Fast Algorithms for Spherical Harmonic Expansions

Vladimir Rokhlin* and Mark Tygert†

Research Report YALEU/DCS/RR-1309
December 17, 2004

Abstract

An algorithm is introduced for the rapid evaluation at appropriately chosen nodes on the two-dimensional sphere S^2 in $\mathbb{R}^3$ of functions specified by their spherical harmonic expansions (known as the inverse spherical harmonic transform), and for the evaluation of the coefficients in spherical harmonic expansions of functions specified by their values at appropriately chosen points on S^2 (known as the forward spherical harmonic transform). The procedure is numerically stable and requires an amount of CPU time proportional to $N(\log N)\log(1/\varepsilon)$, where N is the number of nodes in the discretization of S^2, and ε is the precision of computations. The performance of the algorithm is illustrated via several numerical examples.

1 Introduction

Spherical harmonic expansions are a well-understood and widely used tool of applied mathematics; they are encountered, *inter alia*, in weather and climate modeling, in the representation of gravitational, topographic, and magnetic data in geophysics, in the numerical solution of certain partial differential equations, etc. The role of spherical harmonic expansions in the solution of the Laplace equation in three dimensions is similar to the role played by Fourier series expansions in two dimensions.

The spherical harmonic expansion of a function f in $\mathcal{L}^2(S^2)$ is the series of the form

$$f(\theta,\, \varphi) = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \alpha_l^m \, P_l^{|m|}(\cos\theta)\, e^{im\varphi}, \tag{1}$$

where $(\theta,\, \varphi)$ are the standard spherical coordinates on the two-dimensional sphere S^2 in $\mathbb{R}^3$, $0 \le \theta < \pi$ and $0 \le \varphi < 2\pi$, and P_l^m is the associated Legendre function of degree l and order m. While the functions $\{P_l^{|m|}(\cos\theta)\, e^{im\theta}\}$ constitute a basis of $\mathcal{L}^2(S^2)$ that is orthogonal, that is,

$$\int_{-1}^{1} P_k^{|m|}(x)\, P_l^{|m|}(x)\, dx = 0 \tag{2}$$

*Partially supported by the U.S. DoD under AFOSR Grant #F49620–03–C–0041.
†Partially supported by the U.S. DoD under a 2001 NDSEG Fellowship.

when $l \neq k$, their norms are not equal to 1; in fact, they are so badly normalized as to be virtually unusable in numerical calculations (see Subsection 2.1 below for a detailed discussion of the associated Legendre functions). Therefore, it is customary to replace expansions of the form (1) with expansions of the form

$$f(\theta, \varphi) = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \alpha_l^m \, \overline{P}_l^{|m|}(\cos\theta) \, e^{im\varphi}, \tag{3}$$

where $\overline{P}_l^{|m|}$ denotes the normalized version of the associated Legendre function $P_l^{|m|}$, defined on $[-1, 1]$ via the formula

$$\overline{P}_l^{|m|}(x) = (-1)^{|m|} \sqrt{\frac{2l+1}{2} \frac{(l-|m|)!}{(l+|m|)!}} \, P_l^{|m|}(x), \tag{4}$$

so that

$$\int_{-1}^{1} \left(\overline{P}_l^{|m|}(x) \right)^2 dx = 1. \tag{5}$$

In numerical practice, the series (3) is truncated after a finite number of terms, leading to expressions of the form

$$f(\theta, \varphi) \sim \sum_{l=0}^{N} \sum_{m=-l}^{l} \alpha_l^m \, \overline{P}_l^{|m|}(\cos\theta) \, e^{im\varphi}; \tag{6}$$

(6) is viewed as an approximation to the function f, and N is called the order of the expansion (6). Obviously, the expansion (6) contains $(N+1)^2$ terms; the order N required to obtain a prescribed accuracy of the approximation is determined by the complexity of the function f.

Frequently, the need arises to evaluate the coefficients in an expansion of the form (6) for a function f given by a table of its values at a collection of appropriately chosen nodes on S^2; conversely, given the coefficients in (6), one often needs to evaluate f at a collection of points on S^2. The former is usually called the forward spherical harmonic transform, and the latter is known as the inverse spherical harmonic transform. A standard discretization of S^2 is the "tensor product," consisting of all pairs of the form (θ_k, φ_j), with equispaced nodes $\theta_0, \theta_1, \ldots, \theta_{N-1}, \theta_N$ discretizing the interval $[0, \pi]$, defined by the formula

$$\theta_k = \frac{\pi(k+1/2)}{N+1}, \tag{7}$$

and equispaced nodes $\varphi_0, \varphi_1, \ldots, \varphi_{2N-1}, \varphi_{2N}$ discretizing the interval $[0, 2\pi]$, defined by the formula

$$\varphi_j = \frac{2\pi(j+1/2)}{2N+1}. \tag{8}$$

This leads immediately to numerical schemes for both the forward and inverse spherical harmonic transforms costing $\mathcal{O}(N^3)$ operations. Indeed, given a function f defined on S^2 by the formula (6), one can rewrite (6) in the form

$$f(\theta, \varphi) = \sum_{m=-N}^{N} e^{im\varphi} \sum_{l=|m|}^{N} \alpha_l^m \, \overline{P}_l^{|m|}(\cos\theta). \tag{9}$$

For a fixed value of θ, each of the inner sums in (9) contains no more than $N+1$ terms, and there are $2N+1$ such sums (one for each value of m); since the inverse spherical harmonic transform involves $N+1$ values $\theta_0, \theta_1, \ldots, \theta_{N-1}, \theta_N$, the cost of evaluating all inner sums in (9) is $\mathcal{O}(N^3)$. Once all inner sums have been evaluated, evaluation of each outer sum costs $\mathcal{O}(N)$ operations (since each of them contains $2N+1$ terms), and there are $\mathcal{O}(N^2)$ such sums to be evaluated, leading to $\mathcal{O}(N^3)$ CPU time requirements for the evaluation of all outer sums in (9). The cost of the evaluation of the whole inverse spherical harmonic transform (in the form (9)) is the sum of the costs for the inner and outer sums, and is also $\mathcal{O}(N^3)$; a virtually identical calculation shows that the cost of evaluating of the forward harmonic transform is also $\mathcal{O}(N^3)$.

A trivial modification of the scheme described in the preceding paragraph uses the Fast Fourier Transform to evaluate the outer sums in (9), roughly halving the CPU time requirements of the whole procedure. Several other considerations (see, for example, [2]) can be used to reduce the CPU time requirements by a further factor of four or so, but there is no simple trick for reducing the asymptotic CPU time requirements of the whole spherical harmonic transform (either forward or inverse) below N^3. In this paper, we introduce algorithms for both forward and inverse spherical harmonic transforms with CPU time requirements proportional to $N^2(\log N)\log(1/\varepsilon)$, where ε is the precision of computations.

The algorithm of this paper is a procedure for the rapid evaluation of the inner sums in expressions of the form (9). It is based principally on two observations, as follows:

1. The differential equations defining the functions $\overline{P}_l^m$ with arbitrary positive integer m are very close to the differential equations defining the functions $\overline{P}_l^1$ and $\overline{P}_l^2$.

2. There exist fast algorithms for decomposing functions into and reconstructing functions from sums of the forms

$$f(x) = \sum_{l=0}^{N} p_l^1 \, \overline{P}_l^1(x), \tag{10}$$

$$f(x) = \sum_{l=0}^{N} p_l^2 \, \overline{P}_l^2(x). \tag{11}$$

We use the connections between the functions $\overline{P}_l^m$ with arbitrary positive integer m and the functions $\overline{P}_l^1$ and $\overline{P}_l^2$, to apply rapidly to arbitrary vectors the matrices converting between expansions of the forms (10) and (11) and expansions of the form

$$f(x) = \sum_{l=0}^{N} p_l^m \, \overline{P}_l^m(x). \tag{12}$$

This step utilizes the observation made in [4], that the $N \times N$ matrix of eigenvectors of the sum of a diagonal matrix and a semiseparable matrix (see Subsection 2.4 below for the definition of a semiseparable matrix) can be applied to an arbitrary vector of length N for a cost proportional to $N(\log N)\log(1/\varepsilon)$ operations, where ε is the precision of computations.

During the last several years, the interest in fast transforms has been growing, stimulated by the combination of recent progress in fast algorithms of various kinds with the importance of the Fast Fourier Transform in computational mathematics, electrical engineering, etc., and by the success of various types of multilevel computational techniques. In particular, several prior attempts have been made to construct numerically stable fast spherical harmonic transforms. It is not the purpose of this paper to provide an exhaustive survey of literature on the subject; we refer the reader to the publications [10], [9], [15], [13], [12] for details.

The structure of this paper is as follows. In Section 2, we summarize a number of facts (from both mathematical and numerical analysis) to be used in the rest of the paper; all of the content of Section 2 is either well-known or follows easily from well-known facts. In Section 3, we build the analytical apparatus to be used in the construction of the algorithms of this paper. Section 4 contains an informal description of the algorithm, and in Section 5 the procedure is described in detail. The performance of the scheme is illustrated with numerical examples in Section 6, and Section 7 contains a discussion of possible applications of the approach of this paper in other environments.

2 Mathematical and Numerical Preliminaries

In this section, we summarize several facts from mathematical and numerical analysis. Please note that in this section and throughout this paper, the variable x always takes arbitrary values in $[-1,\,1]$, θ takes values in $[0,\,\pi]$, and φ takes values in $[0,\,2\pi]$. We will always use the term "eigenvector" to mean "normalized eigenvector."

2.1 Spherical Harmonics and Associated Legendre Functions

In this subsection, we summarize a number of properties of spherical harmonics and associated Legendre functions; all of these can be found, for example, in [1].

