proof Property 1 can easily be deduced utilizing by using integration by parts and noting that the functions $\phi_k^{(s)}(x)$ decay to zero as $|x| \to \infty$.

Property 2 is a highly nontrivial result that is provable using several properties of Jacobi polynomials. Most of the calculations are straightforward once a list of Jacobi polynomial properties has been compiled. However, there are some difficulties whose resolutions rely on a couple of fortuitous properties: first, that $\frac{\mathrm{d}x(\theta)}{\mathrm{d}\theta}(\theta) = 1 + \cos\theta$, i.e. that the map we have chosen to take $\theta \to x$ has a Jacobian with a particular form. Second, that

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\left[(\sin\theta)\,\tilde{P}_n^{(\alpha+1,\beta+1)}(\cos\theta)\right]$$

is a sparse combination of $\tilde{P}_n^{(\alpha,\beta+1)}(\cos\theta)$, which is not a trivial result; in fact, this is exactly the result of Lemma 5.1. Our less than ideal method to show it uses brute-force calculation with the compiled list of Jacobi polynomial properties.

Property 3 can be derived from the second property. The key ingredient is Gerschgorin's Theorem. Using the explicit entries for the constants $\tau_{k,l}^{(s)}$ given in Theorem D.1 we can show that for each k satisfying $|k| \geq 2$ the following crude bounds hold:

$$\begin{aligned} |\tau_{k,k}| &\leq n + 2s, \\ |\tau_{k,-k}| + |\tau_{k,k^\vee}| + |\tau_{k,-k^\vee}| + |\tau_{k,k^\wedge}| + |\tau_{k,-k^\wedge}| &\leq n + 3s + 2, \end{aligned}$$

where $n := |k| - 1$. Gerschgorin's Theorem can now be used to define a region in the complex plane in which all the eigenvalues lie. By the above properties, this region has distance from the origin at most $2n + 5s + 2$. Once we consider the necessary relationship between n, k, and N, the result is proven. (It is interesting, but not necessary, to note that the eigenvalues all lie on the imaginary axis due to the skew-Hermitian property of S.) □

Remark 5 While the $O(N)$ maximum eigenvalue does depend on s, the proportionality factor is empirically around 2, not 5 as given in the theorem. See Table 6.2.

The sparsity pattern we have derived for the derivatives of these functions (property 2 of the above theorem) is illustrated in Figure 6.6. In addition, numerical values for the maximum eigenvalues of the stiffness matrix (property 3) are given in Table 6.2. The sparsity of the stiffness matrix is important for fast computations of derivatives for spectral methods for solving PDEs, and the $O(N)$ maximum eigenvalue of the stiffness matrix indicates that we can take a relatively large timesteps for time-dependent problems. Finally, the skew-symmetry of the stiffness matrix easily leads to energy conservation for the Galerkin discretization of hyperbolic conservation

laws.

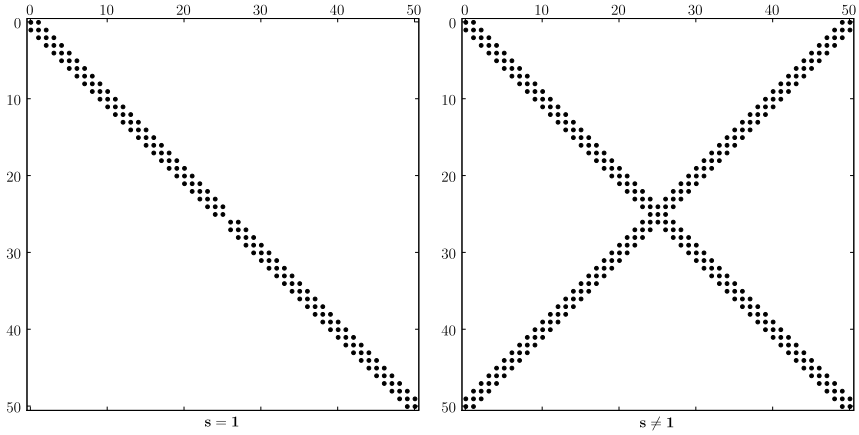

Figure 6.6: Sparsity plots for stiffness matrices of the weighted Wiener rational functions $\phi_k^{(s)}$. The sparsity patterns are representative of property 2 in Theorem 6.1 for $s = 1$ (left) and all $s \neq 1$ (right). The $s = 1$ sparsity pattern has been derived previously [27], and the expressions for the $\tau_{k,l}$ in Appendix D with $s = 1$ reduce to the pattern above.

s\N	11	50	101	250	501
0.6	7.31	43.76	91.50	237.60	483.75
1.0	7.99	44.51	92.28	238.39	484.54
6.0	15.96	53.75	101.81	248.14	494.40
$\pi^2 \approx 9.87$	21.72	60.67	109.05	255.63	501.99
15.5	29.73	70.45	119.40	266.44	512.99

Table 6.2: Maximum eigenvalue of the $N \times N$ stiffness matrix S for the Wiener rational functions $\phi_k^{(s)}$. The results adhere to the asymptotic bound given in property 3 of Theorem 6.1.

We also note that the unweighted functions $\Phi_k^{(s)}(x)$ also have a sparsity result for derivatives. In fact, it is one of the key ingredients in proving Theorem 6.1. We have not explicitly shown it here because we believe the result for the weighted functions $\phi_k^{(s)}(x)$ to be more useful; however, the unweighted function result can be found in Appendix D.

Chapter Seven

Other Approximation Methods

We have devoted the previous chapter to deriving properties of a novel basis set for approximation of a function over an infinite interval. Other classical approximation techniques exist, which we delineate in this section. Numerical comparisons between these methods and Wiener expansion methods will be performed in Section 8.

7.1 Hermite Expansions

The Hermite polynomial (e.g. [60]), denoted $H_n : \mathbb{R} \to \mathbb{R}$, is the n^{th} order polynomial satisfying the orthogonality condition

$$\int_{\mathbb{R}} H_n H_m e^{-x^2} \mathrm{d}x = \sqrt{\pi} 2^n n! \delta_{m,n}, \qquad m, n \in \mathbb{N}_0.$$

Orthogonality under the weight function $w_H = e^{-x^2}$ allows us to define approximations as partial sums of weighted Hermite polynomials that converge in that norm. However, since the Hermite polynomials are unbounded at infinity, the pointwise quality of the approximant is not very good [36].

Although the above polynomials are the standard type, there are generalized Hermite polynomials for which there is a class parameter $\mu \in \left(-\frac{1}{2}, \infty\right)$, similar to the class parameters $(\alpha, \beta) \in (-1, \infty)^2$ for the Jacobi polynomials. We denote these polynomials as $H_n^{(\mu)}(x)$. The weight function for this class of polynomials is

$$w_H^{(\mu)} = |x|^{2\mu} e^{-x^2}.$$

We define the square root of the weight function as

$$\sqrt[*]{w_H^{(\mu)}} = x^{\mu} e^{-x^2/2}.$$

As with the Wiener basis functions, we can shift the weight onto the basis functions and define the Hermite functions as

$$h_n^{(\mu)}(x) := \sqrt[*]{w_H^{(\mu)}} H_n^{(\mu)}(x). \tag{7.1}$$

The $h_n^{(\mu)}$ are orthogonal in the unweighted L^2 inner product, and decay exponentially at $x = \pm\infty$ due to the weight function. We put a tilde $\tilde{()}$ on these functions to denote their L^2-normalized versions. Both of these basis sets satisfy orthogonality relations

$$\begin{aligned} \int_{\mathbb{R}} \tilde{H}_m^{(\mu)} \tilde{H}_n^{(\mu)} w_H^{(\mu)} \mathrm{d}x &= \delta_{m,n} \\ \int_{\mathbb{R}} \tilde{h}_m^{(\mu)} \tilde{h}_n^{(\mu)} \mathrm{d}x &= \delta_{m,n}. \end{aligned}$$

The Hermite polynomials are complete in $L^2\left(\mathbb{R}, \mathbb{R}; w_H^{(\mu)}\right)$, and the Hermite functions with $\mu = 0$ are complete in $L^2(\mathbb{R}, \mathbb{R})$. We shall only use the standard Hermite polynomials/functions with $\mu = 0$.

The Hermite functions are used extensively in simulations when a function decays exponentially. With these types of rapidly decaying functions, they usually exhibit exponential convergence. The Hermite polynomials are not used very often for approximation due to the fact that they grow wildly at infinity. It is common for many functions to show exponential convergence in the $L^2_{w_H^{(\mu)}}$ norm when expanded in Hermite polynomials. However, it is very often the case that such an approximation is suboptimal in the L^∞ norm, or even in the 'eyeball' norm when x is a moderate distance from 0. In such cases, one may use the Hermite functions, but this is problematic if the function to be approximated does not decay exponentially.

We detail some properties of Hermite polynomials/functions in Appendix A.2.

7.2 Whittaker Cardinal Functions

The Sinc (or 'Whittaker Cardinal', or 'cardinal sine') functions [68] form a basis for the infinite interval that are orthogonal under the equidistant interpolation projector (as well as the L^2 norm). In contrast to the Hermite/Laguerre approach, these functions can be defined in terms of the grid points, and thus form an interpolation of the grid points. By *a priori* defining a grid-spacing parameter h, the sinc functions can be defined as

$$C_{j;h}(x) = \operatorname{sinc}\left(\frac{x - jh}{h}\right).$$

The sinc function is defined as

$$\operatorname{sinc}(x) = \frac{\sin(\pi x)}{\pi x}.$$

Then for N equispaced grid points

$$x_j = jh,$$

for j in some subset of $\mathbb{Z}$ with cardinality N, the sinc approximation given N nodal values $f(x_j) = f_j$ is

$$f_N = \sum_{j=1}^{N} f_j C_{j;h}(x).$$

The cardinal functions satisfy $C_{k;h}(x_j) = \delta_{k,j}$. Naturally the scaling h can be modified to linearly expand or contract the interpolation grid (and also the cardinal functions). There have been recent attempts to show some quasi-optimal interpolation properties of the functions [59]; however, a relatively sobering result is that exponential decay is not sufficient to guarantee convergence of the expansion, although simultaneous modification of h and N can ameliorate this problem. One major appeal of these

functions comes from the fact that the Whittaker Cardinal functions are bandlimited in the Fourier-frequency domain, and from the convenient property that they are orthogonal with respect to the interpolation operator on equidistant nodes x_k; thus obtaining modal coefficients from equispaced nodal evaluations is trivial. In addition, the cardinal functions are L^2-orthogonal:

$$\langle C_{j;h}, C_{k;h} \rangle = h\delta_{j,k}.$$

While many basis sets have cardinal interpolation (i.e. Lagrange interpolating) functions, this set is special in both the simplicity of the collocation points and in the concise elegance of the basis functions. This has led to numerous investigations and applications of these functions [50].

7.3 Mapped Chebyshev Functions

The main idea for our generalization of Wiener's original rational basis is using a 'well-behaved' mapping to transform functions on a finite interval to those on an infinite interval. This basic idea is classical [36]. Indeed one of the more popular mappings that has gained momentum in the literature are the so-called 'mapped Chebyshev' functions/polynomials.

In order to further generalize the mapped Chebyshev functions, we will briefly restate their derivation. We begin with the Jacobi polynomials $P_n^{(\alpha,\beta)}(r)$ on $r \in [-1, 1]$. Mapping via $r = \cos\theta$ to $\theta \in [0, \pi]$ yields trigonometric polynomials. We now 'stretch' the domain to $\Theta \in [-\pi, \pi]$ via the mapping $\Theta = 2\theta - \pi$. Finally, we utilize the usual linear fractional map $e^{i\Theta} = \frac{i-x}{i+x}$ to obtain functions on the real line $x \in \mathbb{R}$.

For all $s, t > \frac{1}{2}$, this results in the functions $\mathrm{PB}_n^{(s,t)}(x)$, defined as

$$\mathrm{PB}_n^{(s,t)}(x) = \tilde{P}_n^{((2s-3)/2,(2t-3)/2)}\left(\frac{x}{\sqrt{1+x^2}}\right),$$

are orthonormal on the real line under the weight

$$w_{\mathrm{PB}}^{(s,t)} = \left[1 - \frac{x}{\sqrt{1+x^2}}\right]^{(2s-3)/2}\left[1 + \frac{x}{\sqrt{1+x^2}}\right]^{(2t-3)/2},$$

and the weighted functions

$$\mathrm{pb}_n^{(s,t)} := \sqrt{w_{\mathrm{PB}}^{(s,t)}}\,\mathrm{PB}_n^{(s,t)},$$

are orthonormal under the unweighted inner product. When $s = t = 1$, the functions $\mathrm{PB}_n^{(s,t)}$ coincide with the mapped Cheyshev polynomial $\mathrm{TB}_n(x)$ introduced in [10] and subsequently developed in [12] and [14], although the original idea of applying spectral expansions over finite intervals to solving problems over infinite intervals seems to come from [39]. In any case, the mapped Jacobi functions $\mathrm{pb}_n^{(s,t)}$ decay like $\frac{1}{|x|^s}$ for $x \to -\infty$ and $\frac{1}{|x|^t}$ for $x \to +\infty$. The advantage of these functions is that the decay can be different at $\pm\infty$. Also, others have already explored some convergence theory in function spaces [7] and applications to differential equations [71] for the Chebyshev case $s = t = 1$.

For the application of solving differential equations we might consider the maximum eigenvalue of the stiffness matrix for the weighted mapped Jacobi polynomials $\mathrm{pb}_n^{(s,t)}$; a similar consideration was given in Section 6.8 for the generalized Wiener rational basis. For simplicity, we take $s = t$ and tabulate results in Table 7.1. The maximum eigenvalue of the stiffness matrix for this basis set is on the same order as that for the generalized Wiener rational functions, given in Table 6.2. However, the stiffness

matrix for the functions $\mathrm{pb}_n^{(s,t)}$ is a dense matrix; there is no sparsity result, so it is more expensive computationally to take derivatives using this basis set.

s/N	11	50	101	250	501
0.6	6.20	41.56	90.22	235.32	482.44
1.0	6.56	41.95	90.61	235.71	482.83
6.0	10.17	46.48	95.34	240.57	487.75
$\pi^2 \approx$9.87	12.35	49.72	98.85	244.27	491.52
15.5	15.01	54.12	103.75	249.56	496.96

Table 7.1: Maximum eigenvalues (magnitude) of the Galerkin stiffness matrix for the mapped Jacobi polynomials $\mathrm{pb}_n^{(s,s)}(x)$. Compare with Table 6.2.

Note that because all of these mapped types of polynomials and the generalized Wiener basis we have presented ultimately stem from Jacobi polynomials and mappings of similar character, all these basis sets are related in some fashion. To relate the mapped Jacobi functions to the generalized Wiener rational functions, we have

$$\mathrm{PB}_n^{(s,s)}(x) \propto \mathrm{Re}\left\{\Phi_n^{(s)}\left(\frac{x+\sqrt{x^2+1}-1}{x-\sqrt{x^2+1}+1}\right)\right\}$$

In Table 7.2 we relate the unweighted functions to the generalized Wiener rational basis, up to multiplicative constants.

There has been extensive use of 'mapped Chebyshev polynomials' [15], [11], [14]. These are a specific case of the basis functions we have developed in this section. Although we seek to compare expansions in these functions against generalized Wiener expansion methods, we shall not compare against the 'Higgins' [45] or 'Christov' [27] functions given in Table 7.2 since these are special cases of the functions we've already developed.

Previous function	Name/classification	Interval	Reference	Relation
TB_n	Cheyshev rational functions (1st)	$\mathbb{R}$	[15], [10], [15]	$\widetilde{\mathrm{PB}}_n^{(1,1)}$
SB_n/UB_n	Chebyshev rational functions (2nd)	$\mathbb{R}$	[15], [15], [14]	$\widetilde{\mathrm{PB}}_n^{(2,2)}$
C_n/CC_n	Christov functions (even)	$\mathbb{R}$, $[0,\infty)$	[27], [15]	$\mathrm{Im}\left\{\tilde{\phi}_n^{(1,0)}\right\}$
S_n/SC_n	Christov functions (odd)	$\mathbb{R}$, $[0,\infty)$	[27], [15]	$\mathrm{Re}\left\{\tilde{\phi}_n^{(1,0)}\right\}$
CH_n	Higgins functions (even)	$\mathbb{R}$, $[0,\infty)$	[15]	$\mathrm{Re}\left\{\tilde{\Phi}_n^{(1,0)}\right\}$
SH_n	Higgins functions (odd)	$\mathbb{R}$, $[0,\infty)$	[15]	$\mathrm{Im}\left\{\tilde{\Phi}_n^{(1,0)}\right\}$
ρ_k	(Complex) Higgins functions	$\mathbb{R}$	[45], [27]	$\tilde{\Phi}_k^{(1,0)}$
σ_k	(Complex) Wiener rational functions	$\mathbb{R}$	[69], [27]	$\tilde{\phi}_k^{(1,0)}$
TL_n	Half-infinite Chebyshev rational functions	$[0,\infty)$	[11]	——

Table 7.2: Relationship between orthogonal functions in previous work and the current bases presented.

7.4 Laguerre Functions

The generalized Laguerre polynomials/functions, like the Hermite case, satisfy a singular Sturm-Liouville eigenvalue problem defined over the semi-infinte strip $x \in [0,\infty)$. The class parameter is $\alpha \in (-1,\infty)$. The weight function is $w_L^{(\alpha)} = x^\alpha e^{-x}$, and we define the square root as

$$\sqrt[*]{w_L^{(\alpha)}} = x^{(\alpha/2)} e^{-x/2}.$$

We denote the Laguerre polynomials as $L_n^{(\alpha)}(x)$, and the Laguerre functions are defined as

$$l_n^{(\alpha)}(x) = \sqrt[*]{w_L^{(\alpha)}} L_n^{(\alpha)}(x). \tag{7.2}$$

These basis sets satisfy the orthogonality conditions

$$\begin{aligned} \int_{\mathbb{R}} \tilde{L}_m^{(\alpha)} \tilde{L}_n^{(\alpha)} w_L^{(\alpha)} \mathrm{d}x &= \delta_{m,n} \\ \\ \int_{\mathbb{R}} \tilde{l}_m^{(\alpha)} \tilde{l}_n^{(\alpha)} \mathrm{d}x &= \delta_{m,n}. \end{aligned}$$

The Laguerre polynomials are complete in $L^2\left(\mathbb{R}, \mathbb{R}; w_L^{(\mu)}\right)$, and the Laguerre functions are complete in $L^2(\mathbb{R}, \mathbb{R})$.

The Laguerre polynomials/functions have similar characteristics to the Hermite polynomials/functions when used as an expanding basis. (In fact, one can write Hermite polynomials in terms of Laguerre polynomials.) We compile properties of the Laguerre basis in Appendix A.3.

Chapter Eight

Numerical Examples

8.1 Function Approximation

We ask the following question in this section for some function $f : \mathbb{R} \to \mathbb{R}$. Given a basis set $\{\rho_n(x)\}_{n\in\mathbb{N}}$ and a natural number N, how well does

$$\mathcal{P}_N^\rho f(x) = \sum_{n=0}^{N-1} \hat{f}_n \rho_n(x),$$

approximate $f(x)$? (Here the modal coefficients $\hat{f}_n$ are defined using the L^2 projection operator $\mathcal{P}_N^\rho$.) In other words, is the basis set ρ_n is a good basis set from a function-analytic point of view.

For all of our tests in this section, we will use the generalized Wiener basis set $\left(\phi_k^{(s)}\right)$, the Hermite function basis set (h_n), the Whittaker cardinal (sinc) functions $(C_k(x))$, and the mapped Jacobi functions $\left(\mathrm{pb}_n^{(s,t)}\right)$. Unless specified otherwise, the parameter $s = t = 1$. All these functions have L^2 orthogonality and thus we can measure L^2 norms in the same way.

There remains the issue of how to choose an affine scaling parameter for each basis set to keep the comparisons as unbiased as possible. We scale each of the basis function sets in the following way: we associate with each basis set a 'canonical' interpolation nodal set for each N. For the Hermite functions h_n it is the Gauss quadrature grid; the mapped Jacobi functions use the Jacobi-Gauss quadrature grid mapped to the real line; the sinc functions have their natural equispaced interpolation points; the Wiener rational functions have the quadrature nodes $x\left(\theta_n^{(s-1)}\right)$ as given in section 6.5. All of these canonical interpolation sets serve as a measure of the resolution that N basis functions can accomodate. In an attempt to put all basis sets on equal footing, we fix N and define a scaling parameter L on the real line; next we affinely

map each basis set so that half of the canonical interpolation nodes lie inside the interval prescribed by $|x| \leq L$. Roughly speaking, we scale the basis functions so that about 'half' the resolution for N modes is contained in $[-L, L]$. We change L for any given function f to accomodate any special structure f may exhibit, but L is uniform with respect to the choice of basis set.

The efficacy of an expansion depends heavily on the function f we take. We recall from Section 2.3 that there is very strong theory coupling Sobolev-type regularity of f with approximation quality on the finite interval. The generalized Wiener functions are maps of Fourier series or orthogonal polynomials, so we expect a similar phenomenon on the infinite interval. Therefore, we first consider dependence on the regularity of f.

8.1.1 Function regularity

We consider the following five functions:

$$\begin{aligned} f_\infty(x) &= e^{-(x-0.5)^2} \\ f_0(x) &= \operatorname{sgn}(x-0.5)e^{-x^2} \\ f_1(x) &= |x-0.5|e^{-x^2} \\ f_2(x) &= \operatorname{sgn}(x-0.5)x^2e^{-x^2} \\ f_3(x) &= |x-0.5|^3e^{-x^2}. \end{aligned}$$

Plots of these functions are given Figure 8.1. The subscript on each of these functions denotes how many L^2 derivatives each function has. We will use these test cases to determine the effectiveness of the basis functions and their interpolation sets.

