The Application of the Fast Fourier Transform to Jacobi Polynomial expansions

by Akil C. Narayan and Jan S. Hesthaven

Abstract

We present an algorithm for transforming modal coefficients of one Jacobi Polynomial class to the modal coefficients of another class. This transformation is invertible and is efficient for certain pairs of Jacobi Polynomials classes. When one of the classes corresponds to the Chebyshev case, the Fast Fourier Transform can be used to very quickly compute modal coefficients for a wide variety Jacobi Polynomial expansions. Numerical results are presented that illustrate the computational and accuracy advantage of our method over standard quadrature methods

1 Introduction

The classical Jacobi Polynomials $P_n^{(\alpha,\beta)}$ are a family of orthogonal polynomials [10] that have been used extensively in many applications for their ability to approximate general classes of functions. They are a class of polynomials that encompasses the Chebyshev, Legendre, and Gegenbauer/ultraspheric polynomials. In addition, they have a very close connection to the Associated Legendre functions that are widely used in the Spherical Harmonic expansions.

Jacobi polynomial expansions have been used in a variety of applications, some of which are the resolution of the Gibbs' phenomenon [6], electrocardiogram data compression [11], and the solution to differential equations cite. 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 paper, we show that for a variety of classes of Jacobi polynomials, one can use the FFT to perform spectral transformations.

In section 2 we give an overview of Jacobi polynomials and the relevant properties needed for our discussion. Section 3 is devoted to a theoretical description of the method and includes most of the major results. Section 4 is a special application of the results from section 3 to Chebyshev-like systems where the FFT may be exploited. Finally, section 5 gives some numerical examples.

2 Jacobi Polynomials

For a comprehensive treatement of Jacobi polynomials and their properties, [10], [4], and [1] prove to be excellent references. All properties and results in this section are taken from these references. Jacobi polynomials are one of the two linearly independent solutions to the linear, second-order, singular Sturm-Liouville differential equation

$$-\frac{\mathrm{d}}{\mathrm{d}r}\left[(1-r)^{\alpha+1}(1+r)^{\beta+1)}\frac{\mathrm{d}\rho}{\mathrm{d}r}\right]-n(n+\alpha+\beta+1)(1-r)^{\alpha}(1+r)^{\beta}\rho=0, \qquad r\in[-1,1]. \tag{2.1}$$

The parameters α, β are restricted to take real values in the interval $(-1,\infty)$. The monic Jacobi polynomial of order n, as a solution to (2.1) is written as $P_n^{(\alpha,\beta)}(x)$. We define the space $\mathcal{B}_N=\mathrm{span}\{x^n\colon 0\le n\le N\}$. For any α, $\beta>-1$, they are orthogonal under a weighted L^2 inner product:

$$\int_{-1}^{1} P_m^{(\alpha,\beta)}P_n^{(\alpha,\beta)}(1-r)^{\alpha}(1+r)^{\beta}\,\mathrm{d}r=h_n^{(\alpha,\beta)}\delta_{m,n}, \tag{2.2}$$

where $\delta_{m,n}$ is the Kronecker delta function, and $h_n^{(\alpha,\beta)}$ is a normalization constant given in the appendix, equation (A.1). We take the integral (2.2) in the Lebesgue sense. We define the weight function

$$\omega^{(\alpha,\beta)}(r) = (1-r)^{\alpha}\,(1+r)^{\beta}, \tag{2.3}$$

and denote angled brackets $\langle\,\cdot\,,\cdot\,\rangle_{(\alpha,\beta)}$ as the inner product of equation (2.2). I.e.,

$$\langle f, g\rangle_{(\alpha,\beta)} = \int_{-1}^{1} f(r)\,g(r)\,\omega^{(\alpha,\beta)}\,\mathrm{d}r.$$

This inner product induces a norm $\|\cdot\|_{(\alpha,\beta)}$ on the space $L^2_{(\alpha,\beta)} := \big\{f\colon[-1,1]\to\mathbb{R}\colon f$ measurable, $\|f\|_{(\alpha,\beta)}<\infty\big\}$, where 'measurable' means with respect to Lebesgue measure. The Jacobi polynomials of class (α,β) are complete and orthogonal in $L^2_{(\alpha,\beta)}$.

The monic Jacobi polynomials satisfy various relations and can be expressed in terms of the Hypergeometric function, and also satisfy a generalized Rodrigues relation for all $m\le n$:

$$\binom{2n+\alpha+\beta}{n}\omega^{(\alpha,\beta)}\,P_n^{(\alpha,\beta)} = \frac{(-1)^m}{2^{m-n}(n-1)\cdots(n-m+1)}\,\frac{\mathrm{d}^m}{\mathrm{d}x^m}\Big[\omega^{(\alpha+m,\beta+m)}\,P_{n-m}^{(\alpha+m,\beta+m)}\Big].$$

Although the formulae for the monic orthogonal polynomials are often easier to write down than those for other normalizations, we shall prefer to work with the $L^2_{(\alpha,\beta)}$-normalized polynomials. To this end, we define

$$\tilde{P}_n^{(\alpha,\beta)} = \frac{P_n^{(\alpha,\beta)}}{\sqrt{h_n^{(\alpha,\beta)}}},$$

which are orthonormal under the weight $\omega^{(\alpha,\beta)}$.

