# Fast and backward stable transforms between spherical harmonic expansions and bivariate Fourier series

Richard Mikaël Slevinsky[a,*]

[a]*Department of Mathematics, University of Manitoba, Winnipeg, Manitoba, Canada*

**Abstract**

A rapid transformation is derived between spherical harmonic expansions and their analogues in a bivariate Fourier series. The change of basis is described in two steps: firstly, expansions in normalized associated Legendre functions of all orders are converted to those of order zero and one; then, these intermediate expressions are re-expanded in trigonometric form. The first step proceeds with a butterfly factorization of the well-conditioned matrices of connection coefficients. The second step proceeds with fast orthogonal polynomial transforms via hierarchically off-diagonal low-rank matrix decompositions. Total pre-computation requires at best $O(n^3 \log n)$ flops; and, asymptotically optimal execution time of $O(n^2 \log^2 n)$ is rigorously proved via connection to Fourier integral operators.

*Keywords:* Spherical harmonics, connection coefficients, hierarchical matrices, semiseparable matrices, butterfly factorization, interpolative decomposition
*2010 MSC:* 33C55, 65F99, 65Y99.

## 1. Introduction

Spherical harmonics are the natural basis for square integrable functions on the sphere in the sense of Lebesgue. As eigenfunctions of the Laplace–Beltrami operator, spherical harmonics are mathematically elegant; and with orthonormalization, stable numerical methods are available for tensor calculus. Another basis for the sphere is derived by doubling the co-latitudinal angle and extending the original function (or data) with block-mirror centrosymmetry. A bivariate Fourier basis results, and by the grace of the fast Fourier transform (FFT) comes rapid evaluation at points equispaced-in-angle. Through nonuniform FFTs, rapid evaluation is extended to bivariate grids with little to no regard for their distribution [1, 2, 3, 4]. Townsend, Wilber, and Wright [5] have recently experimented with low-rank compression techniques for bivariate Fourier series on the sphere.

Several numerical methods exist for the purposes of the synthesis and analysis of spherical harmonic expansions; that is, the rapid evaluation of spherical harmonic expansions at points (nearly) equispaced-in-angle and the determination of expansion coefficients by data at such points. Healy et al. [6] derive the first modern approach based on the so-called split-Legendre functions and the Driscoll–Healy sampling theorem [7]. Suda and Takami [8] and Kunis and Potts [9] accelerate the implied polynomial interpolation by using the Fast Multipole Method (FMM). On the basis of classical WKB asymptotics, Mohlenkamp [10] accelerates synthesis and analysis by constructing localized numerical approximations that incorporate phase information. More recently, Rokhlin and Tygert [11] combine the theory of spectral connection matrices of associated Legendre functions with Chandrasekaran and Gu's fast eigensolver for symmetric block-diagonal plus semiseparable matrices [12] to derive the first fast algorithms with asymptotically optimal pre-computation, though the optimal complexity is predicted "only for absurdly large degrees." Tygert completes his trilogy with significant improvements to the first methodology [13] and a new method [14] based on the butterfly algorithm [15, 16]. Tygert's method has been implemented in several real-world applications [17, 18, 19]. Thus for a

---

new contribution, it is as important to maximize the practicality of the spherical harmonic basis as it is to completely eliminate pre-computation.

Shortly after the turn of the 20th century, Schuster published foundational work on the Fourier coefficients of spherical harmonics [20]. In the computer era, Hofsommer and Potters [21] show how one may implement the conversion in practice, and the Fourier coefficients have been more recently studied as well [22, 23]. Although numerical evidence is presented to suggest recursion-through-order is stable in the downward direction, extended precision is still required and the $O(n^3)$ complexity of the conversion is not asymptotically optimal. Conversion from spherical harmonics to bivariate Fourier series is not one-to-one. In order to make sense of inversion, a weighted least-squares solution is proposed in [21]. However, when the problem is properly subdivided, a more natural least-squares problem awaits.

The present work may be viewed as an extension of the butterfly algorithm to the connection problem between spherical harmonic expansions and their bivariate Fourier series. A change of basis implies several advantages over the full synthesis and analysis. Firstly, certain subproblems are well-conditioned and backward stable algorithms are derived for their numerical solution; this obviates the necessity for extended precision [13, 14], as Gaussian quadratures are not part of the algorithm. Secondly, in the light of the connection problem, a so-called *rank property* of the matrices of connection coefficients is rigorously proved, establishing the asymptotically optimal complexity $O(n^2 \log^2 n)$. This rank property is also universal in its proof of Tygert's acceleration of synthesis and analysis as well [14]. Lastly, the formulation as a connection problem presents the possibility to improve the pre-computation by requiring a mere skeleton of the full plan as fast and backward stable algorithms may convert between expansions in neighbouring orders. The main technological asset of the butterfly algorithm is the interpolative decomposition; algorithms for interpolative decompositions include [24, 25]. An open source implementation is available in the software package `FastTransforms.jl` [26].

Two prominent numerical methods for evolving stiff time-dependent partial differential equations are exponential integrators such as the fourth-order Runge–Kutta scheme ETDRK4 [27], and implicit-explicit (IMEX) Runge–Kutta methods such as those developed in [28]. While IMEX schemes are versatile when considering more general stiff linear operators, their order may drop to one in the stiff limit. On the other hand, exponential integrators deal with the stiffness exactly, but an efficient implementation requires a basis that diagonalizes the stiff linear operator. In the basis of spherical harmonics, certain linear differential operators such as the Laplace–Beltrami operator are diagonal. Similarly, common nonlinearities are diagonalized by pointwise evaluation on a grid. Thus a method that utilizes both spherical harmonics and bivariate Fourier series and may stably convert between representations is ideal for time evolution.

The conversion of expansions in associated Legendre functions to those involving zeroth and first orders is useful for the solution of partial differential equations with initial conditions in the spherical harmonic basis and whose partial differential operators on the sphere have low splitting rank. On the sphere, this offers the advantage of resolution in an isometric Hilbert space. A global spectral method for such problems is derived by Townsend and Olver [29].

That spherical harmonics diagonalize certain singular integral operators on the surface of the sphere could be the basis for extending the ultraspherical spectral method for singular integral equations [30], which results in highly-structured linear systems, to multi-sphere scattering with spectral accuracy.

## 2. Fundamentals

Let $\mu$ be a positive Borel measure on $D \subset \mathbb{R}^n$. The inner product:

$$\langle f, g \rangle = \int_D \overline{f(x)} g(x) \, \mathrm{d}\mu(x), \tag{1}$$

where $\overline{f(x)}$ denotes complex conjugation, induces the norm $\|f\|_2 = \sqrt{\langle f, f \rangle}$ and the associated Hilbert space $L^2(D, \mathrm{d}\mu(x))$. In case of ambiguity, the notation $\langle f, g \rangle_{\mathrm{d}\mu}$ is used to distinguish between different measures.

Let $\mathbb{S}^2 \subset \mathbb{R}^3$ denote the unit 2-sphere, $\theta \in [0, \pi]$ the co-latitudinal angle, $\varphi \in [0, 2\pi)$ the longitudinal angle, and $\mathrm{d}\Omega = \sin\theta \, \mathrm{d}\theta \, \mathrm{d}\varphi$ the measure generated by the solid angle $\Omega$ subtended by a spherical cap. Then, any function

$f \in L^2(\mathbb{S}^2, d\Omega)$ may be expanded in spherical harmonics:

$$f(\theta, \varphi) = \sum_{\ell=0}^{+\infty} \sum_{m=-\ell}^{+\ell} f_\ell^m Y_\ell^m(\theta, \varphi) = \sum_{m=-\infty}^{+\infty} \sum_{\ell=|m|}^{+\infty} f_\ell^m Y_\ell^m(\theta, \varphi), \tag{2}$$

where the expansion coefficients are:

$$f_\ell^m = \frac{\langle Y_\ell^m, f \rangle}{\langle Y_\ell^m, Y_\ell^m \rangle}. \tag{3}$$

Let $\mathbb{N}_0$ denote the non-negative integers. Bandlimiting Eq. (2) to $\ell \le n \in \mathbb{N}_0$ results in the best degree-$n$ trigonometric polynomial approximation of $f \in L^2(\mathbb{S}^2, d\Omega)$.

Using the Condon–Shortley phase convention [31], orthonormal spherical harmonics are given by:

$$Y_\ell^m(\theta, \varphi) = \frac{e^{im\varphi}}{\sqrt{2\pi}} \underbrace{i^{m+|m|} \sqrt{(\ell + \tfrac{1}{2}) \frac{(\ell - m)!}{(\ell + m)!}} P_\ell^m(\cos\theta)}_{\tilde{P}_\ell^m(\cos\theta)}, \qquad \ell \in \mathbb{N}_0, \quad -\ell \le m \le \ell, \tag{4}$$

and we will find the connection between associated Legendre functions and ultraspherical polynomials [32, §18.11.1] particularly useful:

$$P_\ell^m(\cos\theta) = (-2)^m (\tfrac{1}{2})_m (\sin\theta)^m C_{\ell-m}^{(m+\frac{1}{2})}(\cos\theta). \tag{5}$$

In Eq. (4), the notation $\tilde{P}_\ell^m$ is used to denote orthonormality for fixed $m$ in the sense of $L^2([-1, 1], dx)$. In Eq. (5), $(x)_n = \frac{\Gamma(x+n)}{\Gamma(x)}$ is the Pochhammer symbol [33].

### 2.1. The spherical harmonic connection problem

Expansions in families of orthogonal functions may be related by a change of basis. This defines the classical connection problem.