For all the calculations, we compute the modes via a 10^4-point $\phi_k^{(1)}(x)$ quadrature rule. This is also how we calculate L^2 and L^∞ errors. These errors are given in Tables 8.1.1-8.1.1. The results of these tables mimic exactly what we expect to see from knowledge of convergence estimates versus regularity on the finite interval. For f_i, $i = 0, 1, 2, 3$ we see consistent $(i + 1/2)^{\text{th}}$-order convergence in the L^2 norm and i^{th}-order convergence in the L^∞ norm for every approximation. Note that f_i has exactly $i + 1/2$ distributional derivatives with L^2 membership, thus the extra $1/2$ order of convergence in the L^2 norm. It is interesting to note that the Hermite and Sinc approximations of f_∞ stagnate around $N = 40$. This could be the product of many factors; among them are numerical precision errors and ill-chosen refinement procedure (i.e. choice of the scaling parameter).

For a given number of basis functions, the Wiener basis seems to have L^2 error that is approximately half of the mapped Chebyshev error. In the context of solutions to differential equations, the Wiener basis set has a sparse stiffness matrix with provably $O(N)$ eigenvalues; the mapped Jacobi function basis set has a full stiffness matrix (although the second derivative Galerkin matrix is sparse). Both of the basis sets are amenable to FFT usage.

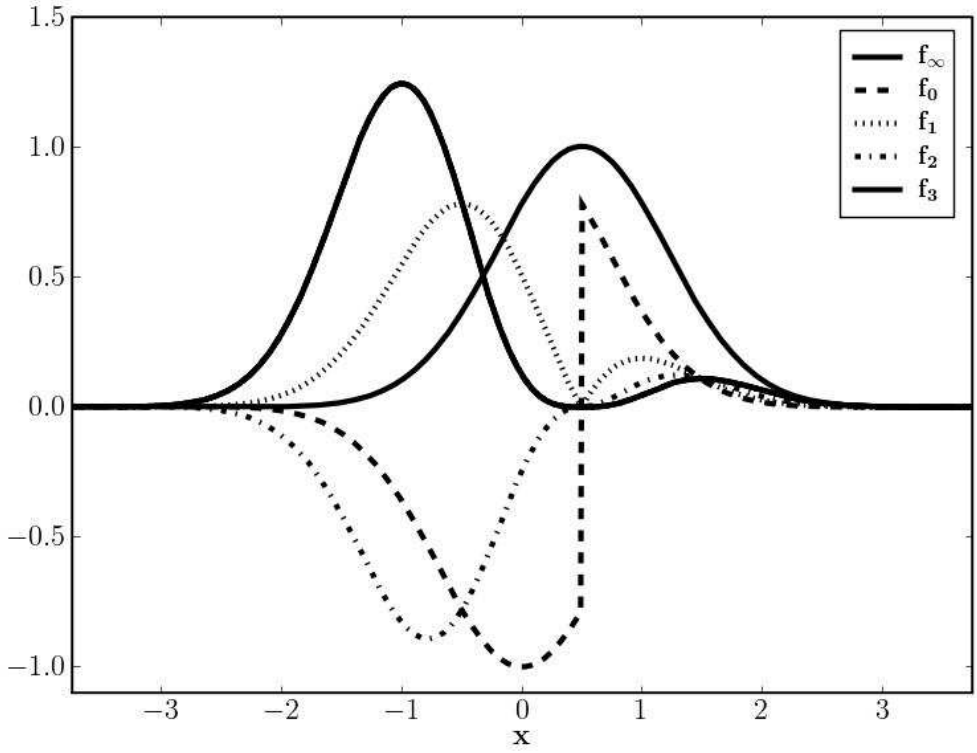

Figure 8.1: Plots of the functions f_0, f_1, f_2, f_3, and f_∞ on $x \in [-3, 3]$. The subscript on the function indices denotes their integer-valued differentiable regularity.

f_∞

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	2.455e-04	7.221e-07	8.409	5.153e-11	13.77	3.281e-15	13.94
$\mathrm{pb}_n^{(1,1)}$	6.178e-04	1.193e-06	9.017	1.097e-10	13.41	2.212e-15	15.60
h_n	5.446e-10	1.194e-11	5.511	1.800e-10	——	3.158e-10	——
C_k	2.476e-06	3.747e-07	2.724	6.471e-07	——	8.109e-07	——

f_0

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	3.026e-01	2.188e-01	0.4677	1.565e-01	0.4832	1.113e-01	0.4914
$\mathrm{pb}_n^{(1,1)}$	3.208e-01	2.271e-01	0.4983	1.585e-01	0.5193	1.120e-01	0.5003
h_n	3.508e-01	2.457e-01	0.5139	1.733e-01	0.5031	1.225e-01	0.5005
C_k	6.561e-01	4.369e-01	0.5865	3.010e-01	0.5377	2.098e-01	0.5210

f_1

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	2.113e-02	7.867e-03	1.425	2.870e-03	1.455	1.032e-03	1.475
$\mathrm{pb}_n^{(1,1)}$	2.605e-02	8.477e-03	1.619	3.036e-03	1.482	1.062e-03	1.516
h_n	3.490e-02	1.140e-02	1.614	3.933e-03	1.535	1.378e-03	1.513
C_k	5.853e-02	1.787e-02	1.711	6.039e-03	1.565	2.097e-03	1.526

f_2

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	4.136e-03	7.636e-04	2.437	1.414e-04	2.433	2.568e-05	2.461
$\mathrm{pb}_n^{(1,1)}$	5.106e-03	9.218e-04	2.470	1.504e-04	2.615	2.649e-05	2.506
h_n	9.414e-03	1.383e-03	2.767	2.361e-04	2.550	4.145e-05	2.510
C_k	2.114e-02	2.470e-03	3.098	3.888e-04	2.667	6.563e-05	2.567

f_3

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	1.512e-03	1.217e-04	3.635	1.142e-05	3.414	1.046e-06	3.448
$\mathrm{pb}_n^{(1,1)}$	1.935e-03	1.458e-04	3.730	1.302e-05	3.485	1.117e-06	3.543
h_n	4.769e-03	2.964e-04	4.008	2.396e-05	3.629	2.055e-06	3.543
C_k	1.424e-02	5.345e-04	4.735	3.997e-05	3.741	3.325e-06	3.588

Table 8.1: L^2 errors of the functions f_i for various basis projections. $\phi_k^{(1)}$: Wiener basis set; $\mathrm{pb}_n^{(1,1)}$: mapped Jacobi function basis; h_n: Hermite functions; C_k: Sinc basis.

f_∞

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	3.561e-04	8.025e-07	8.793	4.522e-11	14.12	1.333e-14	11.73
$\mathrm{pb}_n^{(1,1)}$	3.736e-04	7.519e-07	8.956	5.803e-11	13.66	5.440e-15	13.38
h_n	3.007e-10	1.196e-11	4.652	2.584e-10	——	5.922e-10	——
C_k	9.922e-07	8.020e-07	0.3071	1.835e-06	——	2.768e-06	——

f_0

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	8.392e-01	8.058e-01	0.05860	7.874e-01	——	7.737e-01	——
$\mathrm{pb}_n^{(1,1)}$	8.256e-01	8.050e-01	0.03655	7.876e-01	——	7.735e-01	——
h_n	8.590e-01	8.144e-01	0.07683	7.919e-01	——	7.771e-01	——
C_k	1.397e+00	1.415e+00	——	1.421e+00	——	1.419e+00	——

f_1

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	6.550e-02	3.226e-02	1.022	1.602e-02	1.010	7.949e-03	1.011
$\mathrm{pb}_n^{(1,1)}$	6.709e-02	3.217e-02	1.060	1.622e-02	0.9883	7.998e-03	1.020
h_n	8.083e-02	3.911e-02	1.047	1.927e-02	1.021	9.534e-03	1.016
C_k	8.731e-02	3.583e-02	1.285	1.672e-02	1.100	8.105e-03	1.045

f_2

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	7.546e-03	1.709e-03	2.143	4.119e-04	2.053	1.016e-04	2.020
$\mathrm{pb}_n^{(1,1)}$	7.107e-03	1.771e-03	2.005	4.115e-04	2.106	1.015e-04	2.020
h_n	1.254e-02	2.528e-03	2.310	5.951e-04	2.087	1.458e-04	2.030
C_k	2.609e-02	5.619e-03	2.215	1.306e-03	2.105	3.164e-04	2.045

f_3

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	2.767e-03	2.839e-04	3.285	3.439e-05	3.045	4.262e-06	3.012
$\mathrm{pb}_n^{(1,1)}$	2.144e-03	2.802e-04	2.936	3.560e-05	2.977	4.340e-06	3.036
h_n	5.069e-03	5.098e-04	3.314	5.985e-05	3.091	7.317e-06	3.032
C_k	1.712e-02	9.332e-04	4.198	9.207e-05	3.341	1.038e-05	3.149

Table 8.2: L^∞ errors of the functions f_i for various basis projections. These errors are computed as the maximum pointwise error on a 10^4-point $s = 1$ Wiener grid. $\phi_k^{(1)}$: Wiener basis set; $\mathrm{pb}_n^{(1,1)}$: mapped Jacobi function basis; h_n: Hermite functions; C_k: Sinc basis.

8.1.2 Function decay

The purpose of this section is to explore the effect that the decay of the function has on convergence rates for expansions. The functions we consider here are

$$
\begin{aligned}
g_\infty &= e^{-(x-0.5)^2} \\
g_0 &= \frac{1}{((x-0.5)^2+1)^{0.3}} \\
g_1 &= \frac{1}{(x-0.5)^2+1} + e^{-x^2} \\
g_2 &= \frac{\arctan(x)}{x^2+1} \\
g_3 &= \frac{1}{((x-0.5)^2+1)^3}
\end{aligned}
$$

The L^2 and L^∞ errors are shown in Tables 8.3 and 8.4, respectively. As expected, the convergence rates for the exponentially decaying Gaussian g_∞ are very good for most of the expansions. The Wiener and Mapped Jacobi expansions converge quickly; the Hermite and Sinc convergences are extremely quick but then stagnate, presumably due to precision problems. The function g_0 is in L^2, but decays slower than $1/x$. We do not really expect any of the approximations to produce very good results, and we see exactly this, although the $\text{pb}^{(s,t)}$ approximation seems to do the best in terms of the magnitude of the error.

The function g_1 is a bit of an anomaly: its behavior at infinity is dictated by the rational decay rate $1/x^2$ but we add a Gaussian to the function to examine the behavior. Because the function does not decay exponentially, the Hermite function and Sinc expansions do a relatively poor job of resolving the function. Even the mapped functions $\text{pb}^{(s,t)}$ seem to have a bit of a problem. However, the Wiener basis set successfully captures the features of the function and is able to converge down to machine precision for 160 modes. The function g_2 also decays like $1/x^2$

but is multiplied by an arctan function. Both the mapped Jacobi functions $\mathrm{pb}^{(s,t)}$ and the Wiener approximation converge more quickly than the Sinc and Hermite approximations, albeit only of order 1.5.

Finally, function g_3 decays like $1/x^6$; even with this strong decay, the Hermite and Sinc approximations are inferior to the mapped Jacobi and Wiener sets, which both approximate the function well. The functions g_i that we have defined obviously do not cover the entire gamut of functions one encounters in problems of scientific computing interest, but we have made an attempt to present functions of fairly general and diverse characteristics and shown that the Wiener basis functions $\phi_k^{(s)}$ and the mapped Jacobi functions $\mathrm{pb}_n^{(s,t)}$ are superior in these instances to the Hermite and Sinc approximations. We can postulate that this behavior is very similar for any general function that does not decay exponentially. Although we have presented a case where the Wiener expansion is superior to the mapped Jacobi expansion (g_1), it is difficult to claim a clear advantage for the basis set in this comparison.

g_∞

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	2.455e-04	7.221e-07	8.409	5.153e-11	13.77	3.281e-15	13.94
$\mathrm{pb}_n^{(1,1)}$	6.178e-04	1.193e-06	9.017	1.097e-10	13.41	2.212e-15	15.60
h_n	5.446e-10	1.194e-11	5.511	1.800e-10	——	3.158e-10	——
C_k	2.476e-06	3.747e-07	2.724	6.471e-07	——	8.109e-07	——

g_0

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	1.750e+00	1.598e+00	0.1314	1.448e+00	0.1426	1.300e+00	0.1554
$\mathrm{pb}_n^{(1,1)}$	1.100e+00	9.586e-01	0.1987	8.243e-01	0.2177	6.943e-01	0.2476
h_n	2.418e+00	2.418e+00	——	2.418e+00	——	2.419e+00	——
C_k	2.451e+00	2.466e+00	——	2.473e+00	——	2.476e+00	——

g_1

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	1.769e-04	4.763e-08	11.86	1.538e-13	18.24	7.918e-15	4.280
$\mathrm{pb}_n^{(1,1)}$	2.137e-03	6.781e-04	1.656	2.348e-04	1.530	8.219e-05	1.515
h_n	5.270e-02	5.311e-02	——	5.316e-02	——	5.318e-02	——
C_k	6.469e-02	6.887e-02	——	7.110e-02	——	7.222e-02	——

g_2

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	2.521e-03	9.535e-04	1.403	3.488e-04	1.451	1.255e-04	1.474
$\mathrm{pb}_n^{(1,1)}$	3.059e-03	1.026e-03	1.577	3.620e-04	1.503	1.279e-04	1.501
h_n	7.411e-02	7.576e-02	——	7.646e-02	——	7.682e-02	——
C_k	9.284e-02	9.837e-02	——	1.013e-01	——	1.028e-01	——

g_3

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	2.832e-03	2.918e-06	9.923	1.064e-12	21.39	2.451e-15	8.763
$\mathrm{pb}_n^{(1,1)}$	5.903e-03	7.856e-06	9.553	9.279e-12	19.69	2.057e-13	5.495
h_n	2.020e-02	1.036e-04	7.607	2.109e-05	2.296	2.120e-05	——
C_k	4.666e-02	2.900e-04	7.330	6.609e-05	2.134	6.907e-05	——

Table 8.3: L^2 errors of the functions g_i for various basis projections. $\phi_k^{(1)}$: Wiener basis set; $\mathrm{pb}_n^{(1,1)}$: mapped Jacobi function basis; h_n: Hermite functions; C_k: Sinc basis.

g_∞

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	3.561e-04	8.025e-07	8.793	4.522e-11	14.12	1.333e-14	11.73
$\mathrm{pb}_n^{(1,1)}$	3.736e-04	7.519e-07	8.956	5.803e-11	13.66	5.440e-15	13.38
h_n	3.007e-10	1.196e-11	4.652	2.584e-10	——	5.922e-10	——
C_k	9.922e-07	8.020e-07	0.3071	1.835e-06	——	2.768e-06	——

g_0

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	7.410e-02	4.852e-02	0.6110	3.195e-02	0.6026	2.108e-02	0.6003
$\mathrm{pb}_n^{(1,1)}$	5.431e-02	3.474e-02	0.6447	2.234e-02	0.6365	1.432e-02	0.6423
h_n	3.146e-01	3.246e-01	——	3.320e-01	——	3.373e-01	——
C_k	4.177e-01	4.366e-01	——	4.463e-01	——	4.512e-01	——

g_1

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	4.039e-04	1.278e-07	11.63	9.339e-14	20.38	3.153e-14	1.567
$\mathrm{pb}_n^{(1,1)}$	5.752e-04	6.720e-05	3.097	1.633e-05	2.041	4.029e-06	2.019
h_n	2.276e-02	2.472e-02	——	2.618e-02	——	2.728e-02	——
C_k	4.195e-02	4.684e-02	——	5.013e-02	——	5.188e-02	——

g_2

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	4.306e-04	1.009e-04	2.094	2.506e-05	2.009	6.253e-06	2.003
$\mathrm{pb}_n^{(1,1)}$	7.042e-04	1.002e-04	2.813	2.501e-05	2.003	6.250e-06	2.000
h_n	3.143e-02	3.456e-02	——	3.682e-02	——	3.847e-02	——
C_k	5.877e-02	6.515e-02	——	6.931e-02	——	7.149e-02	——

g_3

	20	40	order	80	order	160	order
$\phi_k^{(1)}$	5.805e-03	6.356e-06	9.835	2.417e-12	21.33	9.663e-15	7.967
$\mathrm{pb}_n^{(1,1)}$	5.146e-03	6.696e-06	9.586	1.824e-12	21.81	1.998e-14	6.512
h_n	1.817e-02	8.534e-05	7.735	1.958e-05	2.124	2.151e-05	——
C_k	3.671e-02	2.262e-04	7.342	9.754e-05	1.214	1.075e-04	——

Table 8.4: L^∞ errors of the functions g_i for various basis projections. These errors are computed as the maximum pointwise error on a 10^4-point $s = 1$ Wiener grid. $\phi_k^{(1)}$: Wiener basis set; $\mathrm{pb}_n^{(1,1)}$: mapped Jacobi function basis; h_n: Hermite functions; C_k: Sinc basis.

8.1.3 Effect of the decay parameter s

We have not yet used a decay parameter other than $s = 1$. To illustrate how the parameter s affects expansions of functions, we'll expand the the functions g_i considered above using different values of the decay parameter s for the Wiener basis set $\phi_k^{(s)}$. Although we could also do the same for the mapped Jacobi functions $\mathrm{pb}_n^{(s,t)}$, the purpose of this section is to explore the properties of the former basis set, not of the latter. We have used an interpolating Vandermonde matrix method to compute these approximations; although we have chosen to use $s = 1, 2, 5, 10$, we will not employ the FFT algorithm from Section 6.7; this decision will be addressed in the closing remarks of this section.

The results of the experiment are tabulated in Tables 8.5 and 8.6. As usual we have produced these tables using a dense (10,000-point) $s = 1$ quadrature. We see that for exponentially decaying functions (g_∞), the absolute value of the error is not affected very strongly by the choice of basis expansion decay parameter s. This is an encouragement for using the generalized Wiener basis for functions that potentially decay exponentially. The tables for function g_0 show the expected (dreary) convergence rates. There seems to be some convergence, but it is hardly anything encouraging to speak of.

For the expansion in g_1, we see that the rate of decay adversely affects the expansion in $s = 5$, but leaves $s = 1, 2$ unchanged. This is commensurate with the rate of decay of g_1, which is like x^{-2}. Finally, g_3 has a relatively steep rate of decay, x^{-6}, and this allows even the $s = 10$ expansion to capture the necessary features.

The conclusion is that the rate of decay is intimately connected with the rate of convergence of these expansions. If a traditional interpolating Vandermonde approach

g_∞

s/N	20	40	order	80	order	160	order
1	2.455e-04	7.221e-07	8.409	5.153e-11	13.77	3.755e-15	13.74
2	3.477e-05	3.868e-07	6.490	2.475e-11	13.93	3.821e-15	12.66
5	9.030e-05	1.555e-07	9.182	7.185e-12	14.40	3.989e-15	10.81
10	3.848e-04	1.107e-07	11.76	2.010e-12	15.75	4.274e-15	8.878

g_0

s/N	20	40	order	80	order	160	order
1	1.750e+00	1.598e+00	0.1310	1.448e+00	0.1424	1.300e+00	0.1554
2	1.721e+00	1.582e+00	0.1211	1.440e+00	0.1361	1.296e+00	0.1518
5	2.089e+00	1.980e+00	0.07724	1.851e+00	0.09709	1.708e+00	0.1163
10	2.238e+00	2.144e+00	0.06244	2.013e+00	0.09086	1.851e+00	0.1212

g_1

	20	40	order	80	order	160	order
1	1.769e-04	4.763e-08	11.86	1.521e-13	18.26	6.339e-15	4.584
2	1.605e-04	2.830e-08	12.47	4.436e-14	19.28	4.940e-15	3.167
5	6.477e-03	4.123e-03	0.6518	1.987e-03	1.053	8.219e-04	1.273
10	1.678e-02	1.068e-02	0.6512	5.308e-03	1.009	2.039e-03	1.380

g_3

	20	40	order	80	order	160	order
1	2.832e-03	2.918e-06	9.923	1.063e-12	21.39	8.358e-15	6.991
2	2.861e-03	2.226e-06	10.33	8.238e-13	21.37	8.406e-15	6.615
5	7.578e-04	6.360e-06	6.897	1.672e-12	21.86	9.442e-15	7.468
10	6.539e-03	9.014e-06	9.503	2.793e-09	11.66	8.681e-11	5.008

Table 8.5: L^2 errors of the functions g_i for Wiener basis projections with $s = 1, 2, 5$, and 10. We have omitted results for g_2 because they do not provide any novel information.

is used, the approximation will converge, but slowly.

We stress again that the results in the table above do *not* use the FFT algorithm to compute modal coefficients. If the FFT algorithm is used, one will pay an extreme price for poorly guessing the wrong rate of decay for an unknown function.