All orthogonal polynomials 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):

$$(1-r)\,\tilde{P}_n^{(\alpha,\beta)} = \mu_{n,0}^{(\alpha,\beta)}\,\tilde{P}_n^{(\alpha-1,\beta)} - \mu_{n,1}^{(\alpha,\beta)}\tilde{P}_{n+1}^{(\alpha-1,\beta)}, \tag{2.4}$$

$$(1+r)\tilde{P}_n^{(\alpha,\beta)} = \mu_{n,0}^{(\beta,\alpha)}\,\tilde{P}_n^{(\alpha,\beta-1)} + \mu_{n,1}^{(\beta,\alpha)}\tilde{P}_{n+1}^{(\alpha,\beta-1)}, \tag{2.5}$$

$$\tilde{P}_n^{(\alpha,\beta)} = \nu_{n,0}^{(\alpha,\beta)}\tilde{P}_n^{(\alpha+1,\beta)} - \nu_{n,-1}^{(\alpha,\beta)}\tilde{P}_{n-1}^{(\alpha+1,\beta)}, \tag{2.6}$$

$$\tilde{P}_n^{(\alpha,\beta)} = \nu_{n,0}^{(\beta,\alpha)}\tilde{P}_n^{(\alpha,\beta+1)} + \nu_{n,-1}^{(\beta,\alpha)}\tilde{P}_{n-1}^{(\alpha,\beta+1)}, \tag{2.7}$$

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. The formulae (2.4)-(2.7) are the main ingredients for our results.

Lastly we cover the spectral expansion of a function $f(r)$ in Jacobi polynomials. For any $f\in L^2_{(\alpha,\beta)}$ we define the modal coefficients

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

We also have a Parseval-type relation:

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

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 define the projection operator

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

Due to completeness and orthogonality, this operator satisfies the relations

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

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

3 Jacobi-Jacobi transformations

The purpose of the 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 very quickly and with very little effort. Of course, the speed of this transformation comes with a natural price: this can only be done for certain values of $|\delta - \beta|$ and $|\gamma - \alpha|$.

We begin by proving a lemma that is an inductive application of equations (2.4) and (2.5):

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

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

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

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:

$$\tilde{P}_n^{(\alpha,\beta)} = \sum_{m=0}^{A} N_{m,n}^{(\alpha,A,\beta)} \tilde{P}_{n-m}^{(\alpha+A,\beta)} \tag{3.3}$$

$$\tilde{P}_n^{(\alpha,b)} = \sum_{m=0}^{B} (-1)^m N_{m,n}^{(\beta,B,\alpha)} \tilde{P}_{n-m}^{(\alpha,\beta+B)} \tag{3.4}$$

Proof. The proofs of (3.1) and (3.2) are accomplished via the demotion relations (2.4) and (2.5): repeated application of (2.4) to the left-hand side of (3.1), and repeated application of (2.5) to the left-hand side of (3.2) yields the desired result.

The relations (3.3) and (3.4) are proven in exactly the same fashion using the promotion relations (2.6) and (2.7). □

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 for us to derive such formulae in terms of the $\mu_{n,i}^{(\alpha,\beta)}$ and $\nu_{n,i}^{(\alpha,\beta)}$, but it is relatively complicated and of little value for us. The main use of lemma 3.1 is in the proof of the following theorem:

Theorem 3.2.
For $A, B \in \mathbb{N}_0$ and $\alpha, \beta > -1$, let $f \in L^2_{(\alpha,\beta)}$. Then

$$\hat{f}_n^{(\alpha+A,\beta+B)} = \sum_{m=0}^{A+B} \Lambda_{m,n}^{(\alpha,A,\beta,B)} \hat{f}_{n+m}^{(\alpha,\beta)}, \qquad n \geq 0 \tag{3.5}$$

for some constants $\Lambda_{m,n}^{(\alpha,A,\beta,B)}$.

Proof. We first 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 $\Lambda_{m,n}^{(\alpha,A,\beta,B)}$ such that

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

Following this we note that $\hat{f}_n^{(\alpha,\beta)}$ is well-defined because f has membership 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

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

□

In other words, theorem 3.2 says that we can express modes of higher-class Jacobi expansions in terms of linear combinations of lower-class Jacobi expansions. Roughly speaking, in order to convert an expansion of class (α_0, β_0) to one of class (α_1, β_1), we require about $O(\alpha_1 - \alpha_0 + \beta_1 - \beta_0)$ computations, as long as $\alpha_1 - \alpha_0$ and $\beta_1 - \beta_0$ are both natural numbers.

The above result is likely known by many authors in this field due to the voluminous literature on Jacobi polynomials. However, we have not seen it written down explicitly, and so we present the above theorem. As in lemma 3.1, we do not present explicit formulae for the constants $\Lambda_{m,n}^{(\alpha,A,\beta,B)}$. Instead, we present the following algorithm for transforming N (α, β) modes into N $(\alpha+A, \beta+B)$ modes.

