Denoising on Sphere via Large Spherical t-designs and Spherical Framelets

Xiaosheng Zhuang
Department of Mathematics,
City University of Hong Kong
Hong Kong, SAR China
xzhuang7@cityu.edu.hk

Yuchen Xiao
Department of Mathematics
City University of Hong Kong
Hong Kong, SAR China
yc.xiao@my.cityu.edu.hk

***Abstract*—In this paper, we investigate the spherical t-designs with large value of t for function approximation, construction of spherical framelets, and the important task of spherical signal processing. Based on the spherical framelet systems and the fast framelet transform algorithms, we propose an effective denoising scheme for spherical signal denoising that utilizes the nice properties of spherical t-designs with large t value. We provide numerical results of signal/image denoising on several data sets.**

***Index Terms*—Tight framelets, spherical framelets, fast spherical harmonic transforms, fast spherical framelet transforms, spherical t-designs, Wendland functions, ETOPO1, spherical signals/images, image/signal denoising.**

I. Introduction

There are many real-world applications for signal and image processing on the unit sphere $\mathbb{S}^d := \{\boldsymbol{x} \in \mathbb{R}^{d+1} | \, \|\boldsymbol{x}\| = 1\}$, where $\|\cdot\|$ is the Euclidean norm, such as the satellite signals and global navigation in engineering, the climate change estimation in geography, the planets study in astronomy, the 360° panoramic images and videos in virtual reality, and so on. The distributions of points on the sphere play a key role in such applications. One of the most important point configurations on the sphere is the so-called *spherical t-design* point sets, which have profound significance in both theoretical aspects (such as in approximation theory, geometry, and combinatorics) and applications as mentioned above. Recently, it has been applied in image reconstruction and signal recovery on the sphere [1]. We refer to the comprehensive survey of Bannai and Bannai [2] for the past five decades' research on spherical t-designs. The concept of spherical t-design was established by Delsarte, Goethals and Seidel [3] in 1977, which said that a finite point set $X_N := \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\} \subset \mathbb{S}^d$ is a spherical t-design if the following quadrature rule (numerical integration)

$$\frac{1}{N}\sum_{i=1}^{N} p(\boldsymbol{x}_i) = \frac{1}{\omega_d}\int_{\mathbb{S}^d} p(\boldsymbol{x})\,\mathrm{d}\omega(\boldsymbol{x}) \tag{1}$$

holds for any polynomial $p \in \Pi_t^d$, where ω_d is the surface area of $\mathbb{S}^d$, $\Pi_t^d := \Pi_t(\mathbb{S}^d)$ is the space of polynomials on $\mathbb{S}^d$ with degree at most t and $\omega(\boldsymbol{x})$ denotes the surface measure on $\mathbb{S}^d$. In this paper, we restrict our attention to $\mathbb{S}^2$.

Numerical methods as computer-assisted proofs for computational spherical t-designs have developed as non-linear equations and optimization problems [4]–[7]. In numerical analysis, Hardin and Sloane [8] constructed a sequence of putative spherical t-designs with $\frac{1}{2}t^2 + o(t^2)$ points. Chen et. al in [4], [9] verified the existence of spherical t-designs with $(t+1)^2$ points for small t. Womersley [7] constructed symmetric spherical t-designs with $N = \frac{t^2+t+4}{2}$ for t up to 325. Gräf and Potts [6] computed numerical spherical t-designs by fast spherical Fourier transforms for $t \leq 1000$ with $N \approx \frac{t^2}{2}$ points.

Sloan and Womersley [5] introduced a nonnegative quantity called variational characterization of spherical t-design, which is

$$A_{N,t}(X_N) := \frac{4\pi}{N^2}\sum_{\ell=1}^{t}\sum_{m=-\ell}^{\ell}|\sum_{i=1}^{N} Y_\ell^m(\boldsymbol{x}_i)|^2,$$

where Y_ℓ^m is spherical harmonic with degree ℓ and order m. Note that $Y_0^0 = \frac{1}{\sqrt{4\pi}}$ and $\Pi_t := \Pi_t^2 = \mathrm{span}\{Y_\ell^m \,|\, (\ell, m) \in \mathcal{I}_t\}$ with the index set $\mathcal{I}_t := \{(\ell, m) \,|\, \ell = 0, \ldots, t; m = -\ell, \ldots, \ell\}$. They showed that X_N is a spherical t-design if and only if $A_{N,t}(X_N) = 0$ (cf. Theorem 3 in [5]).

Multiscale representation systems including wavelets, framelets, curvelets, shearlets, etc., are well developed for exploiting the sparsity of Euclidian data [10]–[15]. On the sphere, spherical wavelets was studied in [16]–[19] . Extension of wavelets/framelets on the sphere with more desirable properties, such as localized property, tight frame property, symmetry, directionality, etc., were further studied in [18]–[21] and many references therein. Based on hierarchical partitions, area-regular spherical Haar tight framelets were constructed in [22]. In [23], a general framework for the construction of tight framelets on a compact smooth Riemannian manifolds was proposed and fast framelet filter bank transforms are developed based on quadrature rules on the sphere.

