Working title: Topics in Multivariate Approximation and Interpolation 1
K. Jetter et al., Editors
© 2005 Elsevier B.V. All rights reserved

Computational Aspects of Radial Basis Function Approximation

Holger Wendland

Institut für Numerische und Angewandte Mathematik, Universität Göttingen
Lotzestr. 16-18, 37083 Göttingen, Germany

Abstract

This paper gives an overview on numerical aspects of multivariate interpolation and approximation by radial basis functions. It comments on the correct choice of the basis function. It discusses the reduction of complexity by different methods as well as the problem of ill-conditioning. It is aimed to be a user's guide to an efficient employment of radial basis functions for the reconstruction of multivariate functions.

Key words: multivariate approximation theory, scattered data approximation, positive definite kernels, domain decomposition, multilevel, multipole expansions, partition of unity
2000 MSC: 46G05, 46B22, 28B05

1. Introduction

Over the past decades radial basis functions or, more generally, (conditionally) positive definite kernels have very successfully been used for reconstructing multivariate functions from scattered data. This success is mainly based upon the following facts:

(i) Radial basis functions can be used in *any* space dimension.
(ii) They work for arbitrarily *scattered data*, bearing no regularity at all.
(iii) They allow interpolants of *arbitrary* smoothness.
(iv) The interpolants have a simple structure, which makes RBFs in particular interesting to users outside mathematics.

However, these positive properties do not come for free. For example, building a smooth interpolant using a smooth basis function leads also to an ill-conditioned linear system that has to be solved. Moreover, since most basis functions are globally supported, a large number of interpolation points leads to an unacceptable complexity concerning both space and time.

For these reasons recent research concentrated on resolving these problems. Fast methods for evaluating and computing an RBF interpolant have been developed and thoroughly investigated. Smoothing techniques have been employed to regularize ill-conditioned systems and to smooth out measurement errors.

It is the goal of this paper to describe the most successful approaches in this research direction. Naturally, the selection is biased by this author's point of view and, even if the attempt is made to cover as many methods as possible, there is no guarantee that some important once are missing.

Nonetheless, this paper should help the reader to understand which basis function and which efficient method she or he should employ for her or his particular reconstruction problem. It should also encourage the reader to consult the literature pointed out in the bibliography for further studying.

This paper is organized as follows. In the next section we will give a short review on interpolation and approximation by (conditionally) positive definite kernels. The second section is devoted to efficient methods, while the last section deals with stabilizing and regularization.

2. Scattered Data Interpolation and Approximation

A general scattered data interpolation problem can be described as follows. Suppose we are given N distinct *data sites* $X = \{x_1, \dots, x_N\}$ situated in a bounded region $\Omega \subseteq \mathbb{R}^d$. Attached to each data site x_j comes a *data value* f_j, which might represent a measured quantity at site x_j. Finally, suppose that it is known, or at least assumed, that the function values f_j are generated by an unknown function f, i.e. $f_j = f(x_j)$. It might also be possible that the data values are generated by functionals other than point evaluations. Examples include derivative or cell average information, but we will not pursue this here, although, to a large extent, the theory can be developed in a more general context (see [1,2] and the references therein).

The information given so far (the data sites, the data values and the fact that the data values are generated by a function) is not enough to built a trustworthy interpolant or approximant to the unknown function, since it provides only very *localized* information. To have a decent reconstruction of the unknown function on all of Ω and not only at X also certain *global* information is necessary. This global information can often be provided by certain *smoothness* assumptions on the unknown target function f. This is equivalent to assuming the function to belong to a certain, normed function space $\mathcal{H}$. With this information at hand, it is possible and natural to look for the solution of

$$\min \{\|s\|_{\mathcal{H}} : s \in \mathcal{H} \text{ with } s|X = f|X\}, \tag{1}$$

or, if the data values are known to contain noise, of

$$\min \left\{ \sum_{j=1}^{N} [s(x_j) - f(x_j)]^2 + \lambda \|s\|_{\mathcal{H}}^2 : s \in \mathcal{H} \right\}, \tag{2}$$

where $\lambda > 0$ is a certain *smoothing parameter*, which has to be chosen carefully, to balance between interpolation and approximation.

2.1. *Reproducing Kernel Hilbert Spaces*

To discuss the solutions to both problems we have to make two more assumptions on the function space. The first assumption is a natural one. Since we want to work with point evaluation functionals, it is reasonable to assume that point evaluation functionals are continuous on $\mathcal{H}$, i.e. that for every $x \in \Omega$ there exists a constant $C_x > 0$ with

$$|f(x)| \leq C_x \|f\|_{\mathcal{H}}, \qquad \text{for all } f \in \mathcal{H}.$$

Our second assumption is not that natural but it will simplify the theory dramatically and provides no severe restriction in applications. We will assume that our function space $\mathcal{H}$ is a *Hilbert space* of functions.

A Hilbert space $\mathcal{H}$ of functions $f : \Omega \to \mathbb{R}$ with continuous point evaluation functionals is known to be a *reproducing kernel Hilbert space* (RKHS) (see e.g. [3]).

Definition 1. A *reproducing kernel Hilbert space* $\mathcal{H}$ is a Hilbert space of functions $f : \Omega \to \mathbb{R}$, which has a unique kernel $\Phi : \Omega \times \Omega \to \mathbb{R}$, satisfying
(i) $\Phi(\cdot, x) \in \mathcal{H}$ for all $x \in \Omega$,
(ii) $f(x) = (f, \Phi(\cdot, x))_{\mathcal{H}}$ for all $x \in \Omega$ and all $f \in \mathcal{H}$.