3.1 Computing the transformation

We collect the modes $\left\{\hat{f}_n^{(\alpha,\beta)}\right\}_n$ into the vector $\boldsymbol{\hat{f}}^{(\boldsymbol{\alpha},\boldsymbol{\beta})}$. Theorem 3.2 makes it clear that there exists an $N \times (N+A+B)$ matrix $\hat{R}$, dependent on $\alpha, \beta, A,$ and B such that

$$\boldsymbol{\hat{f}}^{(\boldsymbol{\alpha}+\boldsymbol{A},\boldsymbol{\beta}+\boldsymbol{B})} = \hat{R}\boldsymbol{\hat{f}}^{(\boldsymbol{\alpha},\boldsymbol{\beta})}. \tag{3.6}$$

This relation transports $N + A + B$ modes from an expansion in the polynomials $\tilde{P}^{(\alpha,\beta)}$ to N modes from an expansion in $\tilde{P}^{(\alpha+A,\beta+B)}$. We do not (yet) know the explicit entries of the matrix $\hat{R}$, but we do know that $\hat{R}$ is a sparse matrix, banded upper-triangular, with about $(A+B)(N+A+B)$ non-zero entries. In order to construct the matrix $\hat{R}$, essentially write out the proof of lemma 3.1 via induction. To this end, we define the following set of matrices: let $U^{(\alpha,\beta)}$ and $V^{(\alpha,\beta)}$ be sparse bidiagonal matrices with entries defined by

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

$$\begin{array}{rcl} U^{(\alpha,\beta)}_{N+A+B,N+A+B} & = & \mu^{(\alpha,\beta)}_{n,0} \\ V^{(\alpha,\beta)}_{N+A+B,N+A+B} & = & \mu^{(\beta,\alpha)}_{n,0} \end{array}$$

The matrix $U^{(\alpha,\beta)}$ transforms the modes of an $(\alpha-1,\beta)$ expansion to those of an (α,β) expansion whenever $\alpha>0$. Simiarly, $V^{(\alpha,\beta)}$ 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 (2.4) and (2.5). 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, as we will see later, we can actually characterize this corruption.

We can now define a square $(N+A+B)\times(N+A+B)$ matrix R as

$$R=\prod_{b=1}^{B} V^{(\alpha+A,\beta+b)} \prod_{a=1}^{A} U^{(\alpha+a,\beta)} \tag{3.7}$$

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

Proposition 3.3.
The matrix R defined by equation (3.7) is invertible and positive-definite.

Proof. R is upper-triangular, and thus the diagonal entries are the eigenvalues. By utilizing equation (3.7) and expressions (A.2) in the appendix, we have, for all $n=1,2,\dots,N+A+B$

$$R_{n,n} \;=\; \prod_{b=1}^{B} \mu^{(\beta+b,\alpha+A)}_{n,0} \prod_{a=1}^{A} \mu^{(\alpha+a,\beta)}_{n,0} \;>\; 0$$

□

Proposition 3.3 tells us that the matrix R is invertible, which means we can travel back and forth between the modes $\hat{f}_n^{(\alpha+A,\beta+B)}$ and $\hat{f}_n^{(\alpha,\beta)}$. In practice, this can be accomplished via back-substitution because R is upper-triangular. The back substitution has a sequential computational cost $O(N(A+B))$, similar to the cost of the formation of the matrix.

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

However, in contrast to the relation presented in (3.6), we would like an invertible transformation. The reason is that we will eventually use this transform as an ingredient in a modal/nodal transformation, so we'd like an invertible transformation for this. The matrix $\hat{R}$ is not invertible (it's not even square), but of course R is invertible, so we shall frequently use that matrix. (As mentioned before, one should be aware that the last $A+B$ modes are not the true modes, but some version of them.)

We note that the constants $\Lambda_{m,n}^{(\alpha,A,\beta,B)}$ in theorem (3.2) are the entries of the matrix R. Thus, we have developed an inexpensive algorithm to compute these constants, even if we don't give explicit formulae for them.

3.2 Properties of the transformation

In the following, we will replace all instances of $N+A+B$ with just N. Indeed this is a trivial thing to do by simply augmenting the value of N accordingly.

Given that the last $A+B$ modes $\left\{\hat{f}_n^{(\alpha+A,\beta+B)}\right\}_{n=N-A-B}^{N-1}$ are not the true modes, we might wonder how the expansion

$$f^{(\alpha+A,\beta+B)}(r) = \sum_{n=0}^{N-1} \hat{f}_n^{(\alpha+A,\beta+B)} \tilde{P}_n^{(\alpha+A,\beta+B)}(r) \tag{3.8}$$

compares to the original expansion

$$f^{(\alpha,\beta)}(r) = \sum_{n=0}^{N-1} \hat{f}_n^{(\alpha,\beta)} \tilde{P}_n^{(\alpha,\beta)}(r), \tag{3.9}$$

assuming that the modes $\hat{f}_n^{(\alpha,\beta)}$ and $\hat{f}_n^{(\alpha+A,\beta+B)}$ are related by (3.7). It is not trivial, but also not surprising, that these expansions are exactly the same function:

Proposition 3.4.
Let the modes $\left\{\hat{f}_n^{(\alpha,\beta)}\right\}_{n=0}^{N-1}$ be given, and let the modes $\left\{\hat{f}_n^{(\alpha+A,\beta+B)}\right\}_{n=0}^{N-1}$ be derived via the matrix R. Then the expansions $f^{(\alpha,\beta)}$ and $f^{(\alpha+A,\beta+B)}$ given by equations (3.9) and (3.8), respectively, are equal.