The coefficients in the spherical harmonic expansion (6) of a function f in $\mathcal{L}^2(S^2)$ are given by the formula

$$\alpha_l^m = \int_0^\pi \int_0^{2\pi} \overline{P}_l^{|m|}(\cos\theta)\, e^{-im\varphi}\, f(\theta,\,\varphi)\, \sin\theta\; d\varphi\, d\theta. \tag{13}$$

For the forward spherical harmonic transform of order N, we have to compute the coefficients (13) from the values $f(\theta_k,\,\varphi_j)$, where $\theta_0,\,\theta_1,\,\dots,\,\theta_{N-1},\,\theta_N$ are defined in (7), and $\varphi_0, \varphi_1, \dots, \varphi_{2N-1}, \varphi_{2N}$ are defined in (8). For a cost of $\mathcal{O}(N^2 \log N)$, we use the Fast Fourier Transform to obtain the $2(2N+1)(N+1)$ values $g^m(\theta_k)$ and $h^m(\theta_k)$ ($m = -N,\,-N+1,\,\dots,\,N-1,\,N$; $k = 0,\,1,\,\dots,\,N-1,\,N$) of the functions g^m and h^m defined on $[0,\,\pi]$ by the formulae

$$g^m(\theta) = \sum_{j=0}^{2N} \cos(m\,\varphi_j)\, f(\theta,\,\varphi_j), \tag{14}$$

$$h^m(\theta) = \sum_{j=0}^{2N} \sin(m\,\varphi_j)\, f(\theta,\,\varphi_j). \tag{15}$$

We then evaluate the coefficients (13) via the formula

$$\alpha_l^m = \int_0^\pi \overline{P}_l^{|m|}(\cos\theta)\, g^m(\theta)\, \sin\theta\; d\theta - i \int_0^\pi \overline{P}_l^{|m|}(\cos\theta)\, h^m(\theta)\, \sin\theta\; d\theta. \tag{16}$$

To evaluate the integrals in (16), we have to convert the values $f^m(\cos\theta_k)$ into the coefficients p_m^m, p_{m+1}^m, $\ldots$, p_{N-1}^m, p_N^m in expansions of functions f^m on $[-1,\,1]$ of the form

$$f^m(x) = \sum_{l=m}^{N} p_l^m \, \overline{P}_l^m(x), \tag{17}$$

$$p_l^m = \int_{-1}^{1} \overline{P}_l^m(x)\, f^m(x)\, dx, \tag{18}$$

where m is any integer with $0 \le m \le N$.

The principal purpose of this paper is the construction of a "fast" scheme for computing the coefficients (18) from the values $f^m(\cos\theta_k)$, and for computing the inverse of this transformation, that is, for computing the values $f^m(\cos\theta_k)$ of the function f^m defined in (17) from the coefficients (18).

For any nonnegative integers l and m with $m \le l$, the associated Legendre function P_l^m on $[-1,\,1]$ is defined by the formula

$$P_l^m(x) = (-1)^m \sqrt{1-x^2}^{\,m} \, \frac{d^m}{dx^m} P_l(x), \tag{19}$$

where P_l is the Legendre polynomial of degree l. Obviously, P_l^m is a polynomial when m is even and a polynomial multiplied by $\sqrt{1-x^2}$ when m is odd.

For any nonnegative integer m, we define the differential operator L_m by the formula

$$L_m(f)(x) = -\frac{d}{dx}\left((1-x^2)\, \frac{d}{dx} f(x) \right) + \frac{m^2}{1-x^2}\, f(x) \tag{20}$$

for any function f on $[-1,\,1]$ with a continuous second derivative. For any nonnegative integers l and m with $m \le l$, the function $\overline{P}_l^m$ satisfies the differential equation

$$L_m\left(\overline{P}_l^m\right)(x) = l(l+1)\, \overline{P}_l^m(x), \tag{21}$$

where the differential operator L_m is defined in (20).

For any integers l and m with $0 \le m \le l$ and $l > 0$,

$$(2l+1)\, x\, P_l^m(x) = (l+m)\, P_{l-1}^m(x) + (l-m+1)\, P_{l+1}^m(x). \tag{22}$$

For any nonnegative integers l and m with $1 \le m \le l$,

$$\int_{-1}^{1} \overline{P}_l^m(x)\, \frac{1}{1-x^2}\, \overline{P}_l^m(x)\, dx = \frac{2l+1}{2m}. \tag{23}$$

Lemma 2.1 *Suppose that l and m are even integers such that $2 \le m \le l$. Then, there exist $l/2$ real numbers ξ_0, ξ_1, $\ldots$, $\xi_{l/2-2}$, $\xi_{l/2-1}$ such that*

$$\overline{P}_l^m(x) = \sum_{k=0}^{l/2-1} \xi_k\, \overline{P}_{2k+2}^2(x). \tag{24}$$

Lemma 2.2 *Suppose that l and m are integers such that m is even, l is odd, and $2 \le m < l$. Then, there exist $(l-1)/2$ real numbers ξ_0, ξ_1, ..., $\xi_{(l-1)/2-2}$, $\xi_{(l-1)/2-1}$ such that*

$$\overline{P}_l^m(x) = \sum_{k=0}^{(l-1)/2-1} \xi_k \, \overline{P}_{2k+3}^2(x). \tag{25}$$

Lemma 2.3 *Suppose that l and m are integers such that m is odd, l is even, and $1 \le m < l$. Then, there exist $l/2$ real numbers ξ_0, ξ_1, ..., $\xi_{l/2-2}$, $\xi_{l/2-1}$ such that*

$$\overline{P}_l^m(x) = \sum_{k=0}^{l/2-1} \xi_k \, \overline{P}_{2k+2}^1(x). \tag{26}$$

Lemma 2.4 *Suppose that l and m are odd integers such that $1 \le m \le l$. Then, there exist $(l+1)/2$ real numbers ξ_0, ξ_1, ..., $\xi_{(l+1)/2-2}$, $\xi_{(l+1)/2-1}$ such that*

$$\overline{P}_l^m(x) = \sum_{k=0}^{(l+1)/2-1} \xi_k \, \overline{P}_{2k+1}^1(x). \tag{27}$$

2.2 Chebyshev Polynomials

In this subsection, we cite the existence of a fast algorithm for computing with Chebyshev polynomials.

For any nonnegative integer k, we define T_k to be the Chebyshev polynomial of degree k of the first kind, defined by the formula

$$T_k(\cos\theta) = \cos(k\theta) \tag{28}$$

for any real θ, and U_k to be the Chebyshev polynomial of degree k of the second kind, defined by the formula

$$U_k(\cos\theta) = \frac{\sin((k+1)\theta)}{\sin\theta} \tag{29}$$

for any real θ.

The following observation cites the relationship between the Fast Fourier Transform and expansions in series of Chebyshev polynomials.

Observation 2.5 *Suppose that $N \ge 0$ is an integer, c_0, c_1, ..., c_{N-1}, c_N, and u_0, u_1, ..., u_{N-1}, u_N are real numbers, and f and g are the functions on $[-1, 1]$ defined by the formulae*

$$f(x) = \sum_{k=0}^{N} c_k \, T_k(x), \tag{30}$$

$$g(x) = \sum_{k=0}^{N} u_k \sqrt{1-x^2} \, U_k(x). \tag{31}$$

Then, there exists an algorithm which uses $\mathcal{O}(N \log N)$ operations to convert the coefficients c_0, c_1, ..., c_{N-1}, c_N into the values $f(x_0)$, $f(x_1)$, ..., $f(x_{N-1})$, $f(x_N)$, and to convert *the coefficients u_0, u_1, ..., u_{N-1}, u_N into the values $g(x_0)$, $g(x_1)$, ..., $g(x_{N-1})$, $g(x_N)$, where the sampling locations x_0, x_1, ..., x_{N-1}, x_N are defined by the formula*

$$x_k = \cos\left(\frac{\pi(k+1/2)}{N+1}\right). \tag{32}$$

Moreover, there exists an algorithm which uses $\mathcal{O}(N \log N)$ operations to convert the values $f(x_0)$, $f(x_1)$, ..., $f(x_{N-1})$, $f(x_N)$ into the coefficients c_0, c_1, ..., c_{N-1}, c_N, and to convert the values $g(x_0)$, $g(x_1)$, ..., $g(x_{N-1})$, $g(x_N)$ into the coefficients u_0, u_1, ..., u_{N-1}, u_N (see, for example, [14]).

2.3 Associated Legendre Functions of Low Orders

In this subsection, we summarize certain simple relationships between Chebyshev polynomials and associated Legendre functions of orders 1 and 2. These relationships are a straightforward consequence of formulae 7.112.1, 8.339.1, 8.339.2, 8.700.1, 8.752.1, 8.826.1, 8.828.1, 8.832.2, and 8.911.4 of [7].

We define the function Λ on $[0, \infty)$ by the formula

$$\Lambda(z) = \frac{\Gamma\left(z+\frac{1}{2}\right)}{\Gamma(z+1)}, \tag{33}$$

where Γ is the Euler gamma function.

For any integer $n \geq 1$ and $l, k = 0, 1, \ldots, n-2, n-1$, we define the entry $A^{n,1,+}_{l,k}$ of the $n \times n$ matrix $A^{n,1,+}$ by the formulae

$$\begin{aligned} A^{n,1,+}_{l,k} &= -\frac{4l+3}{2(2k+2l+3)(2k-2l-1)}\,\Lambda\left(k-l\right)\,\Lambda\left(\frac{2k+2l+1}{2}\right) \\ &\quad \cdot \sqrt{\frac{4(l+1)(2l+1)}{4l+3}} \end{aligned} \tag{34}$$

when $k \geq l$, and

$$A^{n,1,+}_{l,k} = 0 \tag{35}$$

otherwise (when $k < l$).

For any integer $n \geq 1$ and $l, k = 0, 1, \ldots, n-2, n-1$, we define the entry $A^{n,1,-}_{l,k}$ of the $n \times n$ matrix $A^{n,1,-}$ by the formulae

$$\begin{aligned} A^{n,1,-}_{l,k} &= -\frac{4l+5}{2(2k+2l+5)(2k-2l-1)}\,\Lambda\left(k-l\right)\,\Lambda\left(\frac{2k+2l+3}{2}\right) \\ &\quad \cdot \sqrt{\frac{4(l+1)(2l+3)}{4l+5}} \end{aligned} \tag{36}$$

when $k \geq l$, and

$$A^{n,1,-}_{l,k} = 0 \tag{37}$$

otherwise (when $k < l$).

For any integer $n \geq 1$ and $k, l = 0, 1, \ldots, n-2, n-1$, we define the entry $B^{n,1,+}_{k,l}$ of the $n \times n$ matrix $B^{n,1,+}$ by the formulae

$$B^{n,1,+}_{k,l} = 2\,\frac{2k+1}{\pi}\,\Lambda\,(l-k)\;\Lambda\,(l+k+1)\,\sqrt{\frac{4l+3}{4(l+1)(2l+1)}} \tag{38}$$

when $k \leq l$, and

$$B^{n,1,+}_{k,l} = 0 \tag{39}$$

otherwise (when $k > l$).

For any integer $n \geq 1$ and $k, l = 0, 1, \ldots, n-2, n-1$, we define the entry $B^{n,1,-}_{k,l}$ of the $n \times n$ matrix $B^{n,1,-}$ by the formulae

$$B^{n,1,-}_{k,l} = 4\,\frac{k+1}{\pi}\,\Lambda\,(l-k)\;\Lambda\,(l+k+2)\,\sqrt{\frac{4l+5}{4(l+1)(2l+3)}} \tag{40}$$

when $k \leq l$, and

$$B^{n,1,-}_{k,l} = 0 \tag{41}$$

otherwise (when $k > l$).

For any integer $n \geq 1$, $l = 0, 1, \ldots, n-2, n-1$, and $k = 0, 1, \ldots, n-1, n$, we define the entry $A^{n,2,+}_{l,k}$ of the $n \times (n+1)$ matrix $A^{n,2,+}$ by the formulae

$$\begin{aligned} A^{n,2,+}_{l,k} = \Bigg(4 \;+\; & \frac{2k\,(6(l+1)(2l+3) - 2(2k-1)(2k+1))}{(2k+2l+3)(2k-2l-3)}\,\Lambda\,(k-l-1) \\ & \cdot\;\Lambda\left(\frac{2k+2l+1}{2}\right)\Bigg)\,\sqrt{\frac{4l+5}{8(l+1)(l+2)(2l+1)(2l+3)}} \end{aligned} \tag{42}$$

when $k \geq l+1$, and

$$A^{n,2,+}_{l,k} = 4\,\sqrt{\frac{4l+5}{8(l+1)(l+2)(2l+1)(2l+3)}} \tag{43}$$

otherwise (when $k < l+1$).

For any integer $n \geq 1$, $l = 0, 1, \ldots, n-2, n-1$, and $k = 0, 1, \ldots, n-1, n$, we define the entry $A^{n,2,-}_{l,k}$ of the $n \times (n+1)$ matrix $A^{n,2,-}$ by the formulae

$$\begin{aligned} A^{n,2,-}_{l,k} = \Bigg(4 \;+\; & \frac{(2k+1)\,(6(l+2)(2l+3) - 8k(k+1))}{(2k+2l+5)(2k-2l-3)}\,\Lambda\,(k-l-1) \\ & \cdot\;\Lambda\left(\frac{2k+2l+3}{2}\right)\Bigg)\,\sqrt{\frac{4l+7}{8(l+1)(l+2)(2l+3)(2l+5)}} \end{aligned} \tag{44}$$

when $k \geq l+1$, and

$$A^{n,2,-}_{l,k} = 4\,\sqrt{\frac{4l+7}{8(l+1)(l+2)(2l+3)(2l+5)}} \tag{45}$$

otherwise (when $k < l+1$).

For any integer $n \geq 1$, $k = 0, 1, \ldots, n-1, n$, and $l = 0, 1, \ldots, n-2, n-1$, we define the entry $B^{n,2,+}_{k,l}$ of the $(n+1) \times n$ matrix $B^{n,2,+}$ by the formulae

$$\begin{aligned} B^{n,2,+}_{k,l} &= \frac{2(l+1)(2l+3) - 8k^2}{\pi} \, \Lambda\,(l-k+1) \; \Lambda\,(l+k+1) \\ &\cdot \sqrt{\frac{4l+5}{8(l+1)(l+2)(2l+1)(2l+3)}} \end{aligned} \tag{46}$$

when $k = 0$,

$$\begin{aligned} B^{n,2,+}_{k,l} &= 2\,\frac{2(l+1)(2l+3) - 8k^2}{\pi} \, \Lambda\,(l-k+1) \; \Lambda\,(l+k+1) \\ &\cdot \sqrt{\frac{4l+5}{8(l+1)(l+2)(2l+1)(2l+3)}} \end{aligned} \tag{47}$$

when $0 < k \leq l+1$, and

$$B^{n,2,+}_{k,l} = 0 \tag{48}$$

otherwise (when $k > l+1$).

For any integer $n \geq 1$, $k = 0, 1, \ldots, n-1, n$, and $l = 0, 1, \ldots, n-2, n-1$, we define the entry $B^{n,2,-}_{k,l}$ of the $(n+1) \times n$ matrix $B^{n,2,-}$ by the formulae

$$\begin{aligned} B^{n,2,-}_{k,l} &= \frac{4(l+2)(2l+3) - 4(2k+1)^2}{\pi} \, \Lambda\,(l-k+1) \; \Lambda\,(l+k+2) \\ &\cdot \sqrt{\frac{4l+7}{8(l+1)(l+2)(2l+3)(2l+5)}} \end{aligned} \tag{49}$$

when $k \leq l+1$, and

$$B^{n,2,-}_{k,l} = 0 \tag{50}$$

otherwise (when $k > l+1$).

The following four lemmas are proven via mechanical, but rather tedious manipulations of formulae 7.112.1, 8.339.1, 8.339.2, 8.700.1, 8.752.1, 8.826.1, 8.828.1, 8.832.2, and 8.911.4 of [7].

The following lemma provides explicit expressions for the matrix $B^{n,1,+}$ converting coefficients in linear combinations of associated Legendre functions of order 1 of odd degrees into coefficients in linear combinations of even Chebyshev polynomials of the second kind, scaled by $\sqrt{1-x^2}$, and for the matrix $A^{n,1,+}$ converting the latter into the former.

Lemma 2.6 *Suppose that $n \geq 1$ is an integer, $q = (q_0, q_1, \ldots, q_{n-2}, q_{n-1})^{\mathrm{T}}$ is a real vector, and f is the function on $[-1, 1]$ defined by the formula*

$$f(x) = \sum_{l=0}^{n-1} q_l \, \overline{P}^{1}_{2l+1}(x). \tag{51}$$

Then,

$$f(x) = \sum_{k=0}^{n-1} u_k \sqrt{1-x^2} \, U_{2k}(x), \tag{52}$$

where $u = (u_0,\, u_1,\, \ldots,\, u_{n-2},\, u_{n-1})^{\mathrm{T}}$ *is the real vector defined by the formula*

$$u = B^{n,1,+}\, q, \tag{53}$$

and $B^{n,1,+}$ *is defined in (33), (38), and (39). Furthermore,*

$$q = A^{n,1,+}\, u, \tag{54}$$

where $A^{n,1,+}$ *is defined in (33), (34), and (35).*

The following lemma provides explicit expressions for the matrix $B^{n,1,-}$ converting coefficients in linear combinations of associated Legendre functions of order 1 of even degrees into coefficients in linear combinations of odd Chebyshev polynomials of the second kind, scaled by $\sqrt{1-x^2}$, and for the matrix $A^{n,1,-}$ converting the latter into the former.

Lemma 2.7 *Suppose that* $n \geq 1$ *is an integer,* $q = (q_0,\, q_1,\, \ldots,\, q_{n-2},\, q_{n-1})^{\mathrm{T}}$ *is a real vector, and* f *is the function on* $[-1,\, 1]$ *defined by the formula*

$$f(x) = \sum_{l=0}^{n-1} q_l\, \overline{P}_{2l+2}^{1}(x). \tag{55}$$

Then,

$$f(x) = \sum_{k=0}^{n-1} u_k\, \sqrt{1-x^2}\, U_{2k+1}(x), \tag{56}$$

where $u = (u_0,\, u_1,\, \ldots,\, u_{n-2},\, u_{n-1})^{\mathrm{T}}$ *is the real vector defined by the formula*

$$u = B^{n,1,-}\, q, \tag{57}$$

and $B^{n,1,-}$ *is defined in (33), (40), and (41). Furthermore,*

$$q = A^{n,1,-}\, u, \tag{58}$$

where $A^{n,1,-}$ *is defined in (33), (36), and (37).*

The following lemma provides explicit expressions for the matrix $B^{n,2,+}$ converting coefficients in linear combinations of associated Legendre functions of order 2 of even degrees into coefficients in linear combinations of even Chebyshev polynomials of the first kind, and for the matrix $A^{n,2,+}$ converting the latter into the former.

Lemma 2.8 *Suppose that* $n \geq 1$ *is an integer,* $p = (p_0,\, p_1,\, \ldots,\, p_{n-2},\, p_{n-1})^{\mathrm{T}}$ *is a real vector, and* f *is the function on* $[-1,\, 1]$ *defined by the formula*

$$f(x) = \sum_{l=0}^{n-1} p_l\, \overline{P}_{2l+2}^{2}(x). \tag{59}$$

Then,

$$f(x) = \sum_{k=0}^{n} c_k\, T_{2k}(x), \tag{60}$$

where $c = (c_0, c_1, \ldots, c_{n-1}, c_n)^{\mathrm{T}}$ *is the real vector defined by the formula*

$$c = B^{n,2,+}\, p, \tag{61}$$

and $B^{n,2,+}$ *is defined in (33), (46), (47), and (48). Furthermore,*

$$p = A^{n,2,+}\, c, \tag{62}$$

where $A^{n,2,+}$ *is defined in (33), (42), and (43).*

The following lemma provides explicit expressions for the matrix $B^{n,2,-}$ converting coefficients in linear combinations of associated Legendre functions of order 2 of odd degrees into coefficients in linear combinations of odd Chebyshev polynomials of the first kind, and for the matrix $A^{n,2,-}$ converting the latter into the former.

Lemma 2.9 *Suppose that* $n \geq 1$ *is an integer,* $p = (p_0, p_1, \ldots, p_{n-2}, p_{n-1})^{\mathrm{T}}$ *is a real vector, and* f *is the function on* $[-1, 1]$ *defined by the formula*

$$f(x) = \sum_{l=0}^{n-1} p_l\, \overline{P}_{2l+3}^{2}(x). \tag{63}$$

Then,

$$f(x) = \sum_{k=0}^{n} c_k\, T_{2k+1}(x), \tag{64}$$

where $c = (c_0, c_1, \ldots, c_{n-1}, c_n)^{\mathrm{T}}$ *is the real vector defined by the formula*

$$c = B^{n,2,-}\, p, \tag{65}$$

and $B^{n,2,-}$ *is defined in (33), (49), and (50). Furthermore,*

$$p = A^{n,2,-}\, c, \tag{66}$$

where $A^{n,2,-}$ *is defined in (33), (44), and (45).*

Observation 2.10 *Suppose that* $n \geq 1$ *is an integer. Then, there exists an algorithm which uses* $\mathcal{O}(n \log(n/\varepsilon))$ *operations to apply to an arbitrary vector any of the matrices* $A^{n,1,+}$, $B^{n,1,+}$, $A^{n,1,-}$, $B^{n,1,-}$, $A^{n,2,+}$, $B^{n,2,+}$, $A^{n,2,-}$, *and* $B^{n,2,-}$ *defined in (33)–(50), where* ε *is the precision of computations (see [3]).*

2.4 Semiseparable Matrices

For any integer $n > 0$, a semiseparable real $n \times n$ matrix S is a matrix whose entry $S_{j,k}$ is given by the formulae

$$S_{j,k} = a_j\, b_k \tag{67}$$

when $j \leq k$, and

$$S_{j,k} = a_k\, b_j \tag{68}$$

when $j > k$, where $a = (a_0, a_1, \ldots, a_{n-2}, a_{n-1})^{\mathrm{T}}$ and $b = (b_0, b_1, \ldots, b_{n-2}, b_{n-1})^{\mathrm{T}}$ are real vectors.

Matrices of the form

$$G = D + S, \tag{69}$$

where D is a diagonal real matrix and S is a semiseparable real matrix, will be encountered repeatedly throughout this paper. The matrix U of eigenvectors of the matrix G in (69) will be particularly important; U is orthogonal and diagonalizes G, so that

$$U^{\mathrm{T}} G\, U = \Lambda, \tag{70}$$

where Λ is a diagonal real matrix.

The principal numerical tool of this paper is the following observation, made in [8] and [4].

Observation 2.11 *The matrices U and U^{T} in (70) can be applied to an arbitrary vector of length N for a cost of $\mathcal{O}(N(\log N)\log(1/\varepsilon))$ operations, where ε is the precision of computations.*

Remark 2.12 Strictly speaking, only the numerical apparatus behind Observation 2.11 is constructed in [4]. However, the observation itself is stated explicitly in a very similar environment in [8]. In our implementation, we used a minor modification of the apparatus in [4], to be reported at a later date.

3 Analytical Apparatus

In this section, we construct the principal analytical tools used in this paper.

In Subsection 3.1, we observe that when the function $\overline{P}_l^m$ is represented as a linear combination of functions $\overline{P}_j^1$ or $\overline{P}_j^2$ (depending on whether m is even or odd), the Sturm-Liouville problem (21) becomes an eigenvector problem for the matrix G in (69). Thus, according to Observation 2.11, there exists an algorithm that uses $\mathcal{O}\left(N(\log N)\log(1/\varepsilon)\right)$ operations to apply the matrices U and U^{T} in (70) to arbitrary vectors of length N, where ε is the precision of computations.

In Subsection 3.2, we observe that the problem of evaluating expansions of the form (12) can be reduced to the problem of evaluating expansions of the forms (10) and (11), via the matrices U and U^{T} in (70). These matrices can be applied to arbitrary vectors efficiently, due to Observation 2.11.

3.1 Associated Legendre Differential Equations in Terms of Associated Legendre Functions of Low Orders

For any even integers n and m with $2 \le m \le n$, and for $j, k = 0, 1, \ldots, n/2-2, n/2-1$, we define the entry $G_{j,k}^{n,m}$ of the $n/2 \times n/2$ matrix $G^{n,m}$ by the formula

$$G_{j,k}^{n,m} = \int_{-1}^{1} \overline{P}_{2j+2}^2(x) \;\; L_m\left(\overline{P}_{2k+2}^2\right)(x) \;\; dx, \tag{71}$$

where the differential operator L_m is defined in (20).

For any odd integer n and even integer m with $2 \le m < n$, and for $j,k = 0, 1, \dots, (n-1)/2-2, (n-1)/2-1$, we define the entry $G^{n,m}_{j,k}$ of the $(n-1)/2 \times (n-1)/2$ matrix $G^{n,m}$ by the formula

$$G^{n,m}_{j,k} = \int_{-1}^{1} \overline{P}^2_{2j+3}(x) \; L_m\left(\overline{P}^2_{2k+3}\right)(x) \; dx, \tag{72}$$

where the differential operator L_m is defined in (20).

For any even integer n and odd integer m with $1 \le m < n$, and for $j,k = 0, 1, \dots, n/2-2, n/2-1$, we define the entry $G^{n,m}_{j,k}$ of the $n/2 \times n/2$ matrix $G^{n,m}$ by the formula

$$G^{n,m}_{j,k} = \int_{-1}^{1} \overline{P}^1_{2j+2}(x) \; L_m\left(\overline{P}^1_{2k+2}\right)(x) \; dx, \tag{73}$$

where the differential operator L_m is defined in (20).

For any odd integers n and m with $1 \le m \le n$, and for $j,k = 0, 1, \dots, (n+1)/2-2, (n+1)/2-1$, we define the entry $G^{n,m}_{j,k}$ of the $(n+1)/2 \times (n+1)/2$ matrix $G^{n,m}$ by the formula

$$G^{n,m}_{j,k} = \int_{-1}^{1} \overline{P}^1_{2j+1}(x) \; L_m\left(\overline{P}^1_{2k+1}\right)(x) \; dx, \tag{74}$$

where the differential operator L_m is defined in (20).

The following lemma states that the coefficients in the expansion of the function $\overline{P}^m_l$ in terms of either the functions $\overline{P}^1_j$ or the functions $\overline{P}^2_j$ (depending on whether m is even or odd) are the entries in an eigenvector of the matrix $G^{n,m}$.

Lemma 3.1 *Suppose that m and n are integers such that $1 \le m \le n$.*

Then, when m and n are both even, $l(l+1)$ is an eigenvalue of the matrix $G^{n,m}$ defined in (71), for any even integer l with $m \le l \le n$, and the coordinates $\xi_0, \xi_1, \dots, \xi_{n/2-2}, \xi_{n/2-1}$ of the corresponding eigenvector are the coefficients in the expansion

$$\overline{P}^m_l(x) = \sum_{k=0}^{n/2-1} \xi_k \, \overline{P}^2_{2k+2}(x). \tag{75}$$

When m is even and n is odd, $l(l+1)$ is an eigenvalue of the matrix $G^{n,m}$ defined in (72), for any odd integer l with $m < l \le n$, and the coordinates $\xi_0, \xi_1, \dots, \xi_{(n-1)/2-2}, \xi_{(n-1)/2-1}$ of the corresponding eigenvector are the coefficients in the expansion

$$\overline{P}^m_l(x) = \sum_{k=0}^{(n-1)/2-1} \xi_k \, \overline{P}^2_{2k+3}(x). \tag{76}$$

When m is odd and n is even, $l(l+1)$ is an eigenvalue of the matrix $G^{n,m}$ defined in (73), for any even integer l with $m < l \le n$, and the coordinates $\xi_0, \xi_1, \dots, \xi_{n/2-2}, \xi_{n/2-1}$ of the corresponding eigenvector are the coefficients in the expansion

$$\overline{P}^m_l(x) = \sum_{k=0}^{n/2-1} \xi_k \, \overline{P}^1_{2k+2}(x). \tag{77}$$

When m and n are both odd, $l(l+1)$ is an eigenvalue of the matrix $G^{n,m}$ defined in (74), for any odd integer l with $m \le l \le n$, and the coordinates ξ_0, ξ_1, ..., $\xi_{(n+1)/2-2}$, $\xi_{(n+1)/2-1}$ of the corresponding eigenvector are the coefficients in the expansion

$$\overline{P}_l^m(x) = \sum_{k=0}^{(n+1)/2-1} \xi_k \, \overline{P}_{2k+1}^1(x). \tag{78}$$

Proof. We outline the proof in the case that m and n are both even; the proofs in the other three cases are similar.

Substituting (24) into (21) and using (2) and (5), we obtain from (71) that the numbers ξ_0, ξ_1, ..., $\xi_{l/2-2}$, $\xi_{l/2-1}$ from (24), along with the numbers $\xi_{l/2} = 0$, $\xi_{l/2+1} = 0$, ..., $\xi_{n/2-2} = 0$, $\xi_{n/2-1} = 0$ when $l < n$, are the coordinates of the eigenvector of $G^{n,m}$ with corresponding eigenvalue $l(l+1)$, giving the expansion (75). □

The following lemma states that $G^{n,m}$ is the sum of a diagonal matrix and a semiseparable matrix and provides expressions for the entries of $G^{n,m}$.

Lemma 3.2 *Suppose that m and n are integers such that $1 \le m \le n$.*

Then, the matrix $G^{n,m}$ defined in (71)–(74) has the form

$$G^{n,m} = D + S, \tag{79}$$

where D is a diagonal matrix and S is a semiseparable matrix.

When m and n are both even, D is the diagonal $n/2 \times n/2$ matrix with the diagonal entries $D_{0,0}$, $D_{1,1}$, ..., $D_{n/2-2,n/2-2}$, $D_{n/2-1,n/2-1}$ defined by the formula

$$D_{k,k} = (2k+2)(2k+3), \tag{80}$$

and S is the semiseparable $n/2 \times n/2$ matrix with the entry $S_{j,k}$ defined by the formulae

$$S_{j,k} = a_j \, b_k \tag{81}$$

when $j \le k$, and

$$S_{j,k} = a_k \, b_j \tag{82}$$

otherwise (when $j > k$), where the numbers a_0, a_1, ..., $a_{n/2-2}$, $a_{n/2-1}$ and b_0, b_1, ..., $b_{n/2-2}$, $b_{n/2-1}$ are defined by the formulae

$$a_k = \sqrt{\frac{(2k+1)(2k+2)(2k+3)(2k+4)(4k+5)}{8 \cdot 15}}, \tag{83}$$

$$b_k = (m^2 - 4) \sqrt{\frac{15(4k+5)}{2(2k+1)(2k+2)(2k+3)(2k+4)}}. \tag{84}$$

When m is even and n is odd, D is the diagonal $(n-1)/2 \times (n-1)/2$ matrix with the diagonal entries $D_{0,0}$, $D_{1,1}$, ..., $D_{(n-1)/2-2,(n-1)/2-2}$, $D_{(n-1)/2-1,(n-1)/2-1}$ defined by the formula

$$D_{k,k} = (2k+3)(2k+4), \tag{85}$$

and S is the semiseparable $(n-1)/2 \times (n-1)/2$ matrix with the entry $S_{j,k}$ defined by the formulae

$$S_{j,k} = a_j \, b_k \tag{86}$$

when $j \le k$,

$$S_{j,k} = a_k \, b_j \tag{87}$$

otherwise (when $j > k$), where the numbers a_0, a_1, ..., $a_{(n-1)/2-2}$, $a_{(n-1)/2-1}$ and b_0, b_1, ..., $b_{(n-1)/2-2}$, $b_{(n-1)/2-1}$ are defined by the formulae

$$a_k = \sqrt{\frac{(2k+2)(2k+3)(2k+4)(2k+5)(4k+7)}{14 \cdot 15}}, \tag{88}$$

$$b_k = (m^2 - 4) \sqrt{\frac{7 \cdot 15(4k+7)}{8(2k+2)(2k+3)(2k+4)(2k+5)}}. \tag{89}$$

When m is odd and n is even, D is the diagonal $n/2 \times n/2$ matrix with the diagonal entries $D_{0,0}$, $D_{1,1}$, ..., $D_{n/2-2,n/2-2}$, $D_{n/2-1,n/2-1}$ defined by the formula

$$D_{k,k} = (2k+2)(2k+3), \tag{90}$$

and S is the semiseparable $n/2 \times n/2$ matrix with the entry $S_{j,k}$ defined by the formulae

$$S_{j,k} = a_j \, b_k \tag{91}$$

when $j \le k$, and

$$S_{j,k} = a_k \, b_j \tag{92}$$

otherwise (when $j > k$), where the numbers a_0, a_1, ..., $a_{n/2-2}$, $a_{n/2-1}$ and b_0, b_1, ..., $b_{n/2-2}$, $b_{n/2-1}$ are defined by the formulae

$$a_k = \sqrt{\frac{(2k+2)(2k+3)(4k+5)}{30}}, \tag{93}$$

$$b_k = (m^2 - 1) \sqrt{\frac{15(4k+5)}{2(2k+2)(2k+3)}}. \tag{94}$$

When m and n are both odd, D is the diagonal $(n+1)/2 \times (n+1)/2$ matrix with the diagonal entries $D_{0,0}$, $D_{1,1}$, ..., $D_{(n+1)/2-2,(n+1)/2-2}$, $D_{(n+1)/2-1,(n+1)/2-1}$ defined by the formula

$$D_{k,k} = (2k+1)(2k+2), \tag{95}$$

and S is the semiseparable $(n+1)/2 \times (n+1)/2$ matrix with the entry $S_{j,k}$ defined by the formulae

$$S_{j,k} = a_j \, b_k \tag{96}$$

when $j \le k$, and

$$S_{j,k} = a_k \, b_j \tag{97}$$

otherwise (when $j > k$), where the numbers a_0, a_1, $\dots$, $a_{(n+1)/2-2}$, $a_{(n+1)/2-1}$ and b_0, b_1, $\dots$, $b_{(n+1)/2-2}$, $b_{(n+1)/2-1}$ are defined by the formulae

$$a_k = \sqrt{\frac{(2k+1)(2k+2)(4k+3)}{6}}, \tag{98}$$

$$b_k = (m^2-1)\sqrt{\frac{3(4k+3)}{2(2k+1)(2k+2)}}. \tag{99}$$

Proof. We outline the proof in the case that m and n are both even; the proofs in the other three cases are similar.

We define the entry $D_{j,k}$ of the $n/2 \times n/2$ matrix D by the formula

$$D_{j,k} = \int_{-1}^{1} \overline{P}_{2j+2}^{2}(x) \;\; L_2\left(\overline{P}_{2k+2}^{2}\right)(x) \;\; dx, \tag{100}$$

where the differential operator L_2 is defined in (20), and we define the entry $S_{j,k}$ of the $n/2 \times n/2$ matrix S by the formula

$$S_{j,k} = \int_{-1}^{1} \overline{P}_{2j+2}^{2}(x)\, \frac{m^2-4}{1-x^2}\, \overline{P}_{2k+2}^{2}(x)\, dx. \tag{101}$$

We now show that $G^{n,m} = D + S$, D is diagonal, and S is semiseparable.

Combining (71), (100), (101), and (20), we obtain the decomposition (79).

Substituting (21) into (100), and using (2) and (5), we observe that the matrix D is diagonal, with the diagonal entries given by (80).

In order to obtain the formulae (81)–(84), we define the entry $M_{j,k}$ of the infinite-dimensional matrix M, for $j, k = 0, 1, 2, \dots$, by the formula

$$M_{j,k} = \int_{-1}^{1} \overline{P}_{2j+2}^{2}(x)\, \frac{m^2-4}{1-x^2}\, \overline{P}_{2k+2}^{2}(x)\, dx, \tag{102}$$

and observe that the entry $(M^{-1})_{j,k}$ of the inverse M^{-1} of the matrix M is given by the formula

$$(M^{-1})_{j,k} = \int_{-1}^{1} \overline{P}_{2j+2}^{2}(x)\, \frac{1-x^2}{m^2-4}\, \overline{P}_{2k+2}^{2}(x)\, dx, \tag{103}$$

since M represents the operator acting on functions on $[-1, 1]$ by multiplication by the factor

$$\frac{m^2-4}{1-x^2}, \tag{104}$$

whereas M^{-1} represents the operator acting on functions on $[-1, 1]$ by multiplication by the inverse factor

$$\frac{1-x^2}{m^2-4}. \tag{105}$$

Using (2), (5), (22), and (103), we observe that M^{-1} is tridiagonal. So, M is the inverse of a tridiagonal matrix, and, as such, M is semiseparable (see, for example, [6]). But, for

$j, k = 0, 1, \ldots, n/2-2, n/2-1$, (101) and (102) show that $S_{j,k} = M_{j,k}$, so that S is also semiseparable.

Integrating by parts a few times, while using (2), (5), and (19), together with (101), yields explicit expressions for the leftmost entries $S_{0,0}$, $S_{0,1}$, $\ldots$, $S_{0,n/2-2}$, $S_{0,n/2-1}$ of S. Combining (2), (5), (23), and (101) yields explicit expressions for the diagonal entries $S_{0,0}$, $S_{1,1}$, $\ldots$, $S_{n/2-2,n/2-2}$, $S_{n/2-1,n/2-1}$ of S. Combining all of these explicit expressions with the fact that S is semiseparable, we obtain the formulae (81)–(84). □

3.2 Associated Legendre Expansions of Arbitrary Orders and Associated Legendre Expansions of Low Orders

Lemmas 3.3, 3.4, 3.5, and 3.6 of this subsection follow immediately from Lemma 3.1.

Lemma 3.3 states that the matrix of eigenvectors of the matrix $G^{n,m}$ in (79) converts the coefficients in the expansion of a function f in terms of the functions $\overline{P}^m_m$, $\overline{P}^m_{m+2}$, $\ldots$, $\overline{P}^m_{n-2}$, $\overline{P}^m_n$ into the coefficients in the expansion of the function f in terms of the functions $\overline{P}^2_2$, $\overline{P}^2_4$, $\ldots$, $\overline{P}^2_{n-2}$, $\overline{P}^2_n$. Lemma 3.3 states, moreover, that the adjoint of the matrix of eigenvectors converts the latter coefficients into the former coefficients.

Lemma 3.3 *Suppose that n and m are even integers such that $2 \le m \le n$, $p^m = (p^m_0, p^m_1, \ldots, p^m_{n/2-2}, p^m_{n/2-1})^{\mathrm{T}}$ is a real column vector such that $p^m_{(n-m)/2+1} = 0$, $p^m_{(n-m)/2+2} = 0$, $\ldots$, $p^m_{n/2-2} = 0$, $p^m_{n/2-1} = 0$, and f is the function defined on $[-1, 1]$ by the formula*

$$f(x) = \sum_{j=0}^{(n-m)/2} p^m_j \, \overline{P}^m_{2j+m}(x). \tag{106}$$

Then,

$$f(x) = \sum_{l=0}^{n/2-1} p^2_l \, \overline{P}^2_{2l+2}(x), \tag{107}$$

where $p^2 = (p^2_0, p^2_1, \ldots, p^2_{n/2-2}, p^2_{n/2-1})^{\mathrm{T}}$ is the real vector defined by the formula

$$p^2 = U\, p^m, \tag{108}$$

and U is an $n/2 \times n/2$ matrix of eigenvectors of the symmetric matrix $G^{n,m}$ in (79), with

$$\overline{P}^m_{2j+m}(x) = \sum_{l=0}^{n/2-1} U_{l,j} \, \overline{P}^2_{2l+2}(x) \tag{109}$$

for $j = 0, 1, \ldots, (n-m)/2-1, (n-m)/2$. Moreover,

$$p^m = U^{\mathrm{T}} \, p^2. \tag{110}$$

Lemmas 3.4, 3.5, and 3.6 are analogues of Lemma 3.3 for different conditions on m and n.

Lemma 3.4 *Suppose that n and m are integers such that m is even, n is odd, $2 \le m < n$, $p^m = (p^m_0, p^m_1, \ldots, p^m_{(n-5)/2}, p^m_{(n-3)/2})^{\mathrm{T}}$ is a real column vector such that $p^m_{(n-m+1)/2} = 0$, $p^m_{(n-m+3)/2} = 0, \ldots, p^m_{(n-5)/2} = 0$, $p^m_{(n-3)/2} = 0$, and f is the function defined on $[-1, 1]$ by the formula*

$$f(x) = \sum_{j=0}^{(n-m-1)/2} p^m_j \, \overline{P}^m_{2j+m+1}(x). \tag{111}$$

Then,

$$f(x) = \sum_{l=0}^{(n-3)/2} p^2_l \, \overline{P}^2_{2l+3}(x), \tag{112}$$

where $p^2 = (p^2_0, p^2_1, \ldots, p^2_{(n-5)/2}, p^2_{(n-3)/2})^{\mathrm{T}}$ is the real vector defined by the formula

$$p^2 = U \, p^m, \tag{113}$$

and U is an $(n-1)/2 \times (n-1)/2$ matrix of eigenvectors of the symmetric matrix $G^{n,m}$ in (79), with

$$\overline{P}^m_{2j+m+1}(x) = \sum_{l=0}^{(n-3)/2} U_{l,j} \, \overline{P}^2_{2l+3}(x) \tag{114}$$

for $j = 0, 1, \ldots, (n-m-3)/2, (n-m-1)/2$. Moreover,

$$p^m = U^{\mathrm{T}} \, p^2. \tag{115}$$

Lemma 3.5 *Suppose that n and m are integers such that m is odd, n is even, $1 \le m < n$, $p^m = (p^m_0, p^m_1, \ldots, p^m_{n/2-2}, p^m_{n/2-1})^{\mathrm{T}}$ is a real column vector such that $p^m_{(n-m+1)/2} = 0$, $p^m_{(n-m+3)/2} = 0, \ldots, p^m_{n/2-2} = 0$, $p^m_{n/2-1} = 0$, and f is the function defined on $[-1, 1]$ by the formula*

$$f(x) = \sum_{j=0}^{(n-m-1)/2} p^m_j \, \overline{P}^m_{2j+m+1}(x). \tag{116}$$

Then,

$$f(x) = \sum_{l=0}^{n/2-1} p^1_l \, \overline{P}^1_{2l+2}(x), \tag{117}$$

where $p^1 = (p^1_0, p^1_1, \ldots, p^1_{n/2-2}, p^1_{n/2-1})^{\mathrm{T}}$ is the real vector defined by the formula

$$p^1 = U \, p^m, \tag{118}$$

and U is an $n/2 \times n/2$ matrix of eigenvectors of the symmetric matrix $G^{n,m}$ in (79), with

$$\overline{P}^m_{2j+m+1}(x) = \sum_{l=0}^{n/2-1} U_{l,j} \, \overline{P}^1_{2l+2}(x) \tag{119}$$

for $j = 0, 1, \ldots, (n-m-3)/2, (n-m-1)/2$. Moreover,

$$p^m = U^{\mathrm{T}} \, p^1. \tag{120}$$

Lemma 3.6 *Suppose that n and m are odd integers such that $1 \le m \le n$, $p^m = (p_0^m,\, p_1^m,\, \ldots,\, p_{(n-3)/2}^m,\, p_{(n-1)/2}^m)^{\mathrm{T}}$ is a real column vector such that $p_{(n-m)/2+1}^m = 0$, $p_{(n-m)/2+2}^m = 0$, …, $p_{(n-3)/2}^m = 0$, $p_{(n-1)/2}^m = 0$, and f is the function defined on $[-1, 1]$ by the formula*

$$f(x) = \sum_{j=0}^{(n-m)/2} p_j^m \, \overline{P}_{2j+m}^m(x). \tag{121}$$

Then,

$$f(x) = \sum_{l=0}^{(n-1)/2} p_l^1 \, \overline{P}_{2l+1}^1(x), \tag{122}$$

where $p^1 = (p_0^1,\, p_1^1,\, \ldots,\, p_{(n-3)/2}^1,\, p_{(n-1)/2}^1)^{\mathrm{T}}$ is the real vector defined by the formula

$$p^1 = U\, p^m, \tag{123}$$

and U is an $(n+1)/2 \times (n+1)/2$ matrix of eigenvectors of the symmetric matrix $G^{n,m}$ in (79), with

$$\overline{P}_{2j+m}^m(x) = \sum_{l=0}^{(n-1)/2} U_{l,j}\, \overline{P}_{2l+1}^1(x) \tag{124}$$

for $j = 0, 1, \ldots, (n-m)/2 - 1, (n-m)/2$. Moreover,

$$p^m = U^{\mathrm{T}}\, p^1. \tag{125}$$

4 Informal Description of the Algorithm

In this section, we outline a "fast" algorithm for the conversion of the values $f_0, f_1, \ldots, f_{N-1}, f_N$ of the function f tabulated at the nodes $x_0, x_1, \ldots, x_{N-1}, x_N$, defined by the formula

$$x_k = \cos\left(\frac{\pi(k+1/2)}{N+1}\right), \tag{126}$$

into the coefficients $p_0, p_1, \ldots, p_{N-m-1}, p_{N-m}$ in the expansion

$$f(x) = \sum_{l=0}^{N-m} p_l\, \overline{P}_{l+m}^m(x). \tag{127}$$

(The inverse procedure of converting the coefficients $p_0, p_1, \ldots, p_{N-m-1}, p_{N-m}$ into the values $f_0, f_1, \ldots, f_{N-1}, f_N$ is quite similar, so we omit its description.)

The procedure consists of four steps, described briefly below. Since the procedure when m is odd is virtually the same as when m is even, we describe the procedure only for the case that m is even. For definitiveness, we assume also that N is even.

1. We separate f into its even and odd parts f^+ and f^-, defined by the formulae

$$f^+(x) = \frac{f(x) + f(-x)}{\sqrt{2}}, \tag{128}$$

$$f^-(x) = \frac{f(x) - f(-x)}{\sqrt{2}}. \tag{129}$$

All subsequent processing is performed separately for f^+ and f^-. Since the procedures for f^+ and f^- are virtually identical, we describe only the procedure for the even part f^+.

2. Using the Fast Fourier Transform as in Observation 2.5, we convert the values f_0^+, f_1^+, …, f_{N-1}^+, f_N^+ of the function f^+, defined in (128), at the points x_0, x_1, …, x_{N-1}, x_N, defined in (126), into the coefficients c_0, c_1, …, $c_{N/2-1}$, $c_{N/2}$ in the expansion

$$f^+(x) = \sum_{k=0}^{N/2} c_k \, T_{2k}(x). \tag{130}$$

3. Using Lemma 2.8 and Observation 2.10, we convert the coefficients c_0, c_1, …, $c_{N/2-1}$, $c_{N/2}$ in the expansion (130) into the coefficients p_0^2, p_1^2, …, $p_{N/2-2}^2$, $p_{N/2-1}^2$ in the expansion

$$f^+(x) = \sum_{l=0}^{N/2-1} p_l^2 \, \overline{P}_{2l+2}^2(x). \tag{131}$$

4. Using Lemmas 3.2 and 3.3 and Observation 2.11, we convert the coefficients p_0^2, p_1^2, …, $p_{N/2-2}^2$, $p_{N/2-1}^2$ in the expansion (131) into the coefficients p_0^m, p_1^m, …, $p_{(N-m)/2-1}^m$, $p_{(N-m)/2}^m$ in the expansion

$$f^+(x) = \sum_{l=0}^{(N-m)/2} p_l^m \, \overline{P}_{2l+m}^m(x). \tag{132}$$

Remark 4.1 Clearly, Step 1 takes $\mathcal{O}(N)$ operations. As per Observation 2.5, Step 2 takes $\mathcal{O}(N \log N)$ operations. As per Observation 2.10, Step 3 takes $\mathcal{O}(N \log(N/\varepsilon))$ operations, where ε is the precision of computations. As per Observation 2.11, Step 4 takes $\mathcal{O}(N(\log N)\log(1/\varepsilon))$ operations, where ε is the precision of computations used in this step. All together, Steps 1–4 take $\mathcal{O}(N(\log N)\log(1/\varepsilon))$ operations.

Remark 4.2 To handle f^- when m is even, we substitute Lemma 2.9 for Lemma 2.8 in Step 3 and Lemma 3.4 for Lemma 3.3 in Step 4. For f^- when m is even, we find that $p_{N/2-1}^2 = 0$ and $c_{N/2} = 0$, since, when m is even, f^- is a linear combination of only $N/2 - 1$ functions $\overline{P}_3^2$, $\overline{P}_5^2$, …, $\overline{P}_{N-3}^2$, $\overline{P}_{N-1}^2$ (or, equivalently, of the $N/2$ Chebyshev polynomials of the first kind T_1, T_3, …, T_{N-3}, T_{N-1}).

To handle f^+ when m is odd, we substitute Lemma 2.6 for Lemma 2.8 in Step 3 and Lemma 3.5 for Lemma 3.3 in Step 4.

To handle f^- when m is odd, we substitute Lemma 2.7 for Lemma 2.8 in Step 3 and Lemma 3.6 for Lemma 3.3 in Step 4.

When computing the coefficients (13) in the spherical harmonic expansion (6), we tabulate the function f on $[-1, 1]$ at the same $N+1$ sampling locations x_0, x_1, …, x_{N-1}, x_N when m is odd as when m is even, even though when m is odd, f is a linear combination of only N functions $\overline{P}_1^1$, $\overline{P}_2^1$, …, $\overline{P}_{N-1}^1$, $\overline{P}_N^1$ (or, equivalently, of the N Chebyshev polynomials of the second kind $\sqrt{1-x^2}\,U_0$, $\sqrt{1-x^2}\,U_1$, …, $\sqrt{1-x^2}\,U_{N-2}$, $\sqrt{1-x^2}\,U_{N-1}$, which are scaled by $\sqrt{1-x^2}$).

5 Detailed Description of the Algorithm

In this section, we describe in detail the algorithm described informally in Section 4.

Precomputations.

Comment [Compute all the data that Steps 3 and 4 require that do not depend on f.]
Compute all the data that Step 3 requires to apply fast the matrix $A^{N/2,2,+}$ from Lemma 2.8 (*i.e.*, "compress" the matrix $A^{N/2,2,+}$).
Compute all the data that Step 4 requires to apply fast the adjoint of the matrix U from Lemmas 3.2 and 3.3 (*i.e.*, "compress" the matrix U^{T}).

Step 1.

Comment [Convert the values f_0, f_1, $\dots$, f_{N-1}, f_N into the values f_0^+, f_1^+, $\dots$, f_{N-1}^+, f_N^+.]
do $n = 0, \dots, N$
    Set $f_n^+ = (f_n + f_{N-n})/\sqrt{2}$.
enddo

Step 2.

Comment [Convert the values f_0^+, f_1^+, $\dots$, f_{N-1}^+, f_N^+ into the coefficients c_0, c_1, $\dots$, $c_{N/2-1}$, $c_{N/2}$ of the Chebyshev polynomials T_0, T_2, $\dots$, T_{N-2}, T_N.]
Use the Discrete Cosine Transform (see, for example, [14]) to convert the values f_0^+, f_1^+, $\dots$, f_{N-1}^+, f_N^+ into the coefficients c_0, c_1, $\dots$, $c_{N/2-1}$, $c_{N/2}$.

Step 3.

Comment [Convert the coefficients c_0, c_1, $\dots$, $c_{N/2-1}$, $c_{N/2}$ of the Chebyshev polynomials T_0, T_2, $\dots$, T_{N-2}, T_N into the coefficients p_0^2, p_1^2, $\dots$, $p_{N/2-2}^2$, $p_{N/2-1}^2$ of the associated Legendre functions $\overline{P}_2^2$, $\overline{P}_4^2$, $\dots$, $\overline{P}_{N-2}^2$, $\overline{P}_N^2$.]
Use Observation 2.10 to apply the matrix $A^{N/2,2,+}$ from Lemma 2.8 to the vector $c = (c_0, c_1, \dots, c_{N/2-1}, c_{N/2})^{\mathrm{T}}$, in order to obtain the vector $p^2 = (p_0^2, p_1^2, \dots, p_{N/2-2}^2, p_{N/2-1}^2)^{\mathrm{T}}$.

Step 4.

Comment [Convert the coefficients p_0^2, p_1^2, $\dots$, $p_{N/2-2}^2$, $p_{N/2-1}^2$ of the associated Legendre functions $\overline{P}_2^2$, $\overline{P}_4^2$, $\dots$, $\overline{P}_{N-2}^2$, $\overline{P}_N^2$ into the coefficients p_0^m, p_1^m, $\dots$, $p_{(N-m)/2-1}^m$, $p_{(N-m)/2}^m$ of the associated Legendre functions $\overline{P}_m^m$, $\overline{P}_{m+2}^m$, $\dots$, $\overline{P}_{N-2}^m$, $\overline{P}_N^m$.]
Use Observation 2.11 to apply the adjoint of the matrix U from Lemmas 3.2 and 3.3 to the vector $p^2 = (p_0^2, p_1^2, \dots, p_{N/2-2}^2, p_{N/2-1}^2)^{\mathrm{T}}$ in order to obtain the vector $p^m = (p_0^m, p_1^m, \dots, p_{N/2-2}^m, p_{N/2-1}^m)^{\mathrm{T}}$.
Comment [Only the first $(N-m)/2+1$ entries of p^m interest us; the other entries vanish: $p_{(N-m)/2+1}^m = 0$, $p_{(N-m)/2+2}^m = 0$, $\dots$, $p_{N/2-2}^m = 0$, $p_{N/2-1}^m = 0$.]

6 Numerical Results

The algorithms described in this paper have been implemented in Fortran. Tables 1–8 below report the results of applying the algorithm to functions f defined on $[-1, 1]$ by the formulae

$$f(x) = \sum_{l=0}^{(N-m)/2} \varepsilon_l \, \overline{P}^m_{2l+m}(x) \tag{133}$$

when m is even and degrees are even,

$$f(x) = \sum_{l=0}^{(N-m-2)/2} \varepsilon_l \, \overline{P}^m_{2l+m+1}(x) \tag{134}$$

when m is even and degrees are odd,

$$f(x) = \sum_{l=0}^{(N-m-1)/2} \varepsilon_l \, \overline{P}^m_{2l+m+1}(x) \tag{135}$$

when m is odd and degrees are even, and

$$f(x) = \sum_{l=0}^{(N-m-1)/2} \varepsilon_l \, \overline{P}^m_{2l+m}(x) \tag{136}$$

when m is odd and degrees are odd; ε_0, ε_1, ε_2, ... are randomly generated numbers in the interval $[-1, 1]$. The decomposition algorithm computes from the sample values of f the coefficients in the representation of f as a linear combination of the functions $\overline{P}^m_m$, $\overline{P}^m_{m+1}$, ..., $\overline{P}^m_{N-1}$, $\overline{P}^m_N$; the reconstruction algorithm computes the sample values of f from the coefficients in the representation of f as a linear combination of the functions $\overline{P}^m_m$, $\overline{P}^m_{m+1}$, ..., $\overline{P}^m_{N-1}$, $\overline{P}^m_N$. The CPU times are in seconds. The errors are relative root-mean-square errors.

The columns labeled "'fast' transformation" list the times taken by the algorithm to evaluate one sum of the form (133), (134), (135), or (136).

The columns labeled "third of applying an $N/2 \times N/2$ matrix" list the times, divided by 3, taken to apply an $N/2 \times N/2$ matrix once to a vector of length $N/2$. These times scale quadratically with N; the figures in parentheses are estimates used when the memory required by direct calculations would be excessive.

Remark 6.1 The columns labeled "third of applying an $N/2 \times N/2$ matrix" give some indication of how much time current implementations of the decompositions and reconstructions take to run; these times are believed to be reasonable estimates of how a standard package like Spherepack (see [2]) would perform, using the "semi-naive" algorithm described in [10], for example. Clearly, for sufficiently small N, the "slow" implementation would actually run faster than the "fast" implementation.

The code was compiled with the Lahey-Fujitsu compiler, with optimization flag `--o2`, and run on a 2.8 GHz Intel Pentium Xeon microprocessor with 512 Kb of L2 cache.

No effort was made to optimize the precomputations. To simplify the implementation, precomputations that take $\mathcal{O}(N^2)$ operations were used, even though the techniques described in [4] lead naturally to precomputations that would take only $\mathcal{O}(N \log N)$ operations. The precomputations were run to yield approximately 6 digits of accuracy, running all computations (including the precomputations) in double precision arithmetic.

Observation 6.2 *Asymptotically, the algorithm should require a number of operations proportional to $N(\log N)\log(1/\varepsilon)$ to evaluate one sum of the form (133), (134), (135), or (136), where $\varepsilon \approx 10^{-6}$. The times in the columns labeled "'fast' transformation" appear to be consistent with this estimate.*

7 Generalizations

The algorithm of this paper admits a number of generalizations and extensions. The list below is not intended to be exhaustive; subjects in it are under investigation, and will be reported at a later date.

1. Different discretizations of S^2. Throughout this paper, we have assumed that the meridians on S^2 are discretized in an equispaced manner, *i.e.*, that the points θ_0, θ_1, $\dots$, θ_{N-1}, θ_N subdivide the interval $[0, \pi]$ into equal subintervals. This limitation is easily removed via techniques described in [11], [16], and [5]; however, to maintain numerical stability, the nodes have to satisfy certain quite restrictive criteria. One important collection of nodes that does in fact lead to stable algorithms is given by the formula

$$\theta_k = \cos^{-1}(x_k), \tag{137}$$

where x_0, x_1, $\dots$, x_{N-1}, x_N are Gaussian nodes on the interval $[-1, 1]$ (see, for example, [2]).

Furthermore, the nodes φ_0, φ_1, $\dots$, φ_{2N-1}, φ_{2N} in the discretizations of the parallels on S^2 do not have to be equispaced, provided some form of the non-equispaced Fast Fourier Transform is used; again, numerical stability requires that φ_0, φ_1, $\dots$, φ_{2N-1}, φ_{2N} be fairly close to being equispaced.

2. Associated Laguerre functions. Associated Laguerre functions are defined on the entire the half-line $[0, \infty)$, but are very small outside of a finite interval. The Sturm-Liouville problem that generates the associated Laguerre functions becomes an eigenvector problem for the sum of a diagonal matrix and a semiseparable matrix when discretized using associated Laguerre functions of low orders (very much like the associated Legendre functions).

3. Prolate Spheroidal Wave Functions. The Sturm-Liouville problem that generates the Prolate Spheroidal Wave Functions becomes an eigenvector problem for a tridiagonal matrix when discretized using Legendre polynomials.

4. Associated Prolate Spheroidal Wave Functions. The Sturm-Liouville problem that generates the associated Prolate Spheroidal Wave Functions becomes an eigenvector problem for the sum of a tridiagonal matrix and a semiseparable matrix when discretized using associated Legendre functions of low orders.

8 Acknowledgements

The authors would like to thank R. R. Coifman for many useful and inspiring discussions.

Table 1: Times in seconds and errors for $m = \frac{N+6}{2}$, even degrees, reconstruction

N	m	"fast" trans-formation	third of applying an $\frac{N}{2} \times \frac{N}{2}$ matrix	precomp-utation	relative r.m.s. error
1024	515	.20E-02	.91E-03	.20E+01	.64E-07
2048	1027	.52E-02	.36E-02	.80E+01	.58E-07
4096	2051	.23E-01	.15E-01	.33E+02	.17E-06
8192	4099	.48E-01	.58E-01	.16E+03	.39E-06
16384	8195	.11E+00	.23E+00	.69E+03	.72E-06
32768	16387	.24E+00	.96E+00	.36E+04	.14E-05
65536	32771	.51E+00	(.37E+01)	.14E+05	.52E-06
131072	65539	.11E+01	(.15E+02)	.63E+05	.70E-05

Table 2: Times in seconds and errors for $m = \frac{N+6}{2}$, even degrees, decomposition

N	m	"fast" trans-formation	third of applying an $\frac{N}{2} \times \frac{N}{2}$ matrix	precomp-utation	relative r.m.s. error
1024	515	.19E-02	.91E-03	.30E+01	.73E-08
2048	1027	.49E-02	.36E-02	.80E+01	.70E-08
4096	2051	.22E-01	.15E-01	.32E+02	.76E-08
8192	4099	.47E-01	.58E-01	.16E+03	.11E-07
16384	8195	.11E+00	.23E+00	.68E+03	.13E-07
32768	16387	.23E+00	.96E+00	.35E+04	.15E-07
65536	32771	.50E+00	(.37E+01)	.14E+05	.20E-07
131072	65539	.12E+01	(.15E+02)	.62E+05	.35E-07

Table 3: Times in seconds and errors for $m = \frac{N-4}{2}$, even degrees, reconstruction

N	m	"fast" trans-formation	third of applying an $\frac{N}{2} \times \frac{N}{2}$ matrix	precomp-utation	relative r.m.s. error
1024	510	.20E-02	.91E-03	.20E+01	.77E-07
2048	1022	.53E-02	.36E-02	.90E+01	.72E-07
4096	2046	.23E-01	.15E-01	.38E+02	.65E-07
8192	4094	.49E-01	.58E-01	.17E+03	.15E-06
16384	8190	.11E+00	.23E+00	.78E+03	.41E-06
32768	16382	.24E+00	.96E+00	.34E+04	.54E-06
65536	32766	.52E+00	(.37E+01)	.15E+05	.31E-06
131072	65534	.11E+01	(.15E+02)	.64E+05	.41E-05

Table 4: Times in seconds and errors for $m = \frac{N-4}{2}$, even degrees, decomposition

N	m	"fast" trans-formation	third of applying an $\frac{N}{2} \times \frac{N}{2}$ matrix	precomp-utation	relative r.m.s. error
1024	510	.19E-02	.91E-03	.20E+01	.37E-08
2048	1022	.51E-02	.36E-02	.90E+01	.79E-08
4096	2046	.22E-01	.15E-01	.38E+02	.95E-08
8192	4094	.48E-01	.58E-01	.16E+03	.11E-07
16384	8190	.11E+00	.23E+00	.77E+03	.16E-07
32768	16382	.24E+00	.96E+00	.33E+04	.15E-07
65536	32766	.51E+00	(.37E+01)	.16E+05	.17E-07
131072	65534	.11E+01	(.15E+02)	.62E+05	.21E-07

Table 5: Times in seconds and errors for $m = \frac{N-32}{32}$, odd degrees, reconstruction

N	m	"fast" transformation	third of applying an $\frac{N}{2} \times \frac{N}{2}$ matrix	precomputation	relative r.m.s. error
1024	31	.16E-02	.91E-03	.20E+01	.86E-08
2048	63	.43E-02	.36E-02	.70E+01	.20E-07
4096	127	.20E-01	.15E-01	.28E+02	.16E-06
8192	255	.43E-01	.58E-01	.11E+03	.41E-06
16384	511	.94E-01	.23E+00	.56E+03	.60E-06
32768	1023	.21E+00	.96E+00	.26E+04	.10E-05
65536	2047	.46E+00	(.37E+01)	.14E+05	.14E-05
131072	4095	.10E+01	(.15E+02)	.47E+05	.14E-05

Table 6: Times in seconds and errors for $m = \frac{N-32}{32}$, odd degrees, decomposition

N	m	"fast" transformation	third of applying an $\frac{N}{2} \times \frac{N}{2}$ matrix	precomputation	relative r.m.s. error
1024	31	.15E-02	.91E-03	.20E+01	.71E-08
2048	63	.41E-02	.36E-02	.70E+01	.59E-08
4096	127	.20E-01	.15E-01	.27E+02	.82E-08
8192	255	.42E-01	.58E-01	.12E+03	.76E-08
16384	511	.92E-01	.23E+00	.55E+03	.12E-07
32768	1023	.21E+00	.96E+00	.25E+04	.13E-07
65536	2047	.45E+00	(.37E+01)	.11E+05	.11E-07
131072	4095	.98E+00	(.15E+02)	.46E+05	.20E-07

Table 7: Times in seconds and errors for $m = \frac{15N}{16}$, odd degrees, reconstruction

N	m	"fast" trans-formation	third of applying an $\frac{N}{2} \times \frac{N}{2}$ matrix	precomp-utation	relative r.m.s. error
1024	960	.21E-02	.91E-03	.30E+01	.12E-07
2048	1920	.55E-02	.36E-02	.11E+02	.20E-07
4096	3840	.23E-01	.15E-01	.39E+02	.48E-07
8192	7680	.49E-01	.58E-01	.16E+03	.14E-06
16384	15360	.11E+00	.23E+00	.83E+03	.15E-06
32768	30720	.24E+00	.96E+00	.36E+04	.23E-06
65536	61440	.53E+00	(.37E+01)	.21E+05	.12E-05
131072	122880	.11E+01	(.15E+02)	.62E+05	.15E-05

Table 8: Times in seconds and errors for $m = \frac{15N}{16}$, odd degrees, decomposition

N	m	"fast" trans-formation	third of applying an $\frac{N}{2} \times \frac{N}{2}$ matrix	precomp-utation	relative r.m.s. error
1024	960	.20E-02	.91E-03	.30E+01	.48E-08
2048	1920	.52E-02	.36E-02	.10E+02	.74E-08
4096	3840	.23E-01	.15E-01	.38E+02	.10E-07
8192	7680	.49E-01	.58E-01	.16E+03	.11E-07
16384	15360	.11E+00	.23E+00	.81E+03	.13E-07
32768	30720	.24E+00	.96E+00	.35E+04	.18E-07
65536	61440	.51E+00	(.37E+01)	.16E+05	.16E-07
131072	122880	.11E+01	(.15E+02)	.60E+05	.27E-07

References

[1] M. Abramovitz and I. Stegun, eds., *Handbook of Mathematical Functions*, Dover, New York, 1972.

[2] J. C. Adams and P. N. Swarztrauber, *SPHEREPACK 3.0: A model development facility*, Monthly Weather Review, 127 (1999), pp. 1872–1878.

[3] B. Alpert and V. Rokhlin, *A fast algorithm for the evaluation of Legendre expansions*, SIAM Journal of Scientific and Statistical Computing, 12 (1991), pp. 158–179.

[4] S. Chandrasekaran and M. Gu, *A divide-and-conquer algorithm for the eigendecomposition of symmetric block-diagonal plus semiseparable matrices*, Numerische Mathematik, 96 (2004), pp. 723–731.

[5] A. Dutt, M. Gu, and V. Rokhlin, *Fast algorithms for polynomial interpolation, integration, and differentiation*, Journal of Numerical Analysis, 33 (1996), pp. 1689–1711.

[6] F. R. Gantmacher and M. G. Krein, *Oscillation Matrices and Small Vibrations of Mechanical Systems*, American Mathematical Society Chelsea Publishing, Providence, RI, revised english ed., 2002.

[7] I. S. Gradshteyn and I. M. Ryzhik, *Tables of Integrals, Series, and Products*, Academic Press, San Diego, CA, 6th ed., 2000. Alan Jeffrey and Daniel Zwillinger, eds.

[8] M. Gu and S. C. Eisenstat, *A divide-and-conquer algorithm for the symmetric tridiagonal eigenproblem*, SIAM Journal on Matrix Analysis and Applications, 16 (1995), pp. 172–191.

[9] D. M. Healy, P. J. Kostelec, and D. Rockmore, *Towards safe and effective high-order Legendre transforms with applications to FFTs for the 2-sphere*, Advances in Computational Mathematics, 21 (2004), pp. 59–105.

[10] D. M. Healy, P. J. Kostelec, D. Rockmore, and S. S. B. More, *FFTs for the 2-sphere — improvements and variations*, Journal of Fourier Analysis and Applications, 9 (2003), pp. 341–385.

[11] R. Jacob-Chien and B. K. Alpert, *A fast spherical filter with uniform resolution*, Journal of Computational Physics, 136 (1997), pp. 580–584.

[12] S. Kunis and D. Potts, *Fast spherical Fourier algorithms*, Journal of Computational and Applied Mathematics, 161 (2003), pp. 75–98.

[13] M. J. Mohlenkamp, *A fast transform for spherical harmonics*, Journal of Fourier Analysis and Applications, 5 (1999), pp. 159–184.

[14] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in Fortran 77*, Cambridge University Press, Cambridge, UK, second ed., 1992.

[15] R. Suda and M. Takami, *A fast spherical harmonics transform algorithm*, Mathematics of Computation, 71 (2002), pp. 703–715.

[16] N. Yarvin and V. Rokhlin, *A generalized one-dimensional Fast Multipole Method with application to filtering of spherical harmonics*, Journal of Computational Physics, 147 (1998), pp. 594–609.