In a RKHS the reproducing kernel Φ is always symmetric and positive semi-definite, i.e. for arbitrary point sets $X = \{x_1, \ldots, x_N\}$, the matrices

$$A = A_{\Phi,X} = (\Phi(x_i, x_j))_{i,j} \tag{3}$$

are symmetric and positive semi-definite. This follows immediately from the reproduction property applied to the kernel itself:

$$\Phi(x, y) = (\Phi(\cdot, y), \Phi(\cdot, x))_{\mathcal{H}},$$

and this already ensures that (2) has a unique solution for $\lambda > 0$. However, for guaranteeing a unique solution to (1) as well, we have to make the additional assumption that point evaluation functionals are linearly independent on $\mathcal{H}$. This is equivalent to the fact that all matrices A of the form (3) are positive definite.

Theorem 2. *Let $\mathcal{H} \subseteq C(\Omega)$ be a reproducing kernel Hilbert space with reproducing kernel Φ. Then, for every $f \in \mathcal{H}$, both problems (1) and (2) have a unique solution s_0 and s_λ, respectively. This solution has a representation*

$$s_\lambda = \sum_{j=1}^{N} \alpha_j \Phi(\cdot, x_j),$$

where the coefficient vector $\alpha \in \mathbb{R}^N$ can be determined by solving the linear system

$$(A + \lambda I)\alpha = f|X.$$

Here, A is the matrix (3) and I denotes the identity matrix.

For a proof, we refer the reader to [2] for problem (1) and to [4] for problem (2). From this theorem one can read off that the solutions s_λ, $\lambda > 0$, of (2) converge to the solution $s_0 = I_X f$ of (1) with λ approaching zero.

Our main interest will be in kernels that are radial and hence defined on all of $\mathbb{R}^d$.

Definition 3. A kernel $\Phi : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is said to be *translation invariant*, if it can be written in the form $\Phi(x, y) = \varphi(x - y)$ for all $x, y \in \mathbb{R}^d$ with an even function $\varphi : \mathbb{R}^d \to \mathbb{R}$.

The kernel is said to be *radial* if it can be written as $\Phi(x, y) = \phi(\|x - y\|_2)$, $x, y \in \mathbb{R}^d$, with a function $\phi : [0, \infty) \to \mathbb{R}$.

Typical examples of positive definite and radial kernels are Gaussians $\phi(r) = e^{-r^2}$, inverse multiquadrics $\phi(r) = 1/\sqrt{1 + r^2}$, and compactly supported functions like $\phi(r) = (1 - r)_+^4(4r + 1)$. While the associated Hilbert spaces for the first two kernels are rather small and consist mainly of analytic functions, the RKHS to the latter kernel is a classical Sobolev space.

This is due to the fact that in all practical situations, the RKHS to a positive definite, translation invariant kernel can be characterized by Fourier transformation. Recall that the *Fourier transform* of $f \in L_1(\mathbb{R}^d)$ is defined by

$$\widehat{f}(\omega) := (2\pi)^{-d/2} \int_{\mathbb{R}^d} f(x) e^{-ix^T\omega} dx, \qquad \omega \in \mathbb{R}^d.$$

Then, the RKHS to $\Phi(x, y) = \varphi(x - y) = \phi(\|x - y\|_2)$ can be characterized as

$$\{f \in L_2(\mathbb{R}^d) \cap C(\mathbb{R}^d) : \widehat{f}/\sqrt{\widehat{\varphi}} \in L_2(\mathbb{R}^d)\},$$

which shows that for Gaussians and inverse multiquadrics we are confronted with rather small spaces. The spaces we are mainly interested in are *Sobolev spaces*

$$H^\tau(\mathbb{R}^d) := \{f \in L_2(\mathbb{R}^d) : \widehat{f}(\cdot)(1 + \|\cdot\|_2^2)^{\tau/2} \in L_2(\mathbb{R}^d)\},$$

they possess reproducing kernels of finite smoothness. The Sobolev space $H^\tau(\mathbb{R}^d)$ is known to be a subset of $C(\mathbb{R}^d)$ provided that $\tau > d/2$, which will be our general assumption.

Note that a kernel $\Phi(x,y) = \varphi(x-y)$ with a Fourier transform satisfying

$$c_1(1+\|\omega\|_2^2)^{-\tau} \le \widehat{\varphi}(\omega) \le c_2(1+\|\omega\|_2^2)^{-\tau}$$

has $H^\tau(\mathbb{R}^d)$ as its RKHS and the *native* inner product

$$(f,g)_{\mathcal{H}} := (2\pi)^{-d/2} \int_{\mathbb{R}^d} \widehat{f}(\omega)\overline{\widehat{g}(\omega)}\widehat{\Phi}(\omega)^{-1} d\omega$$

leads to a norm which is equivalent to the Sobolev space norm. In this sense, we will call Φ also a reproducing kernel of $H^\tau(\mathbb{R}^d)$.