Proof. Without loss, we take $\hat{f}_n^{(\alpha,\beta)} = \hat{f}_n^{(\alpha+A,\beta+B)} = 0$ for all $n \geq N$. We first determine the expansion coefficients of the given function $f^{(\alpha,\beta)} \in \mathcal{B}_N$ in the space $L^2_{(\alpha+A,\beta+B)}$ with the functions $\tilde{P}_n^{(\alpha+A,\beta+B)}$:

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

Therefore, since $f^{(\alpha,\beta)} \in \mathcal{B}_N$ and by the definition of $\hat{f}^{(\alpha+A,\beta+B)}$ in relation (3.5):

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

□

4 Exploiting the FFT

Up until this point we have not discussed any real implementation issues: it has all been theory. In this section we explore the use of the FFT for performing Jacobi polynomial spectral transformations. To facilitate discussions, we must introduce more notation.

4.1 Quadrature and interpolation

All the results in this section are well-known and we point to the given references for details.

We shall call an N-point quadrature rule Gaussian under a particular weight function $\omega^{(\alpha,\beta)}$ if it exactly integrates any polynomial in the space $\mathcal{B}_{2N}$ under the weight $\omega^{(\alpha,\beta)}$. Such a quadrature rule always exists and is unique if the weight function is positive. A quadrature rule is defined by N nodes and weights $\{r_n, w_n\}_{n=1}^N$ and we shall write the Gaussian quadrature rule evaluation under weight (α,β) as

$$\int_{-1}^{1} f(r)\, \omega^{(\alpha,\beta)}\, \mathrm{d}r \;\simeq\; Q_N^{(\alpha,\beta)}[f] \;:=\; \sum_{n=1}^{N} f(r_q)\, w_q.$$

With this notation, we can write the definition of a Gaussian quadrature rule as one that satisfies

$$Q_N^{(\alpha,\beta)}[r^n] = \int_{-1}^{1} r^n\, \omega^{(\alpha,\beta)}\, \mathrm{d}r, \qquad n \leq 2N-1.$$

We recall a fundamental result due to Golub and Welsch [5]: the determination of the nodes and weights $\{r_n, w_n\}$ can be accomplished via the computation of the eigenvalues and eigenvectors of a symmetric tridiagonal matrix. A more efficient way is to just compute the eigenvalues (which are the nodes r_n) and then use known formulae [3] to compute the weights w_n. This brings the cost of computation down to $O(N^2)$ operations. We refer to the nodes and weights corresponding to the Gaussian quadrature rule $Q_N^{(\alpha,\beta)}[\,\cdot\,]$ as $\left\{r_n^{(\alpha,\beta)}, w_n^{(\alpha,\beta)}\right\}$.

In many computations one 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 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* [8], 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 $\left\{\hat{f}_n^{(\alpha+A,\beta+B)}\right\}_{n=0}^{N-1}$:

$$\tilde{f}_{n;\mathrm{GQ}}^{(\alpha+A,\beta+B)} = Q_N^{(\alpha+A,\beta+B)}\left[f \tilde{P}_n^{(\alpha+A,\beta+B)}\right] \tag{4.1}$$

$$\tilde{f}_{n;R}^{(\alpha+A,\beta+B)} = \sum_{m=0}^{A+B} \Lambda_{m,n}^{(\alpha,A,\beta,B)} \tilde{f}_{n+m;\mathrm{GQ}}^{(\alpha,\beta)} \tag{4.2}$$

$$\tilde{f}_{n;\mathrm{DQ}}^{(\alpha+A,\beta+B)} = Q_N^{(\alpha,\beta)}\left[f \tilde{P}_n^{(\alpha+A,\beta+B)} w^{(A,B)}\right] \tag{4.3}$$

The modes $\tilde{f}_{n;\mathrm{GQ}}$ are obtained using the Gaussian quadrature native to the expansion class $(\alpha + A, \beta + B)$. The modes $\tilde{f}_{n;R}$ are obtained by performing the matrix multiplication in equation (3.6); i.e. the matrix R 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 $\omega^{(\alpha,\beta)}$ to simulate quadrature under the weight $\omega^{(\alpha+A,\beta+B)}$. 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 $f_i^{(\alpha+A,\beta+B)} = \sum_{n=0}^{N-1} \tilde{f}_{n;i}^{(\alpha+A,\beta+B)} \tilde{P}_n^{(\alpha+A,\beta+B)}$, for i = GQ,R,DQ. Since GQ represents a strict Gaussian quadrature, we know that $f_{\mathrm{GQ}}^{(\alpha+A,\beta+B)} = \mathcal{I}_N^{(\alpha+A,\beta+B)} f$. As a corollary of proposition 3.4, we have the following:

Corollary 4.1. $f_R^{(\alpha+A,\beta+B)} = \mathcal{I}_N^{(\alpha,\beta)} f$.

