Fast summation of radial functions on the sphere

Jens Keiner* Stefan Kunis† Daniel Potts‡

Radial functions are a powerful tool in many areas of multi-dimensional approximation, especially when dealing with scattered data. We present a fast approximate algorithm for the evaluation of linear combinations of radial functions on the sphere $\mathbb{S}^2$. The approach is based on a particular rank approximation of the corresponding Gram matrix and fast algorithms for spherical Fourier transforms. The proposed method takes $\mathcal{O}(L)$ arithmetic operations for L arbitrarily distributed nodes on the sphere. In contrast to other methods, we do not require the nodes to be sorted or pre-processed in any way, thus the pre-computation effort only depends on the particular radial function and the desired accuracy. We establish explicit error bounds for a range of radial functions and provide numerical examples covering approximation quality, speed measurements, and a comparison of our particular matrix approximation with a truncated singular value decomposition.

2000 *AMS Subject Classification.* 65T50, 65F30, 42C10, 33C55, 15A23.
Key words and phrases. Fast discrete summation, Radial basis functions, Zonal functions, FFT, NFFT

1 Introduction

In radial basis function methods in $\mathbb{R}^3$ one approximates functions from $\mathbb{R}^3 \to \mathbb{R}$ by linear combinations of radial symmetric translates $\Phi(\mathbf{y} - \cdot\,)$, $\mathbf{y} \in \mathbb{R}^3$, of a single function $\phi : \mathbb{R}^+ \to \mathbb{R}$, i.e., $\Phi(\mathbf{y} - \cdot\,) : \mathbb{R}^3 \to \mathbb{R}$, $\mathbf{x} \mapsto \phi(\|\mathbf{y} - \mathbf{x}\|_2)$.

We restrict ourselves to the sphere $\mathbb{S}^2 := \{\mathbf{x} \in \mathbb{R}^3 : \|\mathbf{x}\|_2 = 1\} \subset \mathbb{R}^3$ and denote its elements by $\boldsymbol{\xi}, \boldsymbol{\eta}, \ldots$. Note that $\|\boldsymbol{\eta} - \boldsymbol{\xi}\|_2^2 = 2 - 2\boldsymbol{\eta} \cdot \boldsymbol{\xi}$ and that it is convenient to use the *geodetic distance* $\arccos(\boldsymbol{\eta} \cdot \boldsymbol{\xi})$. In particular, a point $\boldsymbol{\xi} \in \mathbb{S}^2$ might in *spherical coordinates* be identified with a vector $(\vartheta, \varphi) \in [0, \pi] \times [-\pi, \pi)$ with $\boldsymbol{\xi} = (\sin\vartheta\cos\varphi, \sin\vartheta\sin\varphi, \cos\vartheta)^\top$ and has geodetic distance $\arccos(\mathbf{e}_3 \cdot \boldsymbol{\xi}) = \vartheta$ to the north pole $\mathbf{e}_3 = (0, 0, 1)^\top$. This gives rise to the spherical counterpart of radial functions, namely the *zonal functions* which depend solely on the geodetic distance of two points on the sphere.

*keiner@math.uni-luebeck.de, University of Lübeck, Institute of Mathematics, 23560 Lübeck, Germany
†kunis@math.uni-luebeck.de, University of Lübeck, Institute of Mathematics, 23560 Lübeck, Germany
‡potts@mathematik.tu-chemnitz.de, Chemnitz Universitiy of Technology, Department of Mathematics, 09107 Chemnitz, Germany

More formally, let a function $K \in \mathrm{L}^1([-1,1])$ be given and define for fixed $\boldsymbol{\eta} \in \mathbb{S}^2$ the $\boldsymbol{\eta}$-zonal function

$$K(\boldsymbol{\eta} \cdot) : \mathbb{S}^2 \to \mathbb{R},\ \boldsymbol{\xi} \mapsto K(\boldsymbol{\eta} \cdot \boldsymbol{\xi}).$$

Of course, every radial function corresponds to a zonal one by means of $K(\boldsymbol{\eta}\cdot) = \phi(\|\boldsymbol{\eta} - \cdot\|_2)$.

The description of the problem reads as follows: Given $D, L \in \mathbb{N}$, a set of arbitrary *source nodes* $\mathcal{Y} := \{\boldsymbol{\eta}_l \in \mathbb{S}^2 : l = 0, \dots, L-1\}$, and a vector of real coefficients $\mathbf{b} := (b_l)_{l=0}^{L-1}$, evaluate the sum

$$f(\boldsymbol{\xi}) := \sum_{l=0}^{L-1} b_l K(\boldsymbol{\eta}_l \cdot \boldsymbol{\xi}) \tag{1.1}$$

on a set of arbitrary *target nodes* $\mathcal{X} := \{\boldsymbol{\xi}_d \in \mathbb{S}^2 : d = 0, \dots, D-1\}$.

The naive approach for evaluating (1.1) leads to an $\mathcal{O}(LD)$ algorithm if we assume that the zonal function $K(\boldsymbol{\eta} \cdot)$ can be evaluated in constant time or that all values $K(\boldsymbol{\eta}_l \cdot \boldsymbol{\xi}_d)$ can be stored in advance. For large L and D the computational effort becomes quickly unaffordable. The *panel clustering method* on the sphere in [5] reduces the computational effort for evaluating (1.1) based on the traditional method of dividing the evaluation into a near- and a far-field: For every zonal function $K(\boldsymbol{\eta}_l \cdot)$, the near-field contribution is calculated exactly whereas the contribution of the far-field is approximated coarsely.

In contrast, we present a simple structured approximate algorithm with arithmetic complexity $\mathcal{O}(D + L)$ which can be easily adapted to different kernels K and is based on the *nonequispaced fast spherical Fourier transform* (*NFSFT*). The NFSFT [12] is a derivative of the stabilised Driscoll-Healy algorithm [3, 13, 8] combined with the fast Fourier transform for nonequispaced nodes (NFFT) as described for example in [14]. Recently, the corresponding adjoint algorithm, necessary within our current approach, was derived and implemented in [10].

The basic idea in what follows is the use of a truncated series expansion of $K(\boldsymbol{\eta} \cdot \boldsymbol{\xi})$ into spherical harmonics which yields a separation of the nodes and hence allows for the construction of the fast algorithm. We prove error estimates to obtain clues about the choice of the involved parameters and present numerical examples.

The remainder of this paper is organised as follows: To keep the text self-contained, we summarise basic definitions and properties of Fourier expansions of zonal functions in Section 2. In Section 3, we propose our fast method for the evaluation of (1.1), whereas Section 4 establishes error estimates for a variety of functions K. Finally, Section 5 presents numerical results.

2 Prerequisites

The *Legendre polynomials* $P_k : [-1,1] \to \mathbb{R}$, $k \in \mathbb{N}_0$, as classical orthogonal polynomials are given by their *Rodrigues formula*

$$P_k(x) := \frac{1}{2^k k!} \frac{\mathrm{d}^k}{\mathrm{d}x^k} \left(x^2 - 1\right)^k.$$

One verifies

$$P_k(\pm 1) = (\pm 1)^k, \quad \max_{x \in [-1,1]} |P_k(x)| = 1, \text{ and } \quad |P_k(\cos\vartheta)| \le \sqrt{\frac{2}{\pi k \sin\vartheta}}, \tag{2.1}$$

where the last inequality holds for $\vartheta \in (0, \pi)$ and $k \geq 1$, see e.g. [1, pp. 773]. Furthermore, two recurrence relations are given by

$$(k+1)\,P_{k+1}(x) = (2k+1)\,xP_k(x) - kP_{k-1}(x) \tag{2.2}$$

and

$$(2k+1)\,P_k(x) = P'_{k+1}(x) - P'_{k-1}(x). \tag{2.3}$$

The more general *associated Legendre functions* P_k^n, $k \in \mathbb{N}_0$, $n \leq k$, are defined by

$$P_k^n(x) := \left(\frac{(k-n)!}{(k+n)!}\right)^{1/2} (1-x^2)^{n/2} \frac{\mathrm{d}^n}{\mathrm{d}x^n} P_k(x).$$

Let the space of real-valued continuous functions on the sphere be decomposed into the direct sum of spaces of *spherical harmonics*. We denote by $\{Y_k^n\}_{k\in\mathbb{N}_0;n=-k,\ldots,k}$ the standard L^2-orthonormal basis of spherical harmonics given by

$$Y_k^n(\boldsymbol{\xi}) = Y_k^n(\vartheta,\varphi) := \sqrt{\frac{2k+1}{4\pi}} P_k^{|n|}(\cos\vartheta)\mathrm{e}^{\mathrm{i}n\varphi}.$$

In particular, these functions fulfil the addition theorem, cf. [4, p. 37],

$$\sum_{n=-k}^{k} \overline{Y_k^n(\boldsymbol{\eta})} Y_k^n(\boldsymbol{\xi}) = \frac{2k+1}{4\pi} P_k(\boldsymbol{\eta}\cdot\boldsymbol{\xi})\,.$$

Thus, the orthogonal expansion of the zonal function $K(\boldsymbol{\eta}\cdot)$ in terms of the basis functions Y_k^n is given by

$$K(\boldsymbol{\eta}\cdot\boldsymbol{\xi}) = \sum_{k=0}^{\infty} \sum_{n=-k}^{k} K^\wedge(k) \overline{Y_k^n(\boldsymbol{\eta})} Y_k^n(\boldsymbol{\xi}) \tag{2.4}$$

where the *Fourier-Legendre coefficients* $K^\wedge(k)$ of K are defined by

$$K^\wedge(k) := 2\pi \int_{-1}^{1} K(x) P_k(x) \mathrm{d}x\,. \tag{2.5}$$

From now on, we shall require that the series (2.4) converges absolutely, i.e., $\sum_{k=0}^{\infty}(2k+1)|K^\wedge(k)| < \infty$. For practical calculations, a finite number of Fourier-Legendre coefficients $K^\wedge(k)$ have to be stored in advance in a pre-computation step. For some kernels these coefficients are known explicitly whereas for others, one might use (2.5) and the recurrence relations (2.2) and (2.3) to derive recursion formulae for the coefficients $K^\wedge(k)$. Further pre-computation methods, like discretising (2.5), and the error they introduce are beyond the scope of the present paper.

3 Fast Summation

Basically, the representation in (2.4) allows the construction of fast algorithms, owing to the separation of the nodes $\boldsymbol{\eta}$ and $\boldsymbol{\xi}$. We simply propose to truncate the series (2.4) at a fixed *cut-off degree* $M \in \mathbb{N}_0$, i.e.,

$$K(\boldsymbol{\eta}\cdot\boldsymbol{\xi}) \approx K_M(\boldsymbol{\eta}\cdot\boldsymbol{\xi}) := \sum_{k=0}^{M} \sum_{n=-k}^{k} K^\wedge(k) \overline{Y_k^n(\boldsymbol{\eta})} Y_k^n(\boldsymbol{\xi}). \tag{3.1}$$

Substituting (3.1) into (1.1) and interchanging the order of summation we finally obtain the approximation

$$f(\boldsymbol{\xi}) \approx f_M(\boldsymbol{\xi}) := \sum_{k=0}^{M} \sum_{n=-k}^{k} K^\wedge(k) \left(\sum_{l=0}^{L-1} b_l \overline{Y_k^n(\boldsymbol{\eta}_l)} \right) Y_k^n(\boldsymbol{\xi}), \tag{3.2}$$

to be evaluated at the D target nodes $\boldsymbol{\xi}_d$.

Our algorithm now works as follows: The expression in the inner brackets can be evaluated by an adjoint nonequispaced fast spherical Fourier transform (adjoint NFSFT) with $\mathcal{O}(L + M^2 \log^2 M)$ arithmetic operations involving the L source nodes $\boldsymbol{\eta}_l$. This is followed by $(M+1)^2$ multiplications with the precomputed Fourier-Legendre coefficients $K^\wedge(k)$, and completed by a NFSFT to evaluate the outer sum at the D target nodes $\boldsymbol{\xi}_d$ with $\mathcal{O}\left(D + M^2 \log^2 M\right)$ arithmetic operations.

Remark 3.1. In matrix-vector notation the original problem (1.1) reads $\mathbf{f} = \mathbf{K}\,\mathbf{b}$, where

$$\begin{aligned} \mathbf{f} &:= \left(f\left(\boldsymbol{\xi}_d\right)\right)_{d=0,\ldots,D-1} \in \mathbb{R}^D, \\ \mathbf{K} &:= \left(K\left(\boldsymbol{\eta}_l \cdot \boldsymbol{\xi}_d\right)\right)_{d=0,\ldots,D-1;l=0,\ldots,L-1} \in \mathbb{R}^{D\times L}. \end{aligned}$$

Our approach is a particular rank $(M+1)^2$ approximation of the matrix $\mathbf{K}$ and takes the form $\mathbf{f}_M = \mathbf{Y}_\mathcal{X}\,\hat{\mathbf{W}}\,\mathbf{Y}_\mathcal{Y}^\intercal\,\mathbf{b}$ with

$$\begin{aligned} \mathbf{f}_M &:= \left(f_M\left(\boldsymbol{\xi}_d\right)\right)_{d=0,\ldots,D-1} \in \mathbb{R}^D, \\ \mathbf{Y}_\mathcal{X} &:= \left(Y_k^n\left(\boldsymbol{\xi}_d\right)\right)_{d=0,\ldots,D-1;k=0,\ldots,M,\,n=-k,\ldots,k} \in \mathbb{C}^{D\times(M+1)^2}, \\ \hat{\mathbf{W}} &:= \operatorname{diag}\left(\hat{\mathbf{w}}\right),\ \hat{\mathbf{w}} := \left(\hat{w}_k^n\right)_{k=0,\ldots,M,\,n=-k,\ldots,k} \in \mathbb{R}^{(M+1)^2},\ \hat{w}_k^n := K^\wedge\left(k\right), \\ \mathbf{Y}_\mathcal{Y} &:= \left(Y_k^n\left(\boldsymbol{\eta}_l\right)\right)_{l=0,\ldots,L-1;k=0,\ldots,M,\,n=-k,\ldots,k} \in \mathbb{C}^{L\times(M+1)^2}. \end{aligned}$$

□

The proposed method is summarised in Algorithm 1.