For example, if we use the FFT algorithm to compute the modal coefficients, then we will see extreme (e.g. $O(10^{10})$) errors in the approximation of g_0 with $s = 5, 10$. To understand why this happens, recall from 6.7 that to use the FFT for these expansions, we must multiply the function nodal evaluations by $\left(\sqrt[*]{w_x^{(s,0)}}\right)^{-1}$, which grows like x^s. Applying this to function g_0, which decays like $x^{-0.6}$, we see that

g_∞

s/N	20	40	order	80	order	160	order
1	3.561e-04	8.025e-07	8.793	4.522e-11	14.12	1.166e-13	8.599
2	2.799e-04	5.747e-07	8.928	3.168e-11	14.15	1.476e-13	7.746
5	2.551e-04	3.209e-07	9.635	1.520e-11	14.37	2.799e-13	5.763
10	5.833e-04	2.752e-07	11.05	7.810e-12	15.10	5.280e-13	3.887

g_0

s/N	20	40	order	80	order	160	order
1	7.410e-02	4.852e-02	0.6110	3.195e-02	0.6027	2.108e-02	0.6002
2	6.969e-02	4.688e-02	0.5719	3.138e-02	0.5793	2.088e-02	0.5876
5	1.301e-01	1.012e-01	0.3629	7.297e-02	0.4716	5.022e-02	0.5390
10	1.869e-01	1.564e-01	0.2574	1.117e-01	0.4847	7.578e-02	0.5603

g_1

s/N	20	40	order	80	order	160	order
1	4.039e-04	1.278e-07	11.63	2.067e-13	19.24	1.747e-13	0.2424
2	6.108e-04	2.063e-07	11.53	2.892e-13	19.44	2.666e-13	0.1176
5	2.690e-03	6.828e-04	1.978	2.212e-04	1.626	6.287e-05	1.815
10	6.860e-03	2.483e-03	1.466	8.478e-04	1.550	2.338e-04	1.858

g_3

s/N	20	40	order	80	order	160	order
1	5.805e-03	6.356e-06	9.835	2.418e-12	21.33	2.141e-13	3.498
2	8.077e-03	9.571e-06	9.721	3.847e-12	21.25	2.215e-13	4.118
5	1.775e-02	3.088e-05	9.167	1.630e-11	20.85	2.841e-13	5.843
10	4.368e-02	1.463e-04	8.222	6.586e-10	17.76	1.472e-11	5.483

Table 8.6: L^∞ errors of the functions g_i for Wiener basis projections with $s = 1, 2, 5,$ and 10. We have omitted results for g_2 because they do not provide any novel information.

we generate very large nodal point evaluations; inserting these into the FFT will clearly give large coefficients. In other words, this produces an ill-conditioned algorithm for determining the modes if our guess on the function decay (s) is very wrong. Perhaps a different way of saying this is that the weighted quadrature rule $\left\{x\left(\theta_n^{(s-1)}\right), \omega_n^{(s-1)}\right\}$ produces far less aliasing error than the unweighted quadrature rule $\left\{x\left(\theta_n^{(s-1)}\right), \Omega_n^{(s-1)}\right\}$ when the function to be approximated has a far slower rate of decay than anticipated.

Using a traditional Vandermonde matrix approach will not produce such large errors. However, making small errors in judgement for the rate of decay (somewhere less than 5 powers of x) does not produce such wild errors, even for a 10,000-point quadrature rule.

The lesson to take away is that the FFT method has the advantage of reducing computation cost from $O(N^2)$ to asymptotically $O(N \log N)$ for $s \in \mathbb{N}$; however, one must make sure that the data set is relatively commensurate with the chosen value of s. (*Underestimating* s, however, does not degrade conditioning of the FFT method.)

8.2 Solving differential equations

The solution of partial differential equations is a hallmark application for the spectral expansion of functions. We first present an example on a semi-infinite interval where we use mapped Jacobi polynomials; secondly we present the solution of the Vlasov-Poisson system using the Wiener rational function basis.

8.2.1 The Semi-infinite Interval: The Wave Equation

In this section we will consider solving a differential equation on the semi-infinite interval $[0, \infty)$. The model equation for our example is

$$\tilde{u}_{tt} = c^2 \Delta \tilde{u}, \qquad (x, t) \in (\Gamma, [0, T]), \tag{8.1}$$

for $\Gamma \subset \mathbb{R}^3$ some exterior domain in $\mathbb{R}^3$. Equation (8.1) is a common problem in electromagnetic problems where we are trying to determine the electromagnetic scattering response of an object, which we surround with e.g. an unstructured finite-element grid. This grid eventually terminates, and we must somehow simulate radiation outflow. There is a vast literature regarding absorbing boundary conditions to complete such a task. These boundary conditions are often mathematically complex and computationally intensive. An alternative is to use an infinite element method, which involves surrounding the finite element grid with several elements that extend infinitely in the radial direction. To use these elements, we must employ basis functions on a semi-infinite interval. To keep the analysis simple, we assume that the scattering object together with the finite element grid forms a sphere of radius R. Γ is then the (infinite) exterior of this sphere.

We endow $\mathbb{R}^3$ with the spherical coordinate system (ρ, ϕ, θ). On the finite domains $\theta \in [0, 2\pi]$ and $\phi \in [0\pi]$ we expand in spherical harmonics to represent our function as

$$\tilde{u}(\rho, \phi, \theta, t) = \sum_{n=0}^{\infty} \sum_{|m| \leq n} Y_{nm}(\theta, \phi) u_{nm}(\rho, t),$$

where $Y_{mn}(\phi, \theta)$ are the spherical harmonic functions, eigenmodes of the negative spherical Laplacian with eigenvalue $n(n+1)$. They are given by

$$Y_{mn}(\theta, \phi) = \frac{e^{im\theta} P_n^{|m|}(\cos\phi)}{\left\| e^{im\theta} P_n^{|m|}(\cos\phi) \right\|_{\sin\phi}},$$

where the normalizing factor is

$$\left\| e^{im\theta} P_n^{|m|}(\cos\phi) \right\|_{\sin\phi}^2 := \frac{4\pi(n+|m|)!}{(2n+1)(n-|m|)!},$$

and $P_n^{|m|}$ is the unnormalized associated Legendre function. With this, we can recast (8.1) as a differential equation for the modes $u_{nm}(\rho, t)$. In the following we shall omit the subscript mn dependence of u. The differential equation is

$$u_{tt} = c^2 \left[u_{\rho\rho} + \frac{2}{\rho} u_\rho - \frac{n(n+1)}{\rho^2} u \right]. \tag{8.2}$$

To determine a harmonic solution, we use the ansatz

$$u(\rho, t) = \cos(c\ t)\hat{u}(\rho).$$

With this ansatz, we see that $\hat{u}$ satisfies the eigenvalue equation

$$\rho^2 \hat{u}'' + 2\rho\hat{u}' + \big(\rho^2 - n(n+1)\big)\hat{u} = 0,$$

which has the spherical Bessel functions as solutions:

$$\hat{u}_{n,1}(\rho) = j_n(\rho) = \sqrt{\tfrac{\pi}{2\rho}} J_{n+1/2}(\rho)$$

$$\hat{u}_{n,2}(\rho) = y_n(\rho) = \sqrt{\tfrac{\pi}{2\rho}} Y_{n+1/2}(\rho),$$

where $J_n(\cdot)$ and $Y_n(\cdot)$ are the Bessel functions of the first and second kinds, respectively. The functions $\hat{u}_{n,1}$ and $\hat{u}_{n,2}$ are the kinds of functions that we will see in the solution to our time-evolution problem. Therefore, we should be interested in how well these functions are approximated by our spectral expansion. The solutions $\hat{u}_{n,1}$ and $\hat{u}_{n,2}$ do not decay exponentially and so we expect in general that the rational Chebyshev and Wiener methods give better behavior than the Laguerre method, but even so the convergence with the latter methods is not very good: second order at best.

We turn now to the numerical solution of (8.2) on the domain $\rho \in [R, \infty)$ with the

following initial and boundary conditions:

$$
\begin{aligned}
u(\rho,0) &= j_n(\rho) \\
u_\rho(\rho,0) &= j_n'(\rho) \\
u_\rho(R,t) &= \begin{cases} j_n'(R)\cos(c\ t), & n>1 \\ \frac{1}{3}\sin(ct), & n=1 \end{cases} \\
u(R,t) &= \begin{cases} 0, & n>0 \\ j_n(R)\cos(ct), & n=0 \end{cases} \\
u_t(R,t) &= \begin{cases} 0, & n>0 \\ -c\sin(c\ t)j_n(R), & n=0 \end{cases} \\
u_t(r,0) &= -c\sin(c\ t)j_n(\rho)
\end{aligned}
$$

These conditions correspond to the exact solution

$$u(\rho,t) = \cos(ct)j_n(\rho).$$

In order to solve this differential equation, we recast it as a system of first-order

equations:

$$v := u_t$$

$$w := u_\rho,$$

leading to the system

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix}_t = \begin{pmatrix} v \\ c^2\left[w_\rho + \frac{2w}{\rho} - \frac{n(n+1)}{\rho^2}u\right] \\ v_\rho \end{pmatrix},$$

with boundary conditions:

$$\left.\begin{pmatrix} u \\ v \\ w \end{pmatrix}\right|_{\rho=R} = \begin{pmatrix} \cos(ct)j_n(R) \\ -c\sin(ct)j_n'(R) \\ R\cos(ct)j_n'(R) \end{pmatrix}.$$

We start with the initial conditions

$$\left.\begin{pmatrix} u \\ v \\ w \end{pmatrix}\right|_{t=0} = \begin{pmatrix} j_n'(\rho) \\ 0 \\ \rho j_n'(\rho) \end{pmatrix}$$

We take $c = R = 1$ and evolve the above first-order hyperbolic system up to $T = 25$, almost four periods. This is a PDE on a semi-infinite interval and not an infinite interval, and so we cannot explicitly use our derived basis $\phi_k^{(s)}$ since these functions are defined on a fully infinite interval. However, this can be remedied by simply taking the even/odd portions of these functions and expanding with those. (Recall from the proof of theorem 5.1 that each set of functions is complete on half an interval with respect to a particular weight.) We approximate u, v, and w with a

modal expansion in $\mathrm{Re}\left\{\phi_n^{(1)}(x)\right\}$ for $x \in [R, \infty)$ and $n \in \mathbb{N}_0$. Note that in the end these are merely mapped Chebyshev polynomials, but they use a completely different mapping than that introduced in [11]. The set $\mathrm{Re}\left\{\phi_n^{(1)}(x)\right\}$ is complete in $L^2([R, \infty), \mathbb{R})$. An exponential filter is employed for the computation. It is also well-known that for problems on an infinite domain, there is a freedom to fix a linear stretching parameter (which we will call L). The value of this parameter usually has a significant influence on the solution. No attempt has been made to optimize this stretch parameter, which we fix at $L = 25$. This corresponds to half of the nodal points being to the left of $x = 25 + R = 26$, and the other half to the right of $x = 26$. We shall measure the quality of our solutions with the L^2-error, which is the unweighted L^2-error for the u expansion. The L^2 error is measured by 10^4-point interpolation on the Gaussian quadrature grid for the $\mathrm{Re}\left\{\phi_n^{(1)}\right\} \in L^2$ functions. (Note that these functions are essentially mapped Chebyshev polynomials.)

Despite the fact that we do not expect the convergence in N to be optimal, we cannot obtain better results from any other method. The Laguerre method consistently produces $O(1)$ errors since we must either choose to expand using Laguerre polynomials, which necessarily produce very highly oscillatory and unbounded derivatives, or using Laguerre functions, which have very poor convergence properties if the function does not decay exponentially. There is also the possibility of the 'algebraically-mapped' Chebyshev polynomials used by e.g. Boyd [11]. This method employs the Chebyshev (or some other Jacobi polynomial) method with a tangent-type map directly from $I_r = [-1, 1]$ to $\mathbb{R}^+$; they are functions TL_n from Table 7.2. This map is very different in character from the map that produces the Wiener functions $\mathrm{Re}\left\{\phi_n^{(1)}\right\}$; in particular it uses a linear fractional map to transform I_r to $\mathbb{R}$. As we have seen, the linear fractional map taking the Fourier series to the generalized Wiener basis preserves many of the properties of the Fourier series. Among other things, it preserves

the $O(N)$ eigenvalues of the stiffness matrix, which leads to a stability timestep restriction of the form $\Delta t \leq CN^{-1}$. We can therefore surmise that the $\Delta t \leq CN^{-2}$ timestep restriction for a Chebyshev polynomial method on the finite interval will carry over to the infinite interval; indeed our numerical results will show this. We will refer to the method using the $\mathrm{TL}_n(x)$ functions as the 'Chebyshev' method in the rest of this example.

To determine which of these basis sets is best suited for this problem, we run the simulation using all three methods with $N = 100$ and compare the qualitative function result as well as the time evolution of the L^2 error in Figure 8.2. For all three cases, the L^2 error was computed by interpolating to an $N = 10000$-point $\mathrm{Re}\left\{\phi_n^{(1)}\right\}$ Gauss grid, and then employing the associated quadrature against the exact solution.

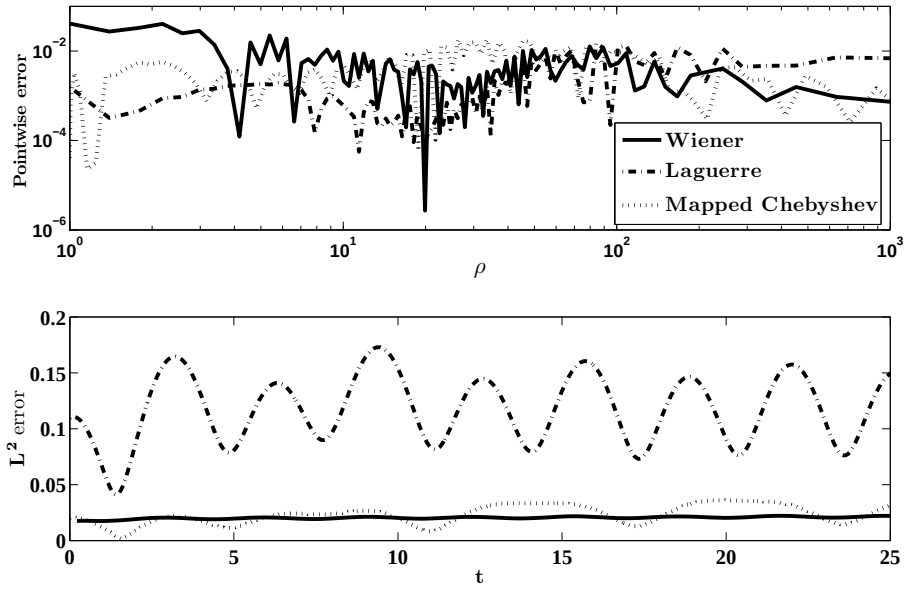

Figure 8.2: Plot of the pointwise error at $T = 25$ (top) and the corresponding time evolution of the L^2 errors (bottom) with the Wiener, Laguerre (function), and mapped Chebyshev methods. All methods use $N = 100$ grid points.

Note that we have chosen $L = 25$ for the mapped Chebyshev and Wiener cases,

which ensures that half of the grid points are to the left of $\rho = 26$. For the Laguerre basis, the choice $L \approx 0.4$ was made, which ensures roughly the same criterion of grid point distribution around $\rho = 26$. The Laguerre case seems to have the smallest L^∞ error in the top figure of Figure 8.2, for small ρ but the pointwise errors for $\rho > 50$ become very large, and one can see from the bottom portion of the figure that the Laguerre case has a bad history of L^2-error. In the pointwise error, the rational Chebyshev basis does quite admirably. However, it is significantly worse than the Wiener basis in the region $10 < \rho < 50$, which is where most of the grid points are. In addition, we will soon see that the time step restriction for the mapped Chebyshev basis is very handicapping.

The Wiener and rational Chebyshev cases give comparable L^2 accuracy, with a notable disadvantage for the Laguerre basis: convergence in N just for the initial condition is extremely slow due to the exponential decay of the basis functions.

The simulations were repeated numerous times with gradually increasing timesteps to ascertain the point of stability. For the fastest simulation, the Wiener basis calculation was completed in about 34 seconds. The Laguerre case has a much stricter CFL requirement and took about 391 seconds. The mapped Chebyshev basis has an even stricter CFL requirement and took about 1101 seconds. The Wiener basis, with its $O(N)$ eigenvalues for the nodal stiffness matrix has the clear advantage here. The mapped Chebyshev basis exhibits the known timestep penalty $\Delta t \leq CN^{-2}$.

The Wiener and Chebyshev bases can use the FFT coupled with a collocation method to increase speed relative to the Laguerre basis, but this was not used for the given cpu times above. This keeps the comparison of the computational times given above just a function of the maximum allowed timestep. Therefore the use of the FFT will

not ameliorate the CFL restriction for the mapped Chebyshev method; it will still perform much slower than the Wiener method.

A summary of the advantages and disadvantages for these three methods is given in table 8.7.

	Accuracy	Time step	FFT
Wiener	(+)	(++)	(+)
Laguerre function	($\circ$)	(−)	(−)
Mapped Chebyshev	(+)	(−−)	(+)

Table 8.7: Qualitative comparision of methods for solving problem (8.2)

8.2.2 The Korteweg-de Vries Equation

We consider the solution of the one-dimensional Korteweg-de Vries (KdV) equation

$$u_t + u_{xxx} + 6uu_x = 0. \tag{8.3}$$

This equation has a colliding two-soliton solution of the form

$$u(x,t) = \frac{2(ff'' - (f')^2)}{f^2}, \tag{8.4}$$

where prime ($'$) denotes partial differentiation with respect to x. $f(x,t)$ is defined for any constants a_1, a_2, b_1, b_2 as

$$\begin{aligned} f &= 1 + b_1 e^{w_1} + b_2 e^{w_2} + \left(\tfrac{a_1 - a_2}{a_1 + a_2}\right) b_1 b_2 e^{w_1 + w_2}, \\ w_1 &= a_1 x - a_1^3 t, \\ w_2 &= a_2 x - a_2^3 t. \end{aligned}$$

For our simulation, we take the parameters to have the values $a_1 = b_1 = b_2 = 1$ and $a_2 = 2$. This produces a solution $u(x,t)$ defined by equation (8.4) that exhibits a faster soliton overtaking a slower one. A graphical depiction of the evolution is shown for $|x| \leq 15$ and $|t| \leq 3.5$ in Figure 8.3.

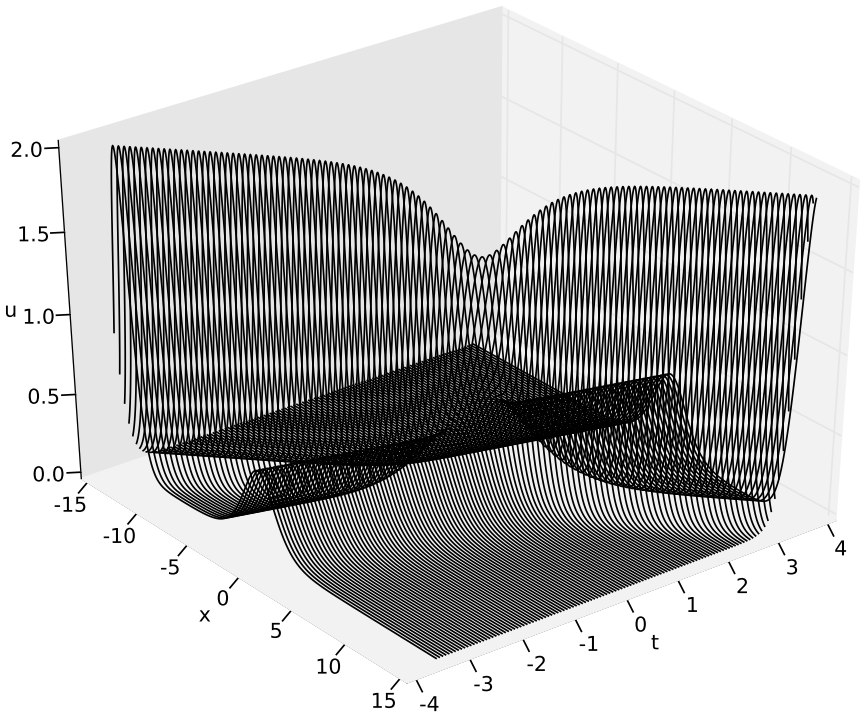

Figure 8.3: Evolution plot of the solution $u(x,t)$ for KdV Equation (8.3) simulated in this example. The solution plotted is the Wiener solution ($s = 1$) with $N = 300$ basis functions.

We solve this problem using:

1. Traditional Fourier series with domain truncation ($\Phi_k^{(0,0)}$)

2. Cardinal/Sinc approximation (C_k)

3. Hermite function approximation (h_n)

4. Mapped Jacobi (Chebyshev) polynomial approximation ($\text{pb}_n^{(s,t)}$ with $s = t = 1$)

5. Wiener basis set with $s = 1, 2$, and 5

The 2-soliton solution decays exponentially at infinity. The purpose of showing results for the Wiener expansion for different values of s is to illustrate the effectiveness of the approximation for $s \neq 1$ for exponential decay. Also note that we have not used domain truncation in combination with a Chebyshev polynomial expansion. It has been noted that Fourier domain truncation is superior to Chebyshev polynomial domain trunction [13]. Although there are other domain truncation methods that are competitive or, in some cases, superior to Fourier domain truncation [49], we assume that the Fourier simulation is a reasonable representative for domain truncation methods in general. We run the simulation from time $t = -3.5$ to $t = 3.5$, which enompasses the behavior shown in Figure 8.3. A spectral exponential filter identical for all methods is used in the computation. (Note that a spectral filter is the same as a spatial one for the Sinc approximation.) We use the FFT whenever possible. We run the temporal evolution with a fourth-order Runge-Kutta method.