The first two expansions GQ and R 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.

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 $f_n^{(\alpha,\beta)}$ using the Gaussian quadrature native to class (α, β). This amounts to performing the operation in equation (4.1) N times, which is an $O(N^2)$ matrix multiplication.

However, given the results in section 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 R. 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 (FFT) [8]. 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.

In the determination of the expansion coefficients $\tilde{f}_{n;\mathrm{GQ}}$ and $\tilde{f}_{n;R}$, we can characterize four types of computations to be peformed:

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 equation (4.1). This is basically a matrix-vector multiplication requiring $O\left(N^2\right)$ computations.

Overhead Time (R)

- 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 R. This is an $O(N(A+B))$ operation.

Online Time (R)

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

To summarize, the GQ method has $O\left(N^2\right)$ complexity for both the overhead and online times. The R method is about $O(N\log N)$ for both, as long as $A+B$ is not very large. The GQ method produces the modes $\tilde{f}_{n;\mathrm{GQ}}^{(\alpha,\beta)}$, and the R method produces the modes $\tilde{f}_{n;R}^{(\alpha,\beta)}$.

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

5 Numerical examples

In this section we will test the theoretical methods developed in section 3 applied to Chebyshev-Jacobi polynomial expansions $\left(-\frac{1}{2}+A,-\frac{1}{2}+B\right)$ as described in section 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 4, we can compute two spectral expansions of f: $f_{\mathrm{GQ}}^{(\alpha,\beta)}$ and $f_R^{(\alpha,\beta)}$. We split the computer work required into the 'overhead', and 'online' divisions, which are defined in section 4.2. By performing the algorithms delineated in section 4.2, we can measure the time required to perform spectral expansions using the GQ and R methods. The computations for online time are performed using C libraries, and those for overhead time are performed using Python wrapped around C linear algebra subroutines.

In figure 5.1 we show the online times for the GQ and R methods for A and B taking values 5 and 10. We see that the online time for the R method with the FFT is relatively insensitive to N, growing only linearly with N. However, the GQ (quadrature) method grows with N^2 and is more expensive than the R method. The time spent for the R method is split almost evenly between the FFT and application of the matrix R. Application of R becomes the dominant factor when A and B become large.

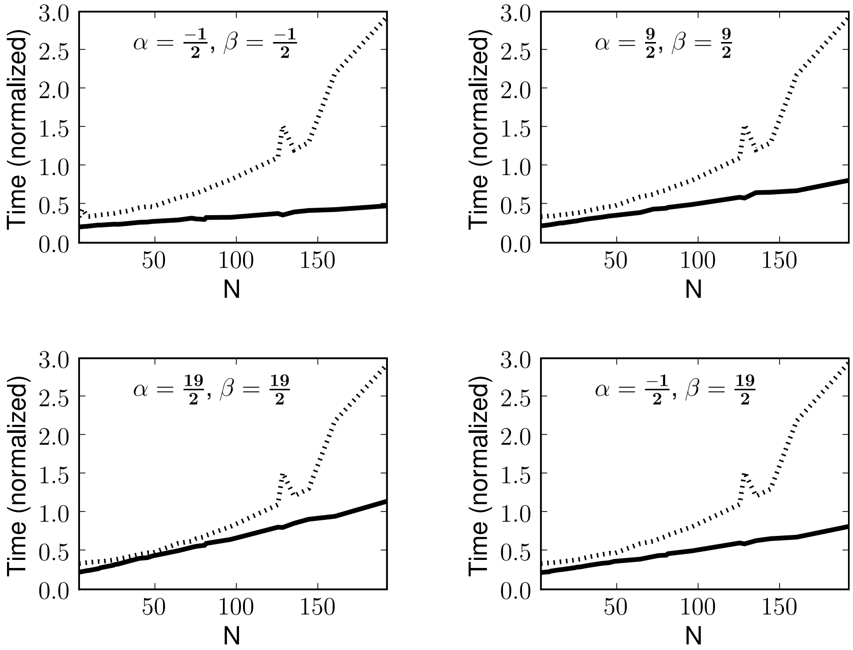

Figure 5.1. Plots of the online times for determining N modal coefficients using N nodal evaluations. FFT+R-matrix calculations are plotted (solid line) vs. quadrature calculations (dashed line) for various expansion classes (α, β).

It is worth that the timings in figure 5.1 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 is undoubtedly faster than a direct application of quadrature, and that this speed advantage remains despite influence by the multiplication by R.

