

# SHAPE-DRIVEN INTERPOLATION WITH DISCONTINUOUS KERNELS: ERROR ANALYSIS, EDGE EXTRACTION, AND APPLICATIONS IN MAGNETIC PARTICLE IMAGING\*

S. DE MARCHI<sup>†</sup>, W. ERB<sup>†</sup>, F. MARCHETTI<sup>‡</sup>, E. PERRACCHIONE<sup>§</sup>, AND M. ROSSINI<sup>¶</sup>

**Abstract.** Accurate interpolation and approximation techniques for functions with discontinuities are key tools in many applications, such as medical imaging. In this paper, we study a radial basis function type of method for scattered data interpolation that incorporates discontinuities via a variable scaling function. For the construction of the discontinuous basis of kernel functions, information on the edges of the interpolated function is necessary. We characterize the native space spanned by these kernel functions and study error bounds in terms of the fill distance of the node set. To extract the location of the discontinuities, we use a segmentation method based on a classification algorithm from machine learning. The results of the conducted numerical experiments are in line with the theoretically derived convergence rates in case that the discontinuities are a priori known. Further, an application to interpolation in magnetic particle imaging shows that the presented method is very promising in order to obtain edge-preserving image reconstructions in which ringing artifacts are reduced.

**Key words.** meshless approximation of discontinuous functions, radial basis function (RBF) interpolation, variably scaled discontinuous kernels (VSDKs), Gibbs phenomenon, segmentation and classification with kernel machines, magnetic particle imaging (MPI)

**AMS subject classifications.** 41A05, 41A25, A1A30, 65D05

**DOI.** 10.1137/19M1248777

**1. Introduction.** Data interpolation is an essential tool in medical imaging. It is required for geometric alignment, registration of images, enhancing the quality on display devices, or reconstructing the image from a compressed amount of data [7, 26, 42]. Interpolation techniques are needed in the generation of images as well as in postprocessing steps. In medical inverse problems as computerized tomography (CT) and magnetic resonance imaging (MRI), interpolation is used in the reconstruction process in order to fit the discrete Radon data into the back-projection step. In single-photon emission computed tomography, regridding the projection data improves the reconstruction quality while reducing acquisition times [41]. In magnetic particle imaging (MPI), the number of calibration measurements can be reduced by interpolation methods [23].

In a general interpolation framework, we are given a finite number of data values sampled from an unknown function  $f$  on a node set  $\mathcal{X} \in \Omega$ ,  $\Omega \subseteq \mathbb{R}^d$ . The goal of

\*Submitted to the journal's Computational Methods in Science and Engineering section March 8, 2019; accepted for publication (in revised form) January 21, 2020; published electronically March 30, 2020.

<https://doi.org/10.1137/19M1248777>

**Funding:** This work was partially supported by GNCS-INdAM and by the European Union's Horizon 2020 research and innovation program ERA-PLANET, grant agreement 689443, via the GEOEssential project.

<sup>†</sup>Dipartimento di Matematica “Tullio Levi-Civita,” Università di Padova, Padova 35121, Italy (demarchi@math.unipd.it, erb@math.unipd.it).

<sup>‡</sup>Dipartimento di Salute della Donna e del Bambino, Università di Padova, Padova 35128, Italy (francesco.marchetti.1@phd.unipd.it).

<sup>§</sup>Dipartimento di Matematica, Università di Genova, Genova 16146, Italy (perracchione@dima.unige.it).

<sup>¶</sup>Dipartimento di Matematica e Applicazioni, Università di Milano–Bicocca, Milano 20125, Italy (milvia.rossini@unimib.it).

every interpolation scheme is to recover, in a faithful way, the function  $f$  on the entire domain  $\Omega$  or on a set of evaluation points. The choice of the interpolation model plays a crucial role for the quality of the reconstruction. If the function  $f$  belongs to the interpolation space itself,  $f$  can be recovered exactly. On the other hand, if the basis of the interpolation space does not reflect the properties of  $f$ , artifacts will usually appear in the reconstruction. In two-dimensional images such artifacts occur, for instance, if the function  $f$  describing the image has sharp edges, i.e., discontinuities across curves in the domain  $\Omega$ . In this case, smooth interpolants get highly oscillatory near the discontinuity points.

This is a typical example of the so-called Gibbs phenomenon. This phenomenon was originally formulated in terms of overshoots that arise when univariate functions with jump discontinuities are approximated by truncated Fourier expansions; see [45]. Similar artifacts arise also in higher-dimensional Fourier expansions and when interpolation operators are used. In medical imaging like CT and MRI, such effects are also known as ringing or truncation artifacts [9].

The Gibbs phenomenon is also a well-known issue for other basis systems like wavelets or splines; see [20] for a general overview. Further, it appears also in the context of radial basis function (RBF) interpolation [17]. The effects of the phenomenon can usually be softened by applying additional smoothing filters to the interpolant. For RBF methods, one can, for instance, use linear RBFs in regions around discontinuities [21]. Furthermore, postprocessing techniques, such as the Gegenbauer reconstruction procedure [19] or digital total variation [33], are available.

**Main contributions.** In this work, we study a shape-driven method to interpolate *scattered* data sampled from discontinuous functions. More precisely, we consider variably scaled kernels (VSKs) [4, 31] to model edges (known or estimated) of a function. Starting from a classical kernel  $K$ , it is possible to define a basis that reflects discontinuities in the data. These basis functions, referred to as variably scaled discontinuous kernels (VSDKs) [11], strictly depend on the given data and provide tailored interpolants to avoid overshoots near the edges of the underlying function. The goal of this work is to present a theoretical analysis of the interpolation spaces and the approximation errors related to these discontinuous basis functions. In the literature, the usage of discontinuous kernels for approximation purposes is rather rare. An alternative construction of discontinuous kernels based on a Mercer decomposition is given in [35].

If the edges of the function  $f$  are explicitly known, we show in this work that the given interpolation model outperforms the classical RBF interpolation and avoids Gibbs artifacts. From the theoretical point of view, we provide two main results. If the kernel  $K$  is algebraically decaying in the Fourier domain, we characterize the native space of the VSDK as a piecewise Sobolev space. This description allows us to derive in a second step convergence rates of the discontinuous interpolation scheme in terms of a given fill distance for the node set in the domain  $\Omega$ . The VSDK convergence rates are significantly better than the convergence rates for standard RBF interpolation. Numerical experiments confirm the theoretical results and indicate that even better rates are possible if the kernel  $K$  involved in the definition of the VSDK is analytic.

In applied problems, such as medical imaging, the edges of  $f$  are usually not a priori known. In this case, we need reliable edge detection or image segmentation algorithms that estimate the position of the edges from the given data. For this reason, we encode in the interpolation method an additional segmentation process based on a classification algorithm that provides the edges via a kernel machine. As labels for the

classification algorithm, we can use thresholds based on function values or on RBF coefficients [30]. The main advantage of this type of edge extraction process is that it works directly for scattered data, in contrast to other edge detection schemes, such as Canny or Sobel detectors, that usually require an underlying grid (cf. [8, 37]).

**Outline.** In section 2, we recall the basic notions on kernel-based interpolation. Then in section 3, we present the theoretical findings on the characterization of the VSDK native spaces (if the discontinuities are known) and Sobolev-type error estimates of the corresponding interpolation scheme. In section 4, numerical experiments underpin the theoretical results and reveal that if the edges are known, the VSDK interpolant outperforms the classical RBF interpolation. We show how the VSDK interpolant behaves with respect to jump parameters and spatial perturbations of the scaling function that models the discontinuities. We review image segmentation via classification and machine learning tools in section 5 and summarize our new approach. In section 6, the novel VSDK interpolation technique incorporating the segmentation algorithm is applied to MPI. Conclusions are drawn in section 7.

**2. Preliminaries on kernel-based interpolation and VSKs.** Kernel-based methods are powerful tools for scattered data interpolation. In the following, we give a brief overview over the basic terminology. For the theoretical background and more details on kernel methods, we refer the reader to [6, 16, 44].

**2.1. Kernel-based interpolation.** For a given set of scattered nodes  $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\} \subseteq \Omega$ ,  $\Omega \subseteq \mathbb{R}^d$ , and values  $f_i \in \mathbb{R}$ ,  $i \in \{1, \dots, N\}$ , we want to find a function  $P_f : \Omega \rightarrow \mathbb{R}$  that satisfies the interpolation conditions

$$(2.1) \quad P_f(\mathbf{x}_i) = f_i, \quad i \in \{1, \dots, N\}.$$

We express the interpolant  $P_f$  in terms of a kernel  $K : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ , i.e.,

$$(2.2) \quad P_f(\mathbf{x}) = \sum_{k=1}^N c_k K(\mathbf{x}, \mathbf{x}_k), \quad \mathbf{x} \in \Omega.$$

If the kernel  $K$  is symmetric and strictly positive definite, the matrix  $A = (A_{ij})$  with the entries  $A_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$ ,  $1 \leq i, j \leq N$ , is positive definite for all possible sets of nodes. In this case, the coefficients  $c_k$  are uniquely determined by the interpolation conditions in (2.1) and can be obtained by solving the linear system  $A\mathbf{c} = \mathbf{f}$ , where  $\mathbf{c} = (c_1, \dots, c_N)^\top$  and  $\mathbf{f} = (f_1, \dots, f_N)^\top$ .