To assess how strict the CFL restriction is for each method, we gradually decrease the time-step from some large value (e.g. $\Delta t = 0.1$) until the first few evolutions steps are stable. These timesteps are tabulated in Table 8.2.2; in addition, we also give the required evolution time using these timesteps. The Sinc method has an extremely restrictive timestep, usually an order of magnitude smaller than all of the other methods. The Fourier and Hermite timesteps are the best of all the methods. The mapped Jacobi functions and Wiener time steps are only smaller by an apparently constant factor of about 2/3.

The total evolution time is smallest using the truncated Fourier method; the Wiener and mapped Jacobi polynomial methods trail slightly behind. The mapped Jacobi polynomial method requires almost twice as much time as an $s = 1$ Wiener method. The Wiener methods become more expensive as s is increased; we expect this due to two factors: firstly the connection problem using the FFT becomes linearly more

Heuristically-determined stable Δt

	N =50	N =100	N =150	N =200	N =300	N =400	N =500
Fourier	7.64e-02	1.06e-02	3.54e-03	1.67e-03	4.85e-04	2.13e-04	1.13e-04
Hermite	6.79e-02	8.20e-03	2.63e-03	1.24e-03	4.69e-04	2.11e-04	1.00e-04
Sinc	6.65e-03	9.26e-04	3.27e-04	1.43e-04	4.19e-05	1.80e-05	——
$\mathrm{pb}_n^{(1,1)}$	5.78e-02	6.05e-03	1.85e-03	8.11e-04	2.98e-04	1.43e-04	7.09e-05
$\phi_k^{(1)}$	5.52e-02	5.96e-03	1.85e-03	8.13e-04	2.98e-04	1.43e-04	7.09e-05
$\phi_k^{(2)}$	5.31e-02	5.86e-03	1.75e-03	8.11e-04	2.98e-04	1.43e-04	7.09e-05
$\phi_k^{(5)}$	5.29e-02	5.79e-03	1.75e-03	8.11e-04	2.93e-04	1.34e-04	7.09e-05

Total evolution time, $t = -3.5, \ldots, 3.5$

	N =50	N =100	N =150	N =200	N =300	N =400	N =500
Fourier	5.45e-01	4.53e+00	1.44e+01	3.47e+01	1.51e+02	3.92e+02	8.64e+02
Hermite	5.15e+00	4.88e+00	2.37e+01	7.05e+01	5.46e+02	2.13e+03	7.81e+03
Sinc	1.40e+00	2.31e+01	1.24e+02	4.63e+02	3.38e+03	——	——
$\mathrm{pb}_n^{(1,1)}$	8.90e-01	9.68e+00	3.72e+01	9.79e+01	3.60e+02	9.95e+02	2.65e+03
$\phi_k^{(1)}$	9.43e-01	9.70e+00	3.49e+01	8.88e+01	2.99e+02	7.25e+02	1.66e+03
$\phi_k^{(2)}$	2.06e+00	2.03e+01	7.45e+01	1.71e+02	5.34e+02	1.26e+03	2.81e+03
$\phi_k^{(5)}$	2.31e+00	2.33e+01	8.35e+01	1.91e+02	6.20e+02	1.51e+03	3.18e+03

Table 8.8: Largest empirically-stable Δt (top) and required evolution time under this timestep (bottom).

expensive in s. Secondly, for all $s > 1$ the stiffness matrix is twice as dense as for $s = 1$. However, there is a considerable (unexpected) increase in required time between $s = 1$ and $s = 2$; this is likely due to some overhead computations that we have not optimized. The Hermite method takes significantly longer to compute, but the required time is on the same order of magnitude as the previously mentioned methods. Finally, the Sinc method takes far longer to compute than all of the others; despite the very great advantage of this method for computing modal coefficients, the stiffness matrix is full, and the stability timestep is highly debilitating.

To quantify the spatial errors, we run simulations for $N \leq 300$ using $\Delta t/4$, where Δt is the stable timestep for $N = 300$ for each basis set. We plot the time evolution of the L^2 and L^∞ errors in Figure 8.4. In Table 8.2.2 we give L^2 and L^∞ errors and

order of convergence for this test.

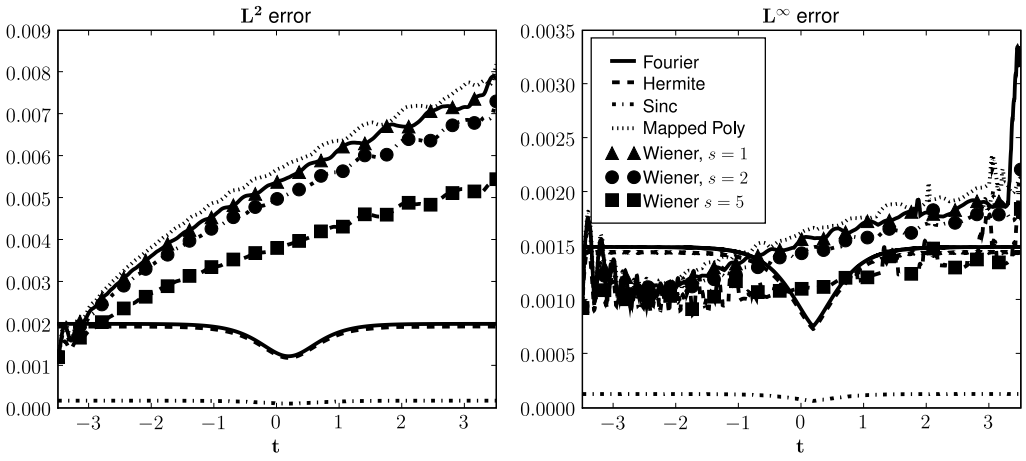

Figure 8.4: Plot of the L^2 (left) and L^∞ (right) errors vs. time for the 2-soliton KdV evolution, $N = 200$.

The results from Table 8.2.2 are very interesting. The Fourier, Hermite, and Sinc methods stagnate past $N \approx 150$; this is likely due to the time evolution error. The Sinc method here actually is the most impressive: it converges very quickly before stagnating. However, we have already noted that the Sinc computation is at least an order of magnitude more expensive than all the other computations. Thus, unless N is small then the Sinc method is simply not practical. The Hermite and Fourier methods both converge quickly, but the Fourier method exhibits an impressively small error for the relatively small value of $N = 100$. In addition, the time required for performing the Hermite computation becomes much more expensive than other methods for large N (see Table 8.2.2). The mapped Jacobi functions and the Wiener basis approximations all converge at roughly the same rate. The $s = 2$, $s = 5$ Wiener approximations are more accurate than the $s = 1$ approximation, but we recall that higher values of s take longer to compute. However, the Wiener method is significantly less accurate than the Fourier domain truncation method for small N. We stress again that all of the above comments pertain to approximations where we have made Δt very small in an attempt to single out spatial errors. However, if

L^2 errors

	N =50	N =100	order	N =150	order	N =200	order	N =300	order
Fourier	1.36e+00	2.43e-03	**9.13**	2.00e-03	0.474	2.00e-03	——	2.00e-03	——
Hermite	——-	3.29e-02	——	2.12e-03	**6.76**	1.94e-03	0.317	1.94e-03	——
Sinc	4.71e-02	1.74e-04	**8.08**	1.74e-04	——	1.74e-04	——	1.74e-04	——
$\text{pb}_n^{(1,1)}$	3.84e+00	5.74e-01	2.74	5.96e-02	**5.59**	8.20e-03	**6.89**	1.24e-03	4.66
$\phi_k^{(1)}$	3.54e+00	5.12e-01	2.79	5.57e-02	**5.47**	7.85e-03	**6.81**	1.23e-03	4.56
$\phi_k^{(2)}$	3.24e+00	4.70e-01	2.78	5.18e-02	**5.44**	7.31e-03	**6.81**	1.24e-03	4.38
$\phi_k^{(5)}$	3.21e+00	3.76e-01	3.09	3.99e-02	**5.54**	5.45e-03	**6.91**	1.22e-03	3.70

L^∞ errors

	N =50	N =100	order	N =150	order	N =200	order	N =300	order
Fourier	9.80e-01	1.48e-03	**9.37**	1.50e-03	——	1.49e-03	——	1.49e-03	——
Hermite	0.00e+00	1.60e-02	——	1.62e-03	**5.65**	1.44e-03	0.400	1.44e-03	——
Sinc	2.38e-02	1.30e-04	**7.52**	1.29e-04	——	1.29e-04	——	1.29e-04	——
$\text{pb}_n^{(1,1)}$	2.18e+00	4.21e-01	2.37	2.39e-02	**7.07**	2.13e-03	**8.41**	9.32e-04	2.04
$\phi_k^{(1)}$	2.20e+00	3.82e-01	2.52	2.26e-02	**6.97**	3.23e-03	**6.77**	9.31e-04	3.06
$\phi_k^{(2)}$	1.88e+00	3.70e-01	2.35	2.21e-02	**6.95**	2.21e-03	**7.99**	9.33e-04	2.13
$\phi_k^{(5)}$	1.91e+00	2.77e-01	2.78	1.87e-02	**6.65**	1.85e-03	**8.05**	9.17e-04	1.73

Table 8.9: L^2 and L^∞ errors for the KdV 2-soliton solution.

one runs the same computation, but uses the values of Δt in Table 8.2.2 for each simulation, we do not see such a large advantage of Fourier domain truncation over the other methods. In fact, for large N, the Wiener and mapped Jacobi methods produce smaller errors than the Fourier method. (Although this is likely due to the fact that the Fourier stability timestep is larger than the Wiener time step, and thus is more susceptible to temporal integration errors.)

We can conclude for this particular example with an exponentially decaying solution that the mapped methods (mapped Jacobi polynomials and Wiener functions) produce very similar results. The Sinc and Hermite methods do produce promising results, but they are suboptimal for reasons of computational effort required. The Fourier domain truncation method seems to produce better results than the others in terms of purely spatial error. However, this is in a case where the finite-interval

periodicity imposed by the Fourier approximation is not a problem at all: in the simulations performed in this section we have chosen to take the stretching parameter $L = 12$, which means that half of the nodes lie inside the interval $|x| \leq 12$ (see Figure 8.3 for an illustration of the structure of the solution). Thus, the truncation error here is inconsequential; we can expect that the truncation error will adversely affect the solution if care is not taken in choosing the scaling parameter.

8.2.3 The Vlasov-Poisson System

In this section we consider a spectral expansion for solving a PDE on an infinite interval. We start with the one-dimensional (1D space) Vlasov-Poisson (collisionless) system for a plasma:

$$\frac{\partial f}{\partial t} + v_x \frac{\partial f}{\partial x} + \frac{q}{m}\left[E_x \frac{\partial f}{\partial v_x} \right] = 0. \tag{8.5}$$

The variables used above are defined below:

$f(x, v_x, t)$	Distribution function for particle species of a single type
v_x	Velocity coordinate, x-direction
E_x	Vector components of the electric field
q	Charge of particle species
m	Mass of particle species

The velocity-space independent variable v_x can take any real value. We will assume that the physical-space independent variable x is defined on a finite interval $[-1, 1]$. The electric and magnetic fields are generated by the particles whose distibution is governed by f. In particular, given a distribution f over phase space, we can derive charge and current densities ρ and J for the single species given by

$$\begin{aligned} \rho(x,t) &= \int f(x,v,t)\mathrm{d}v_x \\ J_x(x,t) &= \int v_x f(x,v_x,t)\mathrm{d}v_x \end{aligned} \tag{8.6}$$

Having defined the charge and current densities, we must now determine how the electric and magnetic fields are computed. Of course, given the inhomogeneous inputs $\rho(x,t)$ and $J_{x,}(x,t)$, we can use these to solve Maxwell's equations (the governing

equations for electromagnetic phenomena) to determine the electric and magnetic fields. In this simplified one-dimensional scheme, Maxwell's equations reduce to

$$\frac{\partial E_x}{\partial t} = -\frac{1}{\varepsilon_0} J_x \tag{8.7}$$

In addition, we must satisfy Gauss's Law:

$$\frac{\partial E_x}{\partial x} = \frac{\rho}{\varepsilon_0} \tag{8.8}$$

Define $u(t) = \big(f, E_{x,}\big)^T$ as the full solution vector. Given a temporal discretization parameter Δt, a general numerical scheme we will consider involves the following steps:

1. Set $t_0 = 0$, $n = 0$.
2. Solve (8.5) and (8.7) simultaneously for $u(t_n + \Delta t)$ given $u(t_n)$
3. Correct for Gauss's Law using (8.8) with a Poisson solver.
4. Update the densities ρ and J_x at time $t_n + \Delta t$ using (8.6).
5. Set $t_{n+1} = t_n + \Delta t$, $n \leftarrow n + 1$, and go back to step 2.

The methods we will use to actually perform steps 2, 3, and 4 will differentiate one method from another. Although there are many challenges to consider in the simulation of (8.5), our desire here is to concentrate on the discretization of the unbounded velocity space. We will consider the following four methods to discretize velocity space:

- (Eulerian) Fourier series with domain truncation

- (Eulerian) Hermite function expansion
- (Eulerian) Wiener function expansion
- (Lagrangian) Particle-in-cell (PIC) approximation

The first three approximations are Eulerian approximations that we are familiar with: we represent $f(v_x)$ at all points in v_x space as a sum of basis functions. The last approximation is a Lagrangian approximation of $f(v_x)$: we fill physical space x with a discrete number of macroparticles endowed with mass, charge, and velocity, which sample the charge density function $f(x, v_x)$ at any given time. In order to connect this Lagrangian description of f to the Eulerian description we require for our discretization of physical space, we employ the popular and well-studied particle-in-cell method, which has gained a favorable reputation for its ability to drastically reduce simulation costs for plasma physics problems [19].

The Fourier method with domain trunction proceeds as follows: for each velocity dimension we select a parameter V serving as the cutoff for the truncation. With this parameter given, we expand the density function $f(v)$ in the variable v_x as

$$f(v_x) = \sum \hat{f}_k \Psi_k^{(0,0)} \left(\frac{V}{\pi} v_x \right), \qquad v_x \in [-V, V],$$

where we recall that the basis functions $\Psi_k^{(0,0)}$ are the canonical Fourier basis functions. Using a Fourier expansion for velocity space has been explored in the past [26] and methods have even been developed to battle difficulties that arise with Fourier expansions in velocity space [48] due to periodicity.

The Hermite and Wiener expansions are taken as L^2-projective Galerkin approximations that we have been familiar with throughout this thesis. In each of the spectral

expansion methods given above, we approximate the density function f as

$$f(x, v_x, t) \simeq f_h(x, v_x, t) = \sum_{k=-K_{v_x}}^{K_{v_x}} \tilde{f}_k(x, t) w_k(v_x).$$

We use this in equation (8.5) and project the result onto a finite-dimensional space in v_x using an L^2 projection; this results in the equation

$$\frac{\partial \tilde{f}_k}{\partial t} + C^w_{kr} \frac{\partial \tilde{f}_r}{\partial x} + \frac{q}{m} \left[E_x S^w_{kr} \tilde{f}_r \right] = 0, \tag{8.9}$$

where repeated indices are summed and the Galerkin stiffness matrix S is defined as

$$S^w_{kl} \;=\; \langle w_k, w_l' \rangle \,, \tag{8.10}$$

The superscript w denotes which basis function to use: the Fourier basis $\Psi^{(0,0)}_k$ (in which case the inner product is only taken over the finite interval $[-V, V]$), the Hermite functions h_n, or the Wiener functions $\phi^{(1)}_k$. For the Fourier basis functions, the stiffness matrix S is diagonal; for the Hermite function basis set it is bidiagonal; the Wiener basis set with $s = 1$ has a tridiagonal stiffness matrix. The matrix C (no relation to the connection coefficient matrix in Chapter 3) is dense for both the Fourier and Wiener cases; however it is again bidiagonal for the Hermite case. Therefore from a strictly Galerkin framework, the Hermite operators are more computationally efficient to apply; however, both the Fourier and Wiener bases have the advantage of the FFT for nodal/modal transformations.

If we arrange the $\tilde{f}_k$ into a vector $\tilde{\boldsymbol{f}}$, then we obtain the matrix-vector PDE system

$$\frac{\partial \tilde{\boldsymbol{f}}}{\partial t} + A \frac{\partial \tilde{\boldsymbol{f}}}{\partial x} + B_1(x, t, \tilde{\boldsymbol{f}}) \tilde{\boldsymbol{f}} = 0. \tag{8.11}$$

In physical space (x coordinate) we employ a discontinuous Galerkin (DG) method [44] for discretization. The discretization of the physical space coordinate using DG in plasma physics simulations is not new and has been investigated and found advantageous [46]. We use an upwind flux, which is not expensive to compute since A is not time-dependent, and we use a collocation method to compute the nonlinear term.

We will simulate a two-stream instability: consider an initial phase-space electron distribution of the form

$$f_0(x, v_x) = K v_x^2 e^{-v_x^2/2} \left(1 + \varepsilon \cos(\pi x)\right),$$

where K is a normalizing constant and $\varepsilon = 10^{-3}$. For this type of initial condition there is a unstable mode that grows which is physically due to two streams of particles with opposite velocities interacting with each other at the same position. This results in vortex-like behavior in phase space at the point of interaction.

In physical space we have 35 elements and use fifth order polynomials on each element. In velocity space we use a total of $N_{v_x} = 128$ modes. For the Fourier spectral method, we truncate the velocity domain to $[-V_x, V_x]$ for $V_x = 20$. The boundary conditions in physical space are periodic, and the boundary conditions in velocity space are the natural boundary conditions imposed by the basis expansions (i.e. periodic for the Fourier expansion, exponentially decaying for the Hermite expansion, algebraically decaying (at least) for the Wiener expansion, and no conditions for PIC).

In Figure 8.6 we show the time evolution of the electrostatic energy for each method. For the spectral methods solving the Vlasov equation, we see that the energy starts

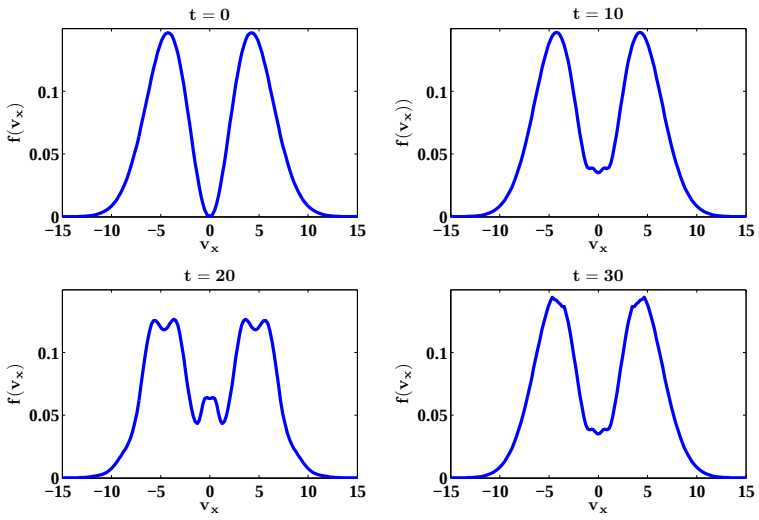

Figure 8.5: Density plots in velocity space for the two-stream instability at times $t = 0, 10, 20, 30$ for the Wiener expansion method.

out low initially and then succumbs to the instability when the unstable mode begins to grow. Each of the methods agrees with the others for short-time into the instability; the behavior longer in time shows some disagreement due to the fine structure exhibited by the density function in velocity space due to the instability (Figure 8.5). Refinement in velocity space suggests that the Hermite and Wiener expansions are the more accurate representations for $N_{v_x} = 128$ rather than the Fourier representation. This makes sense since the Fourier approximation introduces a nonphysical periodic boundary condition at a finite value of v_x. Thus the Wiener and Hermite expansions are the better choice for this problem.

The next natural question to ask is whether the Hermite expansion or the Wiener expansion is more applicable for this type of problem, i.e. whether the results from Figure 8.6 accurate. To assess this, we make use of the particle-in-cell algorithm.

For the PIC method, we show the result for various PIC runs vs the $N_{v_x} = 128$ Wiener expansion result in Figure 8.7. The different PIC runs differ only in the number of

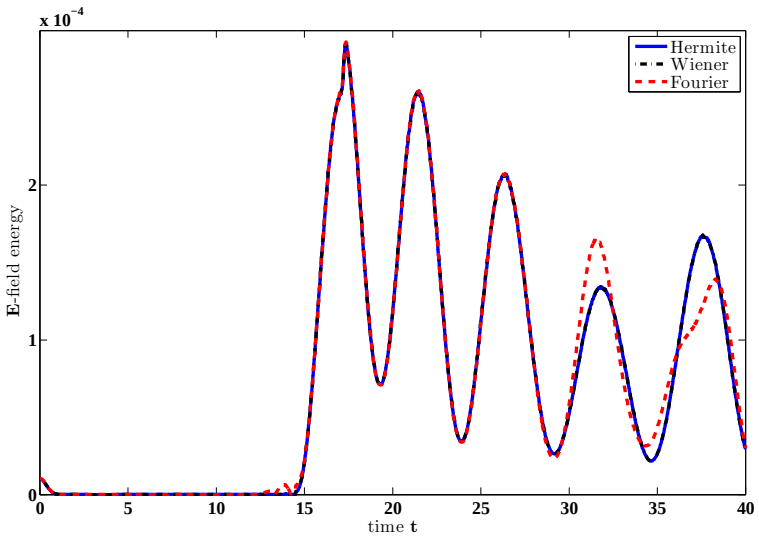

Figure 8.6: Electrostatic E-field energy as a function of time for the two-stream instability solved using each of the spectral methods.

particles used. Three PIC runs were completed with 24,500, 49,000, and 98,000 particles, respectively. For a relatively small number of particles, we see that the instability occurs with decreased strength and this leads to a significant qualitative dynamical deviation. For more particles, 49,000, the initial unstable dynamics are preserved, but they cannot be maintained, and the instability deviates significantly from the spectral result. However, when 98,000 particles are used, we can accurately reproduce the results from the spectral expansion. This indicates that the spectral expansion is a good tool for determining the quality of the plasma physics solutions in general, and may even be used as a method for estimating the error in a particle simulation.