In figure 5.2 we show the same results as in figure 5.1, but we focus on how the parameters A and B affect the online time required for the R 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 matrix R for the R method or the quadrature rule for the GQ method. In figure 5.3 we show these results. We see that even for small N the overhead time required for the R 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 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 $w_n^{(\alpha,\beta)}$, respectively. This requires an additional $O\big(N^2\big)$ 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 equation (4.3) because the Chebyshev-Gauss quadrature rule has an analytic form and is very easy to compute [8]. However, the DQ quadrature becomes more and more inaccurate as A and B are increased. The 'quadrature' overhead time plotted in figure 5.3 is actually that for the DQ method, and not the GQ method.

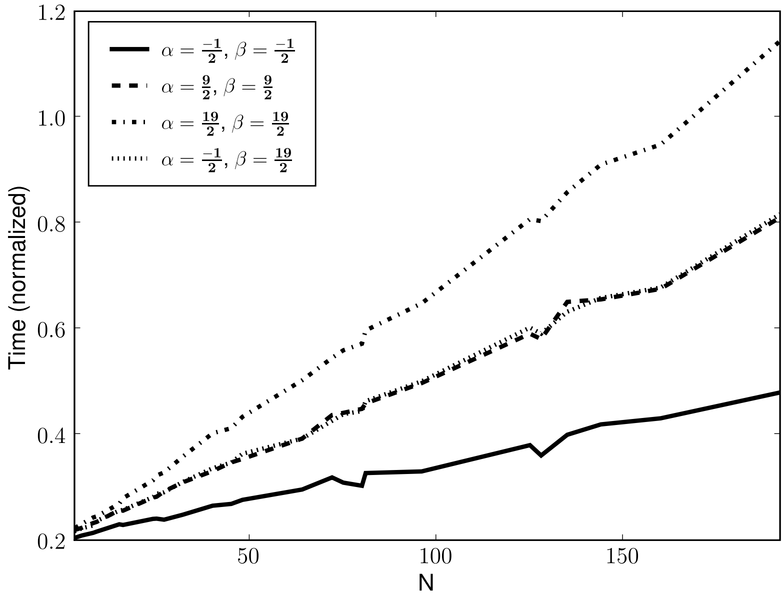

Figure 5.2. Plot of the online computational time required for the R method vs. N. These are the same results as in figure 5.1.

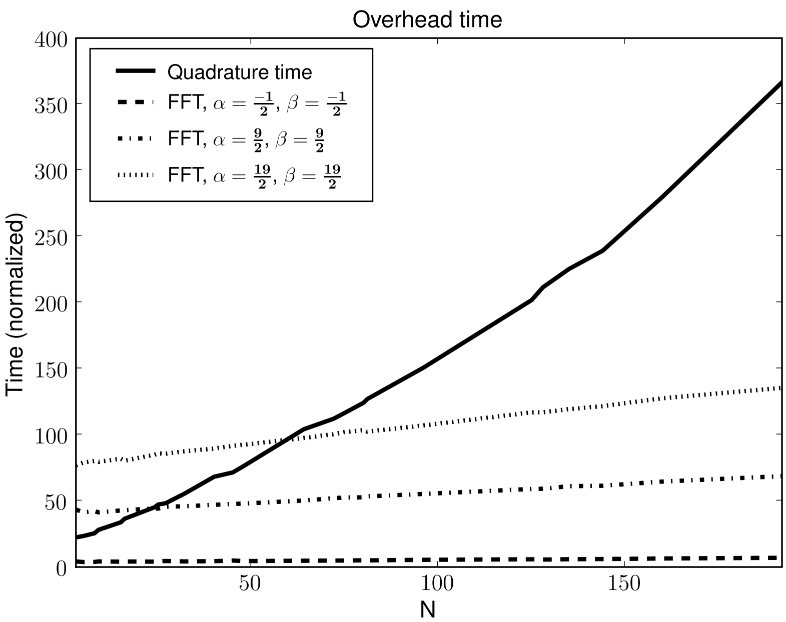

Figure 5.3. Plot of the overhead times vs. number of nodal/modal degrees of freedom N. The overheads are calculated for the f_{GQ} (quadrature) and f_R (FFT) methods.

One final topic we may wish to explore is the accuracy of the expansions. We know from proposition 3.4 that $f_{\mathrm{GQ}} = \mathcal{I}_N^{(\alpha,\beta)}(f)$, and from corollary 4.1 that $f_R = \mathcal{I}_N^{(-1/2,-1/2)}(f)$. It is also well-known that the nodes $\left\{r_n^{(\alpha,\beta)}\right\}_{n=1}^N$ approach the equidistant nodal set as α and β are increased. It is also well-known that the equidistant nodal set for polynomial interpolation has an exponentially growing Lebesgue constant [9], 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 [7], of order $\log(N)$. Therefore, we expect that f_R to be an orders-of-magnitude better pointwise interpolant than f_{GQ} for large α, β.

This suspicion is validated in figure 5.4. We see that for $\alpha = \beta = -\frac{1}{2}$, when f_{GQ} and f_R are the same function, the errors are of order machine precision. When we increase α and β, we see that f_{GQ} develops very bad errors at the boundaries, which is consistent with our observation that this function is an interpolant on nearly-equidistant nodes. However, we also see that the function f_R maintains its pointwise error near machine precision. This is not surprising since we know that for all $(\alpha, \beta) = \left(-\frac{1}{2} + A, -\frac{1}{2} + B\right)$, the function f_R is the the Chebyshev-Gauss interpolant, which has a well-behaved Lebesgue constant.