2.2. *From Thin-plate Splines to Conditionally Positive Definite Kernels*

If $\tau = m$ is a nonnegative integer, the norm on $H^m(\mathbb{R}^d)$ can equivalently be expressed by

$$\|f\|^2_{H^m(\mathbb{R}^d)} = \sum_{k=0}^{m} \sum_{|\alpha|=k} \frac{k!}{\alpha!} \|D^\alpha f\|^2_{L_2(\mathbb{R}^d)}$$

and it is interesting to see that problems (1) and (2) also have a solution if the full norm is replaced by the semi-norm

$$|f|^2_{\mathrm{BL}_m(\mathbb{R}^d)} := \sum_{|\alpha|=m} \frac{m!}{\alpha!} \|D^\alpha f\|^2_{L_2(\mathbb{R}^d)}$$

and minimization is done over all functions from the *Beppo-Levi space*

$$\mathrm{BL}_m := \{f \in C(\mathbb{R}^n) : D^\alpha f \in L_2(\mathbb{R}^n) \text{ for all } |\alpha| = m\}.$$

The solution can be expressed in terms of the so-called *surface* or *thin-plate splines* which are defined as

$$\phi_{d,m}(r) := \begin{cases} \dfrac{\Gamma(\frac{d}{2}-m)}{2^{2m}\pi^{d/2}(m-1)!} r^{2m-d}, & \text{for } d \text{ odd}, \\ \dfrac{(-1)^{m+\frac{d-2}{2}}}{2^{2m-1}\pi^{d/2}(m-1)!(m-d/2)!} r^{2m-d}\log r, & \text{for } d \text{ even}, \end{cases}$$

and form a Green's function to the m-th iterated Laplacian operator. To formulate the result, we introduce $\pi_{m-1}(\mathbb{R}^d)$ as the space of d-variate polynomials of degree at most $m-1$ and let $p_1, \ldots, p_Q$ with $Q = \dim(\pi_{m-1}(\mathbb{R}^d))$ be a basis of $\pi_{m-1}(\mathbb{R}^d)$. Moreover, we say that a set X is $\pi_{m-1}(\mathbb{R}^d)$ *unisolvent*, if the only polynomial $p \in \pi_{m-1}(\mathbb{R}^d)$ with $p|X = 0$ is the zero polynomial. The proof of the following theorem can again be found in [4,2].

Theorem 4. *Suppose that $m > d/2$ and that $X \subseteq \mathbb{R}^d$ is $\pi_{m-1}(\mathbb{R}^d)$ unisolvent. Let $f \in \mathrm{BL}_m(\mathbb{R}^d)$ be the unknown target function. Then, the solution s_λ to*

$$\min \left\{ \sum_{j=1}^{N} [f(x_j) - s(x_j)]^2 + \lambda |s|^2_{\mathrm{BL}_m(\mathbb{R}^n)} : s \in \mathrm{BL}_m(\mathbb{R}^d) \right\}$$

as well as the solution s_0 to

$$\min \left\{ |s|_{\mathrm{BL}_m(\mathbb{R}^n)} : s \in \mathrm{BL}_m(\mathbb{R}^n) \text{ with } s|X = f|X \right\}$$

is given by

$$s_\lambda(x) = \sum_{j=1}^{N} a_j \phi_{d,m}(\|x - x_j\|_2) + \sum_{k=1}^{Q} b_j p_j(x).$$

The coefficients $a \in \mathbb{R}^N$ and $b \in \mathbb{R}^Q$ can be determined as follows. Let $A = (\phi_{d,m}(\|x_i - x_j\|_2)) \in \mathbb{R}^{N \times N}$ and $P = (p_j(x_i)) \in \mathbb{R}^{N \times Q}$. Then a and b are the solutions of

$$\begin{aligned} (A + \lambda I)a + Pb &= f|X \\ P^T b &= 0, \end{aligned}$$

where I denotes the identity matrix.

Thin-plate splines form a particular class of *conditionally positive definite* functions. In general, a function $\varphi : \mathbb{R}^d \to \mathbb{R}$ is said to be conditionally positive definite of order m, if the associated matrices $A_{\varphi,X} \in \mathbb{R}^{N \times N}$ are positive definite on the set

$$\left\{ \alpha \in \mathbb{R}^N : \sum_{j=1}^{N} \alpha_j p(x_j) = 0 \text{ for all } p \in \pi_{m-1}(\mathbb{R}^d) \right\}.$$

Besides thin-plate splines, multiquadrics, defined by $\phi(r) = \sqrt{1 + r^2}$, are the most prominent examples of conditionally positive definite functions. A formal, more general definition for conditionally positive definite kernels is the following one:

Definition 5. Suppose $\mathcal{P}$ is a finite dimensional subspace of $C(\Omega)$, $\Omega \subseteq \mathbb{R}^d$. A continuous symmetric kernel $\Phi : \Omega \times \Omega \to \mathbb{R}$ is said to be conditionally positive definite on Ω with respect to $\mathcal{P}$ if for any N pairwise distinct centers $x_1, \ldots, x_N \in \Omega$ and all $\alpha \in \mathbb{R}^N \setminus \{0\}$ with $\sum_{j=1}^{N} \alpha_j p(x_j) = 0$ for all $p \in \mathcal{P}$, the quadratic form

$$\sum_{j=1}^{N} \sum_{k=1}^{N} \alpha_j \alpha_k \Phi(x_j, x_k)$$

is positive.