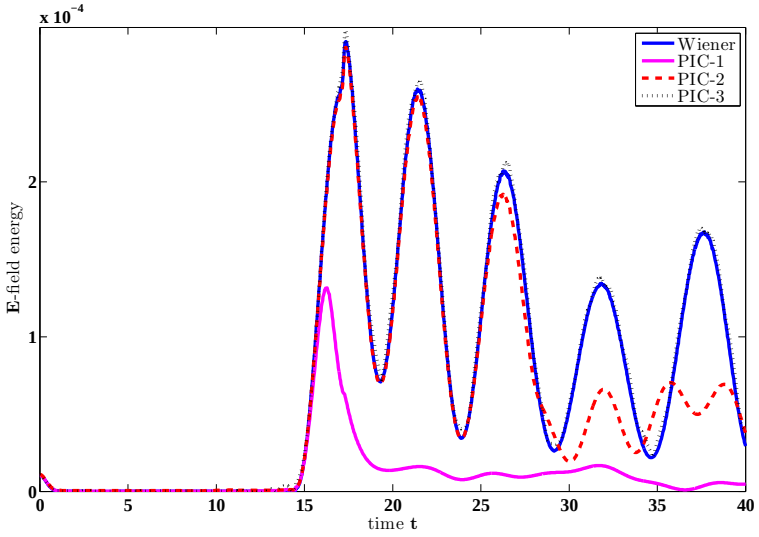

Figure 8.7: Electrostatic E-field as a function of time for three PIC simulations vs the $N_{v_x} = 128$ Wiener spectral method. PIC-1 is a simulation with $24,500$ particles, PIC-2 is with 49,000 particles, and PIC-3 is with 98,000 particles.

Chapter Nine

Conclusion

We have developed the generalized Wiener rational functions $\tilde{\Phi}^{(s)}$ and $\tilde{\phi}^{(s)}$ as basis sets for expanding functions defined over the semi-infinite and infinite real line. These functions are direct maps of a generalized Fourier series on a finite interval. On the finite interval, the generalized Fourier series can make use of the FFT to speed up computations by relying on the properties of Jacobi polynomials. This allows us to use the FFT for computations involving the generalized Wiener functions as well.

We propose that the Wiener basis set is a meritous alternative to existing spectral expansion methods on infinite intervals. The Wiener basis has a tunable rate of decay s for any $s > \frac{1}{2}$; changing the rate of decay can be accomplished in $O(N \log N)$ time if s is an integer, where N is the size of the expansion. This allows one to expand L^2 functions that do not decay exponentially, which has been an Achilles' heel for Hermite function or Sinc expansions. The Wiener basis also has a skew-Hermitian stiffness matrix with maximum eigenvalue $O(N)$ which shows that the timestep restriction for stability in solving time-dependent partial differential equations is not too harsh.

We have shown that the generalized Wiener basis set has value for function approximation and for the solution of differential equations. However, there are still unanswered questions and possible areas for improvement. It remains unclear how to properly make use of the ability to choose the decay parameter s for various problems. No doubt further exploration of applications will provide insight. In addition, we have used the mapped periodic generalized Fourier series as the basis functions of choice; as discussed in Section 5.3, there are alternative trigonometric functions that have have dubious value on the finite interval that may prove very accurate for spectral expansions on the infinite interval. The task of choosing the 'best' finite-interval basis functions to use for expansions on infinite intervals also remains unexplored. More in-depth studies for function approximation and the numerical solution of dif-

ferential equations must also be performed to properly ascertain any advantages the generalized Wiener basis set may have over other methods.

There is always a resolution problem on the infinite interval when expanding in a finite number of basis functions; when features of interest run outside the primary domain of resolution, it is often necessary to resample the solution and translate the basis functions to recapture the necessary resolution. To repeatedly accomplish this in an online manner can be computationally expensive. The linear fractional mapping employed in this thesis to take the unit circle to the real line could be generalized to devise an algorithm to perform the translation operation in $O(N)$ computational time. However, such a possibility needs to be researched.

The Wiener basis functions are derived from generalized Fourier series that in turn are composed of Jacobi polynomials. Because of this, it is natural to wonder if there is a provable relation between function regularity and rate of convergence to parallel the polynomial convergence result from Section 2.3. This possibility also requires further investigation.

We envision application of the topics studied in this thesis to problems outside the scope of general function approximation and spectral expansions. The generalized Fourier series we have developed are useful in performing windowed Fourier transforms since the weight function is a particular windowing function. The surprising nature of optimal Lebesgue constants in one dimension and beyond and the close relationship to Jacobi-Gauss quadratures is an area that requires further investigation. The generalized Wiener rational functions may be used as basis functions in infinite-element approximations, which couple finite-interval and infinite-interval solutions.

The problem of determining an accurate spectral approximation of a function on an infinite interval is a task rife with challenges. Although we cannot claim to have definitively solved any of the classical challenges that arise, we can present the generalized Wiener rational function basis as a consistently accurate and robust basis set for function expansions.

Appendix A

Orthogonal Polynomials

Here we collect information regarding classical orthogonal polynomials. Most of this can be found in e.g. [60], [1], and [34]. Among all properties that orthogonal polynomials satisfy, perhaps the most useful is the three-term recurrence relation

$$p_{n+1} = (x - a_n)p_n - b_n p_{n-1} \tag{A.1}$$

$$\sqrt{b_{n+1}}\tilde{p}_{n+1} = (x - a_n)\tilde{p}_n - \sqrt{b_n}\tilde{p}_{n-1}, \tag{A.2}$$

where p_n denotes the monic polynomial, and $\tilde{p}_n$ denotes the orthonormal polynomial.

A.1 Jacobi Polynomials

The Jacobi polynomials $P_n^{(\alpha,\beta)}$ are polynomial solutions to the differential equation

$$(1 - r^2)\rho'' + [\beta - \alpha - (\alpha + \beta + 2)r]\,\rho' + n(n + \alpha + \beta + 1)\rho = 0, \quad r \in [-1, 1]. \tag{A.3}$$

In canonical Sturm-Liouville form, this reads

$$-\frac{\mathrm{d}}{\mathrm{d}r}\left[(1 - r)^{\alpha+1}(1 + r)^{\beta+1}\rho'\right] - n(n + \alpha + \beta + 1)(1 - r)^{\alpha}(1 + r)^{\beta}\rho = 0. \tag{A.4}$$

One can read off the eigenvalues as $\lambda_n = n(n+\alpha+\beta+1)$. One thus has that the Jacobi polynomials are orthogonal under the weight $w_r^{(\alpha,\beta)}$. We restrict the parameters $\alpha, \beta > -1$ to ensure integrability of the weight function and thus existence of a family of orthogonal polynomials.

In the literature the common way to normalize Jacobi Polynomials is by the charac-

terization

$$\hat{P}_n^{(\alpha,\beta)}(r=1) = \begin{pmatrix} n+\alpha \\ \\ n \end{pmatrix} = \frac{\Gamma(n+\alpha+1)}{\Gamma(\alpha+1)n!},$$

where $\Gamma(\cdot)$ represents the Gamma function. For reference, the norm $\left\|\hat{P}_n^{(\alpha,\beta)}\right\|^2_{w_r^{(\alpha,\beta)}} = \hat{h}_n^{(\alpha,\beta)}$ is defined as

$$h_n^{(\alpha,\beta)} = \frac{2^{\alpha+\beta+1}}{2n+\alpha+\beta+1}\frac{\Gamma(n+\alpha+1)\Gamma(n+\beta+1)}{\Gamma(n+1)\Gamma(n+\alpha+\beta+1)},$$

These orthogonal polynomials are derived from the recurrence relation

$$p_{n+1} = (\hat{a}_n x - \hat{b}_n)p_n - \hat{c}_n p_{n-1},$$

where the constants $\hat{a}_n$, $\hat{b}_n$, and $\hat{c}_n$ are given by

$$\begin{aligned} \hat{a}_n^{(\alpha,\beta)} &= \tfrac{(2n+\alpha+\beta+1)(2n+\alpha+\beta+2)}{2(n+1)(n+\alpha+\beta+1)} \\ \\ \hat{b}_n^{(\alpha,\beta)} &= \tfrac{(\alpha^2-\beta^2)(2n+\alpha+\beta+1)}{2(n+1)(2n+\alpha+\beta)(n+\alpha+\beta+1)} \\ \\ \hat{c}_n^{(\alpha,\beta)} &= \tfrac{(n+\alpha)(n+\beta)(2n+\alpha+\beta+2)}{(n+1)(n+\alpha+\beta+1)(2n+\alpha+\beta)}. \end{aligned}$$

These polynomials have leading coefficient equal to

$$\hat{k}_n^{(\alpha,\beta)} = \begin{cases} 1, & n=0 \\ \\ \frac{\Gamma(2n+\alpha+\beta+1)}{\Gamma(n+\alpha+\beta+1)2^n n!}, & \text{else}. \end{cases}$$

We write the monic Jacobi polynomials as $P_n^{(\alpha,\beta)}(r)$ for any $n \in \mathbb{N}_0$. $P_n^{(\alpha,\beta)}(r)$ is a

solution to (2.3) and for every $n, m \in \mathbb{N}_0$ we have the orthogonality relation

$$\int_{-1}^{1} P_m^{(\alpha,\beta)}(r) P_n^{(\alpha,\beta)}(r)(1-r)^\alpha (1+r)^\beta \mathrm{d}r = h_n^{(\alpha,\beta)} \delta_{m,n},$$

where the monic polynomials have norm $\left\| P_n^{(\alpha,\beta)} \right\|^2 = h_n^{(\alpha,\beta)}$, which is given by

$$h_n^{(\alpha,\beta)} = \begin{cases} \frac{2^{\alpha+\beta+1}\Gamma(\alpha+1)\Gamma(\beta+1)}{\Gamma(\alpha+\beta+2)}, & n=0 \\ \\ \frac{2^{2n+\alpha+\beta+1}(n!)\Gamma(n+\alpha+\beta+1)\Gamma(n+\alpha+1)\Gamma(n+\beta+1)}{\Gamma(2n+\alpha+\beta+1)\Gamma(2n+\alpha+\beta+2)}, & \text{else} \end{cases} \tag{A.5}$$

Furthermore, the leading and second coefficients for the monic Jacobi polynomials are

$$k_n^{(\alpha,\beta)} = 1$$

$$k_n^{(\alpha,\beta)'} = \frac{n(\alpha-\beta)}{2n+\alpha+\beta}.$$

The monic Jacobi polynomials satisfy various relations and can be expressed in terms of the Hypergeometric function ${}_2F_1$, and also satisfy a generalized Rodrigues relation for all $m \leq n$:

$$\begin{pmatrix} 2n+\alpha+\beta \\ n \end{pmatrix} \omega^{(\alpha,\beta)} P_n^{(\alpha,\beta)} = \\ \frac{(-1)^m}{2^{m-n}(n-1)\cdots(n-m+1)} \frac{\mathrm{d}^m}{\mathrm{d}x^m} \left[\omega^{(\alpha+m,\beta+m)} P_{n-m}^{(\alpha+m,\beta+m)} \right]. \tag{A.6}$$

We shall often work with the normalized Jacobi polynomials, $\tilde{P}_n^{(\alpha,\beta)}(r) = \frac{P_n^{(\alpha,\beta)}(r)}{\sqrt{h_n^{(\alpha,\beta)}}}$, which are orthonormal with respect to the associated weight. Naturally these polynomials have the L^2-normalization constant $\tilde{h}_n^{(\alpha,\beta)} \equiv 1$.

For the Jacobi case, the recurrence coefficients in (A.1-A.2) used to generate the

monic and orthonormal polynomials are given by

$$a_n^{(\alpha,\beta)} = \begin{cases} \frac{\beta-\alpha}{(\alpha+\beta+2)}, & n=0 \\ \\ \frac{\beta^2-\alpha^2}{(2n+\alpha+\beta)(2n+\alpha+\beta+2)}, & n>0 \end{cases}$$

$$b_n^{(\alpha,\beta)} = \begin{cases} \frac{2^{\alpha+\beta+1}\Gamma(\alpha+1)\Gamma(\beta+1)}{\Gamma(\alpha+\beta+2)}, & n=0 \\ \\ \frac{4(1+\alpha)(1+\beta)}{(2+\alpha+\beta)^2(3+\alpha+\beta)}, & n=1 \\ \\ \frac{4n(n+\alpha)(n+\beta)(n+\alpha+\beta)}{(2n+\alpha+\beta)^2(2n+\alpha+\beta+1)(2n+\alpha+\beta-1)}, & n>1 \end{cases}$$

Define θ as $r = \cos\theta$. There are some particular choices for α and β which are of special note. These are the Chebyshev polynomials of the first, second, third, and fourth kinds, respectively:

$$t_n \tilde{P}_n^{(-1/2,-1/2)}(r) = T_n(r) = \cos(n\theta) \tag{A.7}$$

$$u_n \tilde{P}_n^{(+1/2,+1/2)}(r) = U_n(r) = \frac{\sin(n+1)\theta}{\sin\theta} \tag{A.8}$$

$$v_n \tilde{P}_n^{(-1/2,+1/2)}(r) = V_n(r) = \frac{\cos\left(n+\frac{1}{2}\right)\theta}{\cos\frac{1}{2}\theta} \tag{A.9}$$

$$w_n \tilde{P}_n^{(+1/2,-1/2)}(r) = W_n(r) = \frac{\sin\left(n+\frac{1}{2}\right)\theta}{\sin\frac{1}{2}\theta} \tag{A.10}$$

where $w_n = v_n = \sqrt{\pi}$; $u_n = \sqrt{\frac{\pi}{2}}$; and $t_n = \sqrt{\frac{\pi}{2}}$ for $n > 0$, and $t_0 = \sqrt{\pi}$.

In Chapter 3 we outlined a method to use the Fast Fourier Transform for computing spectral coefficients of Jacobi Polynomial transforms. The relations requisite for such an algorithm rely on expressions given in (3.1)-(3.4); these equations contain certain constants. These constants are derived in (A.20) and (A.27) below; for reference we reproduce them here:

$$\mu_{n,0}^{(\alpha,\beta)} = \sqrt{\frac{2(n+\alpha)(n+\alpha+\beta)}{(2n+\alpha+\beta)(2n+\alpha+\beta+1)}} \tag{A.11}$$

$$\mu_{n,1}^{(\alpha,\beta)} = \sqrt{\frac{2(n+1)(n+\beta+1)}{(2n+\alpha+\beta+1)(2n+\alpha+\beta+2)}} \tag{A.12}$$

$$\nu_{n,0}^{(\alpha,\beta)} = \sqrt{\frac{2(n+\alpha+1)(n+\alpha+\beta+1)}{(2n+\alpha+\beta+1)(2n+\alpha+\beta+2)}} \tag{A.13}$$

$$\nu_{n,-1}^{(\alpha,\beta)} = \sqrt{\frac{2n(n+\beta)}{(2n+\alpha+\beta)(2n+\alpha+\beta+1)}} \tag{A.14}$$

Using the normalization constant for the monic Jacobi polynomials, we derive, for $n > 0$:

$$\frac{h_n^{(\alpha-1,\beta)}}{h_n^{(\alpha,\beta)}} = \frac{(2n+\alpha+\beta)(2n+\alpha+\beta+1)}{2(n+\alpha)(n+\alpha+\beta)}$$

$$\frac{h_n^{(\alpha,\beta-1)}}{h_n^{(\alpha,\beta)}} = \frac{(2n+\alpha+\beta)(2n+\alpha+\beta+1)}{2(n+\beta)(n+\alpha+\beta)}$$

$$\frac{h_{n-1}^{(\alpha,\beta)}}{h_n^{(\alpha,\beta)}} = \frac{(2n+\alpha+\beta-1)(2n+\alpha+\beta)^2(2n+\alpha+\beta+1)}{4n(n+\alpha)(n+\beta)(n+\alpha+\beta)}$$

We claim that with these three formulas, we have obtained the recurrence constants for a great variety of nontrival relations. To give examples of how we can derive

other types of formulas from these three, we have e.g.

$$\frac{h_{n+1}^{(\alpha-1,\beta)}}{h_n^{(\alpha,\beta)}} = \frac{h_{n+1}^{(\alpha-1,\beta)}}{h_{n+1}^{(\alpha,\beta)}}\frac{h_{n+1}^{(\alpha,\beta)}}{h_n^{(\alpha,\beta)}} = \frac{2(n+1)(n+\beta+1)}{(2n+\alpha+\beta+1)(2n+\alpha+\beta+2)}$$

$$\frac{h_{n+1}^{(\alpha,\beta-1)}}{h_n^{(\alpha,\beta)}} = \frac{h_{n+1}^{(\alpha,\beta-1)}}{h_{n+1}^{(\alpha,\beta)}}\frac{h_{n+1}^{(\alpha,\beta)}}{h_n^{(\alpha,\beta)}} = \frac{2(n+1)(n+\alpha+1)}{(2n+\alpha+\beta+1)(2n+\alpha+\beta+2)}.$$

These new derived formulas may not seem immediately useful, but such expressions form the basis for all the recurrence formulas we now derive for the L^2 normalized Jacobi polynomials from known properties of the monic polynomials:

$$(1-r)P_n^{(\alpha,\beta)} = \gamma_{n,0}^{(\alpha,\beta)}P_n^{(\alpha-1,\beta)} - \gamma_{n,1}^{(\alpha,\beta)}P_{n+1}^{(\alpha-1,\beta)} \tag{A.15}$$

$$(1+r)P_n^{(\alpha,\beta)} = \gamma_{n,0}^{(\beta,\alpha)}P_n^{(\alpha,\beta-1)} + \gamma_{n,1}^{(\beta,\alpha)}P_{n+1}^{(\alpha,\beta-1)} \tag{A.16}$$

$$\left.\begin{aligned}\gamma_{n,0}^{(\alpha,\beta)} &= \frac{2(n+\alpha)(n+\alpha+\beta)}{(2n+\alpha+\beta)(2n+\alpha+\beta+1)} = \frac{h_n^{(\alpha,\beta)}}{h_n^{(\alpha-1,\beta)}}\\ \gamma_{n,1}^{(\alpha,\beta)} &= 1\end{aligned}\right\} \tag{A.17}$$

$$(1-r)\tilde{P}_n^{(\alpha,\beta)} = \mu_{n,0}^{(\alpha,\beta)}\tilde{P}_n^{(\alpha-1,\beta)} - \mu_{n,1}^{(\alpha,\beta)}\tilde{P}_{n+1}^{(\alpha-1,\beta)} \tag{A.18}$$

$$(1+r)\tilde{P}_n^{(\alpha,\beta)} = \mu_{n,0}^{(\beta,\alpha)}\tilde{P}_n^{(\alpha,\beta-1)} + \mu_{n,1}^{(\beta,\alpha)}\tilde{P}_{n+1}^{(\alpha,\beta-1)} \tag{A.19}$$

$$\left.\begin{aligned}\mu_{n,1}^{(\alpha,\beta)} &= \sqrt{\frac{2(n+1)(n+\beta+1)}{(2n+\alpha+\beta+1)(2n+\alpha+\beta+2)}} = \sqrt{\frac{h_{n+1}^{(\alpha-1,\beta)}}{h_n^{(\alpha,\beta)}}} \qquad (A.20)\\ \mu_{n,0}^{(\alpha,\beta)} &= \sqrt{\frac{2(n+\alpha)(n+\alpha+\beta)}{(2n+\alpha+\beta)(2n+\alpha+\beta+1)}} = \sqrt{\frac{h_n^{(\alpha,\beta)}}{h_n^{(\alpha-1,\beta)}}} \qquad (A.21)\end{aligned}\right\}$$

$$P_n^{(\alpha,\beta)} = \delta_{n,0}^{(\alpha,\beta)} P_n^{(\alpha+1,\beta)} - \delta_{n,-1}^{(\alpha,\beta)} P_{n-1}^{(\alpha+1,\beta)} \tag{A.22}$$

$$P_n^{(\alpha,\beta)} = \delta_{n,0}^{(\beta,\alpha)} P_n^{(\alpha,\beta+1)} + \delta_{n,-1}^{(\beta,\alpha)} P_{n-1}^{(\alpha,\beta+1)} \tag{A.23}$$

$$\left.\begin{aligned} \delta_{n,0}^{(\alpha,\beta)} &= 1 \\ \delta_{n,-1}^{(\alpha,\beta)} &= \frac{2n(n+\beta)}{(2n+\alpha+\beta)(2n+\alpha+\beta+1)} = \frac{h_n^{(\alpha,\beta)}}{h_{n-1}^{(\alpha+1,\beta)}} \end{aligned}\right\} \tag{A.24}$$

$$\tilde{P}_n^{(\alpha,\beta)} = \nu_{n,0}^{(\alpha,\beta)} \tilde{P}_n^{(\alpha+1,\beta)} - \nu_{n,-1}^{(\alpha,\beta)} \tilde{P}_{n-1}^{(\alpha+1,\beta)} \tag{A.25}$$

$$\tilde{P}_n^{(\alpha,\beta)} = \nu_{n,0}^{(\beta,\alpha)} \tilde{P}_n^{(\alpha,\beta+1)} + \nu_{n,-1}^{(\beta,\alpha)} \tilde{P}_{n-1}^{(\alpha,\beta+1)} \tag{A.26}$$

$$\left.\begin{aligned} \nu_{n,0}^{(\alpha\beta)} &= \sqrt{\frac{2(n+\alpha+1)(n+\alpha+\beta+1)}{(2n+\alpha+\beta+1)(2n+\alpha+\beta+2)}} = \sqrt{\frac{h_n^{(\alpha+1,\beta)}}{h_n^{(\alpha,\beta)}}} \\ \nu_{n,-1}^{(\alpha,\beta)} &= \sqrt{\frac{2n(n+\beta)}{(2n+\alpha+\beta)(2n+\alpha+\beta+1)}} = \sqrt{\frac{h_n^{(\alpha,\beta)}}{h_{n-1}^{(\alpha+1,\beta)}}} \end{aligned}\right\} \tag{A.27}$$

Putting these things together, we have

$$(1-r^2)P_n^{(\alpha,\beta)} = \sum_{k=0}^{2} \varepsilon_{n,k}^{(\alpha,\beta)} P_{n+k}^{(\alpha-1,\beta-1)} \tag{A.28}$$

$$\left.\begin{aligned} \varepsilon_{n,1}^{(\alpha,\beta)} &= \tfrac{2(\alpha-\beta)(n+\alpha+\beta)}{(2n+\alpha+\beta)(2n+\alpha+\beta+2)} \\ \varepsilon_{n,0}^{(\alpha,\beta)} &= \tfrac{4(n+\alpha)(n+\beta)(n+\alpha+\beta-1)(n+\alpha+\beta)}{(2n+\alpha+\beta-1)(2n+\alpha+\beta)^2(2n+\alpha+\beta+1)} \\ &= \tfrac{h_n^{(\alpha,\beta)}}{h_n^{(\alpha-1,\beta-1)}} \\ \varepsilon_{n,2}^{(\alpha,\beta)} &= -1 \end{aligned}\right\} \tag{A.29}$$

$$(1-r^2)\tilde{P}_n^{(\alpha,\beta)} = \sum_{k=0}^{2} \tilde{\varepsilon}_{n,k}^{(\alpha,\beta)} \tilde{P}_{n+k}^{(\alpha-1,\beta-1)} \tag{A.30}$$

$$\left.\begin{aligned} \tilde{\varepsilon}_{n,0}^{(\alpha,\beta)} &= \sqrt{\varepsilon_{n,0}^{(\alpha,\beta)}} \\ &= \sqrt{\tfrac{4(n+\alpha)(n+\beta)(n+\alpha+\beta-1)(n+\alpha+\beta)}{(2n+\alpha+\beta-1)(2n+\alpha+\beta)^2(2n+\alpha+\beta+1)}} \\ &= \sqrt{\tfrac{h_n^{(\alpha,\beta)}}{h_n^{(\alpha-1,\beta-1)}}} \\ \tilde{\varepsilon}_{n,1}^{(\alpha,\beta)} &= \varepsilon_{n,1}^{(\alpha,\beta)} \sqrt{\tfrac{h_{n+1}^{(\alpha-1,\beta-1)}}{h_n^{(\alpha\beta)}}} = \tfrac{2(\alpha-\beta)\sqrt{(n+1)(n+\alpha+\beta)}}{(2n+\alpha+\beta)(2n+\alpha+\beta+2)} \\ \tilde{\varepsilon}_{n,2}^{(\alpha,\beta)} &= -\sqrt{\tfrac{h_{n+2}^{(\alpha-1,\beta-1)}}{h_n^{(\alpha,\beta)}}} \\ &= -\sqrt{\tfrac{4(n+1)(n+2)(n+\alpha+1)(n+\beta+1)}{(2n+\alpha+\beta+1)(2n+\alpha+\beta+2)^2(2n+\alpha+\beta+3)}} \end{aligned}\right\} \tag{A.31}$$

$$P_n^{(\alpha,\beta)} = \sum_{k=-2}^{0} \eta_{n,k}^{(\alpha,\beta)} P_{n+k}^{(\alpha+1,\beta+1)} \tag{A.32}$$

$$\left.\begin{aligned} \eta_{n,0}^{(\alpha,\beta)} &= 1 \\ \eta_{n,-1}^{(\alpha,\beta)} &= \tfrac{2n(\alpha-\beta)}{(2n+\alpha+\beta)(2n+\alpha+\beta+2)} \\ \eta_{n,-2}^{(\alpha,\beta)} &= \tfrac{-4n(n-1)(n+\alpha)(n+\beta)}{(2n+\alpha+\beta-1)(2n+\alpha+\beta)^2(2n+\alpha+\beta+1)} = \tfrac{h_n^{(\alpha,\beta)}}{h_{n-2}^{(\alpha+1,\beta+1)}} \end{aligned}\right\} \tag{A.33}$$

$$\tilde{P}_n^{(\alpha,\beta)} = \sum_{k=-2}^{0} \tilde{\eta}_{k,n}^{(\alpha,\beta)} P_{n+k}^{(\alpha+1,\beta+1)} \tag{A.34}$$

$$\left.\begin{aligned} \tilde{\eta}_{n,0}^{(\alpha,\beta)} &= \sqrt{\tfrac{h_n^{(\alpha+1,\beta+1)}}{h_n^{(\alpha,\beta)}}} \\ &= \sqrt{\tfrac{4(n+\alpha+1)(n+\beta+1)(n+\alpha+\beta+1)(n+\alpha+\beta+2)}{(2n+\alpha+\beta+1)(2n+\alpha+\beta+2)^2(2n+\alpha+\beta+3)}} \\ \tilde{\eta}_{n,-1}^{(\alpha,\beta)} &= \eta_{n,-1}^{(\alpha,\beta)} \sqrt{\tfrac{h_{n-1}^{(\alpha+1,\beta+1)}}{h_n^{(\alpha,\beta)}}} = \tfrac{2(\alpha-\beta)\sqrt{n(n+\alpha+\beta+1)}}{(2n+\alpha+\beta)(2n+\alpha+\beta+2)} \\ \tilde{\eta}_{n,-2}^{(\alpha,\beta)} &= \eta_{n,-2}^{(\alpha,\beta)} \sqrt{\tfrac{h_{n-2}^{(\alpha+1,\beta+1)}}{h_n^{(\alpha,\beta)}}} = -\sqrt{-\eta_{n,-2}^{(\alpha,\beta)}} \\ &= -\sqrt{\tfrac{4n(n-1)(n+\alpha)(n+\beta)}{(2n+\alpha+\beta-1)(2n+\alpha+\beta)^2(2n+\alpha+\beta+1)}} \end{aligned}\right\} \tag{A.35}$$

$$\tfrac{\mathrm{d}}{\mathrm{d}r}\tilde{P}_n^{(\alpha,\beta)} = \sqrt{n(n+\alpha+\beta+1)}\,\tilde{P}_{n-1}^{(\alpha+1,\beta+1)} = \tilde{\zeta}_n^{(\alpha,\beta)} \tilde{P}_{n-1}^{(\alpha+1,\beta+1)} \tag{A.36}$$

$$
\begin{aligned}
(1-r^2)\frac{\mathrm{d}}{\mathrm{d}r}P_n^{(\alpha,\beta)} &= (1-r^2)nP_{n-1}^{(\alpha+1,\beta+1)} \\
&= n\sum_{k=0}^{2}\varepsilon_{n-1,k}^{(\alpha+1,\beta+1)}P_{n-1+k}^{(\alpha,\beta)} \qquad \text{(A.37)} \\
(1-r^2)\frac{\mathrm{d}}{\mathrm{d}r}\tilde{P}_n^{(\alpha,\beta)} &= (1-r^2)n\sqrt{\frac{h_{n-1}^{(\alpha+1,\beta+1)}}{h_n^{(\alpha,\beta)}}}\tilde{P}_{n-1}^{(\alpha+1,\beta+1)} \\
&= (1-r^2)\sqrt{n(n+\alpha+\beta+1)}\tilde{P}_{n-1}^{(\alpha+1,\beta+1)} \\
&= \sqrt{n(n+\alpha+\beta+1)}\sum_{k=0}^{2}\tilde{\varepsilon}_{n-1,k}^{(\alpha+1,\beta+1)}\tilde{P}_{n-1+k}^{(\alpha,\beta)} \qquad \text{(A.38)}
\end{aligned}
$$

A.2 Hermite polynomials

Another class of polynomials we discuss are the Hermite polynomials. The monic Hermite polynomials $H_n^{(0)}(x)$, $x \in \mathbb{R}$ satisfy

$$\frac{\mathrm{d}^2}{\mathrm{d}x^2}H_n^{(0)} - 2x\frac{\mathrm{d}}{\mathrm{d}x}H_n^{(0)} + 2nH^{(0)} = 0 \qquad \text{(A.39)}$$

In Sturm-Liouville form this reads

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(e^{-x^2}\frac{\mathrm{d}}{\mathrm{d}x}H_n^{(0)}\right) + 2ne^{-x^2}H_n^{(0)} = 0. \qquad \text{(A.40)}$$

From equation (A.40) one can read off that the Hermite polynomials are orthogonal under the weight $w_H^{(0)}(x) = e^{-x^2}$ with eigenvalues $\lambda_n = 2n$. The *generalized* Hermite polynomials are orthogonal under the weight $w_H^{(\alpha)} = |x|^{2\alpha}e^{-x^2}$ for $\alpha > -\frac{1}{2}$. The

recurrence formulae for equations (A.1) and (A.2) are

$$a_n^{(\alpha)} = 0$$

$$b_0^{(\alpha)} = \Gamma\left(\alpha + \tfrac{1}{2}\right)$$

$$b_n^{(\alpha)} = \begin{cases} \frac{n}{2}, & n \text{ even} \\ \\ \frac{n}{2} + \alpha, & n \text{ odd} \end{cases}$$

One of the nice properties of the Hermite polynomials is the convenient expression for differentiation:

$$\frac{\mathrm{d}}{\mathrm{d}x} H_n^{(0)}(x) = n H_{n-1}^{(0)}(x), \qquad \text{(monic)}.$$

The Hermite functions can be defined as

$$h_n^{(0)}(x) = e^{-x^2/2} H_n^{(0)}(x),$$

where $H_n^{(0)}$ can be either the monic or L^2-normalized polynomials and they satisfy the ODE

$$\frac{\mathrm{d}^2}{\mathrm{d}x^2} h_n^{(0)} + h_n^{(0)}(2n + 1 - x^2) = 0. \tag{A.41}$$

The differentiation property for the monic polynomials can be adapted to form a sparse differentiation representation for the L^2-normalized Hermite functions:

$$\frac{\mathrm{d}}{\mathrm{d}x} h_n^{(0)}(x) = \sqrt{\frac{n}{2}} h_{n-1}^{(0)} - \sqrt{\frac{n+1}{2}} h_{n+1}^{(0)}.$$

A.3 Laguerre polynomials

Laguerre polynomials are polynomial solutions to the generalized Laguerre equation

$$xy'' + (\alpha + 1 - x)y' + \lambda_n y = 0, \qquad x \in [0, \infty)$$

In Sturm-Louville form this reads

$$\left[x^{\alpha+1}e^{-x}y'\right]' + \lambda_n x^{\alpha}e^{-x}y = 0, \qquad x \in [0, \infty)$$

When the eigenvalue $\lambda_n = n$ and $\alpha > -1$, Laguerre polynomials $L_n^{(\alpha)}(x)$ are solutions to the above differential equations.

The Laguerre polynomials are usually normalized so that

$$L_n^{(\alpha)}(0) = \begin{pmatrix} n+\alpha \\ n \end{pmatrix}.$$

Given the normalization, the three-term recurrence is

$$(n+1)L_{n+1}^{(\alpha)} = (2n + \alpha + 1 - x)L_n^{(\alpha)} - (n+\alpha)L_{n-1}^{(\alpha)}.$$

The recurrence coefficients for (A.1) and (A.2) are

$$\begin{aligned} a_n^{(\alpha)} &= 2k + \alpha + 1, \\ b_0^{(\alpha)} &= \Gamma(1+\alpha), \\ b_n^{(\alpha)} &= n(n+\alpha), \qquad n > 0. \end{aligned}$$

To parallel the Jacobi case, the Laguerre polynomials also admit promotion/demotion-type formulae:

$$
\begin{aligned}
L_n^{(\alpha)} &= L_n^{(\alpha+1)} - L_{n-1}^{(\alpha+1)}, && n > 0 \\
xL_n^{(\alpha)} &= (n+\alpha)L_n^{(\alpha-1)} - (n+1)L_{n+1}^{(\alpha-1)}, && \alpha > 0 \\
\tfrac{\mathrm{d}}{\mathrm{d}x}L_n^{(\alpha)} &= -L_{n-1}^{(\alpha+1)}
\end{aligned}
$$

Appendix B

Differential Equations

We list in this appendix various systems of differential equations that are frequently derived from the Jacobi differential equation.

B.1 Fourier Differential Equations

The 'even' Fourier differential equation, a direct map of the Jacobi differential equation, is

$$\begin{aligned} &-\tfrac{\mathrm{d}}{\mathrm{d}\theta}\left[w_\theta^{(\gamma,\delta)}\Theta'\right] - n(n+\gamma+\delta)w_\theta^{(\gamma,\delta)}\Theta = 0, && \theta\in[0,\pi] \\ &\Theta'' - \tfrac{\Theta'}{\sin(\theta)}\left[(\gamma+\delta+2)\cos(\theta)+(\gamma-\delta)\right]\Theta' + n(n+\gamma+\delta)\Theta = 0, && \theta\in[0\pi] \end{aligned} \tag{B.1}$$

The solutions are the functions

$$\Theta_{e,n}^{(\gamma,\delta)} = P_n^{(\gamma-1/2,\delta-1/2)}(\cos(\theta)),$$

which are L^2-orthogonal under the weight $w_\theta^{(\gamma,\delta)}$.

The 'odd' Fourier differential equation is a map of the differential equation presented in lemma 2.1 with $a = b = \frac{1}{2}$:

$$\begin{aligned} &-\tfrac{\mathrm{d}}{\mathrm{d}\theta}\left[w_\theta^{(\gamma\delta)}\Theta'\right] + \left[\tfrac{(\gamma+\delta)}{2}(1+\cos^2(\theta)) + (\gamma-\delta)\cos(\theta)\right]\Theta - \\ &\qquad w_\theta^{(\gamma,\delta)}\left[n^2 + \left(n+\tfrac{1}{2}\right)(\gamma+\delta+2)\right]\Theta = 0, && \theta\in[0,\pi] \\ &-\Theta'' + \sin^3(\theta)\left[(\gamma-\delta)+(\gamma+\delta)\cos(\theta)\right]\Theta' + \\ &\qquad \sin^2(\theta)\left[\tfrac{(\gamma+\delta)}{2}(1+\cos^2(\theta)) + (\gamma-\delta)\cos(\theta)\right]\Theta - \\ &\qquad\qquad \left[n^2 + \left(n+\tfrac{1}{2}\right)(\gamma+\delta+2)\right]\Theta = 0, && \theta\in[0,\pi] \end{aligned} \tag{B.2}$$

The solutions are the functions

$$\Theta_{o,n+1}^{(\gamma,\delta)} = \sin(\theta) P_n^{(\gamma+1/2,\delta+1/2)}(\cos(\theta)),$$

which are L^2-orthogonal under the weight $w_\theta^{(\gamma,\delta)}$.

B.2 Rational function differential equations

B.2.1 The Rational Jacobi polynomials $\widetilde{PL}_n^{(s,t)}$

The rational Jacobi polynomials for the semi-infinite interval are introduced in Table 7.2 and used in Section 8.2.1. The map from the Jacobi interval $r \in [-1, 1]$ to the half-strip $x \in [0, \infty]$ is given by

$$\begin{aligned} x &= \tfrac{1+r}{1-r}, \qquad & 1-r &= \tfrac{2}{x+1}, \\ & & & \qquad \tfrac{\mathrm{d}r}{\mathrm{d}x} = \tfrac{2}{(x+1)^2} = \tfrac{1}{2}(1-r)^2 \\ r &= \tfrac{x-1}{x+1}, \qquad & 1+r &= \tfrac{2x}{x+1}, \end{aligned}$$

Under this transformation, the Jacobi differential equation (A.3) can be rewritten as

$$(x+1)^2 \xi'' + (x+1)\left[3 + 2t - x\,(2s-1)\right]\xi' + n(n+2s+2t-1)\xi = 0, \quad s > \frac{1}{2}, t > -\frac{1}{2}.$$

The equation in Sturn-Liouville format reads

$$-\frac{\mathrm{d}}{\mathrm{d}x}\left[\left(\frac{2}{x+1}\right)^{2s-3}\left(\frac{2x}{x+1}\right)^{2t+1}\xi'\right] - \frac{n}{4}\,(n+2s+2t-2)\left(\frac{2}{x+1}\right)^{2s}\left(\frac{2x}{x+1}\right)^{2t}\xi = 0.$$

The solutions are the rational Jacobi polynomials

$$\widetilde{PL}_n^{(s,t)} = \tilde{P}_n^{(2s-2,2t)}\left(\frac{x-1}{x+1}\right),$$

which are orthogonal under the weight

$$w_{\mathrm{PL}}^{(s,t)} = \frac{1}{2}\left(\frac{2}{x+1}\right)^{2s}\left(\frac{2x}{x+1}\right)^{2t}.$$

B.2.2 The rational Jacobi polynomials $\widetilde{PB}_n^{(s,t)}$

The rational Jacobi polynomials over the infinite line $\mathbb{R}$ were introduced in Sections 1.1.3 and 7.3. The map from the Jacobi interval $r \in [-1,1]$ to the real line $x \in \mathbb{R}$ is given by

$$\begin{aligned} r &= \tfrac{x}{\sqrt{1+x^2}}, \qquad & 1-r &= \tfrac{\sqrt{1+x^2}-x}{\sqrt{1+x^2}}, \\ & & \tfrac{\mathrm{d}r}{\mathrm{d}x} &= \tfrac{1}{(1+x^2)^{3/2}} = (1-r^2)^{3/2}. \\ x &= \tfrac{r}{\sqrt{1-r^2}}, \qquad & 1+r &= \tfrac{\sqrt{1+x^2}+x}{\sqrt{1+x^2}}, \end{aligned}$$

Under this transformation the Jacobi differential equation (A.3) becomes

$$\left(1+x^2\right)^2\xi'' + (1+x^2)\left[2x - s - t + (t-s)\sqrt{1+x^2}\right]\xi' + n(n+s+t-2)\xi = 0, \qquad s,t > \frac{1}{2}.$$

The equation in Sturm-Liouville format reads

$$-\tfrac{\mathrm{d}}{\mathrm{d}x}\left[\left(\tfrac{\sqrt{1+x^2}-x}{\sqrt{1+x^2}}\right)^{s-2}\left(\tfrac{\sqrt{1+x^2}+x}{\sqrt{1+x^2}}\right)^{t-2}\xi\right] -$$

$$n(n+s+t-2)\left(\tfrac{\sqrt{1+x^2}-x}{\sqrt{1+x^2}}\right)^{s}\left(\tfrac{\sqrt{1+x^2}+x}{\sqrt{1+x^2}}\right)^{t}\xi = 0$$

The solutions are the rational Jacobi polynomials over the entire interval:

$$\widetilde{PB}_n^{(s,t)} = \tilde{P}_n^{(s-3/2,t-3/2)}\left(\frac{x}{\sqrt{1+x^2}}\right),$$

and are orthogonal under the weight

$$w_{\mathrm{PB}}^{(s,t)} = \left(\frac{\sqrt{1+x^2}-x}{\sqrt{1+x^2}}\right)^{s+3/2}\left(\frac{\sqrt{1+x^2}+x}{\sqrt{1+x^2}}\right)^{t+3/2}$$

Appendix C

Recurrence Relations

This appendix is devoted to deriving the recurrence constants for the recurrence relations for the generalized Fourier basis set given in Section 5.5. Using the orthogonal polynomial three-term recurrence relation (A.2) we can show the following recurrence relation for the Szegö-Fourier functions $\Psi_n^{(\gamma,\delta)}(\theta)$:

$$\begin{aligned}\Psi_{n+1}^{(\gamma)} &= \left[U_n^{(\gamma)}\cos\theta - V_n^{(\gamma)}\right]\Psi_n^{(\gamma)} + \left[U_{-n}^{(\gamma)}\cos\theta - V_{-n}^{(\gamma)}\right]\Psi_{-n}^{(\gamma)} \\ &\quad -W_n^{(\gamma)}\Psi_{n-1}^{(\gamma)} - W_{-n}^{(\gamma)}\Psi_{-(n-1)}^{(\gamma)}.\end{aligned} \tag{C.1}$$

In the following expressions, we make use of the following definitions: for a given $\gamma > -\frac{1}{2}$,

$$\alpha := -\tfrac{1}{2}, \qquad \beta := \gamma - \tfrac{1}{2}.$$

The recurrence constants are then given by

$$U_{\pm n}^{(\gamma)} = \tfrac{1}{2}\left[\sqrt{\frac{1}{b_{n+1}^{(\alpha,\beta)}}} \pm \sqrt{\frac{1}{b_n^{(\alpha+1,\beta+1)}}}\right],$$

$$V_{\pm n}^{(\gamma)} = \pm\tfrac{1}{2}\left[\frac{a_n^{(\alpha,\beta)}}{\sqrt{b_{n+1}^{(\alpha,\beta)}}} + \frac{a_{n-1}^{(\alpha+1,\beta+1)}}{\sqrt{b_n^{(\alpha+1,\beta+1)}}}\right],$$

$$W_{\pm n}^{(\gamma)} = \pm\tfrac{1}{2}\left[\sqrt{\frac{b_n^{(\alpha,\beta)}}{b_{n+1}^{(\alpha,\beta)}}} + \sqrt{\frac{b_{n-1}^{(\alpha+1,\beta+1)}}{b_n^{(\alpha+1,\beta+1)}}}\right].$$

Using the promotion and demotion three-term recurrences (A.30, A.34) we also have the following recurrence relation:

$$\begin{aligned}\Psi_{n+1}^{(\gamma)} &= \left[\tilde{U}_n^{(\gamma)} i\sin\theta - \tilde{V}_n^{(\gamma)}\right]\Psi_n^{(\gamma)} + \left[\tilde{U}_{-n}^{(\gamma)} i\sin\theta - \tilde{V}_{-n}^{(\gamma)}\right]\Psi_{-n}^{(\gamma)} \\ &\quad -\tilde{W}_n^{(\gamma)}\Psi_{n-1}^{(\gamma)} - \tilde{W}_{-n}^{(\gamma)}\Psi_{-(n-1)}^{(\gamma)},\end{aligned} \tag{C.2}$$

where the recurrence constants are given by

$$\tilde{U}^{(\gamma)}_{\pm n} = \frac{1}{2}\left[\frac{1}{\tilde{\eta}^{(\alpha,\beta)}_{n,0}} \mp \frac{1}{\tilde{\varepsilon}^{(\alpha+1,\beta+1)}_{n-1,2}}\right],$$

$$\tilde{V}^{(\gamma)}_{\pm n} = \frac{1}{2}\left[\frac{\tilde{\varepsilon}^{(\alpha+1,\beta+1)}_{n-1,1}}{\tilde{\varepsilon}^{(\alpha+1,\beta+1)}_{n-1,2}} \pm \frac{\tilde{\eta}^{(\alpha,\beta)}_{n,-1}}{\tilde{\eta}^{(\alpha,\beta)}_{n,0}}\right],$$

$$\tilde{W}^{(\gamma)}_{\pm n} = \frac{1}{2}\left[\frac{\tilde{\varepsilon}^{(\alpha+1,\beta+1)}_{n-1,0}}{\tilde{\varepsilon}^{(\alpha+1,\beta+1)}_{n-1,2}} \pm \frac{\tilde{\eta}^{(\alpha,\beta)}_{n,-2}}{\tilde{\eta}^{(\alpha,\beta)}_{n,0}}\right].$$

Finally, putting these last two recurrences together yields

$$D^{(\gamma)}_n \Psi^{(\gamma)}_{n+1} = \left[A^{(\gamma)}_n e^{i\theta} - B^{(\gamma)}_n\right]\Psi^{(\gamma)}_n + \left[A^{(\gamma)}_{-n} e^{-i\theta} - B^{(\gamma)}_{-n}\right]\Psi^{(\gamma)}_{-n} + C^{(\gamma)}_n \Psi^{(\gamma)}_{n-1} + C^{(\gamma)}_{-n}\Psi^{(\gamma)}_{-(n-1)},$$

with the following values for the recurrence coefficients:

$$D_n^{(\gamma)} = \begin{cases} 4\tilde{\varepsilon}_{0,2}^{(\alpha,\beta)}\sqrt{\gamma}, & n=0 \\ 2\tilde{\varepsilon}_{n,2}^{(\alpha,\beta)}\left[\sqrt{n+\gamma}+\sqrt{n}\right], & n>0 \end{cases}$$

$$A_{\pm n}^{(\gamma)} = \begin{cases} \sqrt{2}\left[\sqrt{\gamma+1}\pm 1\right], & n=0 \\ \sqrt{n+\gamma+1}\pm\sqrt{n+1}, & n>0 \end{cases}$$

$$B_{\pm n}^{(\gamma)} = \begin{cases} \frac{2\gamma\sqrt{2}}{\sqrt{\gamma+1}}, & n=0 \\ \frac{-\tilde{\varepsilon}_{n,1}^{(\alpha,\beta)}}{2\sqrt{(n+1)(n+\gamma-1)}}\left(\gamma A_{\pm n}^{(\gamma)}+\left[2\sqrt{n(n+\gamma)}-1\right]A_{\mp n}^{(\gamma)}\right), & n>0 \end{cases}$$

$$C_{\pm n}^{(\gamma)} = \begin{cases} 0, & n=0 \\ \frac{1}{A_0^{(\gamma)}(\gamma+1)}\sqrt{\frac{(2\gamma+1)\gamma}{\gamma+2}}, & n=1 \\ \frac{\gamma\tilde{\varepsilon}_{n,0}^{(\alpha,\beta)}}{\sqrt{(n+\gamma-2)(n+\gamma-1)}A_{n-1}^{(\gamma)}}\left[\sqrt{(n+\gamma)^2-1}-\sqrt{n^2-1}\right], & n>1 \end{cases}$$

Appendix D

Stiffness Matrix

In this appendix we concentrate our efforts on deriving the stiffness result of Theorem 6.1. For notational convenience, we define

$$\alpha := -\tfrac{1}{2}, \quad \beta := s - \tfrac{3}{2}, \quad \tilde{r} := \tfrac{2x}{x^2+1} = \operatorname{sgn}(x)\sqrt{1 - r^2(x)}, \quad \gamma = s - 1$$

$$k^{\vee} = \operatorname{sgn}(k)(|k| - 1), \quad k^{\wedge} = \operatorname{sgn}(k)(|k| + 1), \quad n := |k| - 1$$

We start with (assuming $k \neq 0$):

$$\phi_k^{(s)} = \frac{2^{s/2}}{(x-i)^s}\left[\tilde{P}_{|k|}^{(\alpha,\beta)}(r) + i\tilde{r}\operatorname{sgn}(k)\tilde{P}_{|k|-1}^{(\alpha+1,\beta+1)}(r)\right].$$

From this, we can derive the relation

$$\frac{\mathrm{d}\phi_k^{(s)}}{\mathrm{d}x} = \frac{-s}{x-i}\phi_k^{(s)} + (1+r)\frac{2^{s/2}}{(x-i)^s}\frac{\mathrm{d}\Psi^{(\gamma)}(\theta)}{\mathrm{d}\theta}. \tag{D.1}$$

Noting that

$$\frac{-s}{x-i} = -\frac{s}{2}\left[\tilde{r} + i(1+r)\right] = -\frac{is}{2}\left[1 + e^{-i\theta}\right],$$

then we can easily use the recurrence relations presented in Appendix C to prove the following lemma:

Lemma D.1 *We have the representation:*

$$\tfrac{-s}{(x-i)}\phi_k^{(s)} \quad = \quad \textstyle\sum_{l\in\{\pm k^{\vee},\pm k,\pm k^{\wedge}\}} \chi_l^{(s)}\phi_l^{(s)},$$

where

$$\chi^{(s)}_{\pm k^\vee} = -\frac{i\,\mathrm{sgn}(k)s}{2(2n+\alpha+\beta+2)}\sqrt{\frac{(n+\alpha+1)(n+\beta+1)}{(2n+\alpha+\beta+1)(2n+\alpha+\beta+3)}}\times$$

$$\left(\sqrt{n+\alpha+\beta+1}\pm\mathrm{sgn}(k)\sqrt{n}\right)\left(\sqrt{n+\alpha+\beta+2}+\mathrm{sgn}(k)\sqrt{n+1}\right)$$

$$\chi^{(s)}_{k} = -\frac{is(\beta-\alpha)(\alpha+\beta+1)}{2(2n+\alpha+\beta+2)(2n+\alpha+\beta+4)}-\frac{is}{2}$$

$$\chi^{(s)}_{-k} = -\frac{is(\alpha-\beta)}{2(2n+\alpha+\beta+2)(2n+\alpha+\beta+4)}\left(1+2\,\mathrm{sgn}(k)\sqrt{(n+1)(n+\alpha+\beta+2)}\right)$$

$$\chi^{(s)}_{\pm k^\wedge} = -\frac{i\,\mathrm{sgn}(k)s}{2(2n+\alpha+\beta+4)}\sqrt{\frac{(n+\alpha+2)(n+\beta+2)}{(2n+\alpha+\beta+3)(2n+\alpha+\beta+5)}}\times$$

$$\left(\mathrm{sgn}(k)\sqrt{n+2}\mp\sqrt{n+\alpha+\beta+3}\right)\left(\sqrt{n+\alpha+\beta+2}-\mathrm{sgn}(k)\sqrt{n+1}\right)$$

We now present the next lemma necessary for our stiffness matrix result.

Lemma D.2 *We have*

$$\frac{\mathrm{d}\Phi^{(s)}_k(x)}{\mathrm{d}x} = \sum_{l\in\{\pm k^\vee,\pm k,\pm k^\wedge\}}\sigma^{(s)}_l\Phi^{(s)}_l,$$

with

$$\begin{aligned}
\sigma_{\pm k^\vee}^{(s)} &= i\,\mathrm{sgn}(k)\tfrac{(n+\alpha+\beta+2)}{(2n+\alpha+\beta+2)}\sqrt{\tfrac{(n+\alpha+1)(n+\beta+1)}{(2n+\alpha+\beta+1)(2n+\alpha+\beta+3)}}\times \\
&\quad \left[\sqrt{(n+\alpha+\beta+1)(n+\alpha+\beta+2)} \pm \sqrt{n(n+1)}\right]
\end{aligned}$$

$$\begin{aligned}
\sigma_{\pm k}^{(s)} &= i\,\mathrm{sgn}(k)\begin{cases} \sqrt{(n+1)(n+\alpha+\beta+2)}, & +k \\ \\ \sqrt{(n+1)(n+\alpha+\beta+2)}\tfrac{(\alpha-\beta)(\alpha+\beta+2)}{(2n+\alpha+\beta+2)(2n+\alpha+\beta+4)} & -k \end{cases}
\end{aligned}$$

$$\begin{aligned}
\sigma_{\pm k^\wedge}^{(s)} &= i\,\mathrm{sgn}(k)\tfrac{(n+1)}{(2n+\alpha+\beta+4)}\sqrt{\tfrac{(n+\alpha+2)(n+\beta+2)}{(2n+\alpha+\beta+3)(2n+\alpha+\beta+5)}}\times \\
&\quad \left[\sqrt{(n+1)(n+2)} \pm \sqrt{(n+\alpha+\beta+2)(n+\alpha+\beta+3)}\right]
\end{aligned}$$

The above lemma states that the unweighted Wiener rational functions $\Phi_k^{(s)}$ have derivatives that admit sparse representations in the original basis functions. The proof will show that this is a property that is not shared by the generalized Fourier series functions.

Proof Our intermediate goal is to prove that the derivative of $\Psi_k^{(s-1)}$ is a sparse combination of $\Psi_k^{(s)}$. Note that we can explicitly calculate the expression

$$\begin{aligned}
2\tfrac{\mathrm{d}\Psi_k^{(s-1)}}{\mathrm{d}\theta} &= -\tilde{r}\zeta_{|k|}^{(\alpha,\beta)}\tilde{P}_{|k|-1}^{(\alpha+1,\beta+1)} + \\
&\quad i\,\mathrm{sgn}(k)\left\{\left[b_{|k|-1}^{(\alpha+1,\beta+1)} - \zeta_{|k|-1}^{(\alpha+1,\beta+1)}\varepsilon_{|k|-2,0}^{(\alpha+2,\beta+2)}\right]\tilde{P}_{|k|-2}^{(\alpha+1,\beta+1)} + \right. \\
&\quad \left[a_{|k|-1}^{(\alpha+1,\beta+1)} - \zeta_{|k|-1}^{(\alpha+1,\beta+1)}\varepsilon_{|k|-2,1}^{(\alpha+2,\beta+2)}\right]\tilde{P}_{|k|-1}^{(\alpha+1,\beta+1)} + \\
&\quad \left.\left[b_{|k|}^{(\alpha+1,\beta+1)} - \zeta_{|k|-1}^{(\alpha+1,\beta+1)}\varepsilon_{|k|-2,2}^{(\alpha+2,\beta+2)}\right]\tilde{P}_{|k|}^{(\alpha+1,\beta+1)}\right\},
\end{aligned} \tag{D.2}$$

where we have used the Jacobi recurrence coefficients from the three-term recurrence relation (A.2), the derivative promotion relation (A.36), and the promotion relation (A.34). Note that we have omitted the tilde's on all the recurrence coefficients; this is done to avoid burdensome notation, but one must be cognizant that all of the recurrence coefficients that we use are for the L^2-normalized Jacobi polynomials, not the monic ones. By considering the form of the $\Psi_k^{(s)}$ functions presented in Theorem 5.1, one can determine that the first term is already a sparse combination of $\Psi_k^{(s)}(\theta)$ functions. However, the second term in curly brackets require significant finesse. Our goal is to write this term in curly brackets as a sum of two Jacobi polynomials $\tilde{P}_n^{(\alpha,\beta+1)}$. To do this, we aim to use the promotion relation (A.25). We have the following equalities for $n := |k| - 1$

$$\left[b_{|k|-1}^{(\alpha+1,\beta+1)} - \zeta_{|k|-1}^{(\alpha+1,\beta+1)}\varepsilon_{|k|-2,0}^{(\alpha+2,\beta+2)}\right] =$$

$$-\tfrac{2(n+\alpha+\beta+2)}{2n+\alpha+\beta+2}\sqrt{\tfrac{n(n+\alpha+1)(n+\beta+1)(n+\alpha+\beta+2)}{(2n+\alpha+\beta+1)(2n+\alpha+\beta+3)}},$$

$$\left[a_{|k|-1}^{(\alpha+1,\beta+1)} - \zeta_{|k|-1}^{(\alpha+1,\beta+1)}\varepsilon_{|k|-2,1}^{(\alpha+2,\beta+2)}\right] = \tfrac{(\beta-\alpha)(\alpha+\beta+2+2n(n+\alpha+\beta+3))}{(2n+\alpha+\beta+2)(2n+\alpha+\beta+4)},$$

$$\left[b_{|k|}^{(\alpha+1,\beta+1)} - \zeta_{|k|-1}^{(\alpha+1,\beta+1)}\varepsilon_{|k|-2,2}^{(\alpha+2,\beta+2)}\right] =$$

$$\tfrac{2(n+1)}{2n+\alpha+\beta+4}\sqrt{\tfrac{(n+1)(n+\alpha+2)(n+\beta+2)(n+\alpha+\beta+3)}{(2n+\alpha+\beta+3)(2n+\alpha+\beta+5)}}.$$

Rewriting the above expressions using the promotion recurrence constants from (A.25) gives us

$$\left[b_{|k|-1}^{(\alpha+1,\beta+1)} - \zeta_{|k|-1}^{(\alpha+1,\beta+1)}\varepsilon_{|k|-2,0}^{(\alpha+2,\beta+2)}\right] =$$

$$-\nu_{n,-1}^{(\alpha,\beta+1)}(n+\alpha+\beta+2)\sqrt{\tfrac{2(n+\alpha+1)(n+\alpha+\beta+2)}{(2n+\alpha+\beta+2)(2n+\alpha+\beta+3)}},$$

$$\left[b_{|k|}^{(\alpha+1,\beta+1)} - \zeta_{|k|-1}^{(\alpha+1,\beta+1)}\varepsilon_{|k|-2,2}^{(\alpha+2,\beta+2)}\right] =$$

$$\nu_{n+1,0}^{(\alpha,\beta+1)}(n+1)\sqrt{\tfrac{2(n+1)(n+\beta+2)}{(2n+\alpha+\beta+3)(2n+\alpha+\beta+4)}}.$$

Some algebra allows us to show

$$\left[a_{|k|-1}^{(\alpha+1,\beta+1)} - \zeta_{|k|-1}^{(\alpha+1,\beta+1)}\varepsilon_{|k|-2,1}^{(\alpha+2,\beta+2)}\right] =$$

$$\nu_{n,0}^{(\alpha,\beta+1)}(n+\alpha+\beta+2)\sqrt{\tfrac{2(n+\alpha+1)(n+\alpha+\beta+2)}{(2n+\alpha+\beta+2)(2n+\alpha+\beta+3)}}$$

$$-\nu_{n+1,-1}^{(\alpha,\beta+1)}(n+1)\sqrt{\tfrac{2(n+1)(n+\beta+2)}{(2n+\alpha+\beta+3)(2n+\alpha+\beta+4)}}.$$

Using these last three expressions back in the expansion (D.2) allows us to use the promotion relation (A.25) to write

$$\begin{aligned} 2\tfrac{\mathrm{d}\Psi_k^{(s-1)}}{\mathrm{d}\theta} &= -\tilde{r}\zeta_{|k|}^{(\alpha,\beta)}\tilde{P}_{|k|-1}^{(\alpha+1,\beta+1)} \\ &\quad +i\,\mathrm{sgn}(k)\left\{(n+\alpha+\beta+2)\sqrt{\tfrac{2(n+\alpha+1)(n+\alpha+\beta+2)}{(2n+\alpha+\beta+2)(2n+\alpha+\beta+3)}}\tilde{P}_{|k|-1}^{(\alpha,\beta+1)}+\right. \\ &\quad \left.(n+1)\sqrt{\tfrac{2(n+1)(n+\beta+2)}{(2n+\alpha+\beta+3)(2n+\alpha+\beta+4)}}\tilde{P}_{|k|}^{(\alpha,\beta+1)}\right\}. \end{aligned}$$

To facilitate notation, define

$$f_{n,0}^{(\alpha,\beta)} := (n+\alpha+\beta+2)\mu_{n,0}^{(\alpha+1,\beta+1)},$$

$$f_{n,1}^{(\alpha,\beta)} := (n+1)\mu_{n,1}^{(\alpha+1,\beta+1)}.$$

Our final step is now to use the three-term recurrence relation (A.2) along with the demotion relation (A.19). Doing so, we obtain

$$\begin{aligned} 2(1+r)\frac{\mathrm{d}\Psi_k^{(s-1)}}{\mathrm{d}\theta} &= -\tilde{r}\zeta_{|k|}^{(\alpha,\beta)}\left[\sqrt{b_{|k|}^{(\alpha+1,\beta+1)}}\tilde{P}_{|k|}^{(\alpha+1,\beta+1)} + \left(a_{|k|-1}^{(\alpha+1,\beta+1)}+1\right)\tilde{P}_{|k|-1}^{(\alpha+1,\beta+1)} + \right. \\ &\quad \left. \sqrt{b_{|k|-1}^{(\alpha+1,\beta+1)}}\tilde{P}_{|k|-2}^{(\alpha+1,\beta+1)}\right] + i\,\mathrm{sgn}(k)\left\{f_{n,0}^{(\alpha,\beta)}\gamma_{n,0}^{(\beta+1,\alpha)}\tilde{P}_{|k|-1}^{(\alpha,\beta)} + \right. \\ &\quad \left. + \left(f_{n,0}^{(\alpha,\beta)}\gamma_{n,1}^{(\beta+1,\alpha)} + f_{n,1}^{(\alpha,\beta)}\gamma_{n+1,0}^{(\beta+1,\alpha)}\right)\tilde{P}_{|k|}^{(\alpha,\beta)} + f_{n,1}^{(\alpha,\beta)}\gamma_{n+1,1}^{(\beta+1,\alpha)}\tilde{P}_{|k|+1}^{(\alpha,\beta)}\right\}. \end{aligned}$$

We can see from Theorem 5.1 that $\tilde{r}\tilde{P}_n^{(\alpha+1,\beta+1)}$ and $\tilde{P}_n^{(\alpha,\beta)}$ are sparse combinations of $\Psi_k^{(s)}(\theta)$. Noting that $\frac{\mathrm{d}}{\mathrm{d}x}\Phi_k^{(s)} = (1+r)\frac{\mathrm{d}}{\mathrm{d}\theta}\Psi_k^{(s-1)}$ and making the connection $\Phi_k^{(s)}(x) := \Psi_k^{(s-1)}(\theta(x))$ proves the result.

□

Putting the two lemmas together, we have

Theorem D.1 *The following equality holds:*

$$\frac{\mathrm{d}\phi_k^{(s)}}{\mathrm{d}x} = \sum_{l\in\{\pm k^\vee,\pm k,\pm k^\wedge\}} \tau_l^{(s)}\phi_l^{(s)},$$

where the constants $\tau_l^{(s)}$ *are equal to*

$$\begin{aligned}
\tau_{k,\pm k^\vee} &= \tfrac{i}{2}\sqrt{\tfrac{(n+\alpha+1)(n+\beta+1)}{(2n+\alpha+\beta+1)(2n+\alpha+\beta+3)}}\times \\
&\quad \Big\{\mathrm{sgn}(k)\left(\sqrt{(n+\alpha+\beta+1)(n+\alpha+\beta+2)}\pm\sqrt{n(n+1)}\right)- \\
&\quad \tfrac{\alpha+\beta+2}{(2n+\alpha+\beta+2)}\left(\sqrt{(n+1)(n+\alpha+\beta+1)}\pm\sqrt{n(n+\alpha+\beta+2)}\right)\Big\} \\
\tau_{k,k} &= i\,\mathrm{sgn}(k)\sqrt{(n+1)(n+\alpha+\beta+2)}-\tfrac{is(\beta-\alpha)(\alpha+\beta+1)}{2(2n+\alpha+\beta+2)(2n+\alpha+\beta+4)}-\tfrac{is}{2} \\
\tau_{k,-k} &= -\tfrac{is(\alpha-\beta)}{2(2n+\alpha+\beta+2)(2n+\alpha+\beta+4)} \\
\tau_{k,\pm k^\wedge} &= \tfrac{i}{2}\sqrt{\tfrac{(n+\alpha+2)(n+\beta+2)}{(2n+\alpha+\beta+3)(2n+\alpha+\beta+5)}}\times \\
&\quad \Big\{-\tfrac{s}{2n+\alpha+\beta+4}\left[\sqrt{(n+2)(n+\alpha+\beta+2)}\pm\sqrt{(n+1)(n+\alpha+\beta+3)}\right]+ \\
&\quad \mathrm{sgn}(k)\left[\sqrt{(n+1)(n+2)}\pm\sqrt{(n+\alpha+\beta+2)(n+\alpha+\beta+3)}\right]\Big\}
\end{aligned}$$

Clearly for $|k|=1$ *we have* $\tau_{k,\pm k^\vee}=\tau_{k,0}$ *so that, taking into account the different normalization constant in the definition of* $\phi_k^{(s)}$ *for* $k=0$*:*

$$\tau_{k,0}=\frac{i}{\sqrt{2}}\sqrt{\frac{(\alpha+1)(\beta+1)}{(\alpha+\beta+3)}}\left\{\mathrm{sgn}(k)\sqrt{(\alpha+\beta+2)}-1\right\}$$

And for $k=0$*:*

$$\frac{\mathrm{d}\phi_0^{(s)}}{\mathrm{d}x}=-\frac{i\sqrt{s-\frac{1}{2}}}{2}\left[\frac{1+\sqrt{s}}{\sqrt{1+s}}\phi_{-1}^{(s)}+2\sqrt{s-\frac{1}{2}}\phi_0^{(s)}+\frac{1-\sqrt{s}}{\sqrt{1+s}}\phi_1^{(s)}\right]$$

Proof A combination of (D.1), and Lemmas D.1 and D.2. □

By doing some asymptotic analysis of the constants from Theorem D.1 we can show

$$\begin{aligned} |\tau_{k,k}| &\leq n + 2s \\ |\tau_{k,-k}| + |\tau_{k,k^\vee}| + |\tau_{k,-k^\vee}| + |\tau_{k,k^\wedge}| + |\tau_{k,-k^\wedge}| &\leq 1 + \tfrac{n}{2} + s + \tfrac{s}{2} + \tfrac{n}{2} + s + \tfrac{1}{2} + \tfrac{s}{4} \\ &= n + 3s + 2 \end{aligned}$$

Thus we can establish the following upper bound.

Theorem D.2 *Let S be the $N \times N$ stiffness matrix for the functions $\phi_k^{(s)}$. Then*

$$\lambda_{\max}(S) \leq N + 5s,$$

where $\lambda_{\max}(\cdot)$ denotes the maximum eigenvalue (in magnitude).

Appendix E

Lebesgue-Optimal Values for α

In Chapter 4 we numerically determined the optimal values of α that minimize the Lebesgue constant for one-dimensional interpolation using the symmetric Jacobi Gauss and Gauss-Lobatto nodal points. We tabulate in table E.1 these numerically-computed optimal values of α with significant digits commensurate to their accuracy.

In addition, we developed an algorithm in Section 4.4 for explicit determination of interpolation sets on the two-dimensional triangular simplex. Like the algorithm's predecessor in [66], we rely on an ad-hoc method to cluster the nodes towards the vertices of the simplex. This clustering method requires specification of a scalar γ, which is calculated numerically to minimize the Lebesgue Constant. In Table E.2 we present the values of the scalar γ versus N, the polynomial order on each edge.

N	α^* – GL	α^* – GQ	N	α^* – GL	α^* – GQ	N	α^* – GL	α^* – GQ
			51	0.46795881	-0.54991629	101	0.47296789	-0.53860286
			52	0.46813095	-0.54952619	102	0.47302842	-0.53846907
			53	0.46829347	-0.54915144	103	0.47308739	-0.53833781
4	0.36451576	-0.72484980	54	0.46845540	-0.54878472	104	0.47314620	-0.53820796
5	0.39073923	-0.68742252	55	0.46860868	-0.54843184	105	0.47320352	-0.53808051
6	0.41126533	-0.66018589	56	0.46876139	-0.54808623	106	0.47326069	-0.53795440
7	0.42073672	-0.64366644	57	0.46890628	-0.54775316	107	0.47331645	-0.53783058
8	0.42931116	-0.63020050	58	0.46905064	-0.54742672	108	0.47337205	-0.53770804
9	0.43422334	-0.62069456	59	0.46918791	-0.54711167	109	0.47342631	-0.53758767
10	0.43889222	-0.61251768	60	0.46932468	-0.54680266	110	0.47348043	-0.53746853
11	0.44193695	-0.60624743	61	0.46945498	-0.54650405	111	0.47353326	-0.53735145
12	0.44489274	-0.60068143	62	0.46958482	-0.54621097	112	0.47358595	-0.53723554
13	0.44699052	-0.59618598	63	0.46970875	-0.54592741	113	0.47363743	-0.53712161
14	0.44904847	-0.59211215	64	0.46983224	-0.54564893	114	0.47368876	-0.53700879
15	0.45059765	-0.58870313	65	0.46995031	-0.54537918	115	0.47373893	-0.53689785
16	0.45212605	-0.58556840	66	0.47006797	-0.54511411	116	0.47378897	-0.53678799
17	0.45332713	-0.58287695	67	0.47018065	-0.54485708	117	0.47383789	-0.53667992
18	0.45451598	-0.58037502	68	0.47029293	-0.54460438	118	0.47388669	-0.53657288
19	0.45548112	-0.57818472	69	0.47040063	-0.54435910	119	0.47393442	-0.53646755
20	0.45643834	-0.57613150	70	0.47050796	-0.54411781	120	0.47398203	-0.53636320
21	0.45723542	-0.57430650	71	0.47061103	-0.54388339	121	0.47402862	-0.53626051
22	0.45802694	-0.57258424	72	0.47071376	-0.54365268	122	0.47407509	-0.53615873
23	0.45869955	-0.57103464	73	0.47081256	-0.54342834	123	0.47412058	-0.53605855
24	0.45936800	-0.56956432	74	0.47091102	-0.54320745	124	0.47416596	-0.53595927
25	0.45994550	-0.56822811	75	0.47100583	-0.54299248	125	0.47421039	-0.53586149
26	0.46051975	-0.56695453	76	0.47110033	-0.54278072	126	0.47425472	-0.53576458
27	0.46102269	-0.56578744	77	0.47119142	-0.54257447	127	0.47429815	-0.53566912
28	0.46152301	-0.56467081	78	0.47128222	-0.54237122	128	0.47434147	-0.53557448
29	0.46196628	-0.56364034	79	0.47136984	-0.54217311	129	0.47438393	-0.53548124
30	0.46240735	-0.56265120	80	0.47145719	-0.54197781	130	0.47442628	-0.53538879
31	0.46280195	-0.56173286	81	0.47154157	-0.54178732	131	0.47446781	-0.53529767
32	0.46319470	-0.56084887	82	0.47162567	-0.54159945	132	0.47450923	-0.53520733
33	0.46354904	-0.56002387	83	0.47170700	-0.54141610	133	0.47454986	-0.53511827
34	0.46390178	-0.55922779	84	0.47178808	-0.54123520	134	0.47459039	-0.53502994
35	0.46422234	-0.55848142	85	0.47186655	-0.54105852	135	0.47463015	-0.53494285
36	0.46454150	-0.55775966	86	0.47194478	-0.54088417	136	0.47466982	-0.53485647
37	0.46483342	-0.55708027	87	0.47202056	-0.54071379	137	0.47470874	-0.53477128
38	0.46512409	-0.55642201	88	0.47209611	-0.54054559	138	0.47474758	-0.53468677
39	0.46539146	-0.55580018	89	0.47216935	-0.54038111	139	0.47478571	-0.53460340
40	0.46565770	-0.55519665	90	0.47224238	-0.54021870	140	0.47482374	-0.53452069
41	0.46590382	-0.55462472	91	0.47231325	-0.54005981	141	0.47486109	-0.53443908
42	0.46614894	-0.55406877	92	0.47238390	-0.53990286	142	0.47489836	-0.53435811
43	0.46637655	-0.55354042	93	0.47245249	-0.53974923	143	0.47493496	-0.53427820
44	0.46660324	-0.55302613	94	0.47252089	-0.53959742	144	0.47497149	-0.53419890
45	0.46681460	-0.55253610	95	0.47258735	-0.53944877	145	0.47500736	-0.53412062
46	0.46702511	-0.55205850	96	0.47265363	-0.53930185	146	0.47504316	-0.53404294
47	0.46722210	-0.55160237	97	0.47271806	-0.53915789	147	0.47507834	-0.53396624
48	0.46741831	-0.55115731	98	0.47278232	-0.53901558	148	0.47511345	-0.53389012
49	0.46760252	-0.55073135	99	0.47284484	-0.53887608	149	0.47514797	-0.53381495
50	0.46778603	-0.55031528	100	0.47290719	-0.53873813			

Table E.1: Tabulated values of the numerically-computed optimum α values. GL – Gauss-Lobatto quadrature (boundary nodal sets); GQ – Gauss quadrature (interior nodal sets).

N	γ
3	2.295
4	3.788
5	1.839
6	2.022
7	2.075
8	2.111
9	2.110
10	2.139
11	2.116
12	2.133
13	2.107
14	2.107
15	2.089

Table E.2: Tabulated values of the stretch paramter γ necessary for calculation of the nodal sets presented in Section 4.4.

Bibliography

[1] M. Abramowitz and I. Stegun. *Handbook of mathematical functions*. Dover, 1972.

[2] B.K. Alpert and V. Rokhlin. A fast algorithm for the evaluation of Legendre expansions. *SIAM Journal on Scientific Computing*, 12(1):158–179, 1991.

[3] J.R. Angelos, R.H. Kaufman Jr., M.S. Henry, and T.D. Lenker. *Optimal nodes for polynomial interpolation*, pages 17–20. Academic Press, New York, 1989.

[4] R. Askey. Jacobi polynomial expansions with positive coefficients and imbeddings of projective spaces. *Bulletin of the American Mathematical Society*, 74:301–304, 1968.

[5] R. Askey. *Orthogonal Polynomials and Special Functions*. SIAM, 1975.

[6] G. Ben-yu and Z. Xiao-yong. Spectral method for differential equations of degenerate type on unbounded domains by using generalized Laguerre functions. *Applied Numerical Mathematics*, 57:455–471, 2007.

[7] G. Ben-Yu and W. Zhong-Qing. Modified Chebyshev rational spectral method for the whole line. In *Proceedings of the fourth International Conference on Dynamical Systems and Differential Equations*, pages 365–374, 2002.

[8] C. Bernardi and Y. Maday. *Spectral Methods*, volume 5, pages 209 – 485. Elsevier Science, 1997.

[9] C. De Boor. Proof of the conjectures of Bernstein and Erdös concerning the optimal nodes for polynomial interpolation. *Journal of Approximation Theory*, 24:289–303, 1978.

[10] J.P. Boyd. The optimization of convergence for Chebyshev polynomial methods in an unbounded domain. *Journal of computational physics*, 45:43–79, 1982.

[11] J.P. Boyd. Orthogonal rational functions on a semi-infinite interval. *Journal of Computational Physics*, 70:63–88, 1987.

[12] J.P. Boyd. Spectral methods using rational basis functions on an infinite interval. *Journal of Computational Physics*, 69:112–142, 1987.

[13] J.P. Boyd. Chebyshev domain truncation is inferior to Fourier domain truncation for solving problems on an infinite interval. *Journal of Scientific Computing*, 3(2):109 – 120, 1988.

[14] J.P. Boyd. *Chebyshev and Fourier spectral methods*, volume 49 of *Lecture notes in engineering*. Springer-Verlag, 1989.

[15] J.P. Boyd. The orthogonal rational functions of Higgins and Christov and algebraically mapped Chebyshev polynomials. *Journal of Approximation Theory*, 61:98–105, 1990.

[16] J.P. Boyd. Multipole expansions and pseudospectral cardinal functions: a new generalization of the fast Fourier transform. *Journal of Computational Physics*, 103:184–186, 1992.

[17] J.P. Boyd. A numerical comparison of seven grids for polynomial interpolation on the interval. *Computers and Mathematics with Applications*, 38:35–50, 1999.

[18] J.P. Boyd. Trouble with Gegenbauer reconstruction for defeating Gibbs phenomenon: Runge phenomenon in the diagonal limit of Gegenbauer polynomial approximations. *Journal of Computational Physics*, 204:253–264, 2005.

[19] C.K. Bridsall and A.B. Langdon. *Plasma physics via computer simulation*. Taylor and Francis, first edition, 2004.

[20] A. Bultheel, P. Gonzalez-Vera, E. Hendriksen, and O. Njastad. *Orthogonal Rational Functions*. Cambridge University Press, 1999.

[21] C. Canuto and A. Quarteroni. Approximation results for orthogonal polynomials in sobolev spaces. *Mathematics of Computation*, 38(157):67–86, 1982.

[22] C. Canuto, A. Quarteroni, M.Y. Hussaini, and T.A. Zang. *Spectral Methods: Fundamentals in Single Domains*. Springer, 2006.

[23] D.I. Caruntu. Classical Jacobi polynomial, closed-form solutions for transverse vibrations. *Journal of Sound and Vibration*, 306:467–494, 2007.

[24] Q. Chen and I. Babuska. Approximate optimal points for polynomial interpolation of real functions in an interval and in a triangle. *Computer Methods in Applied Mechanics and Engineering*, 128:405–417, 1995.

[25] Q. Chen and I. Babuska. Adaptive procedure for approximating functions by continuous piecewise polynomials. *Communcations in Numerical Methods in Engineering*, 12:483–495, 1996.

[26] C.Z. Cheng and G. Knorr. The integration of the Vlasov equation in configuration space. *Journal of Computational Physics*, 22:330–351, 1976.

[27] C.I. Christov. A complete orthonormal system of functions in $l^2(-\infty, \infty)$ space. *SIAM Journal of Applied Math*, 42(6):1337–1344, 1982.

[28] C.W. Clenshaw and A.R. Curtis. A method for numerical integration on an automatic computer. *Numerische Mathematik*, 2:197–205, 1960.

[29] A. Cloot and J.A.C. Weideman. An adaptive algorithm for spectral computations on unbounded domains. *Journal of Computational Physics*, 102:398–406, 1992.

[30] J.R. Driscoll and Jr. D.M. Healy. Computing Fourier transforms and convolutions on the 2-sphere. *Advances in Applied Mathematics*, 15:202–250, 1992.

[31] J.R. Driscoll, Jr. D.M. Healy, and D.N. Rockmore. Fast discrete polynomial transforms with applications to data analysis for distance transitive graphs. *SIAM Journal of Scientific Computing*, 26(4):1066–1099, 1997.

[32] W. Gautschi. Orthogonal polynomials and quadrature. *Electronic Transactions on Numerical Analysis*, 9:65–76, 1999.

[33] Walter Gautschi. Gauss-Radau formulae for Jacobi and Laguerre weight functions. *Mathematics and Computers in Simulations*, 54(4-5):403–412, 2000.

[34] Walter Gautschi. *Orthogonal Polynomials: Computation and Approximation*. Numerical Mathematics and Scientific Computation. Oxford University Press, 2004.

[35] G.H. Golub and J.H. Welsch. Calculation of Gauss quadrature rules. *Mathematics of Computation*, 23(106):221–230, 1969.

[36] D. Gottlieb and S.A. Orszag. *Numerical analysis of spectral methods: theory and applications*. CBMS-NSF Regional Conference Series in Applied Mathematics. SIAM, 1977.

[37] D. Gottlieb and C.-W. Shu. On the Gibbs phenomenon iii: recovering exponential accuracy in a sub-interval from a spectral partial sum of a piecewise analytic function. *SIAM Journal on Numerical Analysis*, 33(1):280–290, 1996.

[38] D. Gottlieb and C.-W. Shu. On the Gibbs phenomenon and its resolution. *SIAM Review*, 29(4):644–668, 1997.

[39] C.E. Grosch and S.A. Orszag. Numerical solution of problems in unbounded regions: coordinate transforms. *Journal of Computational Physics*, 25:273–296, 1977.

[40] R. Gunttner. On asymptotics for the uniform norms of the Lagrange interpolation polynomials corresponding to extended Chebyshev nodes. *SIAM Journal of Numerical Analysis*, 25:461–469, 1988.

[41] Wilhelm Heinrichs. Improved Lebesgue constants on the triangle. *Journal of Computational Physics*, 207:625–638, 2005.

[42] J.S. Hesthaven. From electrostatics to almost optimal nodal sets for polynomial interpolation in a simplex. *SIAM Journal of Numerical Analysis*, 35(2):655–676, 1998.

[43] J.S. Hesthaven, S. Gottlieb, and D. Gottlieb. *Spectral methods for time-dependent problems*. Monographs on applied and computational mathematics. Cambridge, 2007.

[44] J.S. Hesthaven and T.W. Warburton. *Nodal discontinuous Galerkin methods: algorithms, analysis, and applications*. Springer, 2008.

[45] J.R. Higgins. *Completeness and basis properties of sets of special functions*. Cambridge University Press, 1977.

[46] G.B. Jacobs and J.S. Hesthaven. High-order nodal discontinuous galerkin particle-in-cell method on unstructured grids. *Journal of Computational Physics*, 214(1):96–121, 2005.

[47] T.A. Kilgore. A characterization of the Lagrange interpolation projection with minimal Tchebysheff norm. *Journal of Approximation Theory*, 24:273–288, 1978.

[48] A.J. Klimas. A method for overcoming the velocity space filamentation problem in collisionless plasma model simulations. *Journal of Computational Physics*, 68:202–226, 1987.

[49] H. Li and H. Ma. Shifted Chebyshev colloation domain truncation for solving problems on an infinite interval. *Journal of Scientific Computing*, 18(2):191 – 213, 2003.

[50] J. Lund and K.L. Bowers. *Sinc Methods for Quadrature and Differential Equations*. SIAM, 1992.

[51] P. Maroni and Z. da Rocha. Connection coefficients between orthogonal polynomials and the canonical sequence: an approach based on symbolic computation. *Numerical Algorithms*, 47:291–314, 2008.

[52] T.M. Mills and S.J. Smith. The Lebesgue constant for Lagrange interpolation. *Numerische Mathematik*, 61:111–115, 1992.

[53] G.V. Milovanovic and A.S. Cvetkovic. Gauss-Radau and Gauss-Lobatto interval quadrature rules for Jacobi weight function. *Numerische Mathematik*, 102:523–542, 2006.

[54] S.A. Orszag. Fast eigenfunction transforms. In G.C. Rota, editor, *Science and Computers, Advances in Mathematics Supplementary Studies*, pages 23–30. Academic Press, 1986.

[55] V. Pan, A. Sadikou, E. Landowne, and O. Tiga. A new approach to fast polynomial interpolation and multipoint evaluation. *Computers and mathematics with applications*, 25, 1993.

[56] M.J. Roth. *Nodal configurations and Voronoi tessellations for triangular spectral elements*. PhD thesis, University of Victoria, 2005.

[57] J.C. Schatzman. Accuracy of the discrete Fourier transform and the fast Fourier transform. *Journal of Scientific Computing*, 17(5):1150–1166, 1996.

[58] J. Shen. Stable and efficient spectral methods in unbounded domains using Laguerre functions. *SIAM Journal of Numerical Analysis*, 38(4):1113–1133, 2000.

[59] M. Sugihara. Near optimality of the sinc approximation. *Mathematics of Computation*, 72(242):767–780, 2002.

[60] Gabor Szego. *Orthogonal Polynomials*. AMS Colloquium Publications, volume XXIII. American Mathematical Society, 1975.

[61] T. Tang. The Hermite spectral method for Gaussian-type functions. *SIAM Journal of Scientific Computing*, 14(3):594–606, 1993.

[62] D. Tchiotsop, D. Wolf, V. Louis-Dorr, and R. Husson. ECG data compression using Jacobi polynomials. In *Proceedings of the 29th annual international conference of the IEEE EMBS*, pages 1863 – 1867, 2007.

[63] L.N. Trefethen. Is Gauss quadrature better than clenshaw-curtis? *SIAM Review*, 50(1):67–87, 2008.

[64] H. Triebel. *Interpolation theory – function spaces – differential operators*. Johann Ambrosius Barth, 1995.

[65] P. Vertresi. Optimal Lebesgue constant for Lagrange interpolation. *SIAM Journal on Numerical Analysis*, 27(5):1322–1331, 1990.

[66] T. Warburton. An explicit construction of interpolation nodes on the simplex. *Journal of Engineering Mathematics*, 56:247–262, 2006.

[67] J.A.C. Weidemann. The eigenvalues of Hermite and rational spectral differentiation matrices. *Numerische Mathematik*, 61:409–431, 1992.

[68] E.T. Whittaker. On the functions which are represented by the expansions of the interpolation theory. *Proceedings of the Royal Society of Edinburgh*, 35:181–194, 1915.

[69] Norbert Wiener. *Extrapolation, interpolation, and smoothing of stationary time series*. MIT Technology Press and John Wiley & Sons, 1949.

[70] Anton Zettl. *Sturm-Liouville Theory*. Mathematical Surveys and Monographs, volume 121. American Mathematical Society, 2005.

[71] W. Zhong-Qing and G. Ben-Yu. A rational approximation and its applications to nonlinear partial differential equations on the whole line. *Mathematical Analysis and Applications*, 274:374–403, 2002.