Moreover, there exists a so-called native space for the kernel  $K$  that is a Hilbert space  $\mathcal{N}_K(\Omega)$  with inner product  $(\cdot, \cdot)_{\mathcal{N}_K(\Omega)}$ , in which the kernel  $K$  is reproducing; i.e., for any  $f \in \mathcal{N}_K(\Omega)$  we have the identity

$$f(\mathbf{x}) = (f, K(\cdot, \mathbf{x}))_{\mathcal{N}_K(\Omega)}, \quad \mathbf{x} \in \Omega.$$

Following [44], we introduce the native space by defining the space

$$H_K(\Omega) = \text{span} \{K(\cdot, \mathbf{y}), \quad \mathbf{y} \in \Omega\}$$

equipped with the bilinear form

$$(2.3) \quad (f, g)_{H_K(\Omega)} = \sum_{i=1}^N \sum_{j=1}^M a_i b_j K(\mathbf{x}_i, \mathbf{y}_j),$$

where  $f, g \in H_K(\Omega)$  with  $f(\mathbf{x}) = \sum_{i=1}^N a_i K(\mathbf{x}, \mathbf{x}_i)$  and  $g(\mathbf{x}) = \sum_{j=1}^M b_j K(\mathbf{x}, \mathbf{y}_j)$ . The

space  $(f, g)_{H_K(\Omega)}$  equipped with  $(f, g)_{H_K(\Omega)}$  is an inner product space with reproducing kernel  $K$  (see [44, Theorem 10.7]). The native space  $\mathcal{N}_K(\Omega)$  of the kernel  $K$  is then defined as the completion of  $H_K(\Omega)$  with respect to the norm  $\|\cdot\|_{H_K(\Omega)} = \sqrt{(\cdot, \cdot)_{H_K(\Omega)}}$ . In particular, for all  $f \in H_K(\Omega)$  we have  $\|f\|_{\mathcal{N}_K(\Omega)} = \|f\|_{H_K(\Omega)}$ .

In this paper we consider radial kernels  $K : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$  of the form

$$(2.4) \quad K(\mathbf{x}, \mathbf{y}) = \phi(\|\mathbf{x} - \mathbf{y}\|_2), \quad \mathbf{x}, \mathbf{y} \in \Omega,$$

based on a continuous scalar function  $\phi : [0, \infty) \rightarrow \mathbb{R}$ . The function  $\phi$  is usually called RBF.

**2.2. VSKs.** VSKs were introduced in [4]. They depend on a *scaling* function  $\psi : \mathbb{R}^d \rightarrow \mathbb{R}$ .

**DEFINITION 2.1.** Let  $K : \mathbb{R}^{d+1} \times \mathbb{R}^{d+1} \rightarrow \mathbb{R}$  be a continuous strictly positive definite kernel. Given a scaling function  $\psi : \mathbb{R}^d \rightarrow \mathbb{R}$ , a VSK  $K_\psi$  on  $\mathbb{R}^d \times \mathbb{R}^d$  is defined as

$$(2.5) \quad K_\psi(\mathbf{x}, \mathbf{y}) := K((\mathbf{x}, \psi(\mathbf{x})), (\mathbf{y}, \psi(\mathbf{y})))$$

for  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ .

The so-given VSK  $K_\psi$  is strictly positive definite on  $\mathbb{R}^d$ . Suitable choices of the scaling function  $\psi$  allow improving stability and recovery quality of the kernel-based interpolation as well as preserving shape properties of the original function; see, e.g., the examples in [4], [11], and [31]. As we assume that the kernel  $K$  in  $\mathbb{R}^{d+1}$  is defined in terms of an RBF  $\phi$ , the VSK has the form

$$(2.6) \quad K_\psi(\mathbf{x}, \mathbf{y}) = \phi\left(\sqrt{\|\mathbf{x} - \mathbf{y}\|_2^2 + |\psi(\mathbf{x}) - \psi(\mathbf{y})|^2}\right).$$

**2.3. VSKs with discontinuities.** Our goal is to introduce interpolation spaces based on discontinuous basis functions on  $\Omega$ . For the definition of these spaces, we use a piecewise continuous scaling function  $\psi$ . The associated VSK  $K_\psi(\mathbf{x}, \mathbf{y})$  is then also only piecewise continuous and denoted as VSDK.

We consider the following setting.

*Assumption 2.2.* We assume the following:

- (i) The bounded set  $\Omega \subset \mathbb{R}^d$  is the union of  $n$  pairwise disjoint sets  $\Omega_i$ ,  $i \in \{1, \dots, n\}$ .
- (ii) The subsets  $\Omega_i$  have a Lipschitz boundary.
- (iii) Let  $\Sigma = \{\alpha_1, \dots, \alpha_n\}$ ,  $\alpha_i \in \mathbb{R}$ . The function  $\psi : \Omega \rightarrow \Sigma$  is piecewise constant so that  $\psi(\mathbf{x}) = \alpha_i$  for all  $\mathbf{x} \in \Omega_i$ . In particular, the discontinuities of  $\psi$  appear only at the boundaries of the subsets  $\Omega_i$ . We assume that  $\alpha_i \neq \alpha_j$  if  $\Omega_i$  and  $\Omega_j$  are neighboring sets.

We point out that the Lipschitz condition in Assumption 2.2(ii) implies an interior cone condition for the subdomains  $\Omega_i$  (cf. [1, page 84]). This is an important prerequisite, in particular for Proposition 3.3 below. The kernel function  $K_\psi(\mathbf{x}, \mathbf{y})$  based on a piecewise constant scaling function  $\psi$  is well defined for all  $\mathbf{x}, \mathbf{y} \in \Omega$ . If  $\mathbf{x}$  and  $\mathbf{y}$  are contained in the same subset  $\Omega_i \subset \Omega$ , then  $K_\psi(\mathbf{x}, \mathbf{y}) = \phi(\|\mathbf{x} - \mathbf{y}\|_2)$ ; i.e., inside the subdomains  $\Omega_i$ ,  $K_\psi$  can be seen as an ordinary RBF kernel based on the

RBF  $\phi$ . We denote the graph of the function  $\psi$  with respect to the domain  $\Omega$  by

$$G_\psi(\Omega) = \{(\mathbf{x}, \psi(\mathbf{x})) \mid \mathbf{x} \in \Omega\} \subset \Omega \times \Sigma.$$

To have a more compact notation for the elements of the graph  $G_\psi(\Omega)$ , we use the shortcuts  $\tilde{\mathbf{x}} = (\mathbf{x}, \psi(\mathbf{x}))$  and  $\tilde{\mathbf{y}} = (\mathbf{y}, \psi(\mathbf{y}))$ . In the same way as in (2.2), we can define an interpolant for the nodes  $\tilde{\mathcal{X}} = \{\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_N\}$  on the graph  $G_\psi(\Omega)$ . Using the kernel  $K$  on  $G_\psi(\Omega) \subset \Omega \times \Sigma \subset \mathbb{R}^{d+1}$ , we obtain an interpolant of the form

$$(2.7) \quad P_f(\tilde{\mathbf{x}}) = \sum_{k=1}^N c_k K(\tilde{\mathbf{x}}, \tilde{\mathbf{x}}_k).$$

Based on this interpolant on  $G_\psi(\Omega)$ , we define the VSDK interpolant  $V_f$  on  $\Omega$  as

$$(2.8) \quad V_f(\mathbf{x}) = P_f(\tilde{\mathbf{x}}) = \sum_{k=1}^N c_k K_\psi(\mathbf{x}, \mathbf{x}_k), \quad \mathbf{x} \in \Omega.$$

The coefficients  $c_1, \dots, c_N$  of the VSDK interpolant  $V_f(\mathbf{x})$  in (2.8) are obtained by solving the linear system of equations

$$(2.9) \quad \begin{pmatrix} K(\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_1) & \cdots & K(\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_N) \\ \vdots & & \vdots \\ K(\tilde{\mathbf{x}}_N, \tilde{\mathbf{x}}_1) & \cdots & K(\tilde{\mathbf{x}}_N, \tilde{\mathbf{x}}_N) \end{pmatrix} \begin{pmatrix} c_1 \\ \vdots \\ c_N \end{pmatrix} = \begin{pmatrix} f_1 \\ \vdots \\ f_N \end{pmatrix}.$$

In fact, the so-obtained coefficients are precisely the coefficients for the interpolant (2.7) for the node points  $\tilde{\mathcal{X}}$  on the graph  $G_\psi(\Omega)$ . Since the kernel  $K : G_\psi(\Omega) \times G_\psi(\Omega) \rightarrow \mathbb{R}$  is strictly positive definite, the system (2.9) admits a unique solution. For the kernel  $K : G_\psi(\Omega) \times G_\psi(\Omega) \rightarrow \mathbb{R}$  and the discontinuous kernel  $K_\psi : \Omega \times \Omega \rightarrow \mathbb{R}$  we can further define the two inner product spaces

$$\begin{aligned} H_K(G_\psi(\Omega)) &= \text{span} \{K(\cdot, \tilde{\mathbf{y}}), \quad \tilde{\mathbf{y}} \in G_\psi(\Omega)\}, \\ H_{K_\psi}(\Omega) &= \text{span} \{K_\psi(\cdot, \mathbf{y}), \quad \mathbf{y} \in \Omega\}, \end{aligned}$$

with the inner products given as in (2.3). For both spaces, we can take the completion and obtain in this way the native spaces  $\mathcal{N}_K(G_\psi(\Omega))$  and  $\mathcal{N}_{K_\psi}(\Omega)$ , respectively. We have the following relation between the two native spaces.

**PROPOSITION 2.3.** *The native spaces  $\mathcal{N}_K(G_\psi(\Omega))$  and  $\mathcal{N}_{K_\psi}(\Omega)$  are isometrically isomorphic.*

In the same way as in [4, Theorem 2], Proposition 2.3 follows from the fact that the two inner product spaces  $H_K(G_\psi(\Omega))$  and  $H_{K_\psi}(\Omega)$  are isometric. Then the same holds true for their respective completion.

### 3. Approximation with discontinuous kernels.

**3.1. Characterization of the native space for VSDKs.** Based on the decomposition of the domain  $\Omega$  described in Assumption 2.2 we define for  $s \geq 0$  and  $1 \leq p \leq \infty$  the following spaces of piecewise smooth functions on  $\Omega$ :

$$\text{WP}_p^s(\Omega) := \{f : \Omega \rightarrow \mathbb{R} \mid f_{\Omega_i} \in \text{W}_p^s(\Omega_i), \quad i \in \{1, \dots, n\}\}.$$

Here,  $f_{\Omega_i}$  denotes the restriction of  $f$  to the subregion  $\Omega_i$ , and  $W_p^s(\Omega_i)$  denote the standard Sobolev spaces on  $\Omega_i$ . As norm on  $WP_p^s(\Omega)$  we set

$$\|f\|_{WP_p^s(\Omega)}^p = \sum_{i=1}^n \|f_{\Omega_i}\|_{W_p^s(\Omega_i)}^p.$$

The piecewise Sobolev space  $WP_p^s(\Omega)$  and the corresponding norm strongly depend on the chosen decomposition of the domain  $\Omega$ . However, for any decomposition of  $\Omega$  in Assumption 2.2, the standard Sobolev space  $W_p^s(\Omega)$  is contained in  $WP_p^s(\Omega)$ . In the following, we assume that the radial kernel  $K$  defining the VSDK  $K_\psi$  has a particular Fourier decay:

$$(3.1) \quad \widehat{\phi(\|\cdot\|)}(\omega) \sim (1 + \|\omega\|_2^2)^{-s-\frac{1}{2}}, \quad s > \frac{d-1}{2}.$$

In order to characterize the native space  $\mathcal{N}_{K_\psi}(\Omega)$ , we need some additional results regarding the continuity of trace and extension operators. The following Lemma 3.1 summarizes some relevant facts from the literature and adapts them to the scope of this article.

LEMMA 3.1. *We have the following relations for extension and trace operators in the native spaces (a) and in the Sobolev spaces (b):*

(a) *Every  $f \in \mathcal{N}_K(G_\psi(\Omega))$  has a natural extension  $Ef \in \mathcal{N}_K(\mathbb{R}^{d+1})$ . Further,*

$$\|Ef\|_{\mathcal{N}_K(\mathbb{R}^{d+1})} = \|f\|_{\mathcal{N}_K(G_\psi(\Omega))}.$$

*For every  $g \in \mathcal{N}_K(\mathbb{R}^{d+1})$ , the trace  $T_{G_\psi(\Omega)}g$  is contained in  $\mathcal{N}_K(G_\psi(\Omega))$ . Further,*

$$\|T_{G_\psi(\Omega)}g\|_{\mathcal{N}_K(G_\psi(\Omega))} \leq \|g\|_{\mathcal{N}_K(\mathbb{R}^{d+1})}.$$

(b) *Let  $s > \frac{1}{2}$ . For every  $g \in W_2^s(\mathbb{R}^{d+1})$ , the trace  $T_{G_\psi(\Omega)}g$  is contained in  $W_2^{s-1/2}(G_\psi(\Omega))$ , and the trace operator  $T_{G_\psi(\Omega)} : W_2^s(\mathbb{R}^{d+1}) \rightarrow W_2^{s-1/2}(G_\psi(\Omega))$  is bounded.*

*Further, there exists a bounded extension operator  $E : W_2^{s-1/2}(G_\psi(\Omega)) \rightarrow W_2^s(\mathbb{R}^{d+1})$  such that  $T_{G_\psi(\Omega)}Ef = f$  for all  $f \in W_2^{s-1/2}(G_\psi(\Omega))$ .*

*Proof.* (a) The proof can be found in [44, Theorems 10.46 and 10.47] or [34, section 9].

(b) This statement is as well mainly based on known results from the literature. We consider the sets  $G_{\alpha_i}(\Omega_i) = \Omega_i \times \{\alpha_i\}$  as subsets of the hyperplane  $\mathbb{R}^d \times \{\alpha_i\} \subset \mathbb{R}^{d+1}$ . As we assume that  $\Omega_i$  has a Lipschitz boundary, there exists a continuous extension from  $W_2^{s-1/2}(G_{\alpha_i}(\Omega_i))$  to  $W_2^{s-1/2}(\mathbb{R}^d \times \{\alpha_i\})$  [32] (this is known as the Calderón extension). As also the corresponding restriction operator is bounded, the trace theorem (see [43, section 2.7.2]) guarantees that statement (b) of lemma holds true with  $G_\psi(\Omega)$  replaced by  $G_{\alpha_i}(\Omega_i)$ . As  $G_\psi(\Omega)$  is the disjoint union of the separated sets  $G_{\alpha_i}(\Omega_i)$ ,  $i \in \{1, \dots, n\}$  (by Assumption 2.2(iii)), we can finally use a partition of unity argument to obtain (b). Note that the results in [32] and [43] are formulated in terms of general Besov and Triebel–Lizorkin spaces. The corresponding results for the simpler Sobolev spaces  $W_2^s(\mathbb{R}^{d+1})$  are therein obtained with  $p = q = 2$ . For  $s \in \mathbb{N}$ , simplified versions of the Calderón extension theorem and the trace theorem are given in [1, Theorems 5.28 and 7.39].  $\square$

We are now ready to prove the following theorem.

**THEOREM 3.2.** *Let Assumption 2.2 hold true, and assume that the continuous strictly positive definite kernel  $K : \mathbb{R}^{d+1} \times \mathbb{R}^{d+1} \rightarrow \mathbb{R}$  based on the RBF  $\phi$  satisfies the decay condition (3.1). Then, for the discontinuous kernel  $K_\psi$ , we have*

$$\mathcal{N}_{K_\psi}(\Omega) = \text{WP}_2^s(\Omega),$$

with the norms of the two Hilbert spaces being equivalent.

*Proof.* We consider the following forward and backward chain of Hilbert space operators:

$$\mathcal{N}_{K_\psi}(\Omega) \xrightleftharpoons[Q_1]{P_5} \mathcal{N}_K(G_\psi(\Omega)) \xrightleftharpoons[Q_2]{P_4} \mathcal{N}_K(\mathbb{R}^{d+1}) \xrightleftharpoons[Q_3]{P_3} \text{W}_2^{s+1/2}(\mathbb{R}^{d+1}) \xrightleftharpoons[Q_4]{P_2} \text{W}_2^s(G_\psi(\Omega)) \xrightleftharpoons[Q_5]{P_1} \text{WP}_2^s(\Omega).$$

In the backward direction, the operators  $P_1, \dots, P_5$  are given as

$$\begin{aligned} P_1 : P_1 f &= \tilde{f}, \quad \text{with } \tilde{f}(\tilde{\mathbf{x}}) = f(\mathbf{x}), \quad \text{for all } \mathbf{x} \in \Omega, \\ P_2 : P_2 f &= Ef, \quad (\text{Extension in the sense of Lemma 3.1 (b)}) \\ P_3 : P_3 f &= f, \\ P_4 : P_4 f &= T_{G_\psi(\Omega)} f, \quad (\text{Trace in the sense of Lemma 3.1 (a)}) \\ P_5 : P_5 \tilde{f} &= f, \quad \text{with } f(\mathbf{x}) = \tilde{f}(\tilde{\mathbf{x}}), \quad \text{for all } \mathbf{x} \in \Omega. \end{aligned}$$

In the forward chain, the operators  $Q_1, \dots, Q_5$  are similarly defined as

$$\begin{aligned} Q_1 : Q_1 f &= \tilde{f}, \quad \text{with } \tilde{f}(\tilde{\mathbf{x}}) = f(\mathbf{x}), \quad \text{for all } \mathbf{x} \in \Omega, \\ Q_2 : Q_2 f &= Ef, \quad (\text{Extension in the sense of Lemma 3.1 (a)}) \\ Q_3 : Q_3 f &= f, \\ Q_4 : Q_4 f &= T_{G_\psi(\Omega)} f, \quad (\text{Trace in the sense of Lemma 3.1 (b)}) \\ Q_5 : Q_5 \tilde{f} &= f, \quad \text{with } f(\mathbf{x}) = \tilde{f}(\tilde{\mathbf{x}}), \quad \text{for all } \mathbf{x} \in \Omega. \end{aligned}$$

All these 10 operators are well defined and continuous:  $P_5$  and  $Q_1$  are isometries by Proposition 2.3.  $P_2$ ,  $Q_2$ , as well as  $P_4$ ,  $Q_4$ , are continuous by Lemma 3.1. Since  $K$  satisfies the condition (3.1), the native space  $\mathcal{N}_K(\mathbb{R}^{d+1})$  is equivalent to the Sobolev space  $\text{W}_2^s(\mathbb{R}^{d+1})$  (see [44, Corollary 10.48]). Therefore, also the identity mappings  $P_3$  and  $Q_3$  are continuous. Finally, the Sobolev norm for a function  $\tilde{f}$  on the graph  $G_\psi(\Omega)$  is a reformulation of the norm of  $f \in \text{WP}_2^s(\Omega)$ . Therefore, also the operators  $P_1$  and  $Q_5$  are isometries.

We can conclude that the concatenations  $P_5 P_4 P_3 P_2 P_1$  and  $Q_5 Q_4 Q_3 Q_2 Q_1$  are continuous operators. Since  $P_5 P_4 P_3 P_2 P_1$  is the inverse to  $Q_5 Q_4 Q_3 Q_2 Q_1$ , these two operators therefore provide an isomorphism between the Hilbert spaces  $\mathcal{N}_{K_\psi}(\Omega)$  and  $\text{WP}_2^s(\Omega)$ .  $\square$

**3.2. Error estimates for VSDK interpolation.** We state first of all a well-known Sobolev sampling inequality for functions vanishing on the subsets  $\mathcal{X} \cap \Omega_i$  which was developed in [28]. For this we introduce the following regional fill distance  $h_i$  on the subset  $\Omega_i$ :

$$h_i = \sup_{\mathbf{x} \in \Omega_i} \inf_{\mathbf{x}_i \in \mathcal{X} \cap \Omega_i} \|\mathbf{x} - \mathbf{x}_i\|_2.$$

**PROPOSITION 3.3** ([28, Theorem 2.12] or [15, Proposition 9]). *Let  $s > 0$  as well as  $1 \leq p < \infty$  and  $1 \leq q \leq \infty$ . Further, let  $m \in \mathbb{N}_0$  such that  $\lfloor s \rfloor > m + d/p$  (for  $p = 1$*

also equality is possible) and  $u$  be a function that vanishes on  $\mathcal{X} \cap \Omega_i$ . Then there is a  $h_0 > 0$  such that for  $h_i \leq h_0$  and for the subregions  $\Omega_i$  satisfying Assumption 2.2(ii) we have the Sobolev inequality

$$\|u\|_{W_q^m(\Omega_i)} \leq C_i h_i^{s-m-d(1/p-1/q)+} \|u\|_{W_p^s(\Omega_i)}.$$

The constant  $C_i > 0$  is independent of  $h_i$ .

The value  $h_0$  in Proposition 3.3 depends on the Sobolev regularity  $s$  and on the interior cone condition of the subsets  $\Omega_i$ ; cf. [28]. As we only have  $n$  different subsets  $\Omega_i$ , we can find a single  $h_0$  for all  $i \in \{1, \dots, n\}$ . The Sobolev sampling inequalities given in Proposition 3.3 allow us to extract the correct power of the fill distance from the smoothness of the underlying error function. Based on these inequalities, a similar analysis can be conducted also on manifolds; see [15]. Further sampling inequalities that can be used as a substitute for Proposition 3.3 can, for instance, be found in [29, Theorem 2.1.1].

We define now the global fill distance

$$h = \max_{i \in \{1, \dots, n\}} h_i$$

and get as a consequence of the regional sampling inequalities in Proposition 3.3 the following Sobolev error estimate.

**THEOREM 3.4.** *Let Assumption 2.2 be satisfied. Further, let  $s > 0$ ,  $1 \leq q \leq \infty$  and  $m \in \mathbb{N}_0$  such that  $\lfloor s \rfloor > m + \frac{d}{2}$ . Additionally, suppose that the RBF  $\phi$  satisfies the Fourier decay (3.1). Then, for  $f \in \text{WP}_2^s(\Omega)$ , we obtain for all  $h \leq h_0$  the error estimate*

$$(3.2) \quad \|f - V_f\|_{\text{WP}_q^m(\Omega)} \leq C h^{s-m-d(1/2-1/q)+} \|f\|_{\text{WP}_2^s(\Omega)}.$$

The constant  $C > 0$  is independent of  $h$ .

*Proof.* By assumption, the function  $f$  is an element of  $\text{WP}_2^s(\Omega)$ . Further, the native space characterization in Theorem 3.2 guarantees that also the VSDK interpolant  $V_f$  is an element of  $\text{WP}_2^s(\Omega)$ . Therefore, we can apply Proposition 3.3 with  $p = 2$  to every subset  $\Omega_i$  and obtain

$$\|f - V_f\|_{W_q^m(\Omega_i)} \leq C_i h_i^{s-m-d(1/2-1/q)+} \|f - V_f\|_{W_2^s(\Omega_i)}, \quad i \in \{1, \dots, n\},$$

with  $h_i \leq h_0$ . Now, using the definition of the piecewise Sobolev space  $\text{WP}_q^m(\Omega)$ , we can synthesize these estimates to obtain

$$(3.3) \quad \|f - V_f\|_{\text{WP}_q^m(\Omega)} \leq C_{\max} h^{s-m-d(1/2-1/q)+} \|f - V_f\|_{\text{WP}_2^s(\Omega)},$$

where  $C_{\max} = \max_{1 \leq i \leq n} C_i$  and  $h = \max_{1 \leq i \leq n} h_i$ . Since  $\text{WP}_2^s(\Omega)$  is equivalent to the native space  $\mathcal{N}_{K_\psi}(\Omega)$ , we can use the fact that the interpolant  $V_f$  is a projection into a subspace of  $\mathcal{N}_{K_\psi}(\Omega)$ . This helps us finalize our bound,

$$\|f - V_f\|_{\text{WP}_2^s} \leq C' \|f - V_f\|_{\mathcal{N}_{K_\psi}(\Omega)} \leq C' \|f\|_{\mathcal{N}_{K_\psi}(\Omega)} \leq C' C'' \|f\|_{\text{WP}_2^s(\Omega)},$$

with the two constants  $C'$ ,  $C''$  describing the bounds for the equivalence of the two Hilbert space norms. The constant  $C$  in (3.2) is then given by  $C = C' C'' C_{\max}$ .  $\square$



*Remark 3.5.* The error estimates in Theorem 3.4 provide a theoretical explanation why VSDK interpolation is superior to RBF interpolation in the spaces  $WP_2^s(\Omega)$ . In these spaces the convergence of the interpolant  $V_f$  toward  $f \in WP_2^s(\Omega)$  depends only on the smoothness  $s$  of  $f$  in the interior of the subsets  $\Omega_i \subset \Omega$  and not on the discontinuities at the boundaries of  $\Omega_i$ . If  $s$  is sufficiently large, the corresponding fast convergence of the interpolation scheme prevents the emergence of Gibbs artifacts in the interpolant  $V_f$ .

#### 4. Numerical experiments.

**4.1. Experimental setup.** In our main application in MPI, we will use samples along Lissajous trajectories as interpolation nodes. For this, we will introduce and use these node sets already for the numerical experiments in this section. As test images we consider the Shepp–Logan phantom and an additional simple geometric phantom. We give a brief description of this experimental setup.

**4.1.1. Lissajous interpolation nodes.** For a vector  $\mathbf{n} = (n_1, n_2) \in \mathbb{N}^2$  with relatively prime frequencies  $n_1, n_2$  and  $\epsilon \in \{1, 2\}$ , the generating curves for the Lissajous nodes are given as

$$(4.1) \quad \gamma_\epsilon^{(\mathbf{n})}(t) = \left( \cos(n_2 t), \cos\left(n_1 t - \frac{\epsilon-1}{2n_2} \pi\right) \right).$$

The Lissajous curve  $\gamma_\epsilon^{(\mathbf{n})}$  is  $2\pi$ -periodic and contained in the square  $[-1, 1]^2$ . If  $\epsilon = 1$ , this curve is degenerate; i.e., it is traversed twice in one period. Further,  $\gamma_1^{(\mathbf{n})}$  with  $\mathbf{n} = (n, n+1)$ ,  $n \in \mathbb{N}$ , are the generating curves of the Padua points [2, 13]. If  $\epsilon = 2$ , then the curve is nondegenerate. If  $n_1 + n_2$  is odd, the curve  $\gamma_2^{(\mathbf{n})}(t)$  in (4.1) can be further simplified in terms of two sine functions and gives a typical sampling trajectory encountered in MPI; see [12, 14, 24, 25]. Using  $\gamma_\epsilon^{(\mathbf{n})}$  as generating curves, we introduce the Lissajous nodes as the sampling points

$$(4.2) \quad \mathbf{LS}_\epsilon^{(\mathbf{n})} = \left\{ \gamma_\epsilon^{(\mathbf{n})} \left( \frac{\pi k}{\epsilon n_1 n_2} \right), \quad k = 0, \dots, 2\epsilon n_1 n_2 - 1 \right\}.$$

In our upcoming tests, we will use the points  $\mathbf{LS}_2^{(\mathbf{n})}$ , with  $n_1, n_2$  relatively prime and  $n_1 + n_2$  odd as underlying interpolation nodes. These node sets were already used in [10, 14, 23] for applications in MPI. The number of points is given by  $\#\mathbf{LS}_2^{(\mathbf{n})} = 2n_1 n_2 + n_1 + n_2$ ; see [12, 14]. The fill distance

$$h_{\mathbf{LS}_2^{(\mathbf{n})}} = \max_{\mathbf{y} \in [-1, 1]^2} \min_{\mathbf{x} \in \mathbf{LS}_2^{(\mathbf{n})}} \|\mathbf{x} - \mathbf{y}\|_2$$

for the nodes  $\mathbf{LS}_2^{(\mathbf{n})}$  in the square  $[-1, 1]^2$  can be computed as

$$(4.3) \quad h_{\mathbf{LS}_2^{(\mathbf{n})}} = \frac{1}{2} \max \left\{ \sqrt{S_{n_1}^2 + \left( \frac{S_{2n_1}^2 + S_{2n_2}^2 - S_{n_1} S_{2n_1}}{S_{2n_2}} \right)^2}, \sqrt{S_{n_2}^2 + \left( \frac{S_{2n_1}^2 + S_{2n_2}^2 - S_{n_2} S_{2n_2}}{S_{2n_1}} \right)^2} \right\}$$

by using the shortcut  $S_n = \sin(\pi/n)$ . This allows us to express the fill distance of  $\mathbf{LS}_2^{(\mathbf{n})}$  directly in terms of the frequency parameters  $n_1$  and  $n_2$ . Further, we have the estimates

$$\frac{1}{2} \max \{S_{n_1}, S_{n_2}\} \leq h_{\mathbf{LS}_2^{(\mathbf{n})}} \leq \max \{S_{2n_1}, S_{2n_2}\} \leq \max \left\{ \frac{\pi}{2n_1}, \frac{\pi}{2n_2} \right\}.$$

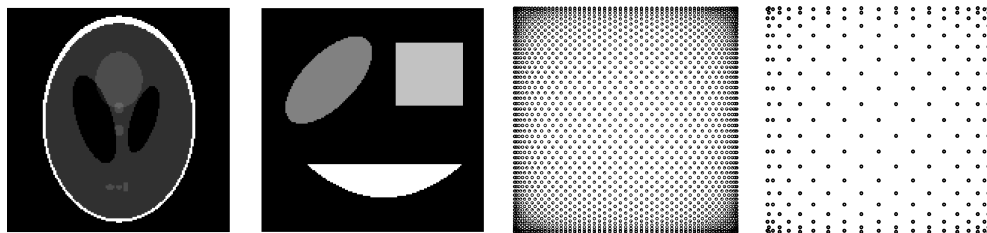


FIG. 1. The Shepp–Logan phantom (left), a geometric phantom (middle left), as well as the Lissajous nodes  $\mathbf{LS}_2^{(33,32)}$  (middle right) and  $\mathbf{LS}_2^{(10,11)}$  (right).

For  $\mathbf{n} = (33, 32)$  and  $\mathbf{n} = (10, 11)$ , the nodes  $\mathbf{LS}_2^{(\mathbf{n})}$  are illustrated in Figure 1 (right). Note that in the interpolation with the VSDKs, only the information at the nodes  $\mathbf{LS}_2^{(\mathbf{n})}$  is of interest, whereas the additional structure of the Lissajous trajectory itself is not relevant.

**4.1.2. Shepp–Logan phantom.** As a main test phantom with sharp edges, we use the Shepp–Logan phantom  $f_{\text{SL}}$  on  $\Omega = [-1, 1]^2$  as introduced in [38]. The function  $f_{\text{SL}} : [-1, 1]^2 \rightarrow [0, 1]$  is defined as a composition of 10 step functions determined by elliptic equations. A discretization of  $f_{\text{SL}}$  on an equidistant  $M \times M$  grid,  $M = 150$ , is displayed in Figure 1 (left).

**4.1.3. Geometric phantom.** As a second phantom we use a geometric composition  $f_G$  of an ellipse  $E = \{\mathbf{x} \mid 2(x_2 + x_1 + 0.05)^2 + 9(x_2 - x_1 - 0.75)^2 \leq 1\}$ , a rectangle  $R = \{\mathbf{x} \mid |x_1 - 0.5| \leq 0.3, |x_2 - 0.3| \leq 0.28\}$ , and a bounded parabola  $P = \{\mathbf{x} \mid x_2 \leq -0.4, x_2 + 0.7 \geq 0.6(x_1 - 0.1)^2\}$ , discretized on a  $M \times M$  grid of size  $M = 150$ . The function  $f_G$  on  $\Omega = [-1, 1]^2$  is given as  $f_G = \chi_E + 1.5\chi_R + 2\chi_P$ , where  $\chi_E$ ,  $\chi_R$  and  $\chi_P$  denote the characteristic functions of  $E$ ,  $R$ , and  $P$ , respectively. The phantom  $f_G$  is illustrated in Figure 1 (middle left).

**4.1.4. Kernels.** For RBF interpolation in  $\mathbb{R}^d$ , as well as for the VSDK interpolation scheme which requires a kernel in  $\mathbb{R}^{d+1}$ , we use the following RBFs (cf. [16, Tables 3.1 and 3.2]):

- (i) The  $C^0$ -Matérn function  $\phi_{\text{Mat},0}(r) = e^{-r}$ . The native space of the corresponding kernel is exactly the Sobolev space  $W_2^s(\mathbb{R}^d)$  with  $s = \frac{d+1}{2}$ . We have  $W_2^{\frac{d+1}{2}}(\mathbb{R}^d) \subset C^0(\mathbb{R}^d)$ .
- (ii) The  $C^2$ -Matérn function  $\phi_{\text{Mat},2}(r) = (1+r)e^{-r}$ . The native space of  $\phi_{\text{Mat},2}(\|\mathbf{x}\|_2)$  is the Sobolev space  $W_2^s(\mathbb{R}^d)$  with  $s = \frac{d+3}{2}$ . Functions in  $W_2^{\frac{d+3}{2}}(\mathbb{R}^d)$  are contained in  $C^1(\mathbb{R}^d)$ .
- (iii) The  $C^4$ -Matérn function  $\phi_{\text{Mat},4}(r) = (3 + 3r + r^2)e^{-r}$ . The radial function  $\phi_{\text{Mat},4}(\|\mathbf{x}\|_2)$  generates the Sobolev space  $W_2^s(\mathbb{R}^d)$  with  $s = \frac{d+5}{2}$ .  $W_2^{\frac{d+5}{2}}(\mathbb{R}^d)$  is contained in  $C^2(\mathbb{R}^d)$ .
- (iv) The Gauss function  $\phi_{\text{Gauss}}(r) = e^{-r^2}$ . This is an analytic function. The native space for the Gauss kernel is contained in every Sobolev space  $W_2^s(\mathbb{R}^d)$ ,  $s \geq 0$ .

With decreasing separation distance of the interpolation nodes, the calculation of the coefficients in (2.9) can be badly conditioned when solving the linear system directly. This is particularly the case when using the Gaussian as underlying kernel.

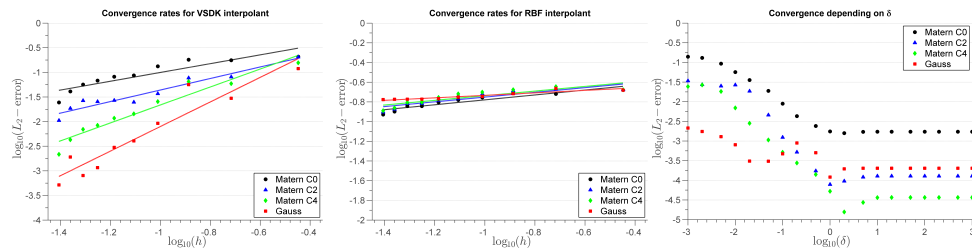


FIG. 2. Convergence rates for interpolating the Shepp–Logan phantom on the nodes  $\mathbf{LS}_2^{(n)}$  using the VSDK scheme with a priori known scaling function  $\psi_\delta$ ,  $\delta = 0.01$ , (left) and the RBF scheme (middle). The  $L_2$ -error for interpolation on the nodes  $\mathbf{LS}_2^{(33,32)}$  for increasing separation distance  $\delta$  is shown on the right-hand side.

In order to stabilize the calculation, we regularized the system (2.9) by adding a small multiple  $\lambda > 0$  of the identity to the interpolation matrix (we chose  $\lambda = 10^{-12}$  in our calculations). Note, however, that in the literature there exist more sophisticated ways to avoid this bad conditioning; see, for instance, [16, Chapters 11–13].

#### 4.2. Experiment 1—Convergence for a priori known discontinuities.

**4.2.1. Description.** For  $n \in \{4, 8, 12, \dots, 40\}$  we interpolate the Shepp–Logan phantom at the Lissajous nodes  $\mathbf{LS}_2^{(n+1,n)}$ . We use the four kernels introduced in section 4.1 and compute RBF as well as VSDK interpolants for all sets of Lissajous nodes. In a log-log diagram we plot the fill distance  $h = h_{\mathbf{LS}_2^{(n)}}$  given in (4.3) against the  $L_2$ -error between the original Shepp–Logan function  $f_{\text{SL}}$  and the interpolant. As an approximation of the continuous  $L_2$ -error we use the root-mean-square error on the equidistant  $M \times M$  discretization grids of the phantoms. The six possible function values  $f_i$ ,  $i \in \{0, \dots, 5\}$  of the Shepp–Logan phantom are given as  $\{0, 0.1, 0.2, 0.3, 0.4, 1\}$ . As a scaling function for the VSDK interpolant, we use

$$\psi_\delta(\mathbf{x}) = \delta f_i \quad \text{if } f_{\text{SL}}(\mathbf{x}) = f_i,$$

with the separation parameter  $\delta \geq 0$ . The scaling function  $\psi_\delta$  gives a  $\delta$ -dependent equidistant separation between the regions  $\Omega_i = \{\mathbf{x} \mid f_{\text{SL}}(\mathbf{x}) = f_i\}$ , in which the phantom is constant. The log-log diagrams for  $\delta = 0.01$  and  $\delta = 0$  (which corresponds to standard RBF interpolation) are given in Figure 2. In Figure 2 (right), we display the behavior of the interpolation error for increasing values of  $\delta$ . The slopes of the error regression lines for  $\delta \in \{0, 0.01, 0.1, 1\}$  are listed in Table 1. The VSDK reconstructions at the nodes  $\mathbf{LS}_2^{(33,32)}$  using the  $C^2$ -Matérn kernel are shown in Figure 3.

**4.2.2. Results and discussion.** This first numerical experiment underpins the theoretical estimates given in Theorem 3.4 and gives further insights into the impact of the scaling function  $\psi_\delta$  on the interpolation. In particular, the influence of the separation parameter  $\delta$  gets visible.

The slope of the regression line for the RBF interpolation (corresponding to  $\delta = 0$  in the VSDK scheme) is for all four kernels between 0.12 and 0.25. This is in line with the low-order smoothness of the Shepp–Logan phantom  $f_{\text{SL}}$  which is contained in the Sobolev space  $W_2^s([-1, 1]^2)$  only for  $s < 1/2$ . On the other hand, with the choice  $\psi_\delta$ ,  $\delta > 0$ , the Shepp–Logan phantom is contained in all piecewise Sobolev spaces  $WP_2^s([-1, 1]^2)$ ,  $s \geq 0$ . As can be seen in Figures 2 and 3 as well as in Table 1, the VSDK scheme is superior to the RBF scheme already for small values of  $\delta$ . For larger

TABLE 1

The slopes of the regression lines for the computed logarithmic  $L_2$ -errors. For the parameters  $\delta = 0.01$  and  $\delta = 0$  the corresponding log-log diagrams are displayed in Figure 2 (left) and (middle).

Kernel	Smoothness $s$ (theoretical slope)	Slope VSDK $\delta = 1$	Slope VSDK $\delta = 0.1$	Slope VSDK $\delta = 0.01$	Slope RBF $\delta = 0$
$C^0$ -Matérn	1.5	2.2569	1.9131	0.8933	0.2493
$C^2$ -Matérn	2.5	2.9859	2.5753	1.1715	0.2403
$C^4$ -Matérn	3.5	3.1380	2.5331	1.8237	0.2355
Gauss	analytic	3.3858	2.8668	2.4959	0.1260



FIG. 3. Reconstruction of the Shepp-Logan phantom for given sampling data at the nodes  $\mathbf{LS}_2^{(33,32)}$  using the VSDK schemes with scaling function  $\psi_\delta$ ,  $\delta \in \{1, 0.1, 0.01\}$  (three images on the left) and the RBF interpolation (right). For all schemes, the  $C^2$ -Matérn kernel is used.

separation values  $\delta$ , the quality of the interpolation increases even more. For  $\delta > 0$ , Theorem 3.4 predicts a theoretical convergence rate for the  $L_2$ -error in order of the smoothness  $s$  of the applied kernel. This impact of the smoothness is visible in the experimentally obtained slopes in Table 1 and Figure 2 but also a dependence on the separation parameter  $\delta$  of the scaling function. For small values  $\delta$ , the slopes for the calculated logarithmic errors are still moderate, while they get larger with increasing  $\delta$ . For large values  $\delta \gtrsim 10$ , the resulting VSDK interpolant stops depending on the parameter  $\delta$ . This dependence of the VSDK interpolation on  $\delta$  is at first glance not visible in the asymptotic error estimate of Theorem 3.4. Nevertheless, as the proof of Theorem 3.4 is based on the equivalence of the native space in Theorem 3.2, the constant  $C$  in the error estimate inherently depends on the scaling function  $\psi_\delta$  and thus on the parameter  $\delta$ .

Note that for a better comparison between the interpolation schemes we used the fill distance  $h_{\mathbf{LS}_2^{(n)}}$  given in (4.3) in our experiments, whereas the fill distance  $h$  used in Theorem 3.4 depends on the decomposition of the domain  $\Omega$  in the disjoint subsets  $\Omega_i$ . In general, we have  $h_{\mathbf{LS}_2^{(n)}} \leq h$ .