The results of Theorem 4 remain true in the situation of a conditionally positive definite kernel, if the Beppo-Levi space is replaced by an adequate, *natural* function space, the so called *native space* $\mathcal{H} = \mathcal{N}_\Phi(\Omega)$ of the kernel Φ. For a thorough discussion of native spaces see [5,6] and Chapter 11 of [2]. Here, only the following information is important. The native space $\mathcal{H}$ is a function space carrying a semi-inner product $(\cdot,\cdot)_\mathcal{H}$ which has the finite dimensional space $\mathcal{P}$ as its kernel. Moreover, the quotient space $\mathcal{H}/\mathcal{P}$ is a Hilbert space. This can be used to equip $\mathcal{H}$ with a new inner product and a new kernel so that $\mathcal{H}$ becomes a reproducing kernel Hilbert space itself. Since this will play a role in a later section we will be more precise here. Suppose $\Xi = \{\xi_1, \ldots, \xi_Q\} \subseteq \Omega$ with $Q = \dim(\mathcal{P})$ is $\mathcal{P}$-unisolvent, meaning that the zero function is the only function from $\mathcal{P}$ that vanishes on Ξ. Suppose further that $p_1, \ldots, p_Q$ form a Lagrange basis for $\mathcal{P}$, meaning in particular $p_j(\xi_i) = \delta_{ij}$. Then, we can define a new inner product

$$(f,g) := (f,g)_\mathcal{H} + \sum_{\ell=1}^{Q} p_\ell(x) p_\ell(y)$$

and a new kernel

$$\begin{aligned} K(x,y) = \Phi(x,y) &- \sum_{k=1}^{Q} p_k(x)\Phi(\xi_k,y) - \sum_{\ell=1}^{Q} p_\ell(y)\Phi(x,\xi_\ell) \\ &+ \sum_{\ell=1}^{Q}\sum_{k=1}^{Q} p_k(x)p_\ell(y)\Phi(\xi_k,\xi_\ell) + \sum_{\ell=1}^{Q} p_\ell(x)p_\ell(y). \end{aligned}$$

Proposition 6. *With this kernel and this inner product the native space $\mathcal{H}$ becomes a RKHS with kernel K.*

Hence, in many situations one can reduce the conditionally positive definite case to the positive definite one.

2.3. *Error Estimates and Condition Numbers*

To understand the numerical behavior of the interpolant or approximant it is essential to have bounds on the approximation error and on the condition number of the interpolation matrix. These bounds are usually expressed employing two different geometric measures. For the approximation error, it is crucial to know how well the data sites X fill the region Ω. This can be measured by the *fill distance*

$$h_{X,\Omega} := \sup_{x\in\Omega} \min_{x_j\in X} \|x - x_j\|_2,$$

which gives the radius of the largest "data-site free ball" in Ω. The condition number, however, will obviously only depend on the data sites X and not on the region Ω. Moreover, if two data sites tend to coalesce then the corresponding interpolation

matrix has two rows which are almost identical. Hence, it is reasonable to measure the condition number in terms of the *separation distance*

$$q_X = \frac{1}{2} \min_{j \neq k} \|x_j - x_k\|_2.$$

In the case of interpolation, i.e. for the solution of (1) interpolation error and condition number are well investigated. See for example [7–18] for error estimates and [19–25,15,2]) for investigations on the condition number. Both lists are far from being complete.

In our situation, when working in $H^\tau(\Omega)$ or $\mathrm{BL}_\tau(\Omega)$, respectively, error estimates for the solutions $s_0 = I_X f$ of (1) are of the form

$$\|D^\alpha (f - I_X f)\|_{L_p(\Omega)} \leq C h_{X,\Omega}^{\tau - |\alpha| - d(1/2 - 1/p)_+} \|f\|_{H^\tau(\Omega)}, \tag{4}$$

provided that $f \in H^\tau(\Omega)$, Ω is a bounded Lipschitz domain, and $\tau > |\alpha| + d/2$.

The condition number of the interpolation process is mainly determined by the smallest eigenvalue of the matrix A, i.e. by

$$\lambda_{\min}(A) = \inf \left\{ \frac{\alpha^T A \alpha}{\alpha^T \alpha} : \alpha \in \mathbb{R}^N \setminus \{0\} \text{ with } P^T \alpha = 0 \right\},$$

where the condition $P^T \alpha = 0$ can be neglected in the case of positive definite kernels. The asymptotic behavior of $\lambda_{\min}(A)$ has extensively been studied over the past years. It is now well known that $\lambda_{\min}(A)$ behaves like

$$\lambda_{\min}(A) \geq C q_X^{2\tau - d}. \tag{5}$$

The interpolation error (4) and the eigenvalue estimate (5) indicate that interpolation with smooth kernels will lead to a good approximation behavior at the price of an ill-conditioned problem.

In the case of the smoothing spline solution of (2) there exists also estimates on the approximation error and on the condition number but we will delay this discussion until later.

2.4. *Choosing an Appropriate Kernel*

By now it should be apparent, that the knowledge on the assumed smoothness of the unknown target function f leads automatically to a specific interpolation problem and hence to a specific kernel. The user should use this smoothness information, if at hand, to set up the correct interpolation problem.

If the precise smoothness of the the target function is unknown, it is better to overestimate the smoothness, as long as no ill-conditioning problems arise, and to work with a kernel that might be smoother than necessary.