Remark 3.2. Replacing the NFSFT algorithms by their slow versions *nonequispaced discrete spherical Fourier transform (NDSFT)* and *adjoint NDSFT*, i.e. direct algorithms which need $\mathcal{O}\left(LM^2\right)$ and $\mathcal{O}\left(DM^2\right)$ arithmetic operations for the multiplications with the matrices $\mathbf{Y}_\mathcal{Y}^\intercal$ and $\mathbf{Y}_\mathcal{X}$, respectively, yields an $\mathcal{O}\left(L+D\right)$ algorithm, too. Nevertheless, the fast algorithms are the key for applications to large data sets since they decouple the cut-off degree M from the numbers of nodes L and D. □

4 Error estimates and examples

Besides the well-known errors appearing in the NFFT computations, see [14], our algorithm causes the following systematic error:

Lemma 4.1. *The proposed approximation f_M obeys for $\mathbf{b} \in \mathbb{C}^L \setminus \{\mathbf{0}\}$ the relative error estimate*

$$\frac{\|f - f_M\|_\infty}{\|\mathbf{b}\|_1} \le \sum_{k>M} \frac{2k+1}{4\pi} \left|K^\wedge(k)\right|.$$

Algorithm 1 Fast summation of zonal functions

Input: $L \in \mathbb{N}$, coefficients $\mathbf{b} \in \mathbb{R}^L$, source nodes $\boldsymbol{\eta}_l \in \mathbb{S}^2$ for $l = 0, \dots, L-1$,
$D \in \mathbb{N}$, target nodes $\boldsymbol{\xi}_d \in \mathbb{S}^2$ for $d = 0, \dots, D-1$,
$M \in \mathbb{N}_0$, Fourier-Legendre coefficients $K^\wedge(k)$ for $k = 0, \dots, M$.

Compute $\tilde{\mathbf{a}} := \mathbf{Y}_{\mathcal{Y}}^{\mathsf{H}} \mathbf{b}$ by an adjoint NFSFT.

Evaluate $\mathbf{a} := \hat{\mathbf{W}}\tilde{\mathbf{a}}$.

Compute $\mathbf{f}_M := \mathbf{Y}_{\mathcal{X}} \mathbf{a}$ by a NFSFT.

Output: $\mathbf{f}_M$ approximating $\mathbf{f} = \mathbf{K}\,\mathbf{b}$.
Complexity: $\mathcal{O}\left(L + D + M^2 \log^2 M\right)$.

Proof. The assertion is obtained by

$$\|f - f_M\|_\infty \leq \sum_{l=0}^{L-1} |b_l| \max_{\boldsymbol{\xi} \in \mathbb{S}^2} \left| \sum_{k>M} \frac{2k+1}{4\pi} K^\wedge(k) P_k\left(\boldsymbol{\eta}_l \cdot \boldsymbol{\xi}\right) \right|$$

and the fact $|P_k(x)| \leq 1$. ∎

Thus, the error decouples into a part solely dependent on the coefficients b_l and a decay condition for the Fourier-Legendre coefficients $K^\wedge(k)$ of the zonal function $K(\boldsymbol{\eta} \cdot)$. In particular, the choice of the cut-off degree M only depends on the desired accuracy and the individual zonal function $K(\boldsymbol{\eta} \cdot)$, but neither on the numbers L and D of source nodes $\boldsymbol{\eta}_l$ and target nodes $\boldsymbol{\xi}_d$, nor on their distribution.

General results for the decay of Fourier-Legendre coefficients are given in [15, 9, 2]. For a fixed set of parameters, the coefficients $K^\wedge(k)$ obey a certain decay rate with respect to k. In contrast, we conversely use the decay of the Fourier-Legendre coefficients with respect to k and the parameters of the zonal function $K(\boldsymbol{\eta} \cdot)$ to determine the cut-off degree M in (3.1).

We consider a 'representative' range of zonal functions. A closed form expression allows for direct evaluation and numerical verification. Depending on the particular zonal function $K(\boldsymbol{\eta} \cdot)$, we use different methods for pre-computing the Fourier-Legendre coefficients $K^\wedge(k)$.

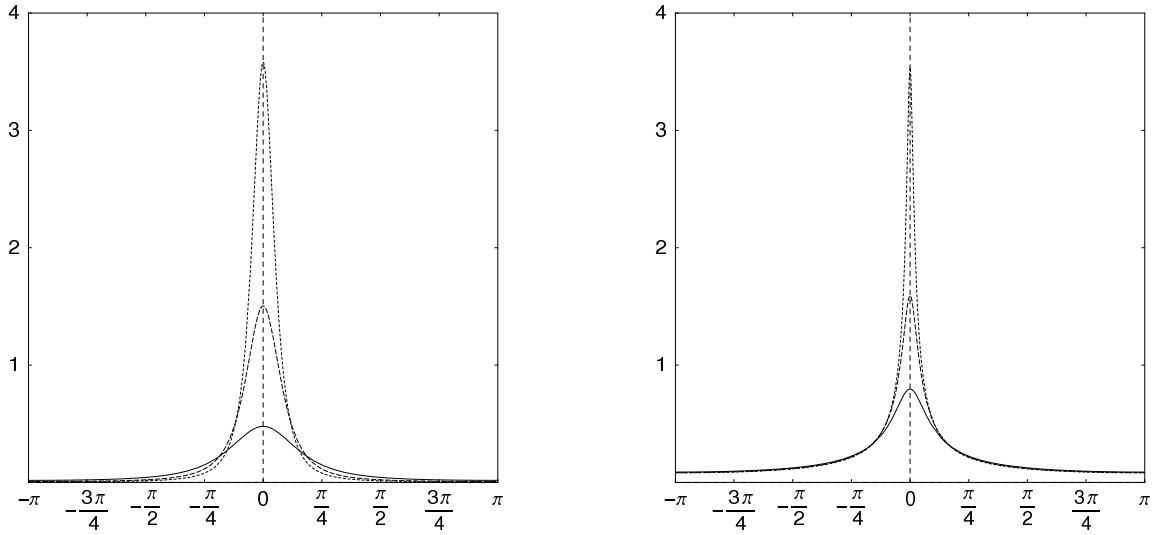
Note furthermore that the spherical convolution lemma [4, pp. 63] yields a simple possibility to trade localisation of a zonal function in spatial domain against the decay of its Fourier-Legendre coefficients, see [15] for details.

4.1 Poisson and Singularity kernel

Definition 4.2. Let $h \in (0, 1)$, then the

1. *Poisson kernel* $Q_h : [-1, 1] \to \mathbb{R}$ is given by

$$Q_h(x) := \frac{1}{4\pi} \frac{1 - h^2}{(1 - 2hx + h^2)^{3/2}},$$



(a) $Q_h(\cos\vartheta)$, $h = 0.5, 0.7, 0.8$.

(b) $S_h(\cos\vartheta)$, $h = 0.8, 0.9, 0.955$.

Figure 4.1: The kernels $Q_h(\cos\vartheta)$ and $S_h(\cos\vartheta)$ for different values of h.

2. and the *singularity kernel* $S_h : [-1, 1] \to \mathbb{R}$ is given by

$$S_h(x) := \frac{1}{2\pi} \frac{1}{(1 - 2hx + h^2)^{1/2}}.$$

□

The parameter h allows for controlling the localisation of $Q_h(\boldsymbol{\eta} \cdot)$, cf. Figure 4.1 (left), and $S_h(\boldsymbol{\eta} \cdot)$, cf. Figure 4.1 (right), around $\boldsymbol{\eta} \in \mathbb{S}^2$, respectively. The Poisson kernel $Q_h(\boldsymbol{\eta} \cdot)$ is a normalised positive function with $\|Q_h(\boldsymbol{\eta} \cdot)\|_{\mathrm{L}^1(\mathbb{S}^2)} = 1$. Further properties of the Poisson kernel $Q_h(\boldsymbol{\eta} \cdot)$ and the singularity kernel $S_h(\boldsymbol{\eta} \cdot)$ with respect to localisation and smoothness are derived in [4, pp. 112].

For both kernels the Fourier-Legendre coefficients $K^\wedge(k)$ are explicitly known, such that we simply state the following lemma:

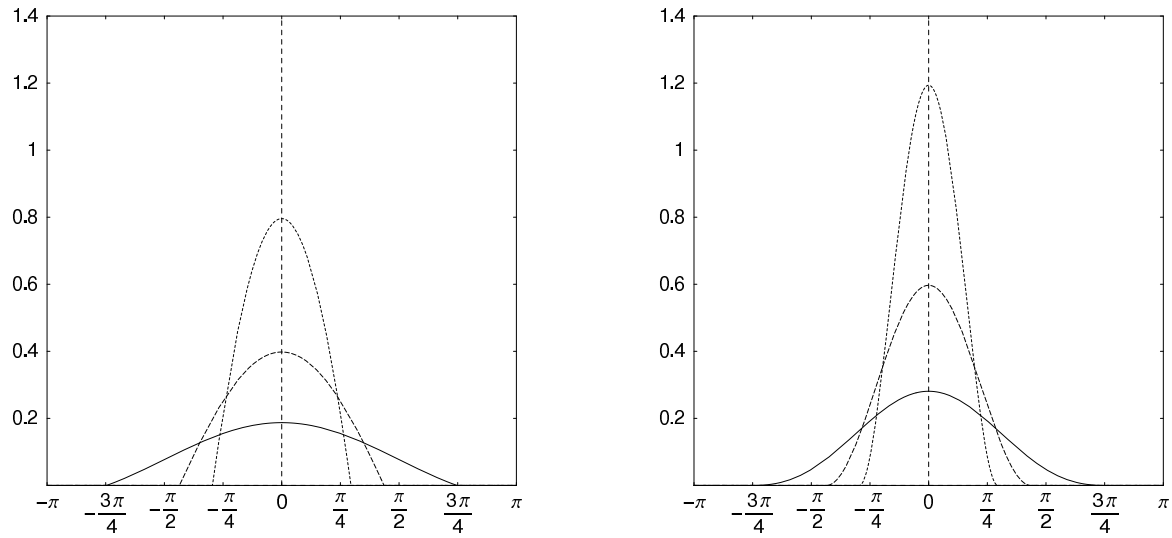
Lemma 4.3.

1. *Using the Poisson kernel $K = Q_h$, cf. Definition 4.2, in our summation algorithm yields a relative error of*

$$\frac{\|f - f_M\|_\infty}{\|\mathbf{b}\|_1} \leq \frac{h^{M+1}}{4\pi} \left(\frac{2M+1}{1-h} + \frac{2}{(1-h)^2} \right). \tag{4.1}$$

2. *Using the singularity kernel $K = S_h$, cf. Definition 4.2, in our summation algorithm yields a relative error of*

$$\frac{\|f - f_M\|_\infty}{\|\mathbf{b}\|_1} \leq \frac{h^{M+1}}{4\pi} \left(\frac{2M+1}{2\,(1-h)} + \frac{4M}{(1-h)^2} + \frac{4}{(1-h)^3} \right). \tag{4.2}$$



(a) $\lambda = 1.0$ (b) $\lambda = 2.0$

Figure 4.2: The locally supported kernel $L_{h,\lambda}(\cos\vartheta)$ for $h = -0.7, 0.2, 0.7$ and different values of λ.

Proof. The Fourier-Legendre coefficients are given by $Q_h^\wedge(k) = h^k$ and $S_h^\wedge(k) = \frac{2}{2k+1}h^k$, respectively, cf. [4, pp. 107]. Using Lemma 4.1 yields the assertions. ■

Simply put, our scheme achieves (almost) accuracy ε for $M \geq \log\varepsilon \,/\, \log h$.

4.2 Locally supported kernel

Definition 4.4. Let $h \in (-1,1)$ and $\lambda \in \mathbb{N}_0$. The locally supported kernel $L_{h,\lambda} : [-1,1] \to \mathbb{R}$, considered in [15], is defined by

$$L_{h,\lambda}(x) := \begin{cases} 0 & \text{if} \quad -1 \leq x \leq h, \\ \frac{\lambda+1}{2\pi(1-h)^{\lambda+1}}\,(x-h)^\lambda & \text{if} \quad h < x \leq 1. \end{cases}$$

□

Figure 4.2 shows the function $L_{h,\lambda}$ for different values h and λ. While the parameter h again controls the localisation in spatial domain, the parameter λ corresponds to the smoothness of $L_{h,\lambda}$ in the endpoint of the support h. We have the following lemma:

Lemma 4.5. *For the locally supported kernel $L_{h,\lambda}$ holds:*

1. *The Fourier-Legendre coefficients $L_{h,\lambda}^\wedge(k)$ can be computed recursively by*

$$L_{h,\lambda}^\wedge(k+1) = \frac{(2k+1)\,h}{k+\lambda+2}L_{h,\lambda}^\wedge(k) - \frac{k-\lambda-1}{k+\lambda+2}L_{h,\lambda}^\wedge(k-1)$$

for $k \in \mathbb{N}$ where $L_{h,\lambda}^\wedge(0) = 1$ and $L_{h,\lambda}^\wedge(1) = \frac{\lambda+1+h}{\lambda+2}$.