In this paper, we explore the applications of spherical t-designs with large value of t and focus on the problem of spherical signal processing using the spherical framelet systems. Based on the truncated spherical framelet systems constructed from the spherical t-designs and the fast transform

This work was supported in part by the Research Grants Council of the Hong Kong Special Administrative Region, China, under Project CityU 11309122 and in part by the City University of Hong Kong under Project 7005603.

algorithms, we propose an effective scheme for spherical signal denoising that utilizes the nice properties of spherical t-designs with large t value. We provide numerical results of signal/image denoising using local thresholding techniques based on a fine-tuned spherical cap [24], [25] restrictions.

II. Spherical t-designs for function approximation

Numerically, a spherical t-design X_N can be found by solving a nonlinear and nonconvex minimization problem:

$$\min_{X_N \subset \mathbb{S}^2} A_{N,t}(X_N), \tag{2}$$

which can be done by using optimization methods such as the line-search methods or the trust-region methods [26]. Using the spiral point sets as the initial point sets, we provide in table I for different t, the number of points $N = (t+1)^2$, the number K_{TR} of the iterations using the trust-region method to reach the final numerical spherical t-design (SPD) point sets with their $\sqrt{A_{N,t}(X_N)}$, $\|\nabla A_{N,t}(X_N)\|_\infty$, and the running time, respectively. For more details on obtaining spherical t-designs using the trust-region method, we refer to [27]. In addition, there are other fast algorithms like manifold version's Newton and Conjugate Gradient method for computing numerical spherical t-designs in [6].

TABLE I
Spherical t-designs X_N (SPD) by the trust-region method.

t	N	K_{TR}	$\sqrt{A_{N,t}(X_N)}$	$\|\nabla A_{N,t}(X_N)\|_\infty$	Time
16	289	264	2.15E-12	7.04E-16	10.51 s
32	1089	567	1.51E-12	7.93E-16	24.61 s
64	4225	1087	1.13E-12	1.27E-15	2.01 min
128	16641	1929	1.55E-12	1.07E-15	11.16 min
256	66049	3234	1.13E-12	1.39E-15	32.50 min
512	263169	6049	1.18E-12	8.64E-15	4.59 h
1024	1050625	9951	1.28E-12	3.80E-15	1.02 d
25	676	422	1.73E-12	6.84E-15	15.38 s
50	2601	764	1.58E-12	9.39E-15	46.52 s
100	10201	1699	1.00E-12	8.51E-16	3.08 min
200	40401	2922	1.16E-12	2.30E-15	26.85 min
400	160801	4980	1.09E-12	4.22E-15	2.29 h
800	641601	8489	1.53E-12	4.18E-14	21.74 h
1600	2563601	18274	1.70E-10	9.26E-14	6.95 d
3200	10246401	22371	1.07E-09	2.22E-12	2.07 mo

Once we obtained the spherical t-design point sets, which are equal weight quadrature rules $Q_{N,t} = (X_N, \boldsymbol{w})$ on $\mathbb{S}^2$ with $\boldsymbol{w} = (w_1, \ldots, w_N)$ and $w_i \equiv \frac{1}{N}$, we can use them for function approximation. For a function $f : \mathbb{S}^2 \to \mathbb{C}$, we define $\boldsymbol{f} := f\big|_{X_N} = (f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_N))$ be a vector for samples of f on X_N.

For a (squared-integrable) signal function $f \in L_2(\mathbb{S}^2)$ on $\mathbb{S}^2$, it can be represented (in L_2-sense) as $f = f_t + g_t$ with $f_t \in \Pi_t$ being the approximation (projection) polynomial and $g_t = f - f_t \notin \Pi_t$ being the residual function. The approximation polynomial f_t can be found through the spherical harmonic basis $\{Y_\ell^m \mid \ell \geq 0, |m| \leq \ell\}$. In fact, by $f_t \in \Pi_t$, we have

$$f(\boldsymbol{x}_i) = f_t(\boldsymbol{x}_i) + g_t(\boldsymbol{x}_i) = \sum_{\ell=0}^{t} \sum_{m=-\ell}^{\ell} \hat{f}_\ell^m Y_\ell^m(\boldsymbol{x}_i) + g_t(\boldsymbol{x}_i).$$

In the vector form, it is equivalent to $\boldsymbol{f} = \boldsymbol{f}_t + \boldsymbol{g}_t = \boldsymbol{Y}_t \hat{\boldsymbol{f}} + \boldsymbol{g}_t$, where the vector $\hat{\boldsymbol{f}} := (\hat{f}_\ell^m)_{(\ell,m)\in\mathcal{I}_t}$ is the *Fourier (spherical harmonic) coefficient* vector and the matrix $\boldsymbol{Y}_t := \boldsymbol{Y}_t(X_N) := (Y_\ell^m(\boldsymbol{x}_i))_{i\in[N],(\ell,m)\in\mathcal{I}_t}$ is of size $N \times (t+1)^2$ with $[N] := \{1, \ldots, N\}$. The *Fourier (spherical harmonic) coefficient* vector $\hat{\boldsymbol{f}}$ can be found through the minimization problem:

$$\min_{\boldsymbol{f}_t \in \Pi_t} \|\boldsymbol{f} - \boldsymbol{f}_t\|. \tag{3}$$

To solve this problem, by $\boldsymbol{f}_t = \boldsymbol{Y}_t \hat{\boldsymbol{f}}$, we aim at finding $\hat{\boldsymbol{f}}$ such that $\boldsymbol{Y}_t \hat{\boldsymbol{f}} = \boldsymbol{f}$. Considering the weight $\boldsymbol{w}$ together, we can solve it by

$$\boldsymbol{Y}_t^\star(\boldsymbol{w} \odot \boldsymbol{Y}_t \hat{\boldsymbol{f}}) = \boldsymbol{Y}_t^\star(\boldsymbol{w} \odot \boldsymbol{f}). \tag{4}$$

Let $\boldsymbol{W} := \mathrm{diag}(\boldsymbol{w})$. Then eq. (4) is actually to solve x for $Ax = b$, where $A = \boldsymbol{Y}_t^\star \boldsymbol{W} \boldsymbol{Y}_t$, $x = \hat{\boldsymbol{f}}$ and $b = \boldsymbol{Y}_t^\star \boldsymbol{W} \boldsymbol{f}$, which can be done by standard conjugate gradient (CG) methods. Here, $\boldsymbol{Y}_t^\star := \overline{\boldsymbol{Y}_t(X_N)}^\top \in \mathbb{C}^{(t+1)^2 \times N}$ is the transpose of complex conjugate of Y_t. Note that the transform operation $\boldsymbol{Y}_t \hat{\boldsymbol{f}}$ and adjoint operation $\boldsymbol{Y}_t^\star \boldsymbol{f}$ can be done through fast spherical harmonic transforms such as the NFSFT [28].

III. Spherical framelets

Spherical t-design point sets can also be used in the construction of semi-discrete spherical tight framelets, which we briefly introduced below.

Let $\Psi := \{\alpha; \beta_1, \ldots, \beta_n\} \subset L_1(\mathbb{R})$ be a set of generating functions associating with a filter bank $\eta := \{a; b_1, \ldots, b_n\} \subset \ell_1(\mathbb{Z})$ and satisfying the relations $\hat{\alpha}(2\xi) = \hat{a}(\xi)\hat{\alpha}(\xi)$ and $\hat{\beta}_s(2\xi) = \hat{b}_s(\xi)\hat{\alpha}(\xi)$, $s \in [n]$, where $\hat{\alpha}(\xi) := \int_{\mathbb{R}} \alpha(x) e^{-2\pi i x \xi} dx$ is the Fourier transform and for a filter (mask) $h = \{h_k\}_{k\in\mathbb{Z}} \subset \mathbb{C}$, the Fourier series $\hat{h}(\xi) := \sum_{k\in\mathbb{Z}} h_k \mathrm{e}^{-2\pi \mathrm{i} k \xi}$. Here $\ell_p(\mathbb{Z})$ is the ℓ_p space on $\mathbb{Z}$.

Let $\mathcal{Q} := \{Q_{N_j} = (X_{N_j}, \boldsymbol{w}_j)\}_j$ with $X_{N_j} := (\boldsymbol{x}_{j,k})_{k\in[N_j]}$ and $\boldsymbol{w}_j = (w_{j,k})_{k\in[N_j]}$ be a sequence of quadrature rules on $\mathbb{S}^2$. We can define the semi-discrete spherical framelets $\varphi_{j,k}(\boldsymbol{x})$ and $\psi_{j,k}^{(s)}(\boldsymbol{x})$ for $s \in [n]$ and for $k \in [N_j]$ as

$$\varphi_{j,k}(\boldsymbol{x}) := \sqrt{w_{j,k}} \sum_{\ell=0}^{\infty} \sum_{m=-\ell}^{\ell} \hat{\alpha}(\frac{\ell}{2^j}) \overline{Y_\ell^m(\boldsymbol{x}_{j,k})} Y_\ell^m(\boldsymbol{x}), \tag{5}$$

$$\psi_{j,k}^{(s)}(\boldsymbol{x}) := \sqrt{w_{j+1,k}} \sum_{\ell=0}^{\infty} \sum_{m=-\ell}^{\ell} \hat{\beta}_s(\frac{\ell}{2^j}) \overline{Y_\ell^m(\boldsymbol{x}_{j+1,k})} Y_\ell^m(\boldsymbol{x}). \tag{6}$$

The semi-discrete spherical framelet system $\mathcal{F}_J(\Psi, \mathcal{Q})$ starting at a scale $J \in \mathbb{Z}$ is

$$\begin{aligned}\mathcal{F}_J(\Psi, \mathcal{Q}) :=& \{\varphi_{J,k} : k \in [N_J]\} \\ & \cup \{\psi_{j,k}^{(s)} : k \in [N_{j+1}], s \in [n]\}_{j=J}^{\infty}.\end{aligned}$$

If $\mathcal{F}_J(\Psi, \mathcal{Q}) \subset L_2(\mathbb{S}^2)$ and $\forall f \in L^2(\mathbb{S}^2)$, in L_2-sense, we have $f = \sum_{k=1}^{N_J} \langle f, \varphi_{J,k} \rangle_{L^2(\mathbb{S}^2)} \varphi_{J,k} + \sum_{j=J}^{\infty} \sum_{k=1}^{N_{j+1}} \sum_{s=1}^{n} \langle f, \psi_{j,k}^{(s)} \rangle_{L^2(\mathbb{S}^2)} \psi_{j,k}^{(s)}$, then $\mathcal{F}_J(\Psi, \mathcal{Q})$ is called a (semi-discrete) tight frame for $L_2(\mathbb{S}^2)$. With the spherical t-designs being equal weight quadrature rules, tightness of $\mathcal{F}_J(\Psi, \mathcal{Q})$ can be easily satisfied. For more details regarding the tightness of such systems, we refer to [23], [27].

In practice, the infinite system $\mathcal{F}_J(\Psi, \mathcal{Q})$ is truncated at certain level and one only needs to use the filter bank association

and the fast spherical harmonic transforms for implementing the fast spherical framelet transforms. We use the system $\mathcal{F}_{J_0}^J(\eta, \mathcal{Q})$ in [27] for the decomposition and reconstruction of spherical signals, see Algorithms 1 and 2.

Algorithm 1 Multi-level Spherical Framelet Transforms: Decomposition

Require: $\{Q_{N_j,t_j} = (X_{N_j}, \boldsymbol{w}_j)\}_{j=J_0}^{J+1}$: quadrature rules; $\boldsymbol{f}_{J+1} = f|_{X_{N_{J+1}}}$: samples of $f \in \Pi_{t_J}$ on the spherical point set $X_{N_{J+1}}$; η: filter bank.

Initialize $\hat{\boldsymbol{f}}_{\boldsymbol{J+1}} = w_{j+1}\boldsymbol{Y}_{t_{J+1}}^{\star}\boldsymbol{f}_{J+1}$.

for j from J to J_0 **do**
  for s from 1 to n **do**
    $\boldsymbol{w}_j^{(s)} = \sqrt{w_{j+1}}\boldsymbol{Y}_{t_{j+1}}[\hat{\boldsymbol{f}}_{j+1} \odot \bar{\hat{\boldsymbol{b}}}_j^{(s)}]$.
  end for
  $\hat{\boldsymbol{f}}_j = [\hat{\boldsymbol{f}}_{j+1} \odot \bar{\hat{\boldsymbol{a}}}_j]{\downarrow_j}$.
end for
$\boldsymbol{v}_{J_0} = \sqrt{w_{J_0}}\boldsymbol{Y}_{t_{J_0}}\hat{\boldsymbol{f}}_{J_0}$.

Ensure: Framelet coefficients $\{\boldsymbol{v}_{J_0}, \boldsymbol{w}_j^{(s)} \,|\, j = J_0, \ldots J; s \in [n]\}$.

Algorithm 2 Multi-level Spherical Framelet Transforms: Reconstruction

Require: $\{Q_{N_j,t_j} = (X_{N_j}, \boldsymbol{w}_j)\}_{j=J_0}^{J+1}$: quadrature rules; $\{\boldsymbol{v}_{J_0}, \boldsymbol{w}_j^{(s)} \,|\, j = J_0, \ldots J; s \in [n]\}$: coefficient sequences; η: filter bank.

Initialize $\hat{\boldsymbol{f}}_{\boldsymbol{J_0}} = \sqrt{w_{J_0}}\boldsymbol{Y}_{t_{J_0}}\boldsymbol{v}_{J_0}$.

for j from J_0 to J **do**
  $\hat{\boldsymbol{f}}_{j+1} = \hat{\boldsymbol{f}}_j{\uparrow_{j+1}} \odot \bar{\hat{\boldsymbol{a}}}_j$
  for s from 1 to n **do**
    $\hat{\boldsymbol{f}}_{j+1} = \hat{\boldsymbol{f}}_{j+1} + [\sqrt{w_{j+1}}\boldsymbol{Y}_{t_{j+1}}^{\star}\boldsymbol{w}_j^{(s)}] \odot \hat{\boldsymbol{b}}_j^{(s)}$.
  end for
end for
$\boldsymbol{f_{J+1}} = w_{j+1}\boldsymbol{Y}_{t_{J+1}}\hat{\boldsymbol{f}}_{J+1}$.

Ensure: $\boldsymbol{f}_{J+1}$: samples of $f \in \Pi_{t_J}$ on the spherical point set $X_{N_{J+1}}$;

IV. Denoising Scheme

Given a noisy function $f_\sigma = f_o + G_\sigma$ on $X_{N_{J+1}}$, where f_o is an unknown underground truth and G_σ is the Gaussian white noisy. Suppose that we have obtained a sequence of spherical t_j-design point sets $\{Q_{N_j,t_j} := (X_{N_j}, \boldsymbol{w}_j)\}_{j=J_0}^{J+2}$. We aim at obtaining the ground truth f_o on $X_{N_{J+1}}$, i.e., the denoised version of f_σ, based on the spherical t-designs and the spherical framelets. We use the following steps.

(S1) Projection. We project f_σ onto Π_{t_J} by using eq. (3) with $t = t_J$ and $X_N = X_{N_{J+1}}$, to obtain $f_\sigma = f + g$ on $X_{N_{J+1}}$ such that $f \in \Pi_{t_J}$ is the projection part and $g = f_\sigma - f$ is the residual part.

(S2) Denoising via spherical framelets. We then use the truncated spherical tight framelet system $\mathcal{F}_{J_0}^J(\eta, \mathcal{Q})$ (see [27]) to decompose f (more precisely, $\boldsymbol{f}_{J+1} = f|_{X_{N_{J+1}}}$, see Algorithm 1) into the framelet coefficient sequences $\{\boldsymbol{v}_{J_0}\} \cup \{\boldsymbol{w}_j^{(s)} \,|\, j = J_0, \ldots, J; s \in [n]\}$. We apply the thresholding techniques for denoising the framelet coefficient sequences $\boldsymbol{w}_j^{(s)}$ of f and the residual g. More precisely, given the framelet coefficient sequence $\boldsymbol{w}_j^{(s)} = (w_{j,k}^{(s)})_{k \in [N_{j+1}]}$, note that $w_{j,k}^{(s)}$ is associated with the point $\boldsymbol{x}_{j+1,k}$. We first normalize it according to the norm $\|\psi_{j,k}^{(s)}\|_{L_2(\mathbb{S}^2)}$ by $\tilde{w}_{j,k}^{(s)} = w_{j,k}^{(s)}/\|\psi_{j,k}^{(s)}\|_{L_2(\mathbb{S}^2)}$. We use local thresholding technique based on spherical caps that updates $\tilde{w}_{j,k}^{(s)}$ to be

$$\check{w}_{j,k}^{(s)} = \begin{cases} \tilde{w}_{j,k}^{(s)} - \operatorname{sgn}(\tilde{w}_{j,k}^{(s)})\tau_{j,k,r}^{(s)}, & |\tilde{w}_{j,k}^{(s)}| \geq \tau_{j,k,r}^{(s)}, \\ 0, & |\tilde{w}_{j,k}^{(s)}| < \tau_{j,k,r}^{(s)}, \end{cases} \tag{7}$$

where $\tau_{j,k,r}^{(s)}$ is a thresholding value determined by

$$\tau_{j,k,r}^{(s)} = \frac{c \cdot \sigma^2}{\sqrt{(\bar{w}_{j,k,r}^{(s)} - \sigma^2)_+}} \tag{8}$$

with c being a constant that is tuned by hand to optimize the performance. Here, $\bar{w}_{j,k,r}^{(s)}$ is the average of the coefficients near $\tilde{w}_{j,k}^{(s)}$ determined by a spherical cap $C(\boldsymbol{x}, r) := \{\boldsymbol{y} \in \mathbb{S}^2 : \|\boldsymbol{x} \times \boldsymbol{y}\| \leq r\}$ of radius r and centered at $\boldsymbol{x} = \boldsymbol{x}_{j+1,k}$. Then we can obtain f_{thr} by the reconstruction algorithm in Algorithm 2 using the thresholded coefficient sequences $\{\boldsymbol{v}_{J_0}\} \cup \{\check{\boldsymbol{w}}_j^{(s)} \,|\, j = J_0, \ldots, J; s \in [n]\}$. Similarly, we can obtained g_{thr} following the local-soft thresholding technique for g. Thus, we obtain the denoised signal $f_{\sigma,thr} = f_{thr} + g_{thr}$.

(S3) Upsampling on a finer point set for further denoising. The signal $f_{\sigma,thr}$ on $\boldsymbol{X} = X_{N_{J+1}}$ could still have noise. To further improve the denoising performance, we upsample $f_{\sigma,thr}$ to f_σ^1 on $\boldsymbol{Y} := X_{N_{J+2}}$. We use the simple nearest neighborhood technique: for each point $\boldsymbol{y} \in \boldsymbol{Y}$, we define $f_\sigma^1(\boldsymbol{y}) = f_{\sigma,thr}(\boldsymbol{x})$ with $\boldsymbol{x} = \arg\min_{\boldsymbol{x} \in \boldsymbol{X}} \|\boldsymbol{x} \times \boldsymbol{y}\|$. Then, for the upsampled f_σ^1, we use the similar procedure in Step (S2) (by replacing $\{Q_{N_j,t_j} := (X_{N_j}, \boldsymbol{w}_j)\}_{j=J_0}^{J+1}$ by $\{Q_{N_j,t_j} := (X_{N_j}, \boldsymbol{w}_j)\}_{j=J_0+1}^{J+2}$) to obtain a further denoised signal $f_{\sigma,thr}^1 = f_{thr}^1 + g_{thr}^1$ on $\boldsymbol{Y}$, through our spherical framelets and the local thresholding techniques.

(S4) Downsampling to the original point set. To see the denoising performance of the above procedure. We downsample $f_{\sigma,thr}^1$ on $\boldsymbol{Y}$ to $f_{\sigma,thr}^0$ on $\boldsymbol{X}$ so that we can evaluate the performance by comparing the final output signal $f_{\sigma,thr}^0$ to the ground truth f_o. We use the following averaging technique. For each $\boldsymbol{x} \in \boldsymbol{X}$, we can find its neigborhood points $\mathcal{N}(\boldsymbol{x}, r)$ in $\boldsymbol{Y}$ in a spherical cap $C(\boldsymbol{x}, r)$, that is $\mathcal{N}(\boldsymbol{x}, r) := C(\boldsymbol{x}, r) \cap \boldsymbol{Y}$. We then define $f_{\sigma,thr}^0(\boldsymbol{x}) = \frac{1}{\#\mathcal{N}(\boldsymbol{x},r)} \sum_{\boldsymbol{y} \in \mathcal{N}(\boldsymbol{x},r)} f_{\sigma,thr}^1(\boldsymbol{y})$. Then we obtain $f_{\sigma,thr}^0$ on $\boldsymbol{X} = X_{N_{J+1}}$. One can then use SNR or PSNR to evaluate the performance.

V. Numerical results

In this section, we present the performance of our denoising scheme in Section IV for noisy data of radial basis functions (Wendland), Earth data (ETOPO1) and images on sphere.

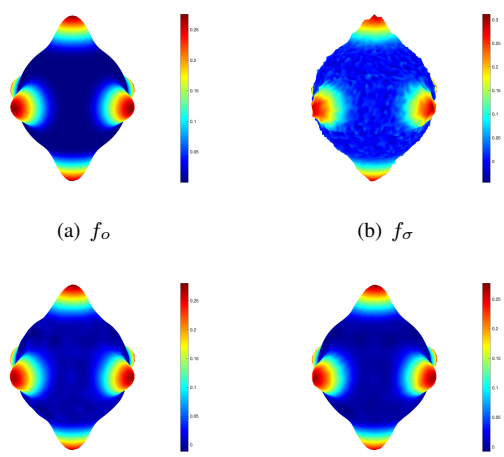
Given spherical t-design point sets $X_{N_0}, X_{N_1}, X_{N_2}, X_{N_3}$ (SPD) corresponding to degree t_0, t_1, t_2, t_3, it corresponds to $J_0 = 0$ and $J = 1$, that is, 2-level decomposition and reconstruction, in Algorithms 1 and 2, and $\boldsymbol{X} = X_{N_2}$ and $\boldsymbol{Y} = X_{N_3}$ in Section IV. We refer to [27] for the details of the filter bank $\eta = \eta_3 = \{a; b_1, b_2, b_3\}$ (3 high-pass filters and 1 low-pass filter), the Wendland functions f_4, the ETOPO1 data set, and the spherical images.

A. Wendland function

For Wendland function f_4, we use $\mathrm{SNR}(f_o, f_r) := 10\log_{10}(\frac{\|f_o\|}{\|f_r - f_o\|})$ to measure the quality of signal denoising of f_σ, where f_r is a reconstruction (denoised) signal. For Wendland function f_4, we let $t_0 = 16, t_1 = 32, t_2 = 64, t_3 = 128$. The results including cost time in two methods are presented in table II, where the results of $f_{\sigma,thr}$ have already shown as best results in [27] under the setting of threshold coefficients $c_f = 1$ and $c_g = 3$ in eq. (8). For $f^0_{\sigma,thr}$, we set $c_f = 1/2$ and $c_g = 3/2$ for (S2), then we set $c^\uparrow_f = 0.01$ and $c^\uparrow_g = 0.01$ as threshold coefficients upsampling denoising for (S3). We can see that results of upsampling denoising scheme $f^0_{\sigma,thr}$ are better than convention denoising method results of $f_{\sigma,thr}$ at about 1 dB. (When set $c_f = 0.9$ and $c_g = 2.9$ for (S2), $c^\uparrow_f = c_f/8$ and $c^\uparrow_g = c_g/8$ for (S3), $f^0_{\sigma,thr}$ will be **25.58** for $\sigma = 0.05\|f_o\|_\infty$.) We also show some figures in fig. 1 related to the results in table II.

TABLE II
Wendland denoising results with respect to different noise level σ.

σ	0.05	0.075	0.1	0.125	0.15	0.175	0.2
f_σ	**13.63**	**10.11**	**7.61**	**5.67**	**4.09**	**2.75**	**1.59**
$f_{\sigma,thr}$ [27]	24.48	21.25	19.03	17.30	15.82	14.49	13.19
Time	0.32s	0.31s	0.31s	0.33s	0.34s	0.31s	0.30s
$f^0_{\sigma,thr}$	**25.19**	**22.53**	**20.16**	**18.28**	**16.98**	**15.58**	**14.34**
Time	1.24s	1.27s	1.36s	1.37s	1.35s	1.34s	1.33s



(a) f_o (b) f_σ (c) $f_{\sigma,thr}$ (d) $f^0_{\sigma,thr}$

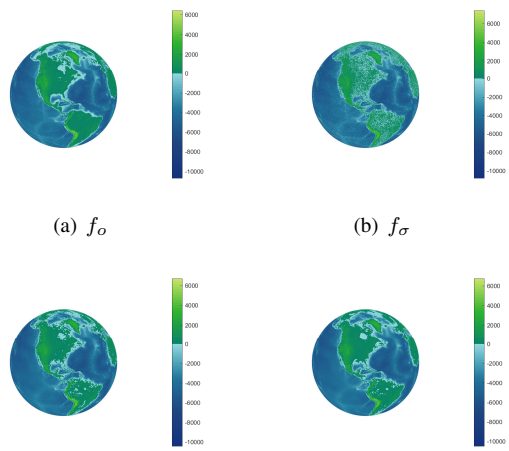
Fig. 1. The behavior of 2-levels framelet decomposition, thresholding and reconstruction for Wendland function f_4 with $\sigma = 0.05$ by η_3 on SPD with $t_0 = 16, t_1 = 32, t_2 = 64, t_3 = 128$.

B. ETOPO1 data

For ETOPO1 data, we use $\mathrm{SNR}(f_o, f_r)$ to measure the quality of signal denoising of f_σ. We let $t_0 = 128, t_1 = 256, t_2 = 512, t_3 = 1024$. We show the result on table III. For $f_{\sigma,thr}$, we set $c_f = c_g = 0.6$ same as [27]. For $f^0_{\sigma,thr}$, we set $c_f = c_g = 0.3$ for (S2), and we set $c^\uparrow_f = c^\uparrow_g = 0.1$ for (S3). We also display some figures for ETOPO1 data in Fig. 2. The results show that the performance of $f^0_{\sigma,thr}$ are better than $f_{\sigma,thr}$.

TABLE III
ETOPO1 denoising results with respect to different noise level σ.

σ	0.05	0.075	0.1	0.125	0.15	0.175	0.2
f_σ	**16.21**	**12.69**	**10.19**	**8.25**	**6.66**	**5.33**	**4.17**
$f_{\sigma,thr}$ [27]	21.06	18.93	17.43	16.25	15.33	14.59	14.01
Time	12.03s	12.60s	13.17s	13.05s	13.26s	14.01s	14.31s
$f^0_{\sigma,thr}$	**21.20**	**19.14**	**17.71**	**16.53**	**15.62**	**14.91**	**14.38**
Time	66.17s	66.51s	68.62s	70.15s	72.23s	78.94s	81.12s



(a) f_o (b) f_σ (c) $f_{\sigma,thr}$ (d) $f^0_{\sigma,thr}$

Fig. 2. The behavior of 2-levels framelet decomposition, thresholding and reconstruction for ETOPO f_σ with $\sigma = 0.05$ by η_3 on SPD with $t_0 = 128, t_1 = 256, t_2 = 512, t_3 = 1024$.

C. Spherical Images

We use PSNR measure the quality of image denoising, which is $\mathrm{PSNR}(f_o, f_r) := 10\log_{10}(\frac{255^2}{\mathrm{MSE}})$ and MSE is the mean squared error which defined as $\mathrm{MSE} = \frac{1}{N}\sum_{\boldsymbol{x}\in X_N}|f_o(\boldsymbol{x}) - f_r(\boldsymbol{x})|^2$. For spherical images, we let $t_0 = 128, t_1 = 256, t_2 = 512, t_3 = 1024$. We show the results in table IV. The rows for $f_{\sigma,thr}$, we set $c_f = 0.6$ and $c_g = 0.5$ the same as those in [27]. For $f^0_{\sigma,thr}$, we set $c_f = 0.3$ and $c_g = 0.25$ for (S2), and we set $c^\uparrow_f = c^\uparrow_g = 0.1$ for (S3). The results show that the performance of $f^0_{\sigma,thr}$ are better than $f_{\sigma,thr}$ except Mandrill for $\sigma = 0.2$.

VI. Conclusions and Final Remarks

In conclusion, spherical t-designs can be used for function approximation and for the construction of spherical framelets.

TABLE IV

Images denoising results. For each images, the first row f_σ is $\text{PSNR}_0 := \text{PSNR}(f_G, f_\sigma)$, the second row $f_{\sigma,thr}$ is $\text{PSNR}(f_G, f_{\sigma,thr})$ values with the denoising scheme in [27], and the third row $f^0_{\sigma,thr}$ is $\text{PSNR}(f_G, f^0_{\sigma,thr})$ values with the denoising scheme in this paper.

Image	σ	0.05	0.075	0.1	0.125	0.15	0.175	0.2
Barbara	f_σ	**26.33**	**22.81**	**20.31**	**18.37**	**16.79**	**15.45**	**14.29**
	$f_{\sigma,thr}$	29.70	27.19	25.48	24.22	23.28	22.57	21.99
	$f^0_{\sigma,thr}$	**29.84**	**27.38**	**25.73**	**24.46**	**23.48**	**22.74**	**22.16**
Boat	f_σ	**26.02**	**22.50**	**20.00**	**18.06**	**16.48**	**15.14**	**13.98**
	$f_{\sigma,thr}$	29.94	27.83	26.34	25.16	24.15	23.31	22.60
	$f^0_{\sigma,thr}$	**30.17**	**27.99**	**26.51**	**25.34**	**24.30**	**23.40**	**22.67**
Hill	f_σ	**26.69**	**23.17**	**20.67**	**18.73**	**17.15**	**15.81**	**14.65**
	$f_{\sigma,thr}$	30.20	28.09	26.66	25.56	24.73	24.07	23.53
	$f^0_{\sigma,thr}$	**30.38**	**28.24**	**26.75**	**25.59**	**24.76**	**24.15**	**23.68**
Lena	f_σ	**26.37**	**22.85**	**20.35**	**18.41**	**16.83**	**15.49**	**14.33**
	$f_{\sigma,thr}$	31.87	29.73	28.14	26.82	25.76	24.90	24.16
	$f^0_{\sigma,thr}$	**32.14**	**30.04**	**28.54**	**27.27**	**26.16**	**25.26**	**24.51**
Man	f_σ	**26.51**	**22.99**	**20.49**	**18.55**	**16.97**	**15.63**	**14.47**
	$f_{\sigma,thr}$	30.67	28.39	26.83	25.65	24.69	23.90	23.27
	$f^0_{\sigma,thr}$	**30.90**	**28.56**	**27.03**	**25.83**	**24.85**	**24.04**	**23.39**
Mandrill	f_σ	**28.17**	**24.65**	**22.15**	**20.22**	**18.63**	**17.29**	**16.13**
	$f_{\sigma,thr}$	29.61	26.82	24.98	23.69	22.74	22.06	**21.54**
	$f^0_{\sigma,thr}$	**29.71**	**26.94**	**25.06**	**23.76**	**22.81**	**22.08**	21.52

Moreover, using the spherical t-designs with large value of t, we apply them for spherical signal/image denoising through the fast framelet transforms and the carefully design denoising schemes. In future, we may consider further improvement of the denoising scheme by using more sophisticated resampling techniques as well the other important tasks such as spherical image inpainting or super-resolution imaging.

REFERENCES

[1] X. Chen and R. S. Womersley, "Spherical designs and nonconvex minimization for recovery of sparse signals on the sphere," *SIAM Journal on Imaging Sciences*, vol. 11, no. 2, pp. 1390–1415, 2018.

[2] E. Bannai and E. Bannai, "A survey on spherical designs and algebraic combinatorics on spheres," *European Journal of Combinatorics*, vol. 30, no. 6, pp. 1392–1425, 2009.

[3] P. Delsarte, J. Goethals, and J. Seidel, "Spherical codes and designs," *Geometriae Dedicata*, vol. 6, no. 3, pp. 363–388, 1977.

[4] X. Chen and R. S. Womersley, "Existence of solutions to systems of underdetermined equations and spherical designs," *SIAM Journal on Numerical Analysis*, vol. 44, no. 6, pp. 2326–2341, 2006.

[5] I. H. Sloan and R. S. Womersley, "A variational characterisation of spherical designs," *Journal of Approximation Theory*, vol. 159, no. 2, pp. 308–318, 2009.

[6] M. Gräf and D. Potts, "On the computation of spherical designs by a new optimization approach based on fast spherical fourier transforms," *Numerische Mathematik*, vol. 119, no. 4, pp. 699–724, 2011.

[7] R. S. Womersley, "Efficient spherical designs with good geometric properties," in *Contemporary Computational Mathematics-A Celebration of the 80th Birthday of Ian Sloan*. Springer, 2018, pp. 1243–1285.

[8] R. H. Hardin and N. J. Sloane, "Mclaren's improved snub cube and other new spherical designs in three dimensions," *Discrete & Computational Geometry*, vol. 15, no. 4, pp. 429–441, 1996.

[9] X. Chen, A. Frommer, and B. Lang, "Computational existence proofs for spherical t-designs," *Numerische Mathematik*, vol. 117, no. 2, pp. 289–305, 2011.

[10] Y. Meyer, "Ondelettes et opérateurs," *I: Ondelettes*, 1990.

[11] I. Daubechies, *Ten lectures on wavelets*. SIAM, 1992.

[12] C. K. Chui, *An introduction to wavelets*. Academic press, 1992, vol. 1.

[13] S. Mallat, *A wavelet tour of signal processing*. Elsevier, 1999.

[14] X. Zhuang, "Digital affine shear transforms: fast realization and applications in image/video processing," *SIAM Journal on Imaging Sciences*, vol. 9, no. 3, pp. 1437–1466, 2016.

[15] Z. Che and X. Zhuang, "Digital affine shear filter banks with 2-layer structure and their applications in image processing," *IEEE Transactions on Image Processing*, vol. 27, no. 8, pp. 3931–3941, 2018.

[16] P. Schröder and W. Sweldens, "Spherical wavelets: Efficiently representing functions on the sphere," in *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, 1995, pp. 161–172.

[17] W. Freeden and U. Windheuser, "Spherical wavelet transform and its discretization," *Advances in Computational Mathematics*, vol. 5, no. 1, pp. 51–94, 1996.

[18] Y. Wiaux, J. D. McEwen, and P. Vielva, "Complex data processing: Fast wavelet analysis on the sphere," *Journal of Fourier Analysis and Applications*, vol. 13, no. 4, pp. 477–493, 2007.

[19] J. D. McEwen, C. Durastanti, and Y. Wiaux, "Localisation of directional scale-discretised wavelets on the sphere," *Applied and Computational Harmonic Analysis*, vol. 44, no. 1, pp. 59–88, 2018.

[20] L. Demanet and P. Vandergheynst, "Directional wavelets on the sphere," *Technical Report R-2001-2, Signal Processing Laboratory (LTS), EPFL, Lausanne*, 2001.

[21] I. Iglewska-Nowak, "Frames of directional wavelets on n-dimensional spheres," *Applied and Computational Harmonic Analysis*, vol. 43, no. 1, pp. 148–161, 2017.

[22] J. Li, H. Feng, and X. Zhuang, "Convolutional neural networks for spherical signal processing via area-regular spherical haar tight framelets," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[23] Y. G. Wang and X. Zhuang, "Tight framelets and fast framelet filter bank transforms on manifolds," *Applied and Computational Harmonic Analysis*, vol. 48, no. 1, pp. 64–95, 2020.

[24] F. Dai and H. Wang, "Positive cubature formulas and marcinkiewicz–zygmund inequalities on spherical caps," *Constructive Approximation*, vol. 31, no. 1, pp. 1–36, 2010.

[25] K. Hesse and R. S. Womersley, "Numerical integration with polynomial exactness over a spherical cap," *Advances in Computational Mathematics*, vol. 36, no. 3, pp. 451–483, 2012.

[26] W. Sun and Y.-X. Yuan, *Optimization theory and methods: nonlinear programming*. Springer Science & Business Media, 2006, vol. 1.

[27] Y. Xiao and X. Zhuang, "Spherical framelets from spherical designs," *arXiv preprint arXiv:2303.05365*, 2023.

[28] S. Kunis and D. Potts, "Fast spherical fourier algorithms," *Journal of Computational and Applied Mathematics*, vol. 161, no. 1, pp. 75–98, 2003.