#### 4.3. Experiment 2—Perturbations of the scaling function.

**4.3.1. Description.** In the second computational experiment, we test the sensitivity of the VSDK interpolation with respect to shifts of the scaling function  $\psi$ . For this experiment, we consider the geometric phantom  $f_G$  and interpolate it with differing scaling functions at the Lissajous nodes  $\mathbf{LS}_2^{(10,11)}$  using the  $C^0$ -Matérn kernel. The corresponding reconstructions are displayed in Figure 4. Starting from the correct scaling function (left), i.e., using  $\psi = f_G$ , the rectangle in the scaling function  $\psi$  is slowly shifted toward the center (in Figure 4, from left to right). The corresponding interpolation errors with respect to the original function  $f_G$  are shown in Figure 5.

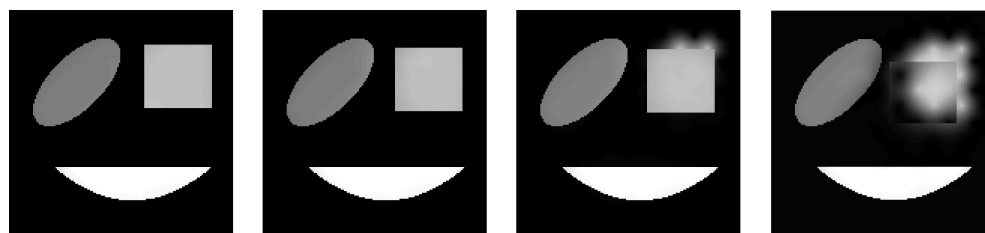


FIG. 4. Reconstruction of the geometric phantom by the VSDK scheme with differing scaling functions. The set of interpolation nodes is  $\mathbf{LS}_2^{(10,11)}$ .

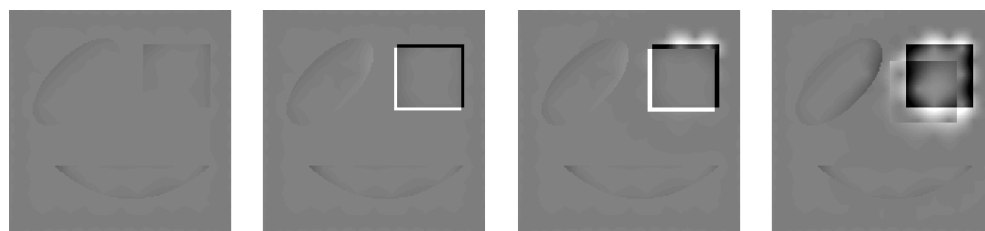


FIG. 5. The differences of the VSDK interpolants in Figure 4 to the original phantom.

**4.3.2. Results and discussion.** The outcome of the VSK interpolation depends sensitively on the choice of the scaling function  $\psi$ . All reconstructions in Figure 4 interpolate the function values on the Lissajous nodes. If the scaling function is correctly chosen (left) or only slightly shifted (middle left), no artifacts are visible in the interpolation. However, the larger the shift of the rectangle in the scaling function  $\psi$  gets, the stronger the artifacts are. In particular, we see that if the values of  $\psi$  do not correspond to the data values on the interpolation nodes, Gibbs-type artifacts appear. Therefore, if the VSDK interpolation scheme is applied in a setting in which the edges are not known, a robust edge estimator is needed. In the next section, we will discuss some possibilities for such an estimator.

**5. Extracting edges from the given data.** We use algorithms from machine learning to obtain a segmentation of the domain  $\Omega$ . In particular, we focus on so-called support vector machines (SVMs) and refer the reader to [36, 39] for a general overview. The main reason to use kernel machines for segmentation is that they can be applied directly to scattered data. Note, however, that the literature on segmentation of images and edge detection is very extensive and gives a lot of further interesting possibilities to obtain a segmentation. We refer to [40] for a general introduction.

**5.1. Segmentation of an image by classification algorithms.** In order to obtain a classification of the entire domain  $\Omega$ , we separate the data values  $(\mathbf{x}_i, f_i)$  into  $n$  classes  $\mathcal{S}_1, \dots, \mathcal{S}_n$  such that all nodes  $\mathbf{x}_i$  in one class  $\mathcal{S}_j$  are precisely contained in  $\Omega_j$ ,  $j \in \{1, \dots, n\}$ . We link every class  $\mathcal{S}_j$  to a value  $\alpha_j \in \mathbb{R}$  and set the label  $z_i = \psi(\mathbf{x}_i) = \alpha_j$  if  $(\mathbf{x}_i, f_i)$  is contained in  $\mathcal{S}_j$ . From the labels  $\mathcal{Z} = \{z_1, \dots, z_N\}$  of the points in  $\mathcal{X}$ , we want to derive now a classification for every  $\mathbf{x} \in \Omega$ .

We give a short description of SVM classification; a more precise introduction can be found, for instance, in [36, 39]. To simplify the considerations, we assume that we only have two classes with possible label values  $\alpha_1 = -1$  and  $\alpha_2 = 1$ . In this case, a decision function  $z$  that allows us to assign to every  $\mathbf{x}$  an appropriate label is given

by

$$z(\mathbf{x}) = \text{sign}(h(\mathbf{x})),$$

where  $h = 0$  describes a hyperplane separating the given measurements. The hyperplane is set up with the help of the so-called kernel trick. By virtue of Mercer's theorem [27], any kernel  $K$  can be decomposed as

$$(5.1) \quad K(\mathbf{x}, \mathbf{y}) = \boldsymbol{\Theta}(\mathbf{x})^\top \boldsymbol{\Theta}(\mathbf{y}) = \sum_{j=1}^{\infty} \Theta_j(\mathbf{x}) \Theta_j(\mathbf{y}), \quad \mathbf{x}, \mathbf{y} \in \Omega,$$

where  $\Theta_j$  are eigenfunctions of the integral operator  $g \rightarrow \int_{\Omega} K(\mathbf{x}, \mathbf{y}) g(\mathbf{y}) d\mathbf{y}$ . The kernel trick consists in mapping the points  $\mathbf{x}_i$  via  $\boldsymbol{\Theta}$  into a (possible) infinite-dimensional Hilbert space and describing the separating hyperplane as

$$h(\mathbf{x}) = \boldsymbol{\Theta}(\mathbf{x})^\top \mathbf{w} + b.$$

The weight  $\mathbf{w}$ , i.e., the unit normal vector to the hyperplane, and the bias  $b$  can be determined by maximizing the gap to both sides of this hyperplane. One standard way to obtain this hyperplane is by solving the optimization problem

$$\max_{\boldsymbol{\beta}} \left( \sum_{k=1}^N \beta_k - \frac{1}{2} \sum_{k=1}^N \sum_{i=1}^N \beta_k \beta_i z_k z_i K(\mathbf{x}_k, \mathbf{x}_i) \right)$$

subject to the constraints

$$\begin{cases} \sum_{k=1}^N \beta_k z_k = 0, \\ 0 \leq \beta_i \leq B, \quad i \in \{1, \dots, N\}. \end{cases}$$

Here, the box constraint  $B$  is simply a regularization parameter [16]. Based on the maximizer of this problem, the decision function  $z$  of the SVM classifier is then given as

$$(5.2) \quad z(\mathbf{x}) = \text{sign}(h(\mathbf{x})) = \text{sign} \left( \sum_{i=1}^N \beta_i z_i K(\mathbf{x}, \mathbf{x}_i) + b \right).$$

The bias  $b$  can be determined as

$$b = \sum_{k=1}^N \beta_k z_k K(\mathbf{x}_k, \mathbf{x}_j),$$

where  $j$  denotes the index of a coefficient  $\beta_j$  which is strictly between 0 and  $B$ .

The classification function  $z(\mathbf{x})$  in (5.2) gives now the desired segmentation of  $\Omega$ : The two sets  $\Omega_1$  and  $\Omega_2$  are defined such that for  $\mathbf{x} \in \Omega_i$  we have  $z(\mathbf{x}) = \alpha_i$ ,  $i \in \{1, 2\}$ . The corresponding discontinuous scaling function  $\psi$  on  $\Omega$  is given as

$$(5.3) \quad \psi(\mathbf{x}) = \begin{cases} \alpha_1 & \text{if } z(\mathbf{x}) = \alpha_1, \\ \alpha_2 & \text{if } z(\mathbf{x}) = \alpha_2, \end{cases}$$

It is straightforward to extend this classification scheme if  $\alpha_1, \alpha_2 \neq \pm 1$ . There are also several strategies to extend this scheme to the case that the number of classes is  $n \geq 2$ . In the literature, this is known as multiclass SVM classification. One usual approach here is to divide the single multiclass problem into multiple binary SVM classification problems.

**5.2. Strategies to set the labels for classification.** In general, the choice of a labeling strategy depends on the application aimed. In the following, we specify a few simple heuristic strategies to extract the labels  $\mathcal{Z}$  from a given data set  $(\mathcal{X}, \mathcal{F})$ .

**5.2.1. Using thresholds on the data values.** If the function  $f$  has discontinuities, these are visible as deviations in the data set  $\mathcal{F}$ . A very simple strategy is therefore to use thresholds for the definition of the labels. If  $a_0 < a_1 < \dots < a_n$  and  $\text{supp}(\mathcal{F})$  is contained in the interval  $[a_0, a_n]$ , we can define  $n$  classes  $\mathcal{S}_1, \dots, \mathcal{S}_n$  by assigning  $(\mathbf{x}_i, f_i)$  to  $\mathcal{S}_j$  if  $a_{j-1} \leq f_i < a_j$ ,  $j \in \{1, \dots, n\}$ .

**5.2.2. Using thresholds on interpolation coefficients.** In [30], it is shown that variations in the expansion coefficients of an RBF interpolation can be used to detect the edges of a function. Thus, in the same way as in the previous strategy, thresholds on the absolute value of the RBF coefficients can be applied to determine the labeling aimed at.

**5.2.3. Automated strategies using k-means clustering.** The given data  $(\mathcal{X}, \mathcal{F})$  can also be segmented using an automated procedure using k-means clustering. If the size  $n$  of classes is known, this method provides  $n$  pairwise disjoint classes  $\mathcal{S}_1, \dots, \mathcal{S}_n$  by minimizing the functional

$$\sum_{j=1}^n \sum_{(\mathbf{x}, h) \in \mathcal{S}_j} |h - \bar{h}_j|.$$

The value  $\bar{h}_j$  denotes the mean of all function values  $h$  inside the class  $\mathcal{S}_j$ . Note that in this case the position  $\mathbf{x}$  of the data is not used to determine the labels.

**5.3. Algorithm for VSDK interpolation with unknown edges.** In the following Algorithm 5.1, we summarize the entire scheme for the computation of a shape-driven interpolant from given function values on a node set  $\mathcal{X}$  and unknown discontinuities. For the interpolation, the VSDK scheme introduced in section 2.3 is used. To estimate the edges of  $f$ , we use the segmentation and labeling procedures described in sections 5.1 and 5.2.

---

**Algorithm 5.1** Shape-driven interpolation with discontinuous kernels.

---

INPUTS: Set of interpolation nodes

$$\mathcal{X} = \{\mathbf{x}_i, i = 1, \dots, N\} \subseteq \Omega,$$

a corresponding set of data values

$$\mathcal{F} = \{f_i = f(\mathbf{x}_i), i = 1, \dots, N\},$$

and the desired evaluation point(s)  $\mathbf{x} \in \Omega$ .

OUTPUTS: VSDK interpolant  $V_f(\mathbf{x})$ ,

Step 1: Extract the labels  $\mathcal{Z}$  for  $\mathcal{X}$  using  $\mathcal{F}$  with a strategy of section 5.2.

Step 2: Train the kernel machine in section 5.1 with the points  $\mathcal{X}$  and the labels  $\mathcal{Z}$  to obtain a prediction (5.3) for the scaling function  $\psi$ .

Step 3: Calculate the coefficients  $c_i$ ,  $i \in \{1, \dots, N\}$ , of the VSDK interpolation by solving (2.9).

Step 4: Evaluate the interpolant  $\mathcal{V}_f(\mathbf{x})$  in (2.8) at  $\mathbf{x} \in \Omega$ .

---

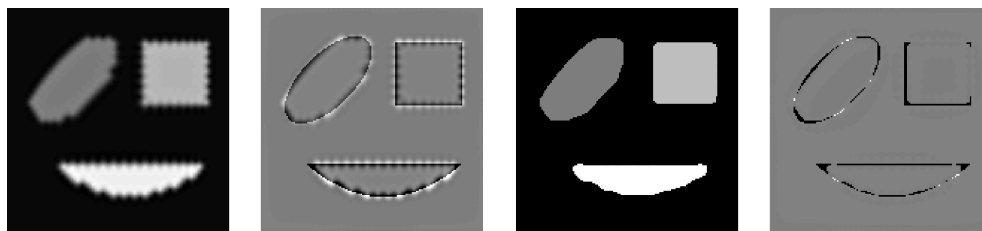


FIG. 6. Comparison of RBF interpolation (left) with the VSDK scheme given in Algorithm 5.1 (right). The interpolation is performed on the nodes  $\mathbf{LS}_2^{(33,32)}$ . In the second and the fourth images (from the left) the respective differences to the original phantom are displayed.

#### 5.4. Numerical example.

**5.4.1. Description.** On the node set  $\mathbf{LS}_2^{(33,32)}$ , we interpolate the geometric phantom  $f_G$  using an ordinary RBF interpolation and the VSDK scheme from Algorithm 5.1. In both cases, we use the  $C^0$ -Matérn function as underlying kernel. The applied edge estimator in Algorithm 5.1 is based on the segmentation method of section 5.1 and the automated labeling described in section 5.2.3. The resulting RBF interpolant and the error with respect to the original phantom are displayed in Figure 6 (left). In Figure 6 (right) the outcome of Algorithm 5.1 and the respective error with respect to  $f_G$  are shown.

**5.4.2. Results and discussion.** In this example, in which we do not use the a priori knowledge of the discontinuities, the VSDK scheme in combination with the edge estimator gives a higher reconstruction quality than an ordinary RBF interpolation. In particular, in the VSDK interpolation the Gibbs phenomenon is not visible, and the errors are more localized at the boundaries of the geometric figures. For a more quantitative comparison, we compute the relative discrete  $L_1$ -errors of the two reconstructions. We obtain

$$\frac{\|f_G - P_{f_G}\|_1}{\|f_G\|_1} \approx 0.1647, \quad \frac{\|f_G - V_{f_G}\|_1}{\|f_G\|_1} \approx 0.1011;$$

i.e., the VSDK interpolant gives a slightly better result with respect to the  $L_1$ -norm.

Again, we want to point out that in case that the discontinuities are not a priori given, the output of the VSDK interpolation strongly depends on the performance of the edge detector. If the edges are detected in a reliable way, also the final VSDK interpolation has a good overall quality. For this compare Figure 6 also with the reconstruction in Figure 4 (left), in which the scaling function with the correct information of the edges was used. On the other hand, as discussed in section 4.3, if the scaling function  $\psi$  is badly chosen, the final reconstruction is also seriously affected by artifacts.

**6. Applications in MPI.** In the early 2000s, B. Gleich and J. Weizenecker [18], invented at Philips Research in Hamburg a new quantitative imaging method called MPI. In this imaging technology, a tracer consisting of superparamagnetic iron oxide nanoparticles is injected and then detected through the superimposition of different magnetic fields. In common MPI scanners, the acquisition of the signal is performed following a generated field free point along a chosen sampling trajectory. The determination of the particle distribution given the measured voltages in the receive coils





FIG. 7. Comparison of different interpolation methods in MPI. The reconstructed data on the Lissajous nodes  $\mathbf{LS}_2^{(33,32)}$  (left) are first interpolated using the polynomial scheme derived in [12] (middle left). Using a scaling function  $\psi$  constructed upon the classification algorithm in section 5.1 (middle right) the second interpolation is performed by the VSDK scheme in Algorithm 5.1 (right).

is an ill-posed inverse problem that can be solved only with proper regularization techniques [24].

Commonly used trajectories in MPI are Lissajous curves [25]. To reduce the amount of calibration measurements, it is shown in [23] that the reconstruction can be restricted to particular sampling points along the Lissajous curves, i.e., the Lissajous nodes  $\mathbf{LS}_2^{(n)}$  introduced in (4.2). By using a polynomial interpolation method on the Lissajous nodes [12] the entire density of the magnetic particles can then be restored. These sampling nodes and the corresponding polynomial interpolation can be seen as an extension of a respective theory on the Padua points [2, 3].

If the original particle density has sharp edges, the polynomial reconstruction scheme on the Lissajous nodes is affected by the Gibbs phenomenon. As shown in [10], postprocessing filters can be used to reduce oscillations for polynomial reconstruction in MPI. In the following, we demonstrate that the usage of the VSDK interpolation method in combination with the presented edge estimator effectively avoids ringing artifacts in MPI and provides reconstructions with sharpened edges.

**6.1. Description.** As a test data set, we consider MPI measurements conducted in [23] on a phantom consisting of three tubes filled with Resovist, a contrast agent consisting of superparamagnetic iron oxide. By the procedure described in [23] we then obtain a reconstruction of the particle density on the Lissajous nodes  $\mathbf{LS}_2^{(33,32)}$ . This reduced reconstruction on the Lissajous nodes is illustrated in Figure 7 (left). A computed polynomial interpolant of these data is shown in Figure 7 (middle left). In this polynomial interpolant some ringing artifacts are visible. In order to obtain the labeling for the classification algorithm, we use the simple thresholding strategy described in section 5.2.1 using one-fifth of the maximal signal strength as a threshold for a binary classification. The scaling function  $\psi$  for the VSDK scheme is then obtained by using the classification algorithm of section 5.1 with a Gauss function for the kernel machine. The resulting scaling function is visualized in Figure 7 (middle right). Using the  $C^0$ -Matérn kernel for the VSDK interpolation, the final interpolant for the given MPI data is shown in Figure 7 (right). For a better visualization of the ringing artifacts, we provide a second illustration of the second and fourth images in Figure 7 with diagonal view direction from the lower left to the upper right corner of the respective image. These illustrations of all cross sections of the polynomial and the VSDK interpolant along this diagonal direction are given in Figure 8.

**6.2. Results and discussion.** In the polynomial interpolation shown in Figure 7 (middle left) ringing artifacts are visible. These artifacts could be removed

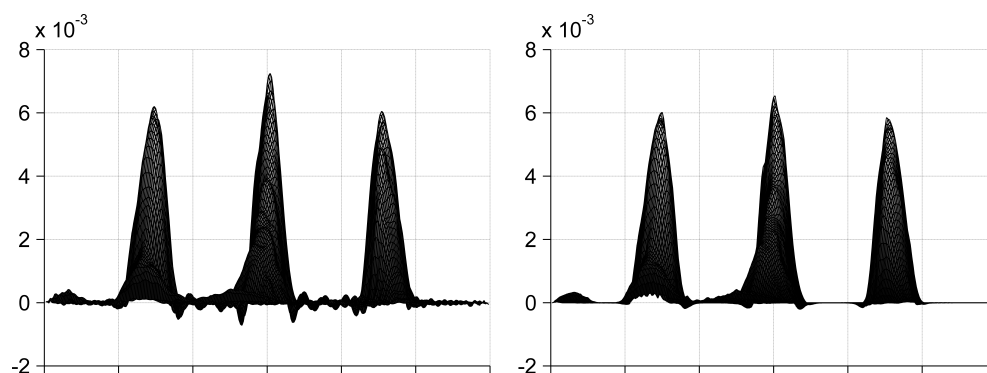


FIG. 8. Comparison of polynomial (left) and VSDK interpolation (right) in MPI. Illustrated are the interpolants in Figure 7 (middle left) and Figure 7 (right) with diagonal view from the lower left to the upper right corner.

by using Algorithm 5.1 for the MPI data instead. As an alternative to the applied manual thresholding strategy, it is also possible to use the automated strategy given in section 5.2.3, in which the k-means algorithm gives the labeling of the data. This second strategy yields classification and reconstruction results that are very close to the ones displayed for the manual strategy in Figure 7.

**7. Conclusions.** To reflect discontinuities of a function or an image in the interpolation of scattered data we studied techniques based on the use of VSDKs. We obtained a characterization and theoretical Sobolev-type error estimates for the native spaces generated by these discontinuous kernels. Numerical experiments confirmed the theoretical convergence rates and investigated the behavior of the interpolants if the scaling function describing the discontinuities is perturbed.

Interpolation with discontinuous kernels can only be conducted if the discontinuities of the function are known. If the discontinuities are not known, sophisticated methods are necessary to approximate the edges from given scattered data. In this work, we used kernel machines, trained with the given data, to obtain the edges and the segmentation of the image.

The results of the VSDK method applied to MPI are promising and show that the Gibbs phenomenon can be sensibly reduced. Work in progress consists in using kernel machines also for regression with VSDKs. This might be of interest when approximating time series with jumps.

**Acknowledgments.** First, we want to thank both referees for their valuable feedback and the constructive comments that helped to improve the quality of this manuscript a lot. This research has been accomplished within Rete Italiana di Approssimazione (RITA).

#### REFERENCES

- [1] R.A. ADAMS AND J. FOURNIER, *Sobolev Spaces*, Academic Press, London, 2003.
- [2] L. BOS, M. CALIARI, S. DE MARCHI, M. VIANELLO, AND Y. XU, *Bivariate Lagrange interpolation at the Padua points: The generating curve approach*, J. Approx. Theory, 143 (2006), pp. 15–25.
- [3] L. BOS, S. DE MARCHI, AND M. VIANELLO, *Polynomial approximation on Lissajous curves in the d-cube*, Appl. Numer. Math., 116 (2017), pp. 47–56.
- [4] M. BOZZINI, L. LENARDUZZI, M. ROSSINI, AND R. SCHABACK, *Interpolation with variably scaled kernels*, IMA J. Numer. Anal., 35 (2015), pp. 199–219.

- [5] M. BUHMANN, *A new class of radial basis functions with compact support*, Math. Comp., 70, 233 (2000), pp. 307–318.
- [6] M. BUHMANN, *Radial Basis Functions: Theory and Implementations*, Cambridge Monogr. Appl. Comput. Math., Cambridge University Press, Cambridge, 2003.
- [7] J.T. BUSHBERG, J.A. SEIBERT, E.M. LEIDHOLDT, AND J.M. BOONE, *The Essential Physics of Medical Imaging*, 2nd ed., Lippincott Williams & Wilkins, Philadelphia, 2001.
- [8] J.F. CANNY, *A computational approach to edge detection*, IEEE TPAMI, 8 (1986), pp. 34–43.
- [9] L.F. CZERVIONKE, J.M. CZERVIONKE, D.L. DANIELS, AND V.M. HAUGHTON, *Characteristic features of MR truncation artifacts*, Am. J. Roentgenol., 151 (1988), pp. 1219–1228.
- [10] S. DE MARCHI, W. ERB, AND F. MARCHETTI, *Spectral filtering for the reduction of the Gibbs phenomenon for polynomial approximation methods on Lissajous curves with applications in MPI*, Dolomites Res. Notes Approx., 10 (2017), pp. 128–137.
- [11] S. DE MARCHI, F. MARCHETTI, AND E. PERRACCHIONE, *Jumping with variably scaled discontinuous kernels (VSDKs)*, BIT, 2019, <https://doi.org/10.1007/s10543-019-00786-z>.
- [12] W. ERB, C. KAETHNER, M. AHLBORG, AND T.M. BUZUG, *Bivariate Lagrange interpolation at the node points of non-degenerate Lissajous nodes*, Numer. Math., 133, 1 (2016), pp. 685–705.
- [13] W. ERB, *Bivariate Lagrange interpolation at the node points of Lissajous curves—The degenerate case*, Appl. Math. Comput., 289 (2016), pp. 409–425.
- [14] W. ERB, C. KAETHNER, P. DENCKER, AND M. AHLBORG, *A survey on bivariate Lagrange interpolation on Lissajous nodes*, Dolomites Res. Notes Approx., 8 (special issue) (2015), pp. 23–36.
- [15] E. FUSELIER AND G. WRIGHT, *Scattered data interpolation on embedded submanifolds with restricted positive definite Kernels: Sobolev error estimates*, SIAM J. Numer. Anal., 50, 3 (2012), pp. 1753–1776.
- [16] G.E. FASSHAUER AND M.J. MCCOURT, *Kernel-Based Approximation Methods Using MATLAB*, World Scientific, Singapore, 2015.
- [17] B. FORNBERG AND N. FLYER, *The Gibbs phenomenon for radial basis functions*, in The Gibbs Phenomenon in Various Representations and Applications, A. Jerri, ed., Sampling Publishing, Potsdam, NY, 2011, pp. 201–224.
- [18] B. GLEICH AND J. WEIZENECKER, *Tomographic imaging using the nonlinear response of magnetic particles*, Nature, 435 (2005), pp. 1214–1217.
- [19] D. GOTTLIEB AND C.W. SHU, *On the Gibbs phenomenon and its resolution*, SIAM Rev., 39 (1997), pp. 644–668.
- [20] A. JERRI, *The Gibbs Phenomenon in Fourier Analysis, Splines and Wavelet Approximations*, Kluwer Academic Publishers, Dordrecht, 1998.
- [21] J.-H. JUNG, *A note on the Gibbs phenomenon with multiquadric radial basis functions*, Appl. Numer. Math., 57 (2007), pp. 213–219.
- [22] J.-H. JUNG, S. GOTTLIEB, AND S.O. KIM, *Iterative adaptive RBF methods for detection of edges in two-dimensional functions*, Appl. Numer. Math., 61 (2011), pp. 77–91.
- [23] C. KAETHNER, W. ERB, M. AHLBORG, P. SZWARGULSKI, T. KNOPP, AND T.M. BUZUG, *Non-equispaced system matrix acquisition for magnetic particle imaging based on Lissajous node points*, IEEE Trans. Med. Imaging, 35 (2016), pp. 2476–2485.
- [24] T. KNOPP AND T.M. BUZUG, *Magnetic Particle Imaging*, Springer-Verlag, Berlin, 2012.
- [25] T. KNOPP, S. BIEDERER, T. SATTEL, J. WEIZENECKER, B. GLEICH, J. BORGERT, AND T.M. BUZUG, *Trajectory analysis for magnetic particle imaging*, Phys. Med. Biol., 54 (2009), pp. 385–397.
- [26] T.M. LEHMANN, C. GONNER, AND K. SPITZER, *Survey: Interpolation methods in medical image processing*, IEEE Trans. Med. Imaging, 18 (1999), pp. 1049–1075.
- [27] J. MERCER, *Functions of positive and negative type and their connection with the theory of integral equations*, Philos. Trans. Roy. Soc., 209 (1909), pp. 415–446.
- [28] F.J. NARCOWICH, J.D. WARD, AND H. WENDLAND, *Sobolev bounds on functions with scattered zeros, with applications to radial basis function surface fitting*, Math. Comp., 74 (2005), pp. 743–763.
- [29] C. RIEGER, *Sampling Inequalities and Applications*, Dissertation, University of Göttingen, 2008.
- [30] L. ROMANI, M. ROSSINI, AND D. SCHENONE, *Edge detection methods based on RBF interpolation*, J. Comput. Appl. Math., 349 (2019), pp. 532–547.
- [31] M. ROSSINI, *Interpolating functions with gradient discontinuities via variably scaled kernels*, Dolomites Res. Notes Approx., 11 (2018), pp. 3–14.
- [32] V.S. RYCHKOV, *On restrictions and extensions of the Besov and Triebel-Lizorkin spaces with respect to Lipschitz domains*, J. Lond. Math. Soc., 60 (1999), pp. 237–257.

- [33] S.A. SARRA, *Digital total variation filtering as postprocessing for radial basis function approximation methods*, Comput. Math. Appl., 52 (2006), pp. 1119–1130.
- [34] R. SCHABACK, *Native Hilbert spaces for radial basis functions. I.*, in New Developments in Approximation Theory (Dortmund, 1998), Internat. Ser. Numer. Math. 132, Birkhäuser, Basel, 1999, pp. 255–282.
- [35] R. SCHABACK AND H. WENDLAND, *Approximation by positive definite kernels*, in Advanced Problems in Constructive Approximation, Birkhäuser, Basel, 2003, pp. 203–222.
- [36] B. SCHÖLKOPF AND A.J. SMOLA, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press, Cambridge, MA, 2002.
- [37] M. SHARIFI, M. FATHY, AND M.T. MAHMOUDI, *A classified and comparative study of edge detection algorithms*, in Proc. Int. Conf. on Inform. Technology: Coding and Computing, Las Vegas, NV, 2002, pp. 117–120.
- [38] L.A. SHEPP AND B.F. LOGAN, *The Fourier reconstruction of a head section*, IEEE Trans. Nucl. Sci., NS-21 (1974), pp. 21–43.
- [39] A.J. SMOLA AND B. SCHÖLKOPF, *A tutorial on support vector regression*, Stat. Comput., 14 (2004), pp. 199–222.
- [40] C.J. SOLOMON AND T.P. BRECKON, *Fundamentals of Digital Image Processing: A Practical Approach with Examples in Matlab*, Wiley-Blackwell, Hoboken, NJ, 2010.
- [41] A. TAKAKI, T. SOMA, A. KOJIMA, K. ASAO, S. KAMADA, M. MATSUMOTO, AND K. MURASE, *Improvement of image quality using interpolated projection data estimation method in SPECT*, Ann. Nucl. Med., 23 (2009), pp. 617–626.
- [42] P. THEVENAZ, T. BLU, AND M. UNSER, *Interpolation revisited*, IEEE Trans. Med. Imaging, 19 (2000), pp. 739–758.
- [43] H. TRIEBEL, *Theory of Function Spaces*, Monogr. Math. 78, Birkhäuser, Basel, 1983.
- [44] H. WENDLAND, *Scattered Data Approximation*, Cambridge Monogr. Appl. Comput. Math., Cambridge University Press, Cambridge, 2005.
- [45] A. ZYGMUND, *Trigonometric Series*, 3rd ed., Vols. I and II combined, Cambridge Mathematical Library, Cambridge University Press, Cambridge, 2002.