A Fast and Accurate Algorithm for Spherical Harmonic Analysis of
the Cosmic Microwave Background Radiation
Kathryn P. Drake∗,† and Grady B. Wright∗

April 25, 2019

Abstract

The Cosmic Microwave Background Radiation (CMBR) represents the first light to travel during the early stages of the universe’s development. This sphere of relic radiation gives the strongest evidence for the Big Bang theory to date, and refined analysis of its angular power spectrum can lead to revolutionary developments in understanding the nature of dark matter and dark energy. Satellites collect CMBR data over a sphere using a Hierarchical Equal Area isoLatitude Pixelation (HEALPix) grid. While this grid gives a quasiuniform discretization of a sphere, it is not well suited for doing fast and accurate spherical harmonic analysis – a central component to computing and analyzing the angular power spectrum of the massive CMBR data sets. In this paper, we present a new method that overcomes these issues through a novel combination of a non-uniform fast Fourier transform, the double Fourier sphere method, and Slevinsky’s fast spherical harmonic transform [25]. The method has a quasi-optimal computational complexity of $O(N \log^2 N)$ with an initial set-up cost of $O(N^{3/2} \log N)$, where $N$ represents the number of points in the HEALPix grid. Additionally, we provide the first analysis of the method used in the current HEALPix software for computing the spherical harmonic coefficients. Numerical results illustrating the effectiveness of the new technique over the current method are also included.

1 Introduction

About 379,000 years after the universe formed, the dense plasma of matter cooled enough to allow photons to move freely through space. Faintly glowing at the edge of the observable universe, this Cosmic Microwave Background Radiation (CMBR) gives the strongest evidence for the Big Bang Theory to date [4]. While the CMBR has been deemed “the most perfect black body ever measured in nature” [27], there are temperature fluctuations on the order of $O(10^{-5})$ that give insight into the primordial universe [20]. These temperature anisotropies are consequences of the initial density distribution of matter, and analyzing them can provide a better understanding of the geometry and composition of the universe [4,13].

Using spacecraft which probe the sky in the microwave and infra-red frequencies, scientists have measured the CMBR temperature differences at small angular scales. These measurements, which are collected on a Hierarchical Equal Area isoLatitude Pixelation (HEALPix) grid, are used to produce high resolution sky maps of the CMBR (Figure 1a). The most recent maps from the Planck mission consist of millions of pixels [21]. Once a sky map is composed, it can then be analyzed by its angular power spectrum. This quantity measures the amplitude of the CMBR temperature fluctuations as a function of wavelength and is used to estimate parameters of the cosmological model for the universe [27]. For example, the confirmation of the first peak in the angular power spectrum affirmed that the universe is spatially flat [14]. The values of the angular spectrum at higher wavelengths are crucial to many aspects of modern cosmology, including the density of dark matter and dark energy in the universe. The CMBR power spectrum (Figure 1b) is calculated from the spherical harmonic coefficients, $b_{\ell m}$, of the sky map as follows:

$$C_\ell = \frac{1}{2\ell+1} \sum_{m} |b_{\ell m}|^2.$$  

∗Department of Mathematics, Boise State University, Boise, ID 83725-1555 (kathryndrake@u.boisestate.edu, gradywright@boisestate.edu).
†Corresponding author.

arXiv:1904.10514v1 [math.NA] 23 Apr 2019
Figure 1: CMBR nine year exposure map from the Planck mission [21] (a) and corresponding (scaled) angular power spectrum (b).

The spherical harmonic conventions used in this work are detailed in Appendix A. While the HEALPix grid and the associated eponymous software were created for gathering and mining CMBR data, some of their properties make it challenging to perform both fast and accurate spherical harmonic analysis, affecting angular power spectrum computations. The focus of this work is on a new method for overcoming these challenges.

In this paper, we introduce a new algorithm for calculating the spherical harmonic coefficients of data collected at the HEALPix points. This technique was motivated by the fast spherical harmonic transform (FSHT), introduced by Slevinsky in [25], which converts the bivariate Fourier coefficients for data on the sphere to spherical harmonic coefficients of the data with near optimal complexity. By synthesizing the Nonuniform fast Fourier transform (NUFFT) [22] and the Double Fourier Sphere (DFS) [26] methods, we give a fast method for obtaining the bivariate Fourier coefficients for functions sampled at the HEALPix grid. By combing these two ideas, we develop a fast and accurate algorithm for spherical harmonic analysis of HEALPix datasets.

This paper is structured in the following manner. In section 2, we offer supporting information on the HEALPix grid as well as details and the first analysis of the current method used in the HEALPix software for computing the spherical harmonic coefficients of CMBR maps. We present the new algorithm for fast spherical harmonic analysis of data collected at the HEALPix points in section 3. Numerical results comparing the presented method with that of the HEALPix software for calculating the angular power spectrum of functions on the sphere are given in section 4. Finally, in section 5, we give some brief conclusions and remarks on future directions of the work.

2 Background and Current Approach

The HEALPix grid[1] and its associated software are the current methods utilized by NASA for collecting and analyzing CMBR data. We explain the relevant details of these techniques in order to support the understanding of the improvements offered in the new algorithm presented in section 3.

2.1 HEALPix Grid

The HEALPix scheme was created to discretize functions on the sphere at high resolutions and was specifically motivated by the need to efficiently perform spherical harmonic transform computations on increasingly large