We remark that all the functions f_R in figure 5.4 are the same function, so we expect all the solid lines in the plot to be identical. We see that this is not the case, which we can attribute to roundoff errors. If one were to expand a function whose spectral coefficients did not decay exponentially, one would see that all the error patterns for f_R are identical, in agreement with proposition 3.4 and corollary 4.1.

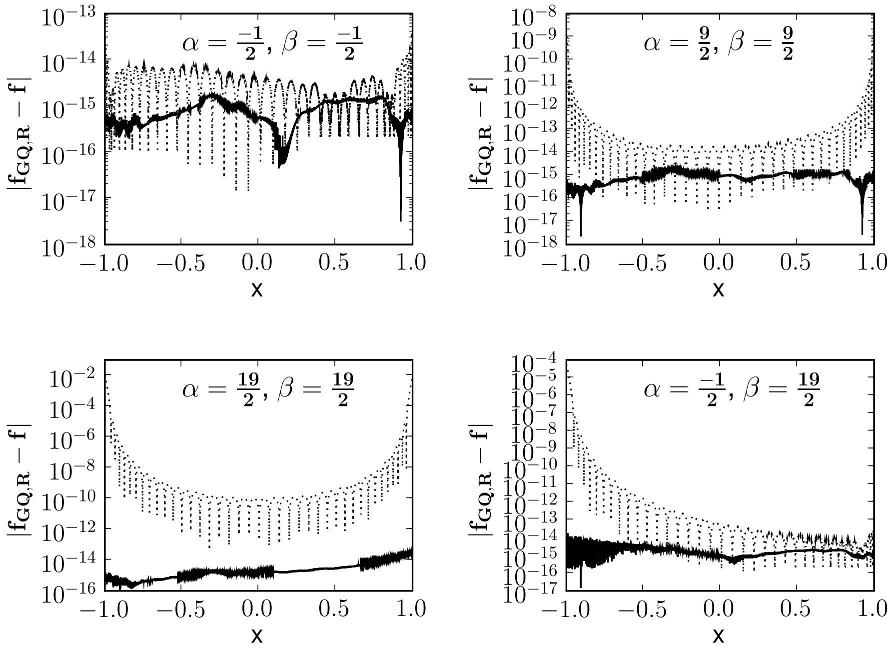

Figure 5.4. Plots of the pointwise errors $|f - f_{\mathrm{GQ}}|$ (dashed line) and $|f - f_R|$ (solid line) with an $N = 40$, class (α, β) modal reconstruction of the FFT approximation f_R and the quadrature approximation f_{GQ}. The grid is a 10^4-point Chebyshev-Gauss set of nodes.

6 Summary

We have presented theory relating the 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 a transformation is possible due to certain equations for promoting and demoting Jacobi class parameters α and β by integral values. This transformation can also be efficiently implemented, transforming N modes in $O(N(A+B))$ 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 to the advantage of speed, we can also claim a victory in accuracy. We have proven that the reconstructed function can be characterized as the Chebyshev-Gauss interpolant. For large A, B, this implies that our reconstructed function is much better behaved than the Jacobi-Gauss $\left(-\frac{1}{2} + A, -\frac{1}{2} + B\right)$ interpolant. From one point of view, the Chebyshev-Gauss interpolant is approximately a filtered version of the Jacobi-Gauss interpolant, wherein a change in the final $A + B$ modes is enough to mold the latter interpolant into the former. On the same topic of accuracy, it is known that using the FFT rather than direct use of direct quadrature is more accurate because fewer calculations overall leads to less roundoff error. This is the reason why f_R is more accurate than f_{GQ} in figure 5.4 for $\alpha = \beta = -\frac{1}{2}$.

Although our numerical examples have concentrated on the Jacobi-Gauss quadrature, the method is equally applicable to the Radau and Lobatto quadratures as well. In addition, fast algorithms for Legendre polynomial expansions also exist [2], which means that this algorithm also yields fast spectral transformations for polynomial expansions in any class of the form (A, B) as well.

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 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 R are nonzero (that is, the first $A + B + 1$ superdiagonals). However, if $A = B$, then the odd superdiagonals are actually zero. This means that an online application of R for Gegenbauer/Ultraspheric expansions can be optimized by a factor of roughly 2, which we have not done in this paper. Because application of R is usually always on the same order of magnitude as application of the FFT, this is 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

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

for the modes $\tilde{f}^{(\alpha, \beta)}$. Then if $\alpha = \beta = -\frac{1}{2}$, one can use the FFT to recover the nodal evaluations.

We provide the Python code used to generate the figures at `http://www.dam.brown.edu/people/anaray/blahblahblah`.