2. *We obtain for* $\lambda \in \mathbb{N}_0$ *and* $k > \lambda + 1$ *the decay rate*

$$\left|L_{h,\lambda}^{\wedge}(k)\right| \leq \frac{2\sqrt{2}}{(2k+1)\sqrt{\pi}} \frac{(\lambda+1)^2}{(1-h)^{2\lambda+1}\sqrt[4]{1-|h|}} \frac{1}{(k-\lambda)^{\lambda+\frac{1}{2}}}.$$

3. *Thus, the relative error of our summation algorithm with this kernel* $K = L_{h,\lambda}$ *is bounded for* $\lambda \in \mathbb{N}$ *and* $M > \lambda$ *by*

$$\frac{\|f - f_M\|_\infty}{\|\mathbf{b}\|_1} \leq \frac{1}{\pi\sqrt{2\pi}} \frac{(\lambda+1)^2}{\lambda - \frac{1}{2}} \frac{1}{(1-h)^{2\lambda+1}\sqrt[4]{1-|h|}} \frac{1}{(M-\lambda)^{\lambda-\frac{1}{2}}}. \tag{4.3}$$

Proof.

1. We apply in (2.5) the recurrence relations (2.2) and (2.3).

2. For $\lambda = 0$, we have

$$\left|L_{h,\lambda}^{\wedge}(k)\right| = \left|2\pi \int_h^1 \frac{\lambda+1}{2\pi(1-h)^{\lambda+1}} (x-h)^\lambda P_k(x)\,\mathrm{d}x\right| \leq \frac{\lambda+1}{(1-h)^{\lambda+1}} \left|\int_h^1 P_k(x)\,\mathrm{d}x\right|.$$

Using (2.3) and (2.1), we obtain

$$\begin{aligned}\left|\int_h^1 P_k(x)\,\mathrm{d}x\right| &= \frac{1}{2k+1}\left|P_{k-1}(h) - P_{k+1}(h)\right| \\ &\leq \frac{1}{2k+1}\sqrt{\frac{2}{\pi \sin\arccos h}}\left(\frac{1}{\sqrt{k-1}} + \frac{1}{\sqrt{k+1}}\right),\end{aligned}$$

and with $\sin\arccos h = \sqrt{1-h^2} \geq \sqrt{1-|h|}$ and $\frac{1}{\sqrt{k-1}} + \frac{1}{\sqrt{k+1}} \leq \frac{2k+1}{k\sqrt{k}}$ finally

$$\left|L_{h,0}^{\wedge}(k)\right| \leq \frac{\sqrt{2}}{\sqrt{\pi}} \frac{\lambda+1}{(1-h)^{\lambda+1}\sqrt[4]{1-|h|}} \frac{1}{k\sqrt{k}}. \tag{4.4}$$

Furthermore, for $\lambda \in \mathbb{N}$, we obtain by applying (2.3) that

$$\left|L_{h,\lambda}^{\wedge}(k)\right| \leq \frac{2}{2k+1} \frac{\lambda+1}{\lambda(1-h)} \max_{|k-k'|\leq 1} \left|L_{h,\lambda-1}^{\wedge}(k')\right|.$$

Iterate this argument and estimate $\frac{2}{2k'+1} \leq \frac{1}{k-\lambda}$ yields

$$\left|L_{h,\lambda}^{\wedge}(k)\right| \leq \frac{2}{2k+1} \frac{\lambda+1}{(1-h)^\lambda} \frac{1}{(k-\lambda)^{\lambda-1}} \max_{|k-k'|\leq \lambda} \left|L_{h,0}^{\wedge}(k')\right|.$$

We finally use (4.4) for $k' = k - \lambda$ to obtain the assertion.

3. We combine 2. with Lemma 4.1.

■

Simply put, our scheme achieves accuracy of order ε for $M \geq \lambda + (1-h)^{-2}\varepsilon^{-1/\lambda}$.

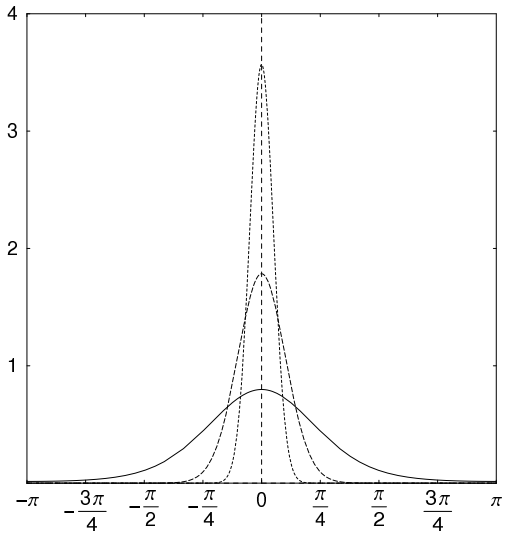


Figure 4.3: The L^2-normalised spherical Gaussian kernel $\gamma(\sigma)G_\sigma(\cos\vartheta)$ with $\gamma(\sigma) := \left(\frac{2\sigma}{\pi}\right)^{1/2}\left(1-\mathrm{e}^{-8\sigma}\right)^{-1/2}$ for $\sigma = 1, 5, 20$.

4.3 Spherical Gaussian kernel

The spherical analogue to the well-known *Gaussian kernel* $\mathrm{e}^{-\sigma x^2}$ is the *spherical Gaussian kernel*, see for example [9]. Figure 4.3 shows the spherical Gaussian kernel G_σ for different values σ.

Definition 4.6. For $\sigma > 0$, the spherical Gaussian kernel $G_\sigma : [-1,1] \to \mathbb{R}$ is given by

$$G_\sigma(x) := \mathrm{e}^{2\sigma x - 2\sigma}.$$

□

Lemma 4.7. *For the spherical Gaussian G_σ holds:*

1. *The Fourier-Legendre coefficients $G_\sigma^\wedge(k)$ are given by*

$$G_\sigma^\wedge(k) = 2\sigma^{-\frac{1}{2}}\mathrm{e}^{-2\sigma}\pi^{\frac{3}{2}} I_{k+\frac{1}{2}}(2\sigma) \tag{4.5}$$

   where $I_{k+\frac{1}{2}}$ denotes the modified Bessel function of first kind, see [1, pp. 437].

2. *Thus, the relative error of our summation algorithm with the spherical Gaussian kernel $K = G_\sigma$ is bounded by*

$$\frac{\|f - f_M\|_\infty}{\|\mathbf{b}\|_1} \leq \frac{\sqrt{\pi\sigma}\,(\mathrm{e}^\sigma - 1)\,\sigma^{M-\frac{1}{2}}}{\Gamma\left(M+\frac{1}{2}\right)}. \tag{4.6}$$

Proof.

1. The relation (2.3) and integration by parts yields the difference equation

$$2\sigma G_\sigma^\wedge(k-1) - 2\sigma G_\sigma^\wedge(k+1) = (2k+1)\, G_\sigma^\wedge(k) \tag{4.7}$$

for $k \in \mathbb{N}$ where $G_\sigma^\wedge(0) = 4\pi\sigma^{-1}\mathrm{e}^{-2\sigma}\sinh\sigma\cosh\sigma$ and $G_\sigma^\wedge(1) = \pi\sigma^{-2}\mathrm{e}^{-2\sigma}(2\sigma\cosh 2\sigma + \sinh\sigma)$. The assertion follows by the definition of the modified Bessel function, cf. [1, pp. 437].

2. Using 1. and

$$\begin{aligned}\frac{2k+1}{4\pi}\left|G_\sigma^\wedge(k)\right| &= \frac{\mathrm{e}^{-2\sigma}\left(k+\frac{1}{2}\right)\sigma^k}{\Gamma(k+1)}\int_{-1}^{1}\mathrm{e}^{2\sigma x}\left(1-x^2\right)^k\mathrm{d}x \\ &\leq \frac{\left(k+\frac{1}{2}\right)\sigma^k}{\Gamma(k+1)}\int_{-1}^{1}\left(1-x^2\right)^k\mathrm{d}x \\ &= \frac{\sqrt{\pi}\sigma^k}{\Gamma\left(k+\frac{1}{2}\right)}\end{aligned}$$

in conjunction with

$$\sum_{k>M}\frac{\sqrt{\pi}\sigma^k}{\Gamma\left(k+\frac{1}{2}\right)} = \frac{\sqrt{\pi\sigma}\mathrm{e}^\sigma}{\Gamma\left(M+\frac{1}{2}\right)}\int_0^\sigma t^{M-\frac{1}{2}}\mathrm{e}^{-t}\mathrm{d}t \leq \frac{\sqrt{\pi\sigma}\left(\mathrm{e}^\sigma-1\right)\sigma^{M-\frac{1}{2}}}{\Gamma\left(M+\frac{1}{2}\right)}$$

yields the assertion.

■

Note that using the difference equation (4.7) in a forward recursion is numerically unstable and we therefore use routines for evaluating Bessel functions provided by the *GNU scientific library (GSL)* [7] in the pre-computation of the Fourier-Legendre coefficients $G^\wedge(k)$ by means of (4.5).

4.4 Matrix approximation

We conclude this section with the following corollary on the approximation of a square matrix $\mathbf{K}$, cf. Remark 3.1, with respect to the matrix p-norm.

Corollary 4.8. *Let $L \in \mathbb{N}$, a set of arbitrary source nodes $\mathcal{Y} = \{\boldsymbol{\eta}_l \in \mathbb{S}^2 : \ l = 0, \ldots, L-1\}$, and a set of arbitrary target nodes $\mathcal{X} := \{\boldsymbol{\xi}_l \in \mathbb{S}^2 : \ l = 0, \ldots, L-1\}$ be given. Then, the proposed approximation $\mathbf{Y}_\mathcal{X}\,\hat{\mathbf{W}}\,\mathbf{Y}_\mathcal{Y}{}^\mathsf{H}$ to the matrix $\mathbf{K} \in \mathbb{R}^{L\times L}$, cf. Remark 3.1, fulfils*

$$\left\|\mathbf{K} - \mathbf{Y}_\mathcal{X}\,\hat{\mathbf{W}}\,\mathbf{Y}_\mathcal{Y}{}^\mathsf{H}\right\|_p \leq L\sum_{k>M}\frac{2k+1}{4\pi}\left|K^\wedge(k)\right|$$

for $1 \leq p \leq \infty$.

Proof. Using Hölder's inequality, we obtain

$$\left\|\mathbf{K} - \mathbf{Y}_\mathcal{X}\,\hat{\mathbf{W}}\,\mathbf{Y}_\mathcal{Y}{}^\mathsf{H}\right\|_p = \max_{\mathbf{b}\in\mathbb{R}^L}\frac{\|\mathbf{f}-\mathbf{f}_M\|_p}{\|\mathbf{b}\|_p} \leq \max_{\mathbf{b}\in\mathbb{R}^L} L^{\frac{1}{p}}L^{1-\frac{1}{p}}\frac{\|\mathbf{f}-\mathbf{f}_M\|_\infty}{\|\mathbf{b}\|_1}\,.$$

The assertion follows by $\|\mathbf{f}-\mathbf{f}_M\|_\infty \leq \|f-f_M\|_\infty$ and the Lemma 4.1. ■

Using for example the Poisson kernel with parameter h, we achieve an approximation of the square matrix $\mathbf{K} \in \mathbb{R}^{L\times L}$ up to an prescribed accuracy ε by choosing the cut-off degree

$$M \geq \frac{|\log \varepsilon| + \log L}{|\log h|}.$$

5 Numerical results

We present numerical examples in order to demonstrate the performance of our approach. All algorithms were implemented in C and tested on an AMD Athlon$^{\text{TM}}$XP 2700+ with 2GB main memory, SuSE-Linux (kernel 2.4.20-4GB-athlon, gcc 3.3) using double precision arithmetic. Moreover, we have used the libraries FFTW 3.0.1 [6], NFFT 2 [11], and a custom NFSFT library which will be part of the next major release of the NFFT library. Throughout our experiments we have applied the NFFT package [11] with pre-computed Kaiser–Bessel functions and an oversampling factor $\rho = 2$. In our tests we have always chosen uniformly distributed pseudo-random source and target nodes $(\vartheta, \varphi) \in [0, \pi] \times [-\pi, \pi)$ and coefficients b_l from $\left[-\frac{1}{2}, \frac{1}{2}\right]$.

We have considered the Poisson kernel Q_h, the singularity kernel S_h, the locally supported kernel $L_{h,\lambda}$, and the spherical Gaussian kernel G_σ.

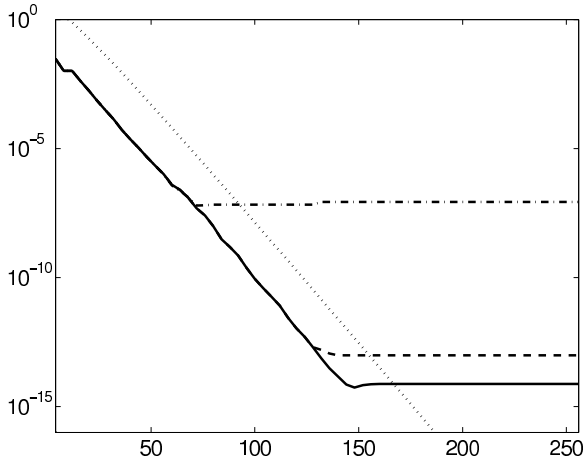
Example 5.1. First, we examine the systematic error due to our approximation (3.1) and the use of the approximate NFSFT algorithms. Figure 5.1 shows the error

$$E_\infty := \frac{\|\mathbf{f} - \mathbf{f}_M\|_\infty}{\|\mathbf{b}\|_1} \quad \approx \quad \frac{\|f - f_M\|_\infty}{\|\mathbf{b}\|_1}$$

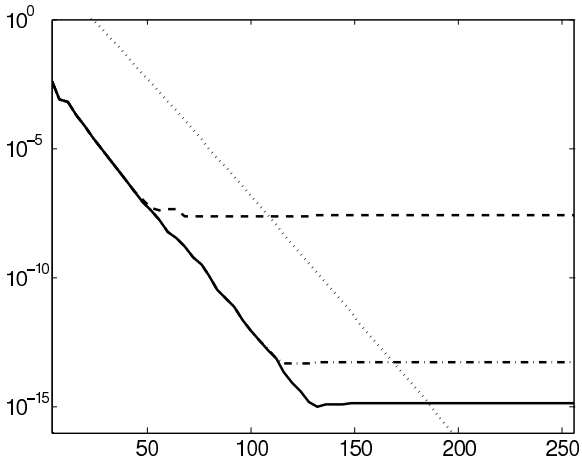
for the mentioned kernels as a function of the cut-off degree M. Here, the vector $\mathbf{f}$ contains the straightforward computed values $f(\boldsymbol{\xi}_d)$ for $d = 0, \ldots, 999$. We compute the vector $\mathbf{f}_M$ by Algorithm 1 where we use the NDSFT and NFSFT, respectively. Furthermore, we plot the error estimates given in (4.1), (4.2), (4.3), and (4.6).

Example 5.2. We now compare the computation time of the straightforward summation (direct alg.), the straightforward summation with pre-computed matrix $\mathbf{K}$ (w/pre-comp.), the fast summation algorithm with NDSFT (FS, NDSFT), and the fast summation algorithm with NFSFT (FS, NFSFT) for increasing $D = L$ and fixed cut-off degree $M = 128$. The CPU time required by the four algorithms is shown in Table 5.1. The last column shows the error $E_\infty = \|\mathbf{f} - \mathbf{f}_M\|_\infty / \|\mathbf{b}\|_1$ where we compute the vector $\mathbf{f}$ using the direct algorithm and the vector $\mathbf{f}_M$ using the NFSFT algorithm. As expected, the fast NDSFT and NFSFT summation algorithms outperform the straightforward algorithms, but with the NFSFT–version considerably faster.

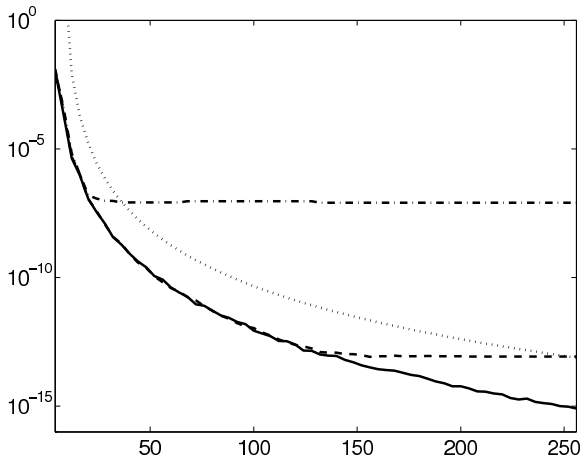
Example 5.3. As an example for the result provided by Corollary 4.8, we finally consider approximations of the matrix $\mathbf{K} \in \mathbb{R}^{L\times L}$ for the Poisson kernel Q_h for $h = 0.1$ and $h = 0.5$, $L = D = 400$ arbitrary source and target nodes $\boldsymbol{\eta}_l$ and $\boldsymbol{\xi}_d$. The particular rank $(M+1)^2$ approximation $\mathbf{Y}_\mathcal{X} \hat{\mathbf{W}} \mathbf{Y}_\mathcal{Y}^\mathsf{H}$ for cut-off degrees $M = 0, \ldots, 19$ is compared to the lth truncated singular value decomposition (TSVD) $\mathbf{S}_l$ for $l = 1, \ldots, L$. Note that for $M = 19$ we have at most an approximation of rank $(M+1)^2 = 400 = L$. Figure 5.2 shows the approximation error in the spectral norm $\|\cdot\|_2$. In particular, we conclude that the proposed approximation scheme performs almost as good as the TSVD while keeping the pre-computation effort and the costs for applying each matrix-factor at considerable lower level.



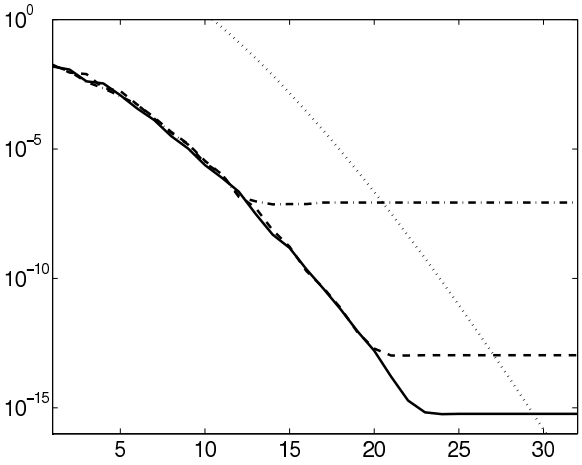
(a) The Poisson kernel Q_h for $h = 0.8$.



(b) The Singularity kernel S_h for $h = 0.8$.



(c) The locally supported kernel $L_{h,\lambda}$ for $h = 0.3$ and $\lambda = 7$.

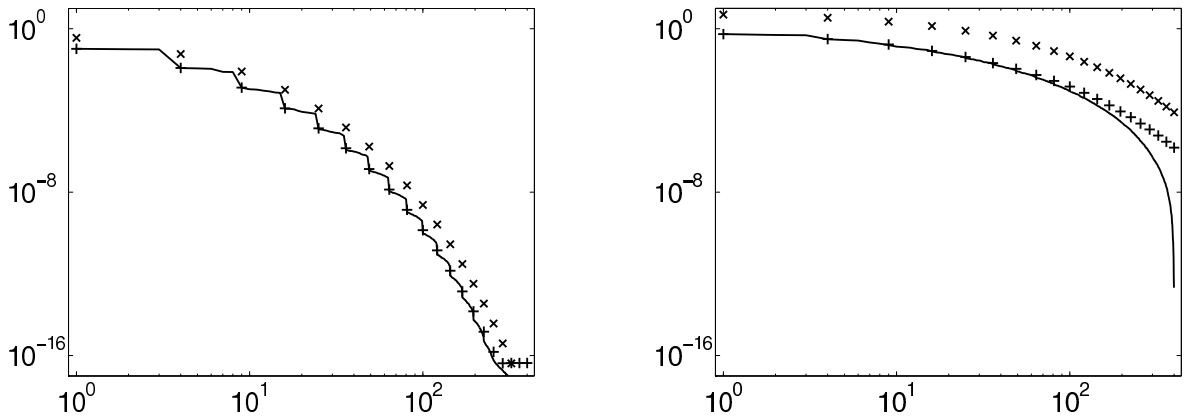


(d) The spherical Gaussian kernel G_σ for $\sigma = 5$.

Figure 5.1: The error E_∞ for $M = 4, 8, \ldots, 256$ in (a) – (c) and $M = 1, 2, \ldots, 32$ in (d), respectively, and $L = D = 1000$: Fast summation with NDSFT (solid), fast summation with NFSFT and NFFT cut-off parameter $m = 3$ (dash-dot), fast summation with NFSFT and NFFT cut-off parameter $m = 6$ (dashed), error estimate for E_∞ (dotted).

$L = D$	direct alg.	w/pre-comp.	FS, NDSFT	FS, NFSFT	error E_∞
2^6	1.0E-05	8.0E-05	1.1E-01	6.2E-01	7.7E-14
2^7	6.0E-05	3.8E-04	2.2E-01	6.2E-01	6.5E-14
2^8	2.5E-04	1.4E-03	4.5E-01	6.2E-01	4.1E-14
2^9	1.0E-03	5.3E-03	8.9E-01	6.3E-01	2.8E-14
2^{10}	4.0E-02	2.1E-02	1.8E+00	6.5E-01	3.6E-14
2^{11}	1.6E+00	8.3E-02	3.6E+00	6.6E-01	1.8E-14
2^{12}	6.4E+00	3.5E-01	7.1E+00	7.2E-01	1.3E-14
2^{13}	2.6E+01	1.4E+00	1.4E+01	8.2E-01	6.7E-15
2^{14}	1.0E+02	*5.6E+00	2.8E+01	1.0E+00	5.5E-15
2^{15}	4.1E+02	*2.2E+01	5.7E+01	1.5E+00	4.0E-15
2^{16}	1.6E+03	*8.9E+01	1.1E+02	2.3E+00	2.9E-15
2^{17}	6.6E+03	*3.6E+02	2.3E+02	4.0E+00	2.4E-15
2^{18}	2.6E+04	*1.4E+03	4.6E+02	7.5E+00	1.9E-15
2^{19}	*1.0E+05	*5.7E+03	9.1E+02	1.4E+01	-
2^{20}	*4.2E+05	*2.3E+04	1.8E+03	2.8E+01	-
2^{21}	*1.7E+06	*9.1E+04	3.6E+03	5.5E+01	-

Table 5.1: CPU-Time and error E_∞ for the fast summation algorithm with cut-off degree $M = 128$, the Poisson kernel $Q_{0.6}$, and NFFT cut-off parameter $m = 6$. Note that we used accumulated measurements in case of small times. Values marked with * were extrapolated owing to CPU-time and memory limitations.



(a) $h = 0.1$ (b) $h = 0.5$

Figure 5.2: Comparison of the proposed approximation with the truncated singular value decomposition (TSVD) for the Poisson kernel Q_h with $h = 0.1$ (left) and $h = 0.5$ (right), and $L = D = 400$ arbitrary source and target nodes. We compare the error of the TSVD $\|\mathbf{K} - \mathbf{S}_l\|_2$ (solid) of rank $l = 1, \ldots, L$ with the error of the proposed approximation $\|\mathbf{K} - \mathbf{Y}_\mathcal{X} \hat{\mathbf{W}} \mathbf{Y}_\mathcal{Y}^\mathsf{H}\|_2$ (+) and the estimate from Corollary 4.8 (×) for cut-off degree $M = 0, \ldots, 19$.

6 Conclusions

A fast approximate algorithm for the evaluation of linear combinations of radial functions on the sphere $\mathbb{S}^2$ has been presented. The proposed method is based on a particular rank approximation of the corresponding Gram matrix $\mathbf{K}$ and uses fast algorithms for spherical Fourier transforms. Under mild assumptions on the radial function our scheme takes $\mathcal{O}(L+D)$ arithmetic operations for L arbitrary source nodes and D arbitrary target nodes.

In contrast to the panel clustering method on the sphere, our approach can be interpreted as an approximation in frequency domain rendering the method particularly useful for moderately smooth functions with large spatial overlap. As an advantage, we do not require the nodes to be sorted or pre-processed in any way, thus quickly adapting to different node distributions. The pre-computation effort only depends on the particular radial function and the desired accuracy.

For a range of zonal functions, we established explicit error bounds relating computational effort to the desired accuracy and the function's parameters. We finally provided numerical examples confirming the theoretical findings with respect to approximation quality and speed. The software for the summation algorithm including all described tests can be obtained from the authors.

Acknowledgement

The second author is grateful for partial support of this work by the German Academic Exchange Service (DAAD) and the warm hospitality during his stay at the Numerical Harmonic Analysis Group, University of Vienna.

References

[1] M. Abramowitz and I. A. Stegun, editors. *Handbook of Mathematical Functions*. National Bureau of Standards, Washington, D.C., 1972.

[2] W. Castell and F. Filbir. Radial basis functions and corresponding zonal series expansions on the sphere. *J. Approx. Theory*, 134:65 – 79, 2005.

[3] J. Driscoll and D. Healy. Computing Fourier transforms and convolutions on the 2–sphere. *Adv. Appl. Math.*, 15:202 – 250, 1994.

[4] W. Freeden, T. Gervens, and M. Schreiner. *Constructive Approximation on the Sphere*. Oxford University Press, Oxford, 1998.

[5] W. Freeden, O. Glockner, and M. Schreiner. Spherical panel clustering and its numerical aspects. *J. of Geodesy*, 72:586 – 599, 1998.

[6] M. Frigo and S. G. Johnson. FFTW, a C subroutine library. http://www.fftw.org/.

[7] GSL - The GNU Scientific Library. http://www.gnu.org/software/gsl/.

[8] D. Healy, P. Kostelec, S. Moore, and D. Rockmore. FFTs for the 2-sphere – Improvements and variations. *J. Fourier Anal. Appl.*, 9:341 – 385, 2003.

[9] S. Hubbert and B. Baxter. Radial basis functions for the sphere. *Progress in Multivariate Approximation*, 137:33 – 47, 2001.

[10] J. Keiner. Fast Spherical Fourier Transforms and Applications. Diploma thesis, Institut für Mathematik, Universität zu Lübeck, 2005.

[11] S. Kunis and D. Potts. NFFT, Softwarepackage, C subroutine library. http://www.math.uni-luebeck.de/potts/nfft, 2002 – 2005.

[12] S. Kunis and D. Potts. Fast spherical Fourier algorithms. *J. Comput. Appl. Math.*, 161:75 – 98, 2003.

[13] D. Potts, G. Steidl, and M. Tasche. Fast and stable algorithms for discrete spherical Fourier transforms. *Linear Algebra Appl.*, 275:433 – 450, 1998.

[14] D. Potts, G. Steidl, and M. Tasche. Fast Fourier transforms for nonequispaced data: A tutorial. In J. J. Benedetto and P. J. S. G. Ferreira, editors, *Modern Sampling Theory: Mathematics and Applications*, pages 247 – 270, Boston, 2001. Birkhäuser.

[15] M. Schreiner. Locally supported kernels for spherical spline interpolation. *J. Approx. Theory*, 89:172 – 194, 1997.