---

[1]The HEALPix grid consists of a collection of pixels of different shapes but the same area. However, for our method we do not exploit this fact and simply treat the center of each pixel as a point with the given value of the pixel.
CMBR datasets \[10\]. While there are many options for discretizing the sphere, there is no known deterministic method that gives a set of quasiumiform points and allows for exact spherical harmonic decompositions of band-limited functions using equal-weight quadrature. Even though the HEALPix grid does not offer optimal complexity for spherical harmonic analyses, it does achieve some efficiency gains over existing point sets. This improvement is accomplished by three properties of the HEALPix points: hierarchical database structure, equal area element partition, and isolatitude distribution of the elements on the sphere. These features can be demonstrated using the base resolution of pixels. The grid resolution is defined using the parameter \( N_{\text{side}} = 2^t, t \in \mathbb{N} \), which creates \( N_{\text{side}}^2 \) equal area divisions of each base pixel. Figure 2 illustrates the base resolution grid in (a), and the next level of resolution refinement in (b) where each base pixel is subdivided further into four equal area pixels. A HEALPix map therefore has \( N = 12N_{\text{side}}^2 \) equal area (but differently shaped) pixels, with the centers placed on \( 4N_{\text{side}} - 1 \) rings of constant latitude. For any \( N_{\text{side}} \), the HEALPix centers (which we call the HEALPix points), are defined algebraically using three regions of the sphere, two polar (\( \mathcal{N} \) and \( \mathcal{S} \)) and one equatorial (\( \mathcal{E} \)) \[15\]. In spherical coordinates, the points in these regions are given as

\[
\begin{align*}
\mathcal{N} & := \left\{ \left( \arccos \left( 1 - \frac{j^2}{3N_{\text{side}}^2} \right), \frac{\pi (k + \frac{1}{2})}{2j} \right) : j = 1, \ldots, N_{\text{side}} - 1, k = 0, \ldots, 4j - 1 \right\} \\
\mathcal{E} & := \left\{ \left( \arccos \left( \frac{2(2N_{\text{side}} - j)}{3N_{\text{side}}} \right), \frac{\pi \left( k + \frac{(j+1) \text{mod} 2}{2} \right)}{2N_{\text{side}}} \right) : j = N_{\text{side}}, \ldots, 3N_{\text{side}}, k = 0, \ldots, 4N_{\text{side}} - 1 \right\} \\
\mathcal{S} & := \left\{ \left( \arccos \left( -\left( 1 - \frac{j^2}{3N_{\text{side}}^2} \right) \right), \frac{\pi (k + \frac{1}{2})}{2j} \right) : j = 1, \ldots, N_{\text{side}} - 1, k = 0, \ldots, 4j - 1 \right\}.
\end{align*}
\]

The final HEALPix point set is \( \mathcal{X} = \mathcal{N} \cup \mathcal{E} \cup \mathcal{S} \). The number of points on each ring varies in the polar regions, with only four points on the rings closest to the north and south poles of the sphere, whereas the rings in the equatorial region have the same number of points. This point distribution is illustrated more clearly in Figure 3, where the HEALPix points are mapped to a latitude-longitude grid.

The biggest advantage for spherical harmonic analysis in the HEALPix scheme lies in the equally-spaced points on each ring of constant latitude. While this aides computation in the longitude direction with FFTs, the misaligned and unequally spaced points in latitude make fast bivariate Fourier analysis impossible without modification. We address this in the new algorithm presented in section 3.

Figure 2: HEALPix grid with resolutions (a) \( N_{\text{side}} = 1 \) and (b) \( N_{\text{side}} = 2 \). (Taken from \[10\])

\[2\]
Figure 3: HEALPix grid on $[0, 2\pi] \times [0, \pi]$, where $\theta$ is latitude, and $\lambda$ is longitude. The point sets in the northern ($N$), equatorial ($E$), and southern ($S$) regions are shown in blue, red, and yellow, respectively.

2.2 HEALPix Software Spherical Harmonic Analysis

The HEALPix software \cite{9} estimates the angular power spectrum in (1) of data at the HEALPix points by first approximating the spherical harmonic coefficients of the data as

$$b_{\ell m} = \frac{4\pi}{N} \sum_{i=1}^{N} Y_{\ell m}^{*} (\lambda_i, \theta_i) f(\lambda_i, \theta_i), \ 0 \leq \ell \leq \ell_{max}, -\ell \leq m \leq \ell,$$

where $(\lambda_i, \theta_i)$ are HEALPix points in longitude-latitude, $f$ is the data, and $Y_{\ell m}$ is a spherical harmonic of degree $\ell$ and order $m$ (see Appendix A for a discussion of the spherical harmonic conventions used in this paper). While the user can input any band limit $\ell_{max}$ for this approximation, the software default is $\ell_{max} = 3N_{side} - 1$. Due to the isolatitude nature of the HEALPix points, this computation is done with $O(N^3)$ complexity as opposed to $O(N^2)$ \cite{10}. Note that $N \sim \ell_{max}^3$, so the complexity of the $b_{\ell m}$ computation is equivalent to $O(\ell_{max})$. The expression (3) is a low-order approximation to the continuous inner product (22) which defines the coefficients, since it uses a simple equal weight quadrature. In order to improve this approximation, the software employs an iterative procedure, which is referred to as a “Jacobi iteration” \cite{10}. We explain this iterative method below in the language of linear algebra in order to show what it converges to.

The analysis operation, defined in (3), produces an approximation to the spherical harmonic coefficients from the data $f$ on the sphere, whereas the synthesis operation reconstructs the data given the spherical harmonic coefficients:

$$\hat{f}(\lambda, \theta) = \sum_{\ell=0}^{\ell_{max}} \sum_{m=-\ell}^{\ell} b_{\ell m} Y_{\ell m} (\lambda, \theta), \ i = 1, \ldots, N$$

Note that we use a hat on $f$ to indicate that computing the spherical harmonic coefficients according to (3) and using them in (4) gives different function values. In matrix-vector notation, we denote (3) and (4) as

**Analysis:** $\mathbf{b} = \mathbf{A}\mathbf{f}$

**Synthesis:** $\mathbf{\hat{f}} = \mathbf{S}\mathbf{b}$,

where $\mathbf{b}$ is the vector of spherical harmonic coefficients and $\mathbf{f}$ and $\mathbf{\hat{f}}$ are the vectors of data values at the HEALPix points. Using this notation, the iterative procedure in the HEALPix software can be written as

$$\mathbf{r}^{(k+1)} = \mathbf{f} - \mathbf{S}\mathbf{b}^{(k)},$$

$$\mathbf{b}^{(k+1)} = \mathbf{b}^{(k)} + \mathbf{A}\mathbf{r}^{(k+1)},$$

(5)
where \( \mathbf{r} \) is the residual vector and \( \mathbf{b}^{(0)} = \mathbf{A}\mathbf{f} \). Substituting the first equation of \((5)\) into the last and using the fact that the Analysis matrix satisfies \( \mathbf{A} = \frac{4\pi}{N} \mathbf{S}^* \), gives the iteration
\[
\mathbf{b}^{(k+1)} = \mathbf{b}^{(k)} + \frac{4\pi}{N} \mathbf{S}^*(\mathbf{f} - \mathbf{S}\mathbf{b}^{(k)}) = \frac{4\pi}{N} \mathbf{S}^*\mathbf{f} + \left( \mathbf{I} - \frac{4\pi}{N} \mathbf{S}^*\mathbf{S} \right) \mathbf{b}^{(k)}.
\] (6)