**Definition 2.1.** *Let* $\{\phi_n(x)\}_{n \in \mathbb{N}_0}$ *be a family of orthogonal functions with respect to* $L^2(\hat{D}, d\hat{\mu}(x))$ *and let* $\{\psi_n(x)\}_{n \in \mathbb{N}_0}$ *be another family of orthogonal functions with respect to* $L^2(D, d\mu(x))$. *The connection coefficients:*

$$c_{\ell,n} = \frac{\langle \psi_\ell, \phi_n \rangle_{d\mu}}{\langle \psi_\ell, \psi_\ell \rangle_{d\mu}}, \tag{6}$$

*allow for the expansion:*

$$\phi_n(x) = \sum_{\ell=0}^{\infty} c_{\ell,n} \psi_\ell(x). \tag{7}$$

**Theorem 2.2.** *Let* $\{\phi_n(x)\}_{n \in \mathbb{N}_0}$ *and* $\{\psi_n(x)\}_{n \in \mathbb{N}_0}$ *be two families of* orthonormal *functions with respect to* $L^2(D, d\mu(x))$. *Then the connection coefficients satisfy:*

$$\sum_{\ell=0}^{\infty} \overline{c_{\ell,m}} c_{\ell,n} = \delta_{m,n}, \tag{8}$$

*where* $\delta_{m,n}$ *is the Kronecker delta function [33, Chap. 24].*

*Proof.* We expand the orthonormal functions $\phi_m$ and $\phi_n$ in the basis $\{\psi_\ell\}_{\ell \in \mathbb{N}_0}$, and integrate the product:

$$\delta_{m,n} = \langle \phi_m, \phi_n \rangle = \int_D \overline{\left( \sum_{\ell=0}^{\infty} c_{\ell,m} \psi_\ell(x) \right)} \left( \sum_{\ell=0}^{\infty} c_{\ell,n} \psi_\ell(x) \right) d\mu(x) = \sum_{\ell=0}^{\infty} \overline{c_{\ell,m}} c_{\ell,n}. \tag{9}$$

$\square$

Since the space of algebraic polynomials of degree at most $n$ is finite-dimensional, if the source and target families of orthogonal functions consist of a weight function multiplying algebraic polynomials, then the expansion of one family in the other basis is finite-dimensional and the connection coefficients populate a matrix. If, in addition, the source and target families are orthonormal, the matrix of connection coefficients has orthonormal columns and thereby unit singular values. Conversion of expansions between families of orthonormal functions in the same finite-dimensional Hilbert space is therefore well-conditioned in the sense of the induced matrix 2-norm.

Any matrix $A \in \mathbb{R}^{m \times n}$, $m \geq n$, with orthonormal columns has the Moore–Penrose pseudoinverse $A^+ = A^\top$. Where the classical connection problem is finite-dimensional but rectangular, the Moore–Penrose pseudoinverse provides a least-squares solution to the inversion problem.

While much is gained by discussing connection coefficients in such generality, we will now consider the connection coefficients pertinent to the spherical harmonics. The normalized associated Legendre functions are a family of orthonormal functions for the same Hilbert space $L^2([-1, 1], \mathrm{d}x)$. In fact, all even-ordered $\tilde{P}_\ell^m(x)$ are polynomials and all odd-ordered $\tilde{P}_\ell^m(x)$ are polynomials multiplied by the weight $\sqrt{1 - x^2}$.

The three-term recurrence relations for the associated Legendre functions may be combined to produce a five-term recurrence relation satisfied by the connection coefficients [34]. However, it is theoretically challenging to prove backward stability of this formulation even though the problem is well-conditioned. Furthermore, the formulation via recurrence relations either requires $O(n^3)$ square roots, $O(n^3)$ flops, and $O(1)$ storage for on-the-fly calculation or $O(n^3)$ flops and $O(n^3)$ storage if the matrices are pre-computed. Either case presents us with a significant compromise. Instead, we will derive a *backward stable* method that either requires $O(n^2)$ square roots, $O(n^3)$ flops, and $O(1)$ storage for on-the-fly calculation or $O(n^3)$ flops and $O(n^2)$ storage if the representation is pre-computed.

**Definition 2.3.** *Let $G_n$ denote the real Givens rotation:*

$$
G_n = \begin{bmatrix}
1 & \cdots & 0 & 0 & 0 & \cdots & 0 \\
\vdots & \ddots & \vdots & \vdots & \vdots & & \vdots \\
0 & \cdots & c_n & 0 & s_n & \cdots & 0 \\
0 & \cdots & 0 & 1 & 0 & \cdots & 0 \\
0 & \cdots & -s_n & 0 & c_n & \cdots & 0 \\
\vdots & & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & \cdots & 0 & 0 & 0 & \cdots & 1
\end{bmatrix},
$$

*where the sines $s_n = \sin\theta_n$ and the cosines $c_n = \cos\theta_n$, for some $\theta_n \in [0, 2\pi)$, are in the intersections of the $n^{\mathrm{th}}$ and $n + 2^{\mathrm{nd}}$ rows and columns, embedded in the identity of a conformable size.*

Let $I_{m \times n}$ denote the rectangular identity matrix with ones on the main diagonal and zeros everywhere else.

**Theorem 2.4.** *The connection coefficients between $\tilde{P}_{n+m+2}^{m+2}(\cos\theta)$ and $\tilde{P}_{\ell+m}^m(\cos\theta)$ are:*

$$
c_{\ell,n}^m = \begin{cases}
(2\ell + 2m + 1)(2m + 2)\sqrt{\dfrac{(\ell + 2m)!}{(\ell + m + \frac{1}{2})\ell!} \dfrac{(n + m + \frac{5}{2})n!}{(n + 2m + 4)!}}, & \text{for} \quad \ell \leq n, \quad \ell + n \text{ even,} \\
-\sqrt{\dfrac{(n + 1)(n + 2)}{(n + 2m + 3)(n + 2m + 4)}}, & \text{for} \qquad \ell = n + 2, \\
0, & \text{otherwise.}
\end{cases}
\tag{10}
$$

*Furthermore, the matrix of connection coefficients $C^{(m)} \in \mathbb{R}^{(n+3) \times (n+1)}$ may be represented via the product of $n$ Givens rotations:*

$$
C^{(m)} = G_0^{(m)} G_1^{(m)} \cdots G_{n-2}^{(m)} G_{n-1}^{(m)} I_{(n+3) \times (n+1)},
$$

*where the sines and cosines for the Givens rotations are given by:*

$$
s_n^m = \sqrt{\frac{(n + 1)(n + 2)}{(n + 2m + 3)(n + 2m + 4)}}, \quad \text{and} \quad c_n^m = \sqrt{\frac{(2m + 2)(2n + 2m + 5)}{(n + 2m + 3)(n + 2m + 4)}}.
\tag{11}
$$

*Proof.* Both recurrence relations involving ultraspherical polynomials [32, §18.9.7 & §18.9.8] are essentially:

$$C_n^{(m+\frac{1}{2})}(\cos\theta) = \frac{2m+1}{2n+2m+1}\left(C_n^{(m+\frac{3}{2})}(\cos\theta) - C_{n-2}^{(m+\frac{3}{2})}(\cos\theta)\right), \quad \text{and}$$

$$\sin^2\theta C_n^{(m+\frac{5}{2})}(\cos\theta) = \frac{(n+2m+3)(n+2m+4)}{(2m+3)(2n+2m+5)}C_n^{(m+\frac{3}{2})}(\cos\theta) - \frac{(n+1)(n+2)}{(2m+3)(2n+2m+5)}C_{n+2}^{(m+\frac{3}{2})}(\cos\theta).$$

Thus, the entire transformation from an expansion in $\sin^2\theta C_n^{(m+\frac{5}{2})}(\cos\theta)$ to an expansion in $C_n^{(m+\frac{1}{2})}(\cos\theta)$ may be represented by the product of the inverse of a square upper-triangular banded matrix and a rectangular banded matrix both with bandwidth 2. The inverse of an upper-triangular banded matrix of bandwidth 2 is a diagonal-plus-upper-semiseparable matrix of semiseparability rank 2. Translating this result into the notation of normalized associated Legendre functions, we arrive at $c_{\ell,n}^m$ in Eq. (10).

The parity in the source and target bases encodes a chessboard pattern of zeros in $C^{(m)}$. Start by applying a Givens rotation from the left to introduce a zero in the third row of the first column. Since the columns of $C^{(m)}$ are orthonormal, $s_0^m = -c_{2,0}^m$. Apply another Givens rotation from the left to introduce a zero in the fourth row of the second column of the conversion matrix. Again, we find that $s_1^m = -c_{3,1}^m$. Due to the orthonormality, the first rotation introduces zeros in every entry of the first row but the first. Similarly, the second rotation introduces zeros in every entry of the second row but the second. Continuing with $n-2$ more Givens rotations, we arrive at $I_{(n+3)\times(n+1)}$. □

To illustrate the order of operations, the matrices of connection coefficients may be represented as:



The arrows indicate which rows are altered by a Givens rotation; rotations nearest the rectangular identity are applied first.

For a real number $x$, we denote floating-point approximations to quantities by $\mathrm{fl}(x)$. Using the IEEE 754-2008 standard model for floating-point arithmetic, whenever $x$ and $y$ are floating-point numbers and $\circledast$ is one of the four operations $+$, $-$, $\times$, or $\div$, we have:

$$\mathrm{fl}(x \circledast y) = (x \circledast y)(1+\delta)^{\pm 1}, \quad \text{where} \quad |\delta| \le \epsilon_{\mathrm{mach}}, \tag{12}$$

where $\epsilon_{\mathrm{mach}}$ is a unit of least precision (ulp). As well, whenever $x$ is a floating-point number:

$$\mathrm{fl}(\sqrt{x}) = \sqrt{x}(1+\delta)^{\pm 1} \quad \text{where} \quad |\delta| \le \epsilon_{\mathrm{mach}}. \tag{13}$$

Since the sines and the cosines for the Givens rotations are derived analytically, rather than one numerically from the other through the relationship $s^2 + c^2 = 1$, the rounding errors committed on a computer with finite-precision arithmetic are on the order of $\epsilon_{\mathrm{mach}}$ multiplied by the relative sizes of the components *independently*. The precise implementation is now discussed.

**Algorithm 2.5.** *The computation of $s_n^m$ and $c_n^m$ is executed as follows:*

1. *Numerators and denominators are calculated in IEEE 64-bit signed integer arithmetic. Since $m \le n$, the largest integers encountered in $s_n^m$ or $c_n^m$ are:*

$$(3n+3)(3n+4) = 9n^2 + 21n + 12,$$

*in the denominators. The denominators, and consequently numerators, are exactly computed so long as $n \le 1,012,333,498$ (computed by solving for the largest integer satisfying $9n^2 + 21n + 12 \le 2^{63} - 1$);*

2. *Exactly computed numerators and denominators are converted to IEEE 64-bit floating-point numbers. This conversion is lossless so long as $n \leq 31,635,420$ (computed by solving for the largest integer satisfying $9n^2 + 21n + 12 \leq 2^{53}$, the largest representable integer by the floating-point number's significand);*

3. *For each sine and cosine, at most one rounding error bounded by one ulp is committed in dividing numerator by denominator; and,*

4. *For each sine and cosine, at most another rounding error bounded by one ulp is committed in computing the square root.*

Due to memory restrictions, a bandlimit of $n = 31,635,420$ is unlikely to be surpassed in our lifetime[1]. Furthermore, there is neither underflow nor overflow. This ensures that an algorithm designed on the application of these Givens rotations is indeed 2-normwise backward stable since component-wise bounds are tighter than bounds on the induced matrix 2-norm. The result of Algorithm 2.5 is that $\mathrm{fl}(G_n^{(m)}) = G_n^{(m)} + E$, where $\|E\|_2 = O(\epsilon_{\mathrm{mach}})$.