It can be shown, that the resulting norm-minimal interpolant still produces optimal approximation rates, which are now determined by the rougher smoothness of f, as long as the data sets are quasi-uniform (see [26,12,27]).

3. Reducing the Complexity

The naive approach to solve the linear systems associated to the interpolation or smoothing problems leads to a complexity of $\mathcal{O}(N^3)$ time and $\mathcal{O}(N^2)$ space. Furthermore, every evaluation needs another $\mathcal{O}(N)$ time. For a large number N of data sites this is unacceptable. Thus more efficient methods are necessary. We will review some of the most promising approaches in this section.

3.1. *The Multipole Expansion*

In the case of a globally supported basis function and a large number of data sites it is impossible to use direct methods for solving the resulting linear equations. Instead, iterative methods have to be employed. For the time being, we do not want to discuss a specific iterative method here. It is only important to realize that the main operation in such an iterative method is a matrix by vector multiplication which takes generally $\mathcal{O}(N^2)$ operations. In our situation, the matrix vector multiplication reduces mainly to the evaluation of N sums of the form

$$s(x) = \sum_{j=1}^{N} \alpha_j \Phi(x, x_j) \tag{6}$$

neglecting the low order polynomial in the case of conditionally positive definite kernels.

Hence, it is crucial to find efficient algorithms that are able to perform this evaluation in less than $\mathcal{O}(N)$ time, preferably in $\mathcal{O}(\log N)$ or even constant time. Obviously we cannot achieve this goal if we want to reproduce the exact value of s at x. But since s is already an approximation to an unknown function, an additional error might be acceptable. Hence, we only try to approximate s up to a certain accuracy $\epsilon > 0$.

In the following, we will call t in $\Phi(x, t)$ a *source point* and x an *evaluation point*. The idea of *multipole expansions* is based on a *far field expansion* of Φ. Suppose all source points are situated in a certain region, also called *panel*, which is centered at a point t_0. Suppose further that we want to evaluate the function (6) at a point x that is sufficiently far away from the source panel. If we can expand Φ in the form

$$\Phi(x, t) = \sum_{k=1}^{p} \phi_k(x)\psi_k(t) + R(x, t) \tag{7}$$

with a remainder R that tends to zero for $\|x - t_0\|_2 \to \infty$ or for $p \to \infty$ if $\|x - t_0\|_2$ is sufficiently large, then we call (7) a *far field expansion* for Φ around the source t_0. Using (7) to evaluate (6) yields

$$s(x) = \sum_{j=1}^{N} \alpha_j \Phi(x, x_j)$$

$$
\begin{aligned}
&= \sum_{j=1}^{N} \alpha_j \sum_{k=1}^{p} \phi_k(x)\psi_k(x_j) + \sum_{j=1}^{N} \alpha_j R(x, x_j) \\
&= \sum_{k=1}^{p} \phi_k(x) \sum_{j=1}^{N} \alpha_j \psi_k(x_j) + \sum_{j=1}^{N} \alpha_j R(x, x_j) \\
&=: \sum_{k=1}^{p} \beta_k \phi_k(x) + \sum_{j=1}^{N} \alpha_j R(x, x_j).
\end{aligned}
$$

Hence, if we use the approximation $\widetilde{s}(x) = \sum_{k=1}^{p} \beta_k \phi_k(x)$ we have an error bound

$$
|s(x) - \widetilde{s}(x)| \le \|\alpha\|_1 \max_{1 \le j \le N} |R(x, x_j)|,
$$

which is small if x is far enough away from the sources x_j. Moreover, each coefficient β_k can be computed in advance in linear time. Thus, if p is much smaller than N, we can consider it as constant and we need therefore $\mathcal{O}(N)$ time to compute the coefficients $\{\beta_k\}$ and constant time for each evaluation of $\widetilde{s}$.

One could say that we have averaged the information given at the sources x_j to one information given at the center t_0 of the panel. In this sense, (7) is a unipole expansion of Φ. Note that in the case of translation invariant kernels it is easy to get the far field expansion of Φ around any t_0 if the far field expansion around 0 is known. To see this, suppose (7) is the far field expansion of $\Phi(x,t) = \Phi(x-t)$ around zero. Then we can write

$$
\Phi(x-t) = \Phi((x-t_0)-(t-t_0)) = \sum_{k=1}^{p} \phi_k(x-t_0)\psi_k(t-t_0) + R(x-t_0, t-t_0)
$$

to derive the far field expansion around t_0.

In general, the evaluation points are close to at least a few of the data sites. To cope with this situation, we have to refine our approach. The solution is a hierarchical subdivision of the region Ω of interest into panels or cells of sources. This is generally achieved using tree-based algorithms. Let us explain the idea by an example. Suppose all data points are contained in the interval $[0,1]$ and suppose we hierarchically divide this interval into panels as shown in Figure 1.

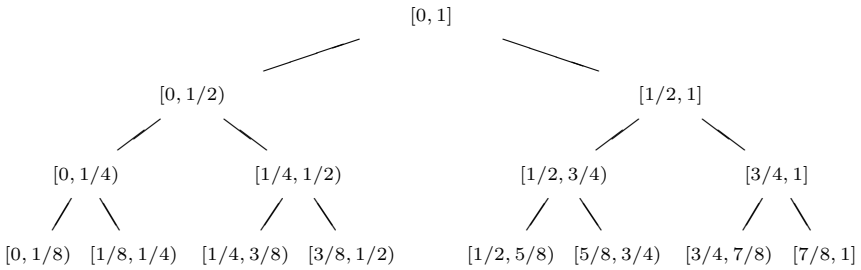

Fig. 1. Hierarchical decomposition of $[0,1]$.