This is just stationary Richardson iteration (or Gradient Decent) with relaxation parameter \( \frac{4\pi}{N} \) applied to the normal equations \( \mathbf{S}^*\mathbf{S}\mathbf{b} = \mathbf{S}^*\mathbf{f} \). Thus, the iterative procedure converges to the least squares solution to \((3)\), provided the spectral radius of \( \mathbf{I} - \frac{4\pi}{N} \mathbf{S}^*\mathbf{S} \) is less than one. The spectral radius also determines the convergence rate. While we cannot prove this bound always holds on the spectral radius, we can numerically check the value. Table I displays the results for several values of \( N \). We see that the spectral radius is much less than 1 and that it appears to decrease with \( N \), indicating the iteration \((6)\) converges rapidly.

| \( N \) | \( \rho (\mathbf{I} - \frac{4\pi}{N} \mathbf{S}^*\mathbf{S}) \) |
|-------|----------------|
| 48    | 0.1986         |
| 192   | 0.0932         |
| 768   | 0.0600         |
| 3072  | 0.0475         |
| 12288 | 0.0421         |

Table 1: Spectral radius of the Richardson iteration matrix from \((6)\) for different values of \( N \).

The default option in the HEALPix software sets the number of iterations of \((6)\) to 3. While this does improve the accuracy of computing the spherical harmonic coefficients, it adds to the cost, as each iteration requires doing an analysis and synthesis (\((3)\) and \((4)\)) at a cost of \( \mathcal{O}(\ell_{\text{max}}^3) \) operations each. Since the solution converges to the least squares solution, one could improve the convergence of the Richardson iteration method by using algorithms like LSQR or Conjugate Gradient on the normal equations \((19)\). We do not take the least squares approach in our method, but instead compute the bivariate Fourier coefficients of the HEALPix data and then use these to obtain the spherical harmonic coefficients.

3 HP2SPH

The algorithm presented here, named HP2SPH, introduces a new way to calculate the spherical harmonic coefficients of data sampled at the HEALPix points \((2)\). The outline for the algorithm is given in Algorithm 1 and each of the pieces are described below.

Algorithm 1 HP2SPH

**Input:** Data sampled at the HEALPix point set of size \( N \), \( \{f_j\}, j = 1, \ldots, N \).

**Output:** Spherical harmonic coefficients, \( \{b_{\ell}^m\}, 0 \leq \ell \leq \ell_{\text{max}}, -\ell \leq m \leq \ell \)

1. Transform the data to a tensor product latitude-longitude grid:
   (i) Upsample the data in longitude from the northern (\( N \)) and southern (\( S \)) point sets using FFT
   (ii) Shift (interpolate) the data from the equatorial (\( E \)) point set so it “lines up” in longitude
2. Compute the bivariate Fourier coefficients:
   (i) Apply the DFS method
   (ii) Apply the inverse NUFFT in latitude
   (iii) Apply the inverse FFT in longitude
3. Obtain the spherical harmonic coefficients via the FSHT
3.1 Step 1: Transform the data to a tensor product latitude-longitude grid

As described in Section 2.1, the HEALPix grid has an unequal number of points on the rings in the northern (N) and southern (S) sets ([2], and the points on the rings in the equatorial (E) set are shifted on every other ring. This structure leads to the pixels being misaligned in latitude. By upsampling the data on the northern and southern points in longitude so that there are an equivalent samples of the data on each ring and shifting the data at equatorial points in longitude, we can use fast algorithms to obtain the bivariate Fourier coefficients of the data as discussed in the next section. On the two polar point sets, we upsample the data using the trigonometric interpolant of the data on each ring of these sets to the non-shifted equally spaced longitude points on the equatorial rings, i.e.,

\[
\lambda_k = \frac{k}{2N_{\text{side}}} \pi, \quad k = 0, \ldots, 4N_{\text{side}} - 1. \tag{7}
\]

We also interpolate the data on the rings in the equatorial point set with shifted longitude points, to these \(\lambda\) values. Figure 4(b) illustrates the upsampling procedure leading to a tensor product latitude-longitude grid of data of size \((4N_{\text{side}} - 1) \times 4N_{\text{side}}\).

We describe the interpolation procedure here for the data in the northern point set \(N\). Consider the latitude values for the northern rings, \(\theta_j = \arccos\left(1 - \frac{j^2}{3N_{\text{side}}^2}\right), \quad j = 1, \ldots, N_{\text{side}}\). We approximate the data in each ring using a trigonometric expansion of the form

\[
f_j(\lambda) = \sum_{n=-2j}^{2j} c_n^{(j)} e^{in\lambda}, \tag{8}
\]

The coefficients in the expansion are determined by enforcing interpolation of the given data values

\[
f\left(\frac{k + \frac{1}{2} \pi}{2j}, \theta_j\right), \quad k = 0, \ldots, 4j - 1.
\]

With the minor algebraic manipulation of (8),

\[
f_j\left(\frac{k + \frac{1}{2} \pi}{2j}\right) = \sum_{n=-2j}^{2j-1} c_n^{(j)} e^{in\frac{k + \frac{1}{2} \pi}{2j}}, \quad k = 0, \ldots, 4j - 1.
\]
we see the interpolation conditions yield the system

$$
\sum_{n=-2j}^{2j-1} c_n^{(j)} e^{i n \frac{k}{2j} \pi} = f \left( \frac{k + \frac{1}{2}}{2j}, \pi, \theta_j \right), \quad k = 0, \ldots, 4j - 1,
$$

which can be computed using the inverse FFT. We can then obtain the Fourier coefficients $c_n^{(j)}$ in (8) for the data at the non-shifted values through simple multiplication. Finally, we pad the vector containing the coefficients $c_n^{(j)}$ with the appropriate number of zeros to get to a total of $4N_{\text{side}}$, so that the expansion in longitude in each ring has the same number of Fourier coefficients. The values of the interpolant on each ring can then be obtained at the upsampled values (7) by applying the FFT on these padded vectors. A similar procedure can be applied to the data on the rings in the southern point set $S$.

On the rings in the equatorial set $E$ where the longitude values are shifted by $\pi(k + \frac{1}{2})/(2N_{\text{side}})$, we compute the Fourier coefficients of the data using (9) with $j = N_{\text{side}}$. We then obtain the coefficients in (8) at the non-shifted points from which the values can be computed using the FFT. No padding or upsampling is needed in this case.

### 3.2 Step 2: Compute Bivariate Fourier Coefficients

Bivariate Fourier analysis for data on the sphere requires the application of the DFS method to obtain periodicity of the data in latitude and to retain spherical symmetry. When we apply this method to the upsampled HEALPix data, there is an issue that the points in latitude are not equally-spaced, making the standard FFT unsuitable. To bypass this issue we use an NUFFT. Both the DFS technique and NUFFT method we use are discussed below for completeness.

#### 3.2.1 Double Fourier Sphere (DFS) Method

A natural approach to representing a function on the sphere is to use a latitude-longitude coordinate transform, defined by

$$
x(\lambda, \theta) = \cos \lambda \sin \theta, \quad y(\lambda, \theta) = \sin \lambda \sin \theta \quad z(\lambda, \theta) = \cos \theta, \quad (\lambda, \theta) \in [0, 2\pi] \times [0, \pi],
$$

which maps the sphere to a rectangular domain. While this transformation allows for performing computations with the function $f(\lambda, \theta) = f(x(\lambda, \theta), y(\lambda, \theta), z(\lambda, \theta))$, it also introduces artificial boundaries at the north and south poles. In addition, the change of variables does not maintain the symmetry of functions on the sphere. Specifically, the transform described in (10) does not preserve the periodicity in the latitude direction, which is necessary for bivariate Fourier analysis to be applicable and for results to make physical sense. These problems are solved by the DFS method.

Originally introduced by Merilee in [16] (see also [26]) the DFS method transforms a function on the sphere to a rectangular grid while maintaining bi-periodicity. This can be thought of as “doubling up” the function $f(\lambda, \theta)$ to form a new function that preserves periodicity in both the latitude and longitude directions. Algebraically, this new function, $\tilde{f}(\lambda, \theta)$, is defined on $[0, 2\pi] \times [0, 2\pi]$ as follows [26]

$$
\tilde{f}(\lambda, \theta) = \begin{cases} 
g(\lambda, \theta), & (\lambda, \theta) \in [0, \pi] \times [0, \pi], \\
h(\lambda - \pi, \theta), & (\lambda, \theta) \in [\pi, 2\pi] \times [0, \pi], \\
h(\lambda, 2\pi - \theta), & (\lambda, \theta) \in [0, \pi] \times [\pi, 2\pi], \\
g(\lambda - \pi, 2\pi - \theta), & (\lambda, \theta) \in [\pi, 2\pi] \times [\pi, 2\pi], 
\end{cases}
$$

where $g(\lambda, \theta) = f(\lambda, \theta)$ and $h(\lambda, \theta) = f(\lambda + \pi, \theta)$ for $(\lambda, \theta) \in [0, \pi] \times [0, \pi]$. Figure 6 illustrates the DFS method applied to the surface of the Earth, which shows the preservation of bi-periodicity in (c). We note that the DFS method can also be easily applied to discrete data sampled at a tensor product latitude-longitude grid using (11), which is what we do for the upsampled HEALPix data. In this case, (11) corresponds to flipping and shifting the data matrix appropriately.
Once the DFS method is applied to a function on the sphere, it can be approximated using a 2D bivariate Fourier expansion:

\[
\tilde{f}(\lambda, \theta) \approx \sum_{j=-\lceil \frac{m}{2} \rceil}^{\lceil \frac{m}{2} \rceil - 1} \sum_{k=-\lceil \frac{n}{2} \rceil}^{\lceil \frac{n}{2} \rceil - 1} C_{jk} e^{ij\theta} e^{ik\lambda}, \tag{12}
\]

where \( m \) and \( n \) represent the number of frequencies in (doubled-up) latitude and longitude, respectively.

Note that the HEALPix grid does not include points at the north and south poles. When applying the DFS to the upsampled data from Step 1, this leads to a relatively large gap in the points in latitude over the poles compared to the other points, which can lead to issues with the inverse NUFFT (see below). To bypass this issue, we construct values at the two poles by using a weighted quadratic least squares fit \[7\] that combines the data from the three rings closest to each pole.

**Remark.** The Fourier coefficients of the upsampled data in longitude are computed in Step 1. These can be used directly in the DFS procedure by applying \[11\] in Fourier space in the \( \lambda \) variable, which amounts to appending the (padded) coefficient matrix from Step 1 with a flipped version of itself that is multiplied by \( e^{i\pi} \). It then only remains to compute the Fourier coefficients in latitude \( \theta \) to obtain the full bivariate Fourier coefficients. This is the focus of the next step.

### 3.2.2 Nonuniform Fast Fourier Transform (NUFFT)

The use of the nonuniform discrete Fourier transform (NUDFT) in many domain sciences has led to the development of algorithms for computing it efficiently. If these algorithms are quasi-optimal requiring \( \mathcal{O}(n \log n) \), then they are referred to as a nonuniform fast Fourier transform (NUFFT). Given a vector \( c \in \mathbb{C}^{n \times 1} \), the one-dimensional NUDFT computes the vector \( f \in \mathbb{C}^{n \times 1} \) defined by

\[
f_j = \sum_{k=0}^{n-1} c_k e^{-2\pi i x_j \omega_k}, \quad 0 \leq j \leq n - 1, \tag{13}
\]

where \( x_j \in [0, 1] \) are the samples and \( \omega_k \in [0, n] \) are the frequencies. If the samples are equispaced \( (x_j = j/n) \) and the frequencies are integer \( (\omega_k = k) \), then the the transform is a uniform DFT, which can be computed
by the FFT in $O(n \log n)$ operations \[6\]. When either the samples are nonequispaced or the frequencies are
noninteger, the FFT does not directly apply without some careful manipulation \[3\].

Ruiz and Townsend \[22\] contributed to the collection of NUFFT algorithms by utilizing low rank matrix
approximations to relate the NUDFT to the uniform DFT. There are three types of NUDFTs and inverse
NUDFTs that they account for in their algorithm: NUDFT-I, which has uniform samples but noninteger
frequencies; NUDFT-II, which has nonuniform samples and integer frequencies; NUDFT-III, which has
both nonuniform samples and nonuniform frequencies \[11\]. Since our HP2SPH method only uses the one-
dimensional inverse NUFFT of the second type, we discuss the NUFFT-II algorithm \[22\].

Given Fourier coefficients, $c \in \mathbb{C}^{n \times 1}$, the NUFFT-II attempts to approximates the matrix-vector product
\[
f = \hat{F}_2 c,
\]
(14)
to machine precision in quasi-optimal complexity. Here $(\hat{F}_2)_{jk} = e^{-2\pi ijx_j k}$, $x_j$ are nonuniform samples
(restricted to be in $[0, 1]$), and $k$ are integer frequencies for $0 \leq j, k \leq n - 1$. Notice that the DFT matrix for
uniform samples and integer frequencies is similarly $F_{jk} = e^{-2\pi ijk/n}$. The key to the NUFFT-II algorithm
described in \[22\] is that if the samples are nearly equispaced, then $\hat{F}_2$ can be related to $F$ with a low rank
matrix. This means that given a rank $K$ approximation which relates $\hat{F}_2$ and $F$, the NUFFT-II can then
be computed using $K$ FFTs with $O(Kn \log n)$ cost. In practice, machine precision can be achieved with
$K = 14$ \[22\].

In the case of the inverse NUFFT-II, Ruiz and Townsend advocate for the use of the conjugate gradient
(CG) method in order to solve the linear system $\hat{F}_2 c = f$ for $c$. Since $\hat{F}_2$ is not positive definite, the CG
method is applied to the normal equations:
\[
\hat{F}_2^* \hat{F}_2 c = \hat{F}_2^* f,
\]
(15)
where $\hat{F}_2^* \hat{F}_2$ is simply a Toeplitz matrix. Therefore, the inverse NUFFT-II can be calculated using the CG
method and a fast Toeplitz multiplication \[8\] in $O(R_{CG} n \log n)$ operations, where $R_{CG}$ is the number of CG
iterations. The following stipulation is placed on the nonuniform function samples to avoid ill-conditioning
in the system \[15\] \[22\]:
\[
\left| x_j - \frac{j}{n} \right| \leq \frac{\gamma}{n}, \quad 0 \leq j \leq n - 1,
\]
(16)
where $0 \leq \gamma \leq 1/2$. When this condition is satisfied, it has been experimentally observed that $R_{CG} \sim 10$
for a large range of $n$.

For the method proposed in this paper, we apply the inverse NUFFT-II in latitude to the DFS upsampled
HEALPix data from Step 2. Unfortunately, the HEALPix points in latitude direction do not meet the
condition \[16\]. To bypass this issue, we instead use a least squares solution to compute fewer coefficients at
higher wave numbers than what the given data may support. We describe this procedure below since it
not discussed in \[22\].

The inverse NUFFT-II method computes first column of the symmetric Toeplitz matrix $\hat{F}_2^* \hat{F}_2$ in \[15\] in
the following manner:
\[
\hat{F}_2^* \hat{F}_2 e_1 = \left( \hat{F}_2^T \hat{F}_2 \right)^* = (\hat{F}_2^T \hat{F}_2)^*.
\]
The last expression above can be computed efficiently by the NUFFT-I algorithm, since the NUDFT-I matrix
is simply the transpose of the NUDFT-II matrix \[22\]. To compute a least squares solution to \[14\] with fewer
coefficients, we simply truncate the first column obtained from the above method to $m < n$ terms and form
the resulting $m \times m$ Toeplitz matrix $\hat{F}_2^* \hat{F}_2$. The right hand side for the least squares solution is obtained by
similarly computing $\hat{F}_2^* f$ and truncating this to $m$ terms.

For the DFS upsampled HEALPix data from Step 2, there are $8N_{side}$ coefficients points in latitude, but
only $4N_{side}$ coefficients in longitude. To keep the number of Fourier modes in both directions (nearly) the
same, we choose $m = 4N_{side} + 1$ as the truncation parameter for the least squares solution for computing
the Fourier coefficients in latitude. This is also a convenient choice since the method in step three for converting
bivariate Fourier coefficients of data on the sphere to spherical harmonic coefficients requires the number of
coefficients in each direction is the same and an odd number (we explain how to convert the coefficients in
longitude to $m = 4N_{side} + 1$ in the next section).
3.3 Step 3: Obtain the spherical harmonic coefficients via the fast spherical harmonic transform (FSHT)

The ubiquitous nature of spherical harmonics analysis for data on the sphere has made the development of a fast spherical harmonic transform a topic of interest for the scientific community. In [25], Slevinsky derives a fast, backward stable method for the transformation between spherical harmonic expansions and their bivariate Fourier series (given in (12)) by viewing it as a change of basis. This relation is defined as a connection problem, and the matrices that arise in the present case are well-conditioned, making them ideal for fast computations. Slevinsky describes the change of basis in two steps: converting expansions in normalized associated Legendre functions to those of only order zero and one, and then re-expressing these in trigonometric form. In other words, it uses spherical harmonic expansions of order zero and one as intermediate expressions between higher-order spherical harmonics and their corresponding bivariate Fourier coefficients.

The first step of the algorithm takes advantage of the fact that the matrix of connection coefficients between the associated Legendre functions of all orders and those of order zero and one can be represented by a product of Givens rotation matrices. This enables the use of the butterfly algorithm, which can be thought of as an abstraction of the algebraic properties of fast Fourier transforms. The term butterfly was introduced in [17], where it was used for analyzing scattering from electrically large surfaces, and then further developed in [18] for use in special function transforms. Slevinsky uses the butterfly algorithm to perform a factorization of the well-conditioned matrices of connection coefficients.

The second step of this method exploits the hierarchical decompositions of the connection coefficient matrices between the associated Legendre functions of order zero and one to the Chebyshev polynomials of the first and second kind. This step quickly computes the fast orthogonal polynomial transforms using an adaptation of the Fast Multipole Method [12] and low-rank matrix approximations. The total pre-computation time for both steps is \( O(\ell_{\text{max}}^3 \log \ell_{\text{max}}) \), and execution time is asymptotically optimal with \( O(\ell_{\text{max}}^2 \log^2 \ell_{\text{max}}) \) operations. This FSHT is implemented in Julia with the package FastTransforms [24] (as are the NUFFT methods from [22] used in Step 2).

The FSHT in FastTransforms assumes the input function has a bivariate Fourier expansion of the form:

\[
\tilde{f}(\lambda, \theta) = \sum_{j=-p}^{p} \sum_{k=-p}^{p} g_{j,k} e^{ij \lambda} \frac{1}{\sqrt{2\pi}} \begin{cases} 
\cos j\theta, & k \text{ even} \\
\sin(j+1)\theta, & k \text{ odd} 
\end{cases}, \tag{17}
\]

Any function on the sphere is required to have these even/odd conditions on its bivariate Fourier coefficients [16]. At the end of step 2 we have obtained the bivariate Fourier expansion of the data of the form:

\[
\tilde{f}(\lambda, \theta) = \sum_{j=-p}^{p} \sum_{k=-p}^{p-1} C_{j,k} e^{ij \lambda}, \tag{18}
\]

where \( p = N_{\text{side}}/2 \). Since we are dealing with real-valued data, we can expand Fourier coefficients array in \( \lambda \) to an odd number of points. The expanded array is defined by:

\[
X_{j,k} = \begin{cases} 
C_{j,k} & \text{if } -p + 1 \leq k \leq p - 1 \\
\frac{1}{2} C_{j,\pm p} & \text{if } k = \pm p 
\end{cases}, \quad -p \leq j, k \leq p.
\]

Using the array \( X \), we can write (18) as:

\[
\tilde{f}(\lambda, \theta) = \sum_{j=0}^{p} \sum_{k=-p}^{p} e^{ik \lambda} \left\{ (X_{j,k} + \overline{X}_{-j,k}) \cos(j\theta) + \frac{1}{i} (X_{j,k} - \overline{X}_{-j,k}) \sin(j\theta) \right\}
\]

\[
= \sum_{j=0}^{p} \sum_{k=-p}^{p} e^{ik \lambda} \begin{cases} 
((X_{j,k} + \overline{X}_{-j,k}) \cos(j\theta), & k \text{ even} \\
((X_{j,k} - \overline{X}_{-j,k}) \sin(j\theta), & k \text{ odd} 
\end{cases},
\]

from which we can obtain the coefficients \( g_{j,k} \) in (17).
The FSHT software takes bivariate Fourier coefficients \( g_j^k \) as input in an array organized as follows:

\[
\begin{pmatrix}
g_0^0 & g_0^{-1} & g_0^{-1} & \cdots & g_0^{-p} & g_0^p \\
g_1^0 & g_1^{-1} & g_1^{-1} & \cdots & g_1^{-p} & g_1^p \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
g_{p-1}^0 & g_{p-1}^{-1} & g_{p-1}^{-1} & \cdots & g_{p-1}^{-p} & g_{p-1}^p \\
g_p^0 & 0 & 0 & \cdots & g_p^{-p} & g_p^p
\end{pmatrix}
\]

The output of the software is the spherical harmonic coefficients of the data arranged in an array of the form

\[
F = \begin{pmatrix}
b_0^0 & b_1^{-1} & b_1^{-1} & b_2^{-2} & b_2^{-2} & \cdots & b_p^{-p} & b_p^p \\
b_1^0 & b_2^{-1} & b_2^{-1} & b_3^{-2} & b_3^{-2} & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
b_{p-2}^0 & b_{p-1}^{-1} & b_{p-1}^{-1} & b_p^{-2} & b_p^{-2} & \cdots & 0 & 0 \\
b_{p-1}^0 & b_p^{-1} & b_p^{-1} & 0 & 0 & \cdots & 0 & 0 \\
b_p^0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0
\end{pmatrix}
\]

The angular power spectrum \( 1 \) can then be computed from this array.

4 Numerical Results

In this section we present a few numerical tests comparing the angular power spectrum \( 1 \) computed by our new method HP2SPH to the spectrum computed by the HEALPix software. The first two tests compare the accuracy of the two methods by applying them to a function with a known power spectrum that is sampled at the HEALPix points. In the third test, we compare the angular power spectrum for a real CMBR data Map. In all cases, we use the default iteration value of 3 in the HEALPix software, as discussed in Section 2.2.

4.1 Numerical Test 1

We choose the function

\[
f(\lambda, \theta) = \sum_{j=1}^{3} c_j (2 - 2x(\lambda, \theta) \cdot x(\lambda_j, \theta_j))^{3/2},
\]

where \( x(\lambda, \theta) = [x(\lambda, \theta) \ y(\lambda, \theta) \ z(\lambda, \theta)] \) from \( 10 \) and the parameters, which we picked randomly, are given by

\[
\{c_1, c_2, c_3\} = \{5, -3, 8\},
\]

\[
\{\lambda_1, \lambda_2, \lambda_3\} = \{0.891498158152027, 2.650004294134628, 5.753735997130328\},
\]

\[
\{\theta_1, \theta_2, \theta_3\} = \{1.232217523107963, 2.059244524372349, 0.537798840821172\}.
\]

The function \( (2 - 2x(\lambda, \theta) \cdot x(\lambda_c, \theta_c))^{3/2} \) is a called a potential spline of order 3/2 centered at \( x(\lambda_c, \theta_c) \) and its spherical harmonic coefficients are given by \( 2 \)

\[
b_{p}^{m} = \frac{18\pi}{(\ell + 5/2)(\ell + 3/2)(\ell + 1/2)(\ell - 1/2)(\ell - 3/2)} Y_{\ell}^{m}(\lambda_c, \theta_c).
\]

We use these values to compute the exact spherical harmonic coefficients of \( f \) and hence its angular power spectrum.

Figure \( 6 \) illustrates how HP2SPH compares to the HEALPix algorithm for computing the angular power spectrum of \( 19 \). We see that both algorithms produce similar results for lower degrees \( \ell \), but that for degrees greater than approximately \( \ell = 50 \) the HEALPix results begin to diverge. This is especially important for the purposes of CMBR analysis, where the peaks of the angular power spectrum at high degrees give insight to various cosmological quantities. To compare the two methods further, we plot in Figure \( 7 \) the absolute error in the angular power spectrum for each degree \( \ell \) computed by the methods. This figure clearly shows improved performance of the HP2SPH method over the HEALPix method.
Figure 6: (Scaled) angular power spectrum of (19) in black against (a) the new HP2SPH method in blue and (b) the method in the HEALPix software in red. Here $N_{\text{side}} = 2^8$, which is $N = 786432$ total points.

Figure 7: Absolute error in the (scaled) angular power spectrum of (19) as a function of degree $\ell$ computed by the HEALPix software (red $\times$’s) and the HP2SPH method (blue $\circ$’s). Here $N_{\text{side}} = 2^8$, which is $N = 786432$ total points.

4.2 Numerical Test 2

In this test, we further explore the accuracy of the two methods, but now focus on how good they are at approximating the power spectrum of data with high frequencies, as occur in the CMBR data. We do this adding several spherical harmonic functions of high degree to the function (19) used in the previous test. The new test function takes the precise form

$$h(\lambda, \theta) = f(\lambda, \theta) + Y^{75}_{300}(\lambda, \theta) + Y^{100}_{425}(\lambda, \theta) + Y^{75}_{550}(\lambda, \theta) + Y^{50}_{600}(\lambda, \theta) + Y^{75}_{700}(\lambda, \theta) + Y^{50}_{750}(\lambda, \theta) + Y^{25}_{800}(\lambda, \theta) + Y^{50}_{850}(\lambda, \theta) + Y^{50}_{900}(\lambda, \theta) + Y^{25}_{950}(\lambda, \theta) + Y^{50}_{1050}(\lambda, \theta) + Y^{50}_{1100}(\lambda, \theta) + Y^{50}_{1150}(\lambda, \theta) + Y^{50}_{1250}(\lambda, \theta).$$

(20)

The power spectrum of this function is the same as (19), but with the value at each degree $\ell$ of appended spherical harmonics increased by $\frac{1}{2\ell+1}$.

Figure 8 displays the error in the angular power spectrum of the two method of $h$. The figure clearly shows
that our new method continues to provide better accuracy than the HEALPix method even in recovering the power spectrum for functions with high frequencies.

4.3 Numerical Test 3

Our final numerical test compares the two methods on the real CMBR map shown in Figure 1. Figure 9 shows the angular power spectrum for this map computed with the two methods. We see from the figure that the new HP2SPH method (in blue) produces a different spectrum from HEALPix method (red). While the shapes of the two spectra are similar for lower degrees, the amplitudes are different. For higher degrees, \( \ell \gtrsim 700 \), the location of the peaks in the spectra begin to differ. Additionally, the HEALPix method spectrum is much smoother than HP2SPH method for degrees \( \ell \gtrsim 100 \), which could be due to the least squares nature of the HEALPix method.

The two previous tests demonstrated that the HP2SPH method is more accurate than the HEALPix method, which implies that this method may be able to offer better information about the location of peaks in CMBR angular power spectrum.

5 Conclusions and Remarks

The numerical tests described in the previous section demonstrate the accuracy of the new HP2SPH method relative to the HEALPix approach for three test problems. This new method utilizes the FFT, NUFFT, and the FSHT to more accurately compute the angular power spectrum of CMBR data in near optimal computational complexity (\( \mathcal{O}(N \log N) \) complexity for \( N \) total nodes). For our next steps, we will work to optimize the implementation of the method, which is currently in Julia, to improve its actual run-time. This will include transcribing our code into a lower-level language like C; efforts in this direction are already underway for the FSHT [23]. In addition to this, we will include the ability to perform Fourier synthesis on a CMBR map, i.e. given an angular power spectrum, we will return the corresponding CMBR map values. For this purpose, our method has another advantage over HEALPix in that we will have the bivariate Fourier coefficients, which will simply make the synthesis an application of the FFT. Finally, we investigate further improvements to the accuracy of the method.
Figure 9: (Scale) angular power spectrum of the CMBR data map displayed in Figure 1(a) with \(N_{\text{side}} = 2^{11}\). The red dots show the (scaled) angular power spectrum computed from the HEALPix software and the blue dots correspond to the spectrum computed by the new HP2SPH method.

Acknowledgements

We are grateful to Ann Almgren for her comments and suggestions on the manuscript. The first author’s work was supported by the SMART Scholarship, which is funded by The Under Secretary of Defense-Research and Engineering, National Defense Education Program / BA-1, Basic Research. The second author’s work was supported by National Science Foundation grant CCF 1717556.

A Spherical Harmonic Conventions

We denote a scalar spherical harmonic of degree \(\ell \geq 0\) and order \(-\ell \leq m \leq \ell\) as \(Y_\ell^m(\lambda, \theta)\), where \(\lambda\) is the azimuth angle and \(\theta\) is the zenith angle. We define these functions as

\[
Y_\ell^m(\lambda, \theta) = \sqrt{\frac{2\ell + 1}{4\pi}} \sqrt{\frac{(\ell - m)!}{(\ell + m)!}} P_\ell^m(\cos \theta) e^{i m \lambda}, \quad m = 0, 1, \ldots, \ell,
\]  

(21)

where \(Y_\ell^m = (-1)^m Y_\ell^{-m}\) for \(m < 0\) and \(P_\ell^m(\cos \theta)\) are the associate Legendre functions. As eigenfunctions of the Laplace-Beltrami operator, spherical harmonics are the natural basis for square integrable functions on the sphere \(\mathbb{S}^2\). In other words, any \(L^2\)-integrable function \(f\) on the sphere can be uniquely represented as

\[
f(\lambda, \theta) = \sum_{\ell=0}^{\infty} \sum_{m=-\ell}^{\ell} b_\ell^m Y_\ell^m(\lambda, \theta),
\]

where the spherical harmonic coefficients, \(b_\ell^m\), are found using the usual \(L^2\)-inner product for scalar functions on the sphere:

\[
b_\ell^m = \langle f, Y_\ell^m \rangle = \int_0^{2\pi} \int_0^\pi f(\lambda, \theta) Y_\ell^m(\lambda, \theta) \sin \theta d\theta d\lambda.
\]  

(22)
References

[1] K. Atkinson and W. Han, Spherical Harmonics and Approximations on the Unit Sphere: An Introduction, Springer Berlin Heidelberg, 2012.

[2] B. J. C. Baxter and S. Hubbert, Radial basis functions for the sphere, in Recent progress in multivariate approximation (Witten-Bommerholz, 2000), vol. 137 of Internat. Ser. Numer. Math., Birkhäuser, Basel, 2001, pp. 33–47.

[3] R. Beatson and L. Greengard, A short course on fast multipole methods, Wavelets Multilevel Methods elliptic PDEs, 1 (1997), pp. 1–37.

[4] C. L. Bennett, D. Larson, J. L. Weiland, N. Jarosik, G. Hinshaw, N. Odegard, K. M. Smith, R. S. Hill, B. Gold, M. Halpern, E. Komatsu, M. R. Nolta, L. Page, D. N. Spergel, E. Wollack, J. Dunkley, A. Kogut, M. Limon, S. S. Meyer, G. S. Tucker, and E. L. Wright, Nine-year Wilkinson Microwave Anisotropy Probe (WMAP) Observations: Final Maps and Results, The Astrophysical Journal Supplement Series, 208 (2013), p. 20.

[5] M. Benzi, G. H. Golub, and J. Liesen, Numerical solution of saddle point problems, Acta Numerica, 14 (2005), pp. 1–137.

[6] J. Cooley and J. Tukey, An algorithm for the machine calculation of complex Fourier series, Math. Comp., 19 (1965), pp. 297–301.

[7] G. E. Fasshauer, Matrix Approximation Methods with MATLAB, World Scientific Publishers, Singapore, 2007.

[8] G. Golub and J. Lee, Matrix Computations, Johns Hopkins University Press, Baltimore, 1996.

[9] K. M. Gorski, B. D. Wandelt, E. Hivon, F. K. Hansen, and A. J. Banday, The HEALPix Primer. arXiv preprint astro-ph/9905275, 1999.

[10] K. M. Gorski, B. D. Wandelt, E. Hivon, F. K. Hansen, A. J. Banday, M. Reinecke, and M. Bartelmann, HEALPix – A framework for high resolution discretization, and fast analysis of data distributed on the sphere, ApJ., 622 (2005), pp. 759–771.

[11] L. Greengard and J. Lee, Accelerating the nonuniform fast Fourier transform, SIAM Rev., 46 (2004), pp. 443–454.

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

[13] G. Hinshaw, D. Larson, E. Komatsu, D. N. Spergel, C. L. Bennett, J. Dunkley, M. R. Nolta, M. Halpern, R. S. Hill, N. Odegard, L. Page, K. M. Smith, J. L. Weiland, B. Gold, N. Jarosik, A. Kogut, M. Limon, S. S. Meyer, G. S. Tucker, E. Wollack, and E. L. Wright, Nine-year Wilkinson Microwave Anisotropy Probe (WMAP) Observations: Cosmological Results, The Astrophysical Journal Supplement Series, 208 (2013), p. 19.

[14] W. Hu and S. Dodelson, Cosmic microwave background anisotropies, Ann. Rev. Astron. Astrophys., 40 (2002), pp. 171–216.

[15] J. Keiner and D. Potts, Fast evaluation of quadrature formulae on the sphere, Math. Comp., 77 (2008), pp. 397–419.

[16] P. E. Merilees, The pseudospectral approximation applied to the shallow water equations on a sphere, Atmosphere, 11 (1973), pp. 13–20.

[17] E. Michielssen and A. Boag, A multilevel matrix decomposition algorithm for analyzing scattering from large structures, IEEE Trans. Antennas and Propagation, 44 (1996), pp. 1086–1093.
[18] M. O’Neil, F. Woolfe, and V. Rokhlin, An algorithm for the rapid evaluation of special function transforms, Appl. Comput. Harmon. Anal., 28 (2010), pp. 203–226.

[19] C. C. Paige and M. A. Saunders, LSQR: An algorithm for sparse linear equations and sparse least squares, ACM Trans. Math. Softw., 8 (1982), pp. 43–71.

[20] P. Paykari and J.-L. Starck, CMB data analysis, Advances in Machine Learning and Data Mining for Astronomy, (2012), pp. 55–87.

[21] Planck Collaboration 2005, Planck: The scientific programme, ESA publication ESA-SCI(2005)/01.

[22] D. Ruiz-Antolin and A. Townsend, A nonuniform fast Fourier transform based on low rank approximation, SIAM J. Sci. Comput., 40 (2018), pp. A529–A547.

[23] R. M. Slevinsky. Personal communication.

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

[25] ———, Fast and backward stable transforms between spherical harmonic expansions and bivariate fourier series, in press at Appl. Comput. Harmon. Anal., (2017).

[26] A. Townsend, H. Wilber, and G. B. Wright, Computing with functions in spherical and polar geometries I. the sphere, SIAM Journal on Scientific Computing, 38 (2016), pp. C403–C425.

[27] M. White, Anisotropies in the CMB, Proceedings of Division of Particles and Fields of the American Physical Society, (1999).