Appendix A Jacobi polynomial properties

This appendix is devoted to providing the formulae necessary to carry out the algorithms presented in this paper. The Jacobi polynomials satisfy recurrence relations (2.4)-(2.7) for the constants $\mu_{n,0/1}^{(\alpha,\beta)}$ and $\nu_{n,0/1}^{(\alpha,\beta)}$. To derive these constants, we note that the usual scaling found for the Jacobi polynomials is that adhering to the criterion

$$\hat{P}_n^{(\alpha,\beta)}(r=1)=\left(\begin{array}{c} n+\alpha \\ n \end{array}\right),$$

where we have introduced the scaled Jacobi polynomials $\hat{P}_n^{(\alpha,\beta)}$, which are the scaled polynomials that usually appear in the literature [1]. From various sources, e.g. [10], we have that these polynomials satisfy promotion and demotion formulae in a form similar to (2.4)-(2.7), but with different constants. With these formulae from the literature, we have that the *monic* Jacobi polynomials $P_n^{(\alpha,\beta)}$ satisfy the same promotion and demotion formulae with the constants:

$$\begin{aligned}
(1-r)P_n^{(\alpha,\beta)} &= \frac{2(n+\alpha)(n+\alpha+\beta)}{(2n+\alpha+\beta)(2n+\alpha+\beta+1)} P_n^{(\alpha-1,\beta)} - P_{n+1}^{(\alpha-1,\beta)} \\
(1+r)P_n^{(\alpha,\beta)} &= \frac{2(n+\beta)(n+\alpha+\beta)}{(2n+\alpha+\beta)(2n+\alpha+\beta+1)} P_n^{(\alpha,\beta-1)} + P_{n+1}^{(\alpha,\beta-1)} \\
P_n^{(\alpha,\beta)} &= P_n^{(\alpha+1,\beta)} - \frac{2n(n+\beta)}{(2n+\alpha+\beta)(2n+\alpha+\beta+1)} P_{n-1}^{(\alpha+1,\beta)} \\
P_n^{(\alpha,\beta)} &= P_n^{(\alpha,\beta+1)} + \frac{2n(n+\alpha)}{(2n+\alpha+\beta)(2n+\alpha+\beta+1)} P_{n-1}^{(\alpha,\beta+1)}
\end{aligned}$$

The monic polynomials have norm $\left\|P_n^{(\alpha,\beta)}\right\|^2 = h_n^{(\alpha,\beta)}$, which is given by

$$h_n^{(\alpha,\beta)} = \frac{2^{2n+\alpha+\beta+1}(n!)\,\Gamma(n+\alpha+\beta+1)\,\Gamma(n+\alpha+1)\,\Gamma(n+\beta+1)}{\Gamma(2n+\alpha+\beta+1)\,\Gamma(2n+\alpha+\beta+2)}, \tag{A.1}$$

where $\Gamma(\,\cdot\,)$ represents the Gamma function. The above formulae allow us to translate the promotion/demotion formulae for the monic polynomials into those for the normalized Jacobi polynomials, as given in equations (2.4)-(2.7). After some algebra, we determine that the constants in those equations are given by

$$\mu_{n,0}^{(\alpha,\beta)} = \sqrt{\frac{2(n+\alpha)(n+\alpha+\beta)}{(2n+\alpha+\beta)(2n+\alpha+\beta+1)}} \tag{A.2}$$

$$\mu_{n,1}^{(\alpha,\beta)} = \sqrt{\frac{2(n+1)(n+\beta+1)}{(2n+\alpha+\beta+1)(2n+\alpha+\beta+2)}} \tag{A.3}$$

$$\nu_{n,0}^{(\alpha,\beta)} = \sqrt{\frac{2(n+\alpha+1)(n+\alpha+\beta+1)}{(2n+\alpha+\beta+1)(2n+\alpha+\beta+2)}} \tag{A.4}$$

$$\nu_{n,-1}^{(\alpha,\beta)} = \sqrt{\frac{2n(n+\beta)}{(2n+\alpha+\beta)(2n+\alpha+\beta+1)}} \tag{A.5}$$

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] C. Bernardi and Y. Maday. *Handbook of Numerical Analysis – Techniques of Scientific Computing*, volume 5, chapter Spectral Methods. Elsevier Science, 1997.

[4] Walter Gautschi. *Orthogonal Polynomials: Computation and Approximation*. Numerical Mathematics and Scientific Computation. Oxford University Press, 2004.

[5] G.H. Golub and J.H. Welsch. Calculation of gauss quadrature rules. *Mathematics of Computation*, 23(106):221–230, 1969.

[6] D. Gottlieb and C.-W. Shu. On the gibbs phenomenon and its resolution. *SIAM Review*, 29(4):644–668, 1997.

[7] 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.

[8] J.S. Hesthaven, S. Gottlieb, and D. Gottlieb. *Spectral methods for time-dependent problems*. Monographs on applied and computational mathematics. Cambridge, 2007.

[9] T.M. Mills and S.J. Smith. The lebesgue constant for lagrange interpolation. *Numerische Mathematik*, 61:111–115, 1992.

[10] Gabor Szego. *Orthogonal Polynomials*. AMS Colloquium Publications, volume XXIII. American Mathematical Society, 1975.

[11] 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.