With every source panel T we associate the part of the function s that corresponds to the sources in that panel by setting

$$s_T = \sum_{x_j \in T} \alpha_j \Phi(\cdot, x_j).$$

Moreover, we also assign the far field expansion $\widetilde{s}_T$ of s_T to the panel T.

Then, the approximation $\widetilde{s}$ to the function s at evaluation point x is computed by adding the associated functions s_T for the panel that contains x itself and all neighboring panels. Those panels that are well-separated from x contribute only by their far field expansion $\widetilde{s}_T$ to $\widetilde{s}$. Here, we say that a point x is *well-separated* from a panel T if it has at least the distance $\operatorname{diam}(T)$ from T. A panel U is called well-separated from a panel T if all points in U are well-separated from T. Since we want to save floating point operations we always use the largest possible source panel for the far field series.

Let us return to the example given in Figure 1. If we want to approximate $s(x)$ for x in the panel $[0, 1/8)$ we form

$$\widetilde{s}(x) = s_{[0,\frac{1}{8})}(x) + s_{[\frac{1}{8},\frac{1}{4})}(x) + \widetilde{s}_{[\frac{1}{4},\frac{3}{8})}(x) + \widetilde{s}_{[\frac{3}{8},\frac{1}{2})}(x) + \widetilde{s}_{[\frac{1}{2},\frac{3}{4})}(x) + \widetilde{s}_{[\frac{3}{4},1]}(x).$$

Note, that we use the two level 2 approximants $\widetilde{s}_{[\frac{1}{2},\frac{3}{4})}$ and $\widetilde{s}_{[\frac{3}{4},1]}(x)$ instead of the four level 3 approximants $\widetilde{s}_{[\frac{1}{2},\frac{5}{8})}, \ldots, \widetilde{s}_{[\frac{7}{8},1]}$. This halves the computational complexity in this case. We can do this because the panels $[1/2, 3/4)$ and $[3/4, 1]$ are well-separated from $[0, 1/4)$, the panel on the same level that contains x. On the other hand, we could not use the approximant $\widetilde{s}_{[\frac{1}{2},1]}$ because its panel $[1/2, 1]$ is not well-separated from the panel $[0, 1/2]$, which is the panel on the same level that contains x.

Similarly, to approximately evaluate $s(x)$ in the panel $[\frac{3}{8}, \frac{1}{2})$ we would use

$$\begin{aligned}\widetilde{s}(x) = {} & s_{[\frac{3}{8},\frac{1}{2})}(x) + s_{[\frac{1}{4},\frac{3}{8})}(x) + s_{[\frac{1}{2},\frac{5}{8})}(x) \\ & + \widetilde{s}_{[0,\frac{1}{8})}(x) + \widetilde{s}_{[\frac{1}{8},\frac{1}{4})}(x) + \widetilde{s}_{[\frac{5}{8},\frac{3}{4})}(x) + \widetilde{s}_{[\frac{3}{4},1]}(x).\end{aligned}$$

This finishes the description of the general idea of the multipole technique to speed up the evaluation of sums like (6). What still has to be done is to find a far field expansion for the underlying kernel. Since this, in general, has to be done for every kernel separately, it would go beyond the scope of this text. Hence, we refer the reader to the constructions, for example, in [28–36].

3.2. *The Domain Decomposition Method*

Now, we want to discuss one particular iterative method for interpolation, which has been introduced in [37]. The idea of this method is to subdivide the original data set into several smaller data sets and to iteratively solve the interpolation equations and to form residuals. To be more precise, let us decompose X in subsets $X_1, \ldots, X_k$. These subsets need not be disjoint but their union must be X. Then the algorithm starts to interpolate on the first set X_1, forms the residual, interpolates

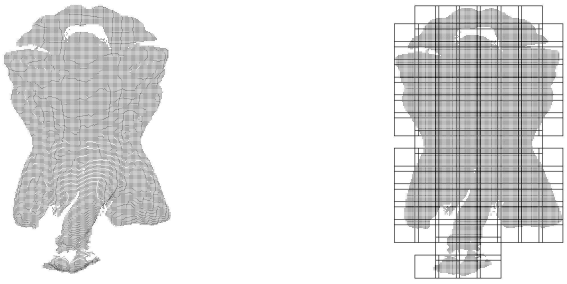

Fig. 2. The test data set consisting of 23574 points (left) and the decomposition (right).

this on X_2 and so on. After k steps one cycle of the algorithm is complete and it starts over again. A more formal description is

(i) Set $f_0 = f$, $s_0 = 0$.
(ii) For $n = 0, 1, 2, \ldots$
  For $r = 1, \ldots, k$
    $f_{nk+r} = f_{nk+r-1} - I_{X_r} f_{nk+r-1}$
    $s_{nk+r} = s_{nk+r-1} + I_{X_r} f_{nk+r-1}$
  If $\|f_{(n+1)k}\|_{L_\infty(X)} < \epsilon$ stop.