Algorithm 2.5 is also a backward stable implementation of left inversion (transposition) of matrices of connection coefficients by transposition of the product of Givens rotations. In fact, the Givens rotations correspond to a structured $QR$ factorization of $C^{(m)}$ and thus their use to solve a least-squares problem enjoys 2-normwise relative backward stability since the condition number of $C^{(m)}$ is 1. Following the text after Higham [35, Theorem 20.3], "The conclusion is that, unless $A$ is very ill-conditioned, the residual $b - A\hat{x}$, [computed by a 2-normwise relative backward stable $QR$ factorization of $A$], will not exceed the larger of the true residual $r = b - Ax$ and a constant multiple of the error in evaluating $\mathrm{fl}(r)$-a very satisfactory result."

## 2.2. The interpolative decomposition

The interpolative decomposition (ID) of a rectangular matrix $A \in \mathbb{R}^{m \times n}$ factorizes the matrix into one whose $k$ columns constitute a subset of unique columns of $A$, and another containing the $k$-by-$k$ identity matrix as a subset and whose remaining entries are not too large. The following lemma justifies the approximation power of an ID for low-rank matrices.

**Lemma 2.6.** *[24, 25, 14] Consider a rectangular matrix $A \in \mathbb{R}^{m \times n}$. Then, for any $k \in \mathbb{N}_0$ with $k \leq \min\{m, n\}$, there exist $A_{\mathrm{CS}} \in \mathbb{R}^{m \times k}$ whose columns constitute a subset of the columns of $A$ and $A_{\mathrm{I}} \in \mathbb{R}^{k \times n}$ such that:*

1. *some subset of the columns of $A_{\mathrm{I}}$ makes up the $k \times k$ identity matrix;*
2. *$\max_{1 \leq i \leq k, 1 \leq j \leq n} \left| (A_{\mathrm{I}})_{i,j} \right| \leq 2$;*
3. *the spectral norm of $A_{\mathrm{I}}$ satisfies $\|A_{\mathrm{I}}\|_2 \leq \sqrt{4k(n-k)+1}$;*
4. *the least singular value of $A_{\mathrm{I}}$ is at least $1$;*
5. *$A_{\mathrm{CS}}A_{\mathrm{I}} = A$ whenever $k = m$ or $k = n$; and,*
6. *when $k < \min\{m, n\}$, the spectral norm of $A - A_{\mathrm{CS}}A_{\mathrm{I}}$ satisfies:*

$$\|A - A_{\mathrm{CS}}A_{\mathrm{I}}\|_2 \leq \sqrt{4k(n-k)+1}\sigma_{k+1},$$

*where $\sigma_{k+1}$ is the $k+1^{\mathrm{st}}$ singular value of $A$.*

The subscript $_{\mathrm{CS}}$ stands for *column skeleton* and $_{\mathrm{I}}$ stands for *interpolation*.

This lemma states that $A \approx A_{\mathrm{CS}}A_{\mathrm{I}}$ provided that $\sigma_{k+1}$ is sufficiently small. For any $\varepsilon > 0$, numerous disparate algorithms [36, 24, 25] can identify the least $k$ such that $\|A - A_{\mathrm{CS}}A_{\mathrm{I}}\|_2 \approx \varepsilon$. The algorithms compute both $A_{\mathrm{CS}}$ and $A_{\mathrm{I}}$ using at most $O(kmn \log n)$ operations, typically requiring only $O(kmn)$ operations. Furthermore, there is abundant empirical evidence that the ID is a numerically stable algorithm for computing (transposed) matrix-vector products.

---

[1]In case it were surpassed, then the algorithm above could easily be modified to compute individual ratios in $s_n^m$ and $c_n^m$ in 64-bit floating-point rather than computing the full numerators and denominators. This modification would still be backward stable but involves a few more sources of rounding errors.

## 2.3. Threefold symmetry

For improved performance, the threefold symmetry in the spherical harmonics is leveraged: negative-ordered layers are treated similarly to positive-ordered layers; even-ordered layers are necessarily treated differently than odd-ordered layers; and, the parity in the bases allows one to perfectly shuffle matrices of connection coefficients to remove the necessity to include storage for the chessboard pattern of zeros. This also halves the semiseparability rank.

Throughout, complexity results are stated for the transformation of all layers of the spherical harmonic transform in terms of the bandlimit $n$. Furthermore, some technical results are proved for even-ordered layers and omitted for odd-ordered layers in order to not overburden the reader and the exposition.

## 3. Rapid algorithms for the conversion between bandlimited spherical harmonic expansions and their bivariate Fourier series

The steps required by the algorithm are illustrated in Figure 1. At first, we derive a new algorithm that converts higher-order layers of the spherical harmonics into expansions with orders zero and one. Then, these coefficients are rapidly transformed into their Fourier coefficients.
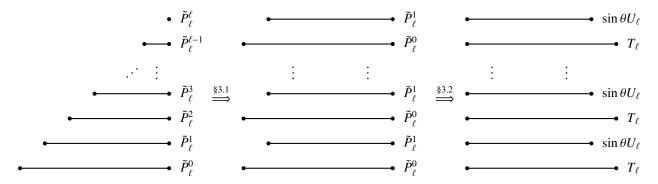


Figure 1: The spherical harmonic transform proceeds in two steps. Firstly, normalized associated Legendre functions are converted to normalized associated Legendre functions of order zero and one. Then, these intermediate expressions are re-expanded in trigonometric form. The first step proceeds with a butterfly factorization of the well-conditioned matrices of connection coefficients. The second step proceeds via the accelerated arithmetic of numerically approximating matrices of connection coefficients by hierarchically off-diagonal low-rank matrices.

## 3.1. The butterfly algorithm

Suppose that $n$ is a positive integer, and $A$ is an $n \times n$ matrix. Suppose further that $\varepsilon$ and $C$ are positive real numbers, and $k$ is a positive integer, such that any contiguous rectangular subblock of $A$ containing at most $Cn$ entries can be approximated to precision $\varepsilon$ by a matrix whose rank is $k$ (using the Frobenius/Hilbert–Schmidt norm to measure the accuracy of the approximation); we will refer to this hypothesis as "*the rank property.*"

The butterfly algorithm is an abstraction of the algebraic properties of divide-and-conquer fast Fourier transforms without their analytic properties. It is ideally suited for matrices resulting from the discretization of an integral operator with kernel $e^{\mathrm{i}xy}$. This bivariate function is known to generate matrices which satisfy the aforementioned *rank property*, whereby the numerical rank of any subblock is proportional to the area in the $xy$-plane from which entries are derived.

We refer the interested reader to [15, 16] for a detailed description of the butterfly algorithm. The following synopsis assumes the aforementioned *rank property*, and is almost identical to the synopsis given in [14]. Notwithstanding, the butterfly algorithm is more generally applicable to other compressible matrices where the rank structure is unknown [14], less clear, or different.

The running-time of the algorithm will be proportional to $k^2/C$; taking $C$ to be roughly proportional to $k$ suffices for many matrices of interest, so ideally $k$ should be small. We will say that two matrices $G$ and $H$ are equal to precision $\varepsilon$, denoted $G \approx H$, to mean that the spectral norm of $G - H$ is $O(\varepsilon)$.

We now explicitly use the rank property for subblocks of multiple heights, to illustrate the basic structure of the butterfly algorithm. Consider any two adjacent contiguous rectangular subblocks $L$ and $R$ of $A$, each containing at most $Cn$ entries and having the same number of rows, with $L$ on the left and $R$ on the right. Due to the rank property, there exists IDs:

$$L \approx L_{\text{CS}}L_{\text{I}}, \quad \text{and} \quad R \approx R_{\text{CS}}R_{\text{I}},$$

where the $k$ columns of $L_{\text{CS}}$ are a subset of the columns of $L$ and the $k$ columns of $R_{\text{CS}}$ are a subset of the columns of $R$, and where all entries in $L_{\text{I}}$ and $R_{\text{I}}$ have absolute value at most 2.

Next, we merge the matrices $L$ and $R$ and split the columns of the result approximately in half, obtaining $T$ on top and $B$ on the bottom:

$$\begin{pmatrix} L & R \end{pmatrix} = \begin{pmatrix} T \\ B \end{pmatrix}.$$

Observe that the matrices $T$ and $B$ each have at most $Cn$ entries (since $L$ and $R$ each have at most $Cn$ entries). Similarly, we merge the matrices $L_{\text{CS}}$ and $R_{\text{CS}}$ and split the columns of the result approximately in half, obtaining $T^{(1)}$ and $B^{(1)}$:

$$\begin{pmatrix} L_{\text{CS}} & R_{\text{CS}} \end{pmatrix} = \begin{pmatrix} T^{(1)} \\ B^{(1)} \end{pmatrix}.$$

As the columns of $L_{\text{CS}}$ and $R_{\text{CS}}$ are columns of the original matrix, so too are the $2k$ columns of $T^{(1)}$ and $B^{(1)}$. Hence, due to the rank property, there exist IDs:

$$T^{(1)} \approx T^{(1)}_{\text{CS}}T^{(1)}_{\text{I}}, \quad \text{and} \quad B^{(1)} \approx B^{(1)}_{\text{CS}}B^{(1)}_{\text{I}}.$$

Combining this multilevel approximation, the top matrix $T$ and the bottom matrix $B$ are approximately:

$$T \approx T^{(1)}_{\text{CS}}T^{(1)}_{\text{I}} \begin{pmatrix} L_{\text{I}} & 0 \\ 0 & R_{\text{I}} \end{pmatrix}, \quad \text{and} \quad B \approx B^{(1)}_{\text{CS}}B^{(1)}_{\text{I}} \begin{pmatrix} L_{\text{I}} & 0 \\ 0 & R_{\text{I}} \end{pmatrix}.$$

If we use $m$ to denote the number of rows in $L$ and $R$, then the number of columns in $L$ and $R$ is at most $Cn/m$, and so the total number of entries in the matrices in the right-hand sides of the first set of IDs can be as large as $2mk + 2k(Cn/m)$, whereas the total number of nonzero entries in the matrices in the right-hand sides of the bi-level representation is at most $mk + 4k^2 + 2k(Cn/m)$. If $m$ is nearly as large as possible – nearly $n$ – and $k$ and $C$ are much smaller than $n$, then $mk + 4k^2 + 2k(Cn/m)$ is about half $2mk + 2k(Cn/m)$. Thus, the bi-level representation is more efficient than that provided by the single-level representation, in terms of the complexity of both storage and matrix arithmetic. This gain in efficiency is due to the rank property holding for subblocks of multiple heights.

Naturally, we may repeat this process of merging adjacent blocks and splitting in half the columns of the result, updating the compressed representations after every split. We start by partitioning $A$ into blocks each dimensioned $n \times \lfloor C \rfloor$ (except possibly for the rightmost block, which may have fewer than $\lfloor C \rfloor$ columns), and then recursively group unprocessed blocks into disjoint pairs, processing these pairs by merging and splitting them into new, unprocessed blocks having fewer rows. The resulting multilevel representation of $A$ allows us to apply $A$ and $A^{\top}$ with precision $\varepsilon$ using just $O((k^2/C)n \log n)$ operations. This is due to the fact that there are $O(\log n)$ levels in the representation and each level except the last will only involve $O(n/C)$ interpolation matrices of dimensions $k \times (2k)$.

In practice, it is crucial to use a compact form for the interpolation matrices where the embedded identity is applied as a column permutation, and to ensure that the relative tolerance of $\varepsilon \max\{m, n\}$ is used to determine the numerical rank, where $m$ and $n$ denote the dimensions of the current subblock under compression.

### 3.1.1. Interpretation as Fourier integral operators

Due to the semiseparability of the matrices of connection coefficients, the analytic rank structure of contiguous subblocks above the $2m^{\text{th}}$ subdiagonal is at most $2m$; straddling this subdiagonal, the ranks of the subblocks are undetermined; and any subblock that is well below the subdiagonal has rank 0. Using only analytic rank structure, it appears that the butterfly algorithm may be ineffective when $m$ is large. However, the analytic structure overlooks the potential for decay in the singular values of the contiguous subblocks in the semiseparable region. To elucidate this numerical rank structure, we interpret the connection coefficients as Fourier integral operators.

A rank-1 (bivariate) function is separable: if $f_1(x, y) = g(x)h(y)$, then $\text{rank}(f_1) = 1$.

Similarly, a rank-$k$ (bivariate) function is a non-trivial sum of $k$ rank-1 functions.

**Definition 3.1.** *A function $f \in L^p([-1, 1]^2)$ is said to have rank $k_\varepsilon$ to precision $\varepsilon > 0$ if:*

$$k_\varepsilon = \inf_{k \in \mathbb{N}_0} \left\{ \inf_{\mathrm{rank}(f_k) \le k} \|f - f_k\|_p \le \varepsilon \|f\|_p \right\}.$$

**Definition 3.2.** *Let $A : L^2([-1, 1]) \to L^2([-1, 1])$ be the integral operator with kernel $f \in L^2([-1, 1]^2)$, given by:*

$$A\{u\}(x) = \int_{-1}^{1} f(x, y)u(y) \, \mathrm{d}y.$$

*The operator $A$ is said to have rank $k_\varepsilon$ to precision $\varepsilon > 0$ if $f$ has rank $k_\varepsilon$ to the same precision.*

We recall a lemma from [37] appearing in a similar form in [16].

**Lemma 3.3.** *Suppose that $\delta$, $\varepsilon$, and $\gamma$ are positive real numbers and $\varepsilon < 1$. Suppose further that the operator $F : L^2([-1, 1]) \to L^2([-1, 1])$ is given by the formula:*

$$F\{h\}(x) = \int_{-1}^{1} e^{\mathrm{i}\gamma xy} h(y) \, \mathrm{d}y.$$

*Then $F$ has rank to precision $\varepsilon$ at most:*

$$k(= k(\epsilon, \delta, \gamma)) = (1 + \delta)\left(\frac{2\gamma}{\pi} + \frac{E}{\delta}\right) + 3,$$

*where:*

$$E(= E(\epsilon, \delta)) = 2\sqrt{2 \ln\left(\frac{4}{\varepsilon}\right) \ln\left(\frac{6\sqrt{1/\sqrt{\delta} + \sqrt{\delta}}}{\varepsilon}\right)}.$$

This result means that if the Fourier transform with kernel $e^{\mathrm{i}kx}$ is restricted to a rectangle in the $kx$-plane, then its rank is bounded by a constant times the area of the rectangle.

For $m \in \mathbb{N}$, the connection coefficients between $\tilde{P}_{\ell+2m}^{2m}$ and $\tilde{P}_n^0$ are given by the inner product:

$$c_{\ell,n}^{2m} = \int_{-1}^{1} \tilde{P}_{\ell+2m}^{2m}(x)\tilde{P}_n^0(x) \, \mathrm{d}x. \tag{14}$$

Using the Fourier transform of normalized Legendre polynomials [38, §7.243 5.]:

$$\int_{-1}^{1} e^{-\mathrm{i}kx} \tilde{P}_n^0(x) \, \mathrm{d}x = 2(-\mathrm{i})^n \sqrt{n + \tfrac{1}{2}} \, j_n(k), \tag{15}$$

we may represent the connection coefficients as the composition of a spherical Bessel integral operator and a Fourier integral operator:

$$c_{\ell,n}^{2m} = \frac{(-\mathrm{i})^n \sqrt{n + \tfrac{1}{2}}}{\pi} \int_{\mathbb{R}} j_n(k) \, \mathrm{d}k \int_{-1}^{1} e^{\mathrm{i}kx} \tilde{P}_{\ell+2m}^{2m}(x) \, \mathrm{d}x, \tag{16}$$

where the order of integration has been reversed. Eq. (16) clearly shows how a subblock of connection coefficients with indices among $\{\ell\} \times \{n\}$ may be represented as the composition of linear operators involving the indices $n$ and variable $k$, variables $k$ and $x$, and variable $x$ and indices $\ell$. As the rank of the composed operator is bounded by the smallest rank in the composition, the relation to a Fourier integral operator is of paramount importance.

In this integral operator setting, we would need to identify bounds on the relevant wavenumbers to invoke Lemma 3.3. Before we proceed, we prove a technical lemma.

**Lemma 3.4.** *For integers $\ell$ and $m$, the summation:*

$$\sum_{j=0}^{[\ell/2]} (2m + 1 + 2\ell - 4j) \frac{\Gamma(\ell + 2m - j + \frac{1}{2})}{\Gamma(\ell + m - j + \frac{3}{2})} \frac{(m)_j}{j!} \frac{\Gamma(\ell + 2m - 2j + 1)}{\Gamma(\ell - 2j + 1)} = \frac{2\sqrt{\pi}\Gamma(\ell + 4m + 1)}{16^m \Gamma(m + \frac{1}{2})\Gamma(\ell + 1)}.$$

*Proof.* Representing $C_\ell^{(2m+\frac{1}{2})}(x)$ in the basis of $C_j^{(m+\frac{1}{2})}(x)$ via [32, §18.18.16]:

$$C_\ell^{(2m+\frac{1}{2})}(x) = \sum_{j=0}^{[\ell/2]} \frac{2m + 1 + 2\ell - 4j}{2m + 1} \frac{(2m + \frac{1}{2})_{\ell-j}}{(m + \frac{3}{2})_{\ell-j}} \frac{(m)_j}{j!} C_{\ell-2j}^{(m+\frac{1}{2})}(x), \tag{17}$$

the summation is obtained by using:

$$C_\ell^{(2m+\frac{1}{2})}(1) = \frac{(4m + 1)_\ell}{\ell!}, \quad \text{and} \quad \frac{\Gamma(2m + \frac{1}{2})\Gamma(2m + 2)}{\Gamma(m + \frac{3}{2})\Gamma(4m + 1)} = \frac{2\sqrt{\pi}}{16^m \Gamma(m + \frac{1}{2})}.$$

$\square$

**Theorem 3.5.** *Let:*

$$k_1(n, \varepsilon) := 2\left(\frac{\varepsilon}{2}\sqrt{\frac{\pi}{2n + 1}}(n + 1)\Gamma(n + \tfrac{3}{2})\right)^{\frac{1}{n+1}},$$

*and let:*

$$k_2(\ell, m, n, \varepsilon) := \frac{1}{8}\left(\frac{2}{\varepsilon}\sqrt{\frac{2n + 1}{\pi}}\sqrt{\frac{(2\ell + 4m + 1)\Gamma(\ell + 4m + 1)}{\Gamma(\ell + 1)}}\frac{1}{m\Gamma(m + \frac{1}{2})}\right)^{\frac{1}{m}}.$$

*Then only integration over $k_1(n, \varepsilon) \le |k| \le k_2(\ell, m, n, \varepsilon)$ contributes to Eq. (16) to precision $\varepsilon > 0$.*

*Proof.* The absolute value of the connection coefficients is bounded by:

$$\left|c_{\ell,n}^{2m}\right| \le \left|\frac{\sqrt{n + \frac{1}{2}}}{\pi}\int_{\mathbb{R}} j_n(k)\,\mathrm{d}k\left|\int_{-1}^1 e^{\mathrm{i}kx}\tilde{P}_{\ell+2m}^{2m}(x)\,\mathrm{d}x\right|\right|.$$

Using the Cauchy–Schwarz inequality, $|\langle e^{-\mathrm{i}kx}, \tilde{P}_{\ell+2m}^{2m}\rangle| \le \sqrt{2}$, and thus:

$$\left|c_{\ell,n}^{2m}\right| \le \left|\frac{\sqrt{2n + 1}}{\pi}\int_{\mathbb{R}} j_n(k)\,\mathrm{d}k\right|.$$

By the inequality:

$$|j_n(k)| \le \frac{\sqrt{\pi}}{2\Gamma(n + \frac{3}{2})}\left|\frac{k}{2}\right|^n, \quad \forall n \in \mathbb{N}_0, \ k \in \mathbb{R},$$

we may readily ascertain the equality:

$$\left|\frac{\sqrt{2n + 1}}{\pi}\int_{-k_1}^{k_1} j_n(k)\,\mathrm{d}k\right| = \varepsilon.$$

Furthermore, an upper bound on $|k|$ may be derived from the fact that the regularity of $\tilde{P}_{\ell+2m}^{2m}(x)\chi_{[-1,1]}(x)$ on the real line dictates the decay rate of its Fourier transform, where $\chi_{[-1,1]}(x)$ is the characteristic function on the interval $[-1, 1]$. As $m$ increases, the polynomial decay to 0 at $\pm 1$ from the interior of the interval $[-1, 1]$ matches 0 from the exterior of the interval to $m^{\text{th}}$ degree, increasing the regularity proportionally.

In particular, using the Fourier transform [38, §7.321]:

$$\int_{-1}^1 e^{\mathrm{i}kx}C_\ell^{(m+\frac{1}{2})}(x)(1 - x^2)^m\,\mathrm{d}x = \frac{\mathrm{i}^\ell\sqrt{\pi}\Gamma(\ell + 2m + 1)j_{\ell+m}(k)}{2^{m-1}\Gamma(\ell + 1)\Gamma(m + \frac{1}{2})k^m},$$

combined with the representation of $C_\ell^{(2m+\frac{1}{2})}(x)$ in the basis of $C_j^{(m+\frac{1}{2})}(x)$ via Eq. (17), we obtain the Fourier transform:

$$\int_{-1}^{1} e^{ikx} P_{\ell+2m}^{2m}(x)\,dx = \frac{2^m}{k^m} \sum_{j=0}^{[\ell/2]} (2m+1+2\ell-4j) \frac{\Gamma(\ell+2m-j+\frac{1}{2})}{\Gamma(\ell+m-j+\frac{3}{2})} \frac{(m)_j}{j!} \frac{i^{\ell-2j}\Gamma(\ell+2m-2j+1)j_{\ell+m-2j}(k)}{\Gamma(\ell-2j+1)}. \quad (18)$$

Combining Lemma 3.4 with the inequality:

$$|j_n(k)| \le \sqrt{\frac{1}{|k|}}, \quad \forall n \in \mathbb{N}_0,\ k \in \mathbb{R},$$

we bound the Fourier transform in absolute value by:

$$\left| \int_{-1}^{1} e^{ikx} \tilde{P}_{\ell+2m}^{2m}(x)\,dx \right| \le \sqrt{\frac{(\ell+2m+\frac{1}{2})\Gamma(\ell+4m+1)}{\Gamma(\ell+1)}} \frac{2\sqrt{\pi}}{8^m\,|k|^{m+\frac{1}{2}}\,\Gamma(m+\frac{1}{2})}.$$

Then:

$$\left\| \frac{\sqrt{n+\frac{1}{2}}}{\pi} \int_{\mathbb{R}\setminus\{|k|<k_2\}} j_n(k)\,dk \left| \int_{-1}^{1} e^{ikx} \tilde{P}_{\ell+2m}^{2m}(x)\,dx \right| \right\|,$$

$$\le \left| \frac{\sqrt{n+\frac{1}{2}}}{\pi} \int_{\mathbb{R}\setminus\{|k|<k_2\}} j_n(k)\,dk \sqrt{\frac{(\ell+2m+\frac{1}{2})\Gamma(\ell+4m+1)}{\Gamma(\ell+1)}} \frac{2\sqrt{\pi}}{8^m\,|k|^{m+\frac{1}{2}}\,\Gamma(m+\frac{1}{2})} \right|,$$

$$\le \sqrt{\frac{(\ell+2m+\frac{1}{2})\Gamma(\ell+4m+1)}{\Gamma(\ell+1)}} \frac{2\sqrt{\pi}}{8^m\Gamma(m+\frac{1}{2})} \left| \frac{\sqrt{n+\frac{1}{2}}}{\pi} \int_{\mathbb{R}\setminus\{|k|<k_2\}} \frac{|j_n(k)|}{|k|^{m+\frac{1}{2}}}\,dk \right|,$$

$$\le \sqrt{\frac{(2\ell+4m+1)\Gamma(\ell+4m+1)}{\Gamma(\ell+1)}} \frac{1}{8^m\Gamma(m+\frac{1}{2})} \sqrt{\frac{2n+1}{\pi}} \left| \int_{\mathbb{R}\setminus\{|k|<k_2\}} \frac{dk}{|k|^{m+1}} \right| = \varepsilon.$$

$\square$

**Remark 3.6.** 1. Theorem 3.5 has a fundamentally different interpretation than [16, Theorem 10], which proves the requisite rank property for Fourier–Bessel transforms with kernel $xJ_\mu(xt)$. With $\mu$ fixed, this theorem is much more closely linked to the Fourier transform due to the oscillatory asymptotics of the Bessel function. In Eq. (16), the two variables related to indices in the matrices of connection coefficients are $n$ and $\ell$, which do not appear as a product.

2. Estimates for the bounds are $k_1(n,\varepsilon) = O(n)$, and for $m = O(\ell)$ we find that $k_2(\ell,m,n,\varepsilon) = O(\ell)$. The limit $m = O(1)$ is known to be compressible due to semiseparability and therefore it is not of interest.

3. It is reasonable to suggest that the estimate for $k_2(\ell,m,n,\varepsilon)$ may be improved upon by using uniform asymptotics of Jacobi polynomials [39, 40] or by refining estimates on the summation in Eq. (18) by incorporating the alternation in sign $i^{\ell-2j}$ and spherical Bessel asymptotics. On the other hand, with the current estimates in place, we may conclude by Theorem 3.5 that the connection coefficients satisfy *a rank property* for successful compression in the butterfly algorithm and we leave the compression to the interpolative decomposition.

4. In fact, $|k| \le k_2(\ell,m,n,\varepsilon)$ is sufficient to prove this rank property but the inclusion of $k_1(n,\varepsilon) \le |k|$ shows that there are certain regions in the $\ell,m,n$-cube where more compression is obtained than others.

5. The rank property revealed by Theorem 3.5 is universal in the sense that it is the same low-rank property that appears in the matrices of associated Legendre functions sampled at Gauss–Jacobi quadrature nodes. These matrices are part of Tygert's synthesis and analysis-based approach [14] that is also accelerated by the butterfly algorithm.

The partitioning involved in the butterfly algorithm with six levels is illustrated in the left panel of Figure 2.
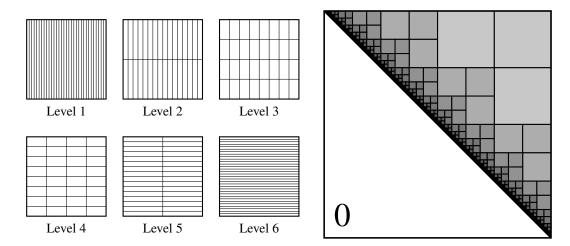
Figure 2: The spherical harmonic transform proceeds in two steps. Left: an illustration of the butterfly algorithm used to accelerate the conversion of expansions in $\tilde{P}_\ell^{2m}$ to expansions in $\tilde{P}_\ell^0$ and similarly for $\tilde{P}_\ell^{2m+1}$ to $\tilde{P}_\ell^1$. Right: an illustration of the hierarchical decomposition of the matrices of connection coefficients between $\tilde{P}_\ell^0$ and $T_\ell$ and $\tilde{P}_{\ell+1}^1$ and $\sin\theta U_\ell$. The opacity illustrates the data-sparsity in the hierarchical decomposition.

### 3.2. An adaptation of the Fast Multipole Method

The right panel of Figure 2 describes the familiar hierarchical decomposition of an upper-triangular matrix that characterizes the Fast Multipole Method [41]. In this schema, all subblocks in the partition are *well-separated* from the main diagonal, relative to their own size. With off-diagonal compression comes accelerated matrix-vector products, among other algebraic properties. Physically, the name is derived from the multipole expansion of an inverse power of the euclidean distance in $\mathbb{R}^n$:

$$\frac{1}{|\mathbf{r} - \mathbf{r}_0|^{2\lambda}} = \frac{1}{\left(r^2 - 2rr_0\cos\theta + r_0^2\right)^\lambda} = \frac{1}{r^{2\lambda}} \sum_{n=0}^{\infty} \left(\frac{r_0}{r}\right)^n C_n^{(\lambda)}(\cos\theta).$$

In physical space, the assumption of subblocks being well-separated from the main diagonal is equivalent to expansion for sufficiently small $r_0/r$, in which it is readily observed to require $O(\log(\varepsilon^{-1}))$ terms in the multipole expansion for approximation to a prescribed accuracy. However, to be precise in this and other settings requires mathematical rigour.

If we refine our hierarchical decomposition of an upper-triangular matrix $A \in \mathbb{R}^{n \times n}$ through $O(\log n)$ levels as indices get closer to the diagonal, all subblocks in the partition may be well-approximated by low-rank matrices. In practice, we stop after the dimensions of the subblocks are comparable to the numerical rank required to guarantee an accuracy on the order of machine precision. This implies that the subblocks are no longer data-sparse and nothing is gained from further partitioning.

In the cases of present interest, another asymptotically smooth function succinctly defines the connection coefficients between $\tilde{P}_\ell^0$ and $T_\ell$ and between $\tilde{P}_\ell^1$ and $\sin\theta U_\ell$.

**Definition 3.7.** *The meromorphic function $\Lambda : \mathbb{C} \to \mathbb{C}$ is defined by:*

$$\Lambda(z) := \frac{\Gamma(z + \frac{1}{2})}{\Gamma(z + 1)}. \tag{19}$$

The full asymptotic expansion of $\Lambda(z)$ is derived in [42], but only a few terms are required in double precision.

**Algorithm 3.8** (Adapted from Appendix B in [43]). *If $z > 9.844{,}75$, then*

$$\Lambda(z) \approx \left(1 - \frac{1}{64(z + \frac{1}{4})^2} + \frac{21}{8{,}192(z + \frac{1}{4})^4} - \frac{671}{524{,}288(z + \frac{1}{4})^6} + \frac{180{,}323}{134{,}217{,}728(z + \frac{1}{4})^8}\right.$$
$$\left. - \frac{20{,}898{,}423}{8{,}589{,}934{,}592(z + \frac{1}{4})^{10}} + \frac{7{,}426{,}362{,}705}{1{,}099{,}511{,}627{,}776(z + \frac{1}{4})^{12}}\right) \Bigg/ \sqrt{z + \frac{1}{4}}. \tag{20}$$

*gives an approximation to* $\Lambda(z)$ *that is accurate to the machine precision,* $\epsilon_{\text{mach}} \approx 2.22 \times 10^{-16}$. *In case* $z \leq 9.844,75$, *then we use the recursion:*

$$\frac{\Lambda(z+1)}{\Lambda(z)} = \frac{z + \frac{1}{2}}{z + 1}, \tag{21}$$

*until the first condition is satisfied.*

Conversion from normalized Legendre polynomials to cosines is given by:

$$\tilde{P}_n^0(\cos\theta) = \sqrt{n + \tfrac{1}{2}} \sum_{\ell=n,-2}^{0} \Lambda(\tfrac{n-\ell}{2})\Lambda(\tfrac{n+\ell}{2})\frac{2 - \delta_{\ell,0}}{\pi} \cos\ell\theta. \tag{22}$$

The inverse relationship is given by:

$$\cos n\theta = -n \sum_{\ell=n,-2}^{0} \frac{\Lambda(\tfrac{n-\ell-2}{2})\Lambda(\tfrac{n+\ell-1}{2})}{(n - \ell)(n + \ell + 1)} \sqrt{\ell + \tfrac{1}{2}}\,\tilde{P}_\ell^0(\cos\theta). \tag{23}$$

Special care must be taken to obtain the limiting values when $n = 0$ and when $\ell = n$.

Similarly, conversion from normalized Legendre polynomials of order 1 to sines is given by:

$$\tilde{P}_{n+1}^1(\cos\theta) = \sqrt{\frac{n + \frac{3}{2}}{(n+1)(n+2)}} \sum_{\ell=n,-2}^{0} \Lambda(\tfrac{n-\ell}{2})\Lambda(\tfrac{n+\ell+2}{2})\frac{2(\ell+1)}{\pi} \sin(\ell+1)\theta, \tag{24}$$

and the inverse relationship is given by:

$$\sin(n+1)\theta = -\sum_{\ell=n,-2}^{0} \frac{(\ell + \tfrac{3}{2})\Lambda(\tfrac{n-\ell}{2})\Lambda(\tfrac{n+\ell+3}{2})}{(n - \ell - 1)(n + \ell + 2)} \sqrt{\frac{(\ell+1)(\ell+2)}{\ell + \frac{3}{2}}}\,\tilde{P}_{\ell+1}^1(\cos\theta). \tag{25}$$

The following definition results in the partitioning in the right panel of Figure 2.

**Definition 3.9** (Keiner [44]). *A square* $S \subset \mathbb{R}^2$ *defined by the formula* $S = [x_0, x_0 + c] \times [y_0, y_0 + c]$ *with* $c > 0$ *is said to be* well-separated *if* $y_0 - x_0 \geq 2c$.

Alpert and Rokhlin [45] prove rigorous bounds on the univariate Chebyshev interpolants to the $\Lambda$ function on a well-separated square $S$. The bounds are obtained for approximation in either variable while the free variable ranges over all possible values it may take in $S$. In the general ultraspherical setting, Keiner [44] shows that the geometric rate of $3^{-k}$ in the original bounds is not optimal. In fact, through analyticity in the open Bernstein ellipse $E_\rho$ with $\rho = 3 + \sqrt{8}$, Keiner shows that the optimal geometric decay is $\rho^{-k}$.

Considering Eq. (22), the Chebyshev–Legendre transform requires approximation of the function [45]:

$$\mathcal{M}(x,y) := \frac{2}{\pi}\Lambda\left(\frac{y-x}{2}\right)\Lambda\left(\frac{y+x}{2}\right),$$

on well-separated squares. In the $x$-variable, this results in deriving bounds on the error of the Chebyshev interpolants to:

$$\mathcal{M}(x(t),y) = \frac{2}{\pi}\Lambda\left(\frac{y - x_0 - \frac{c(t+1)}{2}}{2}\right)\Lambda\left(\frac{y + x_0 + \frac{c(t+1)}{2}}{2}\right), \quad \text{for} \quad t \in [-1,1].$$

**Theorem 3.10** (Keiner [44]). *Let* $\mathcal{M}_k$ *denote the degree-k Chebyshev interpolant to* $\mathcal{M}(x(t),y)$. *Then for* $c > 1$:

$$\sup_{t\in[-1,1]} |\mathcal{M}(x(t),y) - \mathcal{M}_k(x(t),y)| \leq \frac{4M\rho^{-k}}{\rho - 1}, \tag{26}$$

*where:*

$$M = \frac{2\sqrt{2}e^{\frac{5}{3}}}{\pi}, \quad \text{and} \quad \rho = 3 + \sqrt{8}. \tag{27}$$

Thus, a Chebyshev polynomial interpolant of degree $\left\lceil \log_{3+\sqrt{8}} \left( \frac{4\sqrt{2}e^{\frac{5}{3}}}{\pi(1+\sqrt{2})\varepsilon} \right) \right\rceil$ approximates $\mathcal{M}(x(t), y)$ to precision $\varepsilon > 0$ on any well-separated square.

In Alpert and Rokhlin's [45] Chebyshev–Legendre transform, the Lagrange interpolating formula is originally used to approximate subblocks well-separated from the main diagonal. Since then, Higham [46] has proved the numerical stability of the second barycentric formula, popularized by Berrut and Trefethen [47]. This formulation also leads to a lower pre-computation since the Chebyshev barycentric weights $\lambda_\ell$ are known analytically.

Let $x_\ell$ and $\lambda_\ell$ be the pair of Chebyshev points and barycentric weights of the first kind:

$$x_\ell = \cos\left( \frac{2\ell+1}{2k+2}\pi \right), \quad \text{and} \quad \lambda_\ell = (-1)^\ell \sin\left( \frac{2\ell+1}{2k+2}\pi \right), \quad \text{for} \quad \ell = 0, \dots, k.$$

Let $a, b, c, d \in \mathbb{N}_0$ and let $m = b - a + 1$ and $n = d - c + 1$. For each $x \in [a, b]$ and $y \in [c, d]$, consider the degree-$k$ approximation by a polynomial interpolant in the first variable at the mapped Chebyshev points in barycentric form:

$$f(x, y) \approx p_k(x, y) = \frac{\displaystyle\sum_{\ell=0}^{k} \frac{\lambda_\ell f\left( \frac{a+b}{2} + \frac{(b-a)x_\ell}{2}, y \right)}{2x - a - b - (b-a)x_\ell}}{\displaystyle\sum_{\ell=0}^{k} \frac{\lambda_\ell}{2x - a - b - (b-a)x_\ell}}.$$

Consider the matrix that results from sampling the bivariate function $f$ at the integers within the rectangle $[a, b] \times [c, d]$:

$$[F_{a,b}^{c,d}]_{i,j} := f(i, j) \quad \text{for} \quad a \le i \le b, \ c \le j \le d.$$

The matrix-vector product $Fv$, where $v \in \mathbb{R}^n$ may be replaced by an approximation that requires only $O((k+1)(m+n))$ operations when using $p_k(x, y)$. This is because the barycentric formula allows for the storage of $(k+1)n$ function samples:

$$[\tilde{F}_{a,b}^{c,d}]_{\ell,j} := f\left( \frac{a+b}{2} + \frac{(b-a)x_\ell}{2}, j \right) \in \mathbb{R}^{k \times n}, \quad \text{for} \quad 0 \le \ell \le k, \ c \le j \le d,$$

and $m(k+1)$ weights:

$$[W_{a,b}]_{i,j} := \frac{\lambda_j}{2i - a - b - (b-a)x_j} \bigg/ \sum_{\ell=0}^{k} \frac{\lambda_\ell}{2i - a - b - (b-a)x_\ell} \in \mathbb{R}^{m \times k}, \quad \text{for} \quad a \le i \le b, \ 0 \le j \le k.$$

With the function samples $\tilde{F}_{a,b}^{c,d}$ and the barycentric weights $W_{a,b}$:

$$\boxed{F_{a,b}^{c,d}} \approx \boxed{W_{a,b}} \boxed{\tilde{F}_{a,b}^{c,d}}$$

is the result of interpolating each column of $F_{a,b}^{c,d}$ with a degree-$k$ polynomial.

**Remark 3.11.** Other methods exist for fast polynomial transforms, such as the Chebyshev–Legendre [48] and Chebyshev–Jacobi transforms [49], and the Toeplitz-dot-Hankel approach [50], all of which are available in `FastTransforms.jl` [26]. After extensive numerical experiments, the hierarchical approach is recommended for this particular problem because the execution is an order of magnitude faster than the pre-computation, rather than the other way around.

### 3.3. Practical methods to reduce the pre-computation

When a matrix has orthonormal columns, full columns are incompressible in the first step of the butterfly algorithm since the matrix has full rank. Instead of computing IDs of the full columns, the first level of the butterfly algorithm is essentially skipped.

With the ability to convert between neighbouring expansions whose orders differ by two in $O(n)$ operations using Givens rotations, the present pre-computation may be "thinned" at a modest cost to the execution. Let $k_{avg}$ denote the average numerical rank in the butterfly factorization of any given layer. By §3.1, the conversion of one layer of a spherical harmonic expansion to the zeroth or first layer by the butterfly algorithm costs $O(k_{avg}n \log n)$; the average numerical rank is a good indicator because subblocks of lower rank have cheaper arithmetic, and for nearly full-rank subblocks the interpolation matrix is nearly a permutation. Similarly, conversion between all spherical harmonic expansions of orders $m + 2\mu$, for $\mu = 1, \ldots, k_{avg}$ to those of order $m$ costs $O(k_{avg}^2 n)$ by Givens rotations. Thus, if the computational complexities are balanced[2], then a "thin" plan consists of creating the butterfly factorizations only for orders in stride lengths of $O(k_{avg})$. The thin plan, depicted in Figure 3, has a construction cost of $O(1/\log n)$ times the cost of the full pre-computation. In the practical bandlimit of $n < O(10,000)$ due to storage limitations on a standard computer, the thin plan's pre-computation is effectively only one order of magnitude larger than execution.
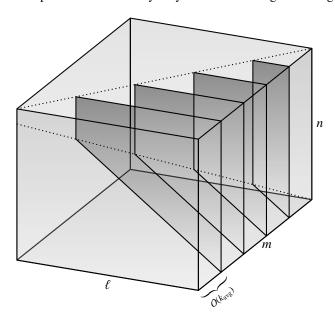


Figure 3: The spherical harmonic transform pre-computation is accelerated by only constructing butterfly factorizations of orders $O(k_{avg})$ apart.

## 4. Numerical discussion

The software package `FastTransforms.jl` [26] written in the JULIA programming language [51, 52] implements the fast and backward stable transforms between spherical harmonic expansions and their bivariate Fourier series. Interpolative decompositions are computed by the `LowRankApprox.jl` package [53]. The implementations are templated in IEEE single and double precision[3], with most matrix-vector multiplications performed by OpenBLAS [54] at the lowest level. All numerical simulations are performed on a MacBook Pro with a 2.8 GHz Intel Core i7-4980HQ processor with $4 \times 256$ KB of L2 cache, 6 MB of L3 cache, and 16 GB of 1600 MHz DDR3 RAM.

Spherical harmonic expansion coefficients $f_\ell^m$ naturally populate a doubly triangular matrix. However, for com-

---

[2]And if $k_{avg} = O(\log n)$ as supported by the right panel of Figure 6.

[3]and consequently extended precision, but this hasn't been tested.

putational purposes, we organize them into the array:

$$F = \begin{pmatrix} f_0^0 & f_1^{-1} & f_1^1 & f_2^{-2} & f_2^2 & \cdots & f_n^{-n} & f_n^n \\ f_1^0 & f_2^{-1} & f_2^1 & f_3^{-2} & f_3^2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ f_{n-2}^0 & f_{n-1}^{-1} & f_{n-1}^1 & f_n^{-2} & f_n^2 & \cdot^{\cdot^{\cdot}} & \vdots & \vdots \\ f_{n-1}^0 & f_n^{-1} & f_n^1 & 0 & 0 & \cdots & 0 & 0 \\ f_n^0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \end{pmatrix} \in \mathbb{R}^{(n+1)\times(2n+1)}.$$

This structure has the advantage of organizing the decay in the coefficients of sufficiently regular functions downward and to the right. The columns of $F$ may be interpreted as having longitudinal basis:

$$\frac{1}{\sqrt{2\pi}}\{1, e^{-i\varphi}, e^{i\varphi}, e^{-i2\varphi}, e^{i2\varphi}, \ldots\}, \quad \text{or} \quad \frac{1}{\sqrt{\pi}}\{\frac{1}{\sqrt{2}}, \sin\varphi, \cos\varphi, \sin 2\varphi, \cos 2\varphi, \ldots\},$$

for a real-to-real transform. In the experiments, test spherical harmonic expansion coefficients are first drawn from a standard normal distribution and subsequently, the columns of $F$ are normalized in $\ell^2$. Numerical results report the maximum $\ell^2$ norm over all columns of the error in transforming the spherical harmonic expansion coefficients to Fourier coefficients and back, averaged over three independent trials to reduce the variance. Because the columns of $F$ are normalized, the errors are relative and absolute.

Figure 4 reports the numerical results for the $O(n^3)$ application of the Givens rotations followed by the rapid conversion from $\tilde{P}_\ell^0$ to $T_\ell$ and $\tilde{P}_\ell^1$ to $\sin\theta U_\ell$. For the sake of comparison, timings are also reported for a 2D discrete cosine transform (DCT) of the same array.
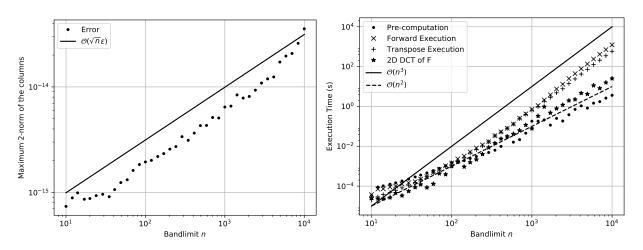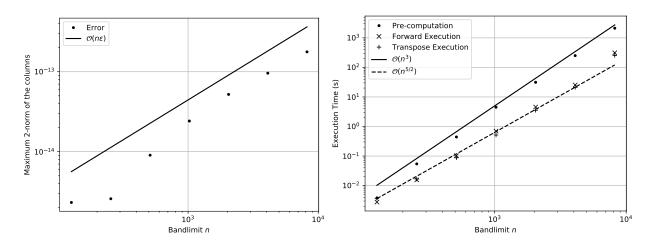


Figure 4: Left: The maximum 2-norm in the columns of a bandlimited spherical harmonic expansion transformed to its bivariate Fourier series and back. Right: the execution times.

Due to backward stability, the Givens rotations exhibit error growth proportional to $O(\sqrt{n}\varepsilon)$. The butterfly factorizations compound this by a factor of $O(\sqrt{n})$, as depicted in Figure 5 for the thin pre-computation. The thinning parameter of 64 is utilized in the experiments and not much has been done to further optimize the thinning. Numerical results in the extant literature show error in the numerical evaluation of spherical harmonic expansions at points on the sphere in the $\ell^\infty$ norm. While the $\ell^2$ norm is more natural for the expansion coefficients in the Hilbert space $L^2(\mathbb{S}^2, d\Omega)$, a very rough conversion would see the present results scaled by $O(\log n/\sqrt{n})$.

Figure 6 shows total memory required to store the thin spherical harmonic pre-computation and the numerical rank statistics. In the right panel of Figure 5, the forward and transpose execution times at a bandlimit of $n = 8,191$ are higher than might be extrapolated by the preceding trend. This is explained by the thin pre-computation requiring 26

Figure 5: Left: The maximum 2-norm in the columns of a bandlimited spherical harmonic expansion transformed to its bivariate Fourier series and back. Right: the execution times.

GB of memory. Since it requires more memory than may be held in the RAM, the execution times are conflated with memory transfer from the flash hard drive. Additionally, it appears that the execution times follow the $O(n^{5/2})$ trend rather than $O(n^2 \log^2 n)$. This is reasonable because the asymptotically optimal complexity of the butterfly algorithm requires a larger bandlimit before becoming fully apparent. This is consistent with Tygert's results [14, Tables 1, 3, & 5] that begin to show the optimal complexity *for a single layer* only for $n \geq O(20,000)$.
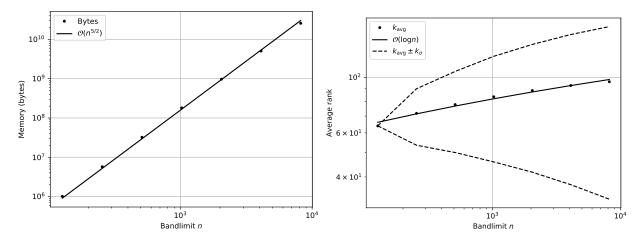


Figure 6: Left: the total number of bytes allocated in the thin spherical harmonic pre-computation. Right: the average numerical rank in the butterfly factorizations bracketed by one standard deviation from the mean.

Tygert has shown [14] that the butterfly algorithm may be performed with $O(n \log n)$ storage; this is an important consideration for a parallel implementation. For a full spherical harmonic transform, ultimately requiring $O(n^2 \log n)$ storage, we can afford to perform the butterfly algorithms on a single pair of matrices of connection coefficients that are updated from layer to layer, one each for even and odd layers.

For parallelization over a modest number of processes, the backward stable Givens rotations may still be used: on the master processor, one updates the matrices of connection coefficients and distributes them to the worker processes for compression. For parallelization over numerous processes, application of the Givens rotations would exacerbate

asynchrony. Backward stability could be sacrificed for the use of the five-term recurrence relations that are satisfied by the connection coefficients, uncoupling successive layers. However, a parallel implementation is beyond the scope of this report.

## 5. Conclusion & outlook

The transforms described in the present contribution are rigorously proved to be asymptotically fast and are fast in practice as well. Based on the theory of orthogonal transformations, certain subproblems are well-conditioned and the implementations are backward stable *ipso facto* the Givens rotations are known analytically. Thus, this is the first pair of fast transforms to claim backward stability without the use of extended precision in some part of the algorithm or another [13, 14, 23], though some algorithms have empirical evidence substantiating stability. In terms of software, the main contribution is to lower the egregious pre-computation found elsewhere to something more reasonable.

We describe two potential methods to overcome an $O(n^3 \log n)$ pre-computation. Firstly, Chebyshev polynomial interpolation has been used to accelerate pre-computation in the butterfly algorithm for Fourier integral operators [55] and may be further recompressed by another butterfly factorization [56]. In the present context, this requires sampling inner products $\langle \tilde{P}_\nu^0, \tilde{P}_{\lambda+2m}^{2m} \rangle$ and $\langle \tilde{P}_\nu^1, \tilde{P}_{\lambda+2m+1}^{2m+1} \rangle$, where $\nu$ and $\lambda$ are real numbers rather than integers. To this end, the results of Theorem 2.4 could be extended to larger integer orders and then potentially analytically continued to non-integer degrees. It would take some creativity to obtain $O(1)$ computation of entries for real-valued $\lambda$ and $\nu$. Secondly, the Givens rotations may potentially be accelerated on their own. It is surprising in this setting that the analytic structure[4] of the Givens rotations is known completely. Multiple layers are represented by the sequence of operations:



Since conversions are rectangular, it is possible to trim the number of Givens rotations via turnovers [57, Theorem 2.1]. Turnovers alone do not produce an optimal complexity; however, the overturned structure presents the potential for acceleration. Another possibility is to accumulate Givens rotations into Householder reflectors and relate these reflections in an approximate sense to the Householder decomposition of the DCT.

This work also extends naturally to the conversion of Zernike polynomials [58], orthogonal on the unit disk, to Fourier–Chebyshev series and other families of bivariate analogues of the Jacobi polynomials [59], orthogonal on a triangle or a deltoid, among other shapes. These families of bivariate orthogonal polynomials may be organized into layers that are orthogonal functions with respect to the same Hilbert space. This structure ensures that the connection problem is well-conditioned, and allows us to formulate backward stable algorithms.

## Acknowledgments

[1] A. Dutt, V. Rokhlin, Fast Fourier transforms for nonequispaced data, SIAM J. Sci. Comput. 14 (1993) 1368–1393.

[2] D. Potts, G. Steidl, Fast summation at nonequispaced knots by NFFTs, SIAM J. Sci. Comput. 24 (2003) 2013–2037.

[3] L. Greengard, J.-Y. Lee, Accelerating the nonuniform fast Fourier transform, SIAM Rev. 46 (2004) 443–454.

[4] D. Ruiz-Antolín, A. Townsend, A nonuniform fast Fourier transform based on low rank approximation, arXiv:1701.04492 (2017).

[5] A. Townsend, H. Wilber, G. B. Wright, Computing with functions in spherical and polar geometries I. The sphere, SIAM J. Sci. Comput. 38 (2016) C403–C425.

---

[4]By analytic structure, we mean that the sines and cosines are given analytically by Eq. (11).

[6] D. M. H. Jr., D. N. Rockmore, P. J. Kostelec, S. Moore, FFTs for the 2-sphere–improvements and variations, J. Fourier Anal. Appl. 9 (2003) 341–385.

[7] J. R. Driscoll, D. M. H. Jr., Computing Fourier transforms and convolutions on the 2-sphere, Adv. Appl. Math. 15 (1994) 202–250.

[8] R. Suda, M. Takami, A fast spherical harmonics transform algorithm, Math. Comp. 71 (2002) 703–715.

[9] S. Kunis, D. Potts, Fast spherical Fourier algorithms, J. Comp. Appl. Math. 161 (2003) 75–98.

[10] M. J. Mohlenkamp, A fast transform for spherical harmonics, J. Fourier Anal. Appl. 5 (1999) 159–184.

[11] V. Rokhlin, M. Tygert, Fast algorithms for spherical harmonic expansions, SIAM J. Sci. Comput. 27 (2006) 1903–1928.

[12] S. Chandrasekaran, M. Gu, A divide-and-conquer algorithm for the eigendecomposition of symmetric block-diagonal plus semiseparable matrices, Numer. Math. 96 (2004) 723–731.

[13] M. Tygert, Fast algorithms for spherical harmonic expansions, II, J. Comp. Phys. 227 (2008) 4260–4279.

[14] M. Tygert, Fast algorithms for spherical harmonic expansions, III, J. Comp. Phys. 229 (2010) 6181–6192.

[15] E. Michielssen, A. Boag, A multilevel matrix decomposition algorithm for analyzing scattering from large structures, IEEE Trans. Antennas Propagat. 44 (1996) 1086–1093.

[16] M. O'Neil, F. Woolfe, V. Rokhlin, An algorithm for the rapid evaluation of special function transforms, Appl. Comput. Harmon. Anal. 28 (2010) 203–226.

[17] M. G. Reuter, M. A. Ratner, T. Seideman, A fast method for solving both the time-dependent Schrödinger equation in angular coordinates and its associated "$m$-mixing" problem, J. Chem. Phys. 131 (2009) 094108–1–094108–7.

[18] D. S. Seljebotn, Wavemoth–fast spherical harmonic transforms by butterfly matrix compression, Astro. J. Suppl. Series 199 (2012) 12.

[19] N. P. Wedi, M. Hamrud, G. Mozdzynski, A fast spherical harmonics transform for global NWP and climate models, Monthly Weather Review 141 (2013) 3450–3461.

[20] A. Schuster, On some definite integrals, and a new method of reducing a function of spherical co-ordinates to a series of spherical harmonics, Phil. Trans. of the Royal Society A 200 (1903) 181–223.

[21] D. J. Hofsommer, M. L. Potters, Table of Fourier coefficients of associated Legendre functions, in: Proc. Koninklijke Nederlandse Akademie van Wetenschappen: Series A: Mathematical Sciences, Vol. 63, 1960, pp. 460–480.

[22] N. Sneeuw, R. Bun, Global spherical harmonic computation by two-dimensional Fourier methods, J. Geod. 70 (1996) 224–232.

[23] C. Gruber, O. Abrykosov, On computation and use of Fourier coefficients for associated Legendre functions, J. Geod. 90 (2016) 525–535.

[24] H. Cheng, Z. Gimbutas, P. G. Martinsson, V. Rokhlin, On the compression of low rank matrices, SIAM J. Sci. Comput. 26 (2005) 1389–1404.

[25] E. Liberty, F. Woolfe, P.-G. Martinsson, V. Rokhlin, M. Tygert, Randomized algorithms for the low-rank approximation of matrices, Proc. Nat. Acad. Sci. 104 (2007) 20167–20172.

[26] R. M. Slevinsky, `https://github.com/MikaelSlevinsky/FastTransforms.jl`, GitHub.

[27] S. M. Cox, P. C. Matthews, Exponential time differencing for stiff systems, J. Comp. Phys. 176 (2002) 430–455.

[28] U. M. Ascher, S. J. Ruuth, R. J. Spiteri, Implicit-explicit Runge–Kutta methods for time-dependent partial differential equations, Appl. Numer. Math. 25 (1997) 151–167.

[29] A. Townsend, S. Olver, The automatic solution of partial differential equations using a global spectral method, J. Comput. Phys. 299 (2015) 106–123.

[30] R. M. Slevinsky, S. Olver, A fast and well-conditioned spectral method for singular integral equations, J. Comp. Phys. 332 (2017) 290–315.

[31] E. U. Condon, G. H. Shortley, The Theory of Atomic Spectra, Cambridge University Press, Cambridge, England, 1951.

[32] F. W. J. Olver, D. W. Lozier, R. F. Boisvert, C. W. Clark (Eds.), NIST Handbook of Mathematical Functions, Cambridge U. P., Cambridge, UK, 2010.

[33] M. Abramowitz, I. A. Stegun, Handbook of Mathematical Functions, Dover, New York, 1965.

[34] H. E. Salzer, A recurrence scheme for converting from one orthogonal expansion into another, Comm. ACM 16 (1973) 705–707.

[35] N. J. Higham, Accuracy and Stability of Numerical Algorithms, 2nd Edition, SIAM, 2002.

[36] M. Gu, S. C. Eisenstat, Efficient algorithms for computing a strong rank-revealing QR factorization, SIAM J. Sci. Comput. 17 (1996) 848–869.

[37] H. J. Landau, H. O. Pollak, Prolate spheroidal wave functions, Fourier analysis and uncertainty – III: the dimension of the space of essentially time- and band-limited signals, Bell System Tech. J. 41 (1962) 1295–1336.

[38] I. S. Gradshteyn, I. M. Ryzhik, Table of Integrals, Series, and Products, Seventh Edition, Elsevier Academic Press, Burlington, MA, 2007.

[39] C. L. Frenzen, R. Wong, A uniform asymptotic expansion of the Jacobi polynomials with error bounds, Can. J. Math. 37 (1985) 979–1007.

[40] X.-X. Bai, Y.-Q. Zhao, A uniform asymptotic expansion for Jacobi polynomials via uniform treatment of Darboux's method, J. Approx. Theory 148 (2007) 1–11.

[41] L. Greengard, V. Rokhlin, A fast algorithm for particle simulations, J. Comp. Phys. 73 (1987) 325–348.

[42] N. Elezović, Asymptotic expansions of central binomial coefficients and Catalan numbers, J. Integer Seq. 17 (2014) 14.2.1.

[43] I. Bogaert, B. Michiels, J. Fostier, $O(1)$ computation of Legendre polynomials and Gauss–Legendre nodes and weights for parallel computing, SIAM J. Sci. Comput. 34 (2012) C83–C101.

[44] J. Keiner, Computing with expansions in Gegenbauer polynomials, SIAM J. Sci. Comput. 31 (2009) 2151–2171.

[45] B. K. Alpert, V. Rokhlin, A fast algorithm for the evaluation of Legendre expansions, SIAM J. Sci. Stat. Comput. 12 (1991) 158–179.

[46] N. J. Higham, The numerical stability of barycentric Lagrange interpolation, IMA J. Numer. Anal. 24 (2004) 547–556.

[47] J.-P. Berrut, L. N. Trefethen, Barycentric Lagrange interpolation, SIAM Rev. 46 (2004) 501–517.

[48] N. Hale, A. Townsend, A fast, simple, and stable Chebyshev–Legendre transform using an asymptotic formula, SIAM J. Sci. Comput. 36 (2014) A148–A167.

[49] R. M. Slevinsky, On the use of Hahn's asymptotic formula and stabilized recurrence for a fast, simple, and stable Chebyshev–Jacobi transform, IMA J. Numer. Anal.

[50] A. Townsend, M. Webb, S. Olver, Fast polynomial transforms based on Toeplitz and Hankel matrices, Math. Comp.

[51] J. Bezanson, S. Karpinski, V. B. Shah, A. Edelman, Julia: a fast dynamic language for technical computing, arXiv:1209.5145 (2012).

[52] J. Bezanson, A. Edelman, S. Karpinski, V. B. Shah, Julia: a fresh approach to numerical computing, arXiv:1411.1607 (2014).

[53] K. L. Ho, `https://github.com/klho/LowRankApprox.jl`, GitHub.

[54] Z. Xianyi, `https://github.com/xianyi/OpenBLAS`, GitHub.

[55] J. Poulson, L. Demanet, N. Maxwell, L. Ying, A parallel butterfly algorithm, SIAM J. Sci. Comput. 36 (2014) C49–C65.

[56] Y. Li, H. Yang, E. R. Martin, K. L. Ho, L. Ying, Butterfly factorization, Multiscale Model. Simul. 13 (2015) 714–732.

[57] J. L. Aurentz, R. Vandebril, D. S. Watkins, Fast computation of the zeros of a polynomial via factorization of the companion matrix, SIAM J. Sci. Comput. 35 (2013) A255–A269.

[58] G. M. Vasil, K. J. Burns, D. Lecoanet, S. Olver, B. P. Brown, J. S. Oishi, Tensor calculus in polar coordinates using Jacobi polynomials, J. Comp. Phys. 325 (2016) 53–73.

[59] T. Koornwinder, Two-variable analogues of the classical orthogonal polynomials, in: R. Askey (Ed.), Theory and Application of Special Functions, 1975, pp. 435–495.