This algorithm approximates the interpolant $I_X f$ up to the specified accuracy. The convergence result is based upon the fact that the interpolant $s_0 = I_X f$ is also the *best approximant* to f from the subspace

$$V_X := \left\{ \sum_{j=1}^{N} \alpha_j \Phi(x, x_j) : P^T \alpha = 0 \right\} + \mathcal{P}.$$

Convergence is achieved under very mild assumptions on the decomposition. The data sets X_j have to be *weakly disjoint* meaning that $X_j \neq Y_j$ and $Y_{j+1} \neq Y_j$ for each $1 \leq j \leq k-1$, where $Y_j = \cup_{i=j}^{k} X_i$, $1 \leq j \leq k$. This is, for example, satisfied, if each X_j contains at least one data site, which is not contained in any other X_i.

Theorem 7. *Let $f \in \mathcal{H}$ be given. Suppose $X_1, \ldots, X_k$ are weakly distinct subsets of $\Omega \subseteq \mathbb{R}^d$. Set $Y_j = \cup_{i=j}^{k} X_i$, $1 \leq j \leq k$. Denote with $s^{(j)}$ the approximant after j completed cycles. Then there exists a constant $c \in (0,1)$ so that*

$$\|s_{f,Y_1} - s^{(n)}\|_{\mathcal{H}} \leq c^n \|f\|_{\mathcal{H}}.$$

For a proof of this theorem and for a more thorough discussion on how the subsets X_j have to be chosen we refer the reader to [37,2].

We want to demonstrate the behavior of this method as well as that of the ones still coming. To this end we applied all methods to the data set represented in the left part of Figure 2.

This data set consists of a moderate number of points; most of them are arranged in a grid like manner. However, as one can clearly see the data set contains

regions without points and the reader should compare the behavior of the numerical methods discussed in this paper in particular in these regions.

The results for the alternating-projection algorithm are shown in Figure 3. The point sets X_k have been chosen as axes parallel boxes containing at most 300 points, see the right part of Figure 2. The choice is not optimal and can be improved by an adaptive domain decomposition algorithm as we will employ it in the next subsection.

The first picture from the left in Figure 3 shows the reconstruction after one outer half cycle. The second and the third show the reconstruction after one completed outer cycle. While the reconstruction seems accurate from above, the side view reveals that the solution is not yet decent. The final two pictures present the reconstruction after 88 cycles where the residuum has dropped below 10^{-6}.

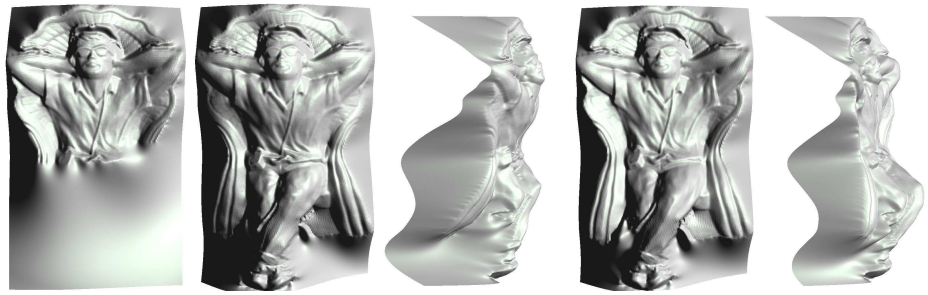

Fig. 3. Alternating-projection reconstruction.

3.3. *The Partition of Unity Idea*

The idea of a partition of unity method is comparable to the idea of the domain decomposition method. In the context of radial basis function interpolation, a partition of unity approach has first been introduced in [38,39].

This time not only the set of data sites is decomposed but the entire region Ω, i.e. we choose an overlapping covering $\{\Omega_j\}_{j=1}^M$ of Ω. Along with this covering we need a partition of unity, i.e. a family of compactly supported, nonnegative, continuous functions $\{w_j\}$ with $\operatorname{supp}(w_j) \subseteq \Omega_j$ and

$$\sum_{j=1}^{M} w_j(x) = 1, \qquad x \in \Omega.$$

Moreover, we choose for every cell Ω_j an approximation space V_j. Then, a function f is approximated on each cell by a local approximant $s_j \in V_j$ and the global approximant is formed by the weighted sum of local approximants:

$$s_f = \sum_{j=1}^{M} s_j w_j. \tag{8}$$

Note that this process respects an interpolation property, i.e. if all s_j are interpolants at $X \cap \Omega_j$ then s_f is an interpolant at the entire set X.

Moreover, even in this general situation, the simple estimate

$$\begin{aligned} |f(x) - s_f(x)| &= \left| \sum_{j=1}^{M} [f(x) - s_j(x)] w_j(x) \right| \\ &\leq \sum_{j=1}^{M} |f(x) - s_j(x)| w_j(x) \\ &\leq \max_{1 \leq j \leq M} \|f - s_j\|_{L_\infty(\Omega_j)} \end{aligned}$$

shows, that the global approximation error is governed by the worst local error. In other words, if the local approximants provide good approximations so will the global one. In our situation, the local approximants are formed as solutions of (1) or (2) where the global data set is replaced by the local data sets $X_j := \Omega_j \cap X$.

The complexity of this approach is governed by the following assumptions.

(i) We need a data structure for our point set such that we can find the data sites in each cell efficiently.

(ii) We need a cell structure such that
- each cell contains only a small number of points,
- each $x \in \Omega$ is contained only in a small number of cells,
- these cells can be found efficiently.

Note that these assumptions lead to the requirement that the number of cells M is proportional to the number N of points. This means in particular that the $\mathcal{O}(N)$ interpolation/approximation problems can be solved in $\mathcal{O}(N)$ time.

Again, tree-like decompositions lead to good data structures in both cases and, employing such data structures, one can show that the operations can be performed in $\mathcal{O}(\log N)$ time with an additional preprocessing step to build the data structure in $\mathcal{O}(N \log N)$ time.

The results for our test data sets are shown in the left part of Figure 4. The right part of that figure indicates how the overlapping boxes are chosen this time.

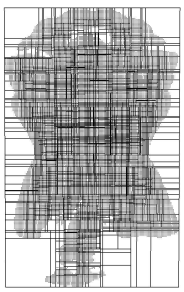

Fig. 4. Partition-of-unity reconstruction.

3.4. *Multilevel and Compactly Supported RBFs*

The numerical methods discussed so far are tailored for globally supported radial basis functions. In particular the domain decomposition method necessarily needs a multipole expansion for an efficient implementation. Even if compactly supported basis functions can be used as globally supported ones, this does not take the local character of these functions into account. Hence, it is time to discuss ideas for efficiently using compactly supported functions.

One possibility is to adjust the support radius as a function of the data density. Instead of using the function Φ one better employs the function $\Phi_\delta = \Phi(\cdot/\delta)$ and chooses δ as a function of the fill distance $h_{X,\Omega}$. The choice $\delta = ch_{X,\Omega}$ with a sufficiently small $c > 0$ will always lead to a sparse matrix. Moreover, since the interpolation matrix using Φ_δ and X coincides with the interpolation matrix using Φ and X/δ, we know from (5) that our scaled problem has a smallest eigenvalue, which behaves like

$$\lambda_{\min} \geq C \left(\frac{q_X}{\delta}\right)^{2\tau-d}.$$

Hence, in the case of quasi-uniform data, the choice $\delta = ch_{X,\Omega}$ leads to a stable interpolation problem.

Unfortunately, it is also possible to show that the error scales in a similar way. To see this, we cannot directly use (4), since the constant there includes a norm equivalence constant, which now depends also on δ. Nonetheless, it is possible to show the following result:

Theorem 8. *Let $\Omega \subseteq \mathbb{R}^d$ be a bounded Lipschitz domain satisfying an interior cone condition. If Φ is the reproducing kernel of $H^\tau(\Omega)$, $\tau = k + \sigma$ with $k > d/2$ and $0 < \sigma \leq 1$, then interpolation with the scaled kernel Φ_δ leads for $0 < \delta \leq 1$ to*

$$\|f - I_{X,\Phi_\delta} f\|_{L_2(\Omega)} \leq C \left(\frac{h_{X,\Omega}}{\delta}\right)^\tau \|f\|_{H^\tau(\Omega)},$$

with a constant $C > 0$ independent of $h_{X,\Omega}$, δ, and f.

Proof. Since the function $u := f - I_{X,\Phi_\delta} f \in W_2^\tau(\Omega)$ vanishes on X, a general result from [14] gives

$$\|f - I_{X,\Phi_\delta} f\|_{L_2(\Omega)} \leq C h_{X,\Omega}^\tau |f - I_{X,\Phi_\delta} f|_{H^\tau(\Omega)}$$

It remains to show that the expression $|f - I_{X,\Phi_\delta} f|_{H^\tau(\Omega)}$ can be bounded by a constant times $\delta^{-\tau}\|f\|_{H^\tau(\Omega)}$. To this end, we assume that the interpolant is given by

$$s := I_{X,\Phi_\delta} f = \sum_{j=1}^N c_j \Phi((\cdot - x_j)/\delta).$$

Moreover, since Ω is a Lipschitz domain, we can extend $f \in H^\tau(\Omega)$ continuously to a function from $H^\tau(\mathbb{R}^d)$. Next, the fractional semi-norm can, up to a constant, be expressed as

$$|s|^2_{H^\tau(\Omega)} = \sum_{|\alpha|=k} \int_\Omega \int_\Omega \frac{|D^\alpha s(x) - D^\alpha s(y)|^2}{\|x-y\|_2^{d+2\sigma}} dxdy,$$

which leads, with $\widetilde{s} = \sum_j c_j \Phi(\cdot - y_j)$, $y_j = x_j/\delta$, to

$$\begin{aligned}
|s|^2_{H^\tau(\Omega)} &\le |s|^2_{H^\tau(\mathbb{R}^d)} \\
&= \delta^{-2k} \sum_{|\alpha|=k} \int_{\mathbb{R}^d} \int_{\mathbb{R}^d} \frac{|D^\alpha \widetilde{s}(x/\delta) - D^\alpha \widetilde{s}(y/\delta)|^2}{\|x-y\|_2^{d+2\sigma}} dxdy \\
&= \delta^{d-2\tau} \sum_{|\alpha|=k} \int_{\mathbb{R}^d} \int_{\mathbb{R}^d} \frac{|D^\alpha \widetilde{s}(x) - D^\alpha \widetilde{s}(y)|^2}{\|x-y\|_2^{d+2\sigma}} dxdy \\
&= \delta^{d-2\tau} |I_{X/\delta,\Phi} f(\delta\cdot)|^2_{H^\tau(\mathbb{R}^d)} \\
&\le \delta^{d-2\tau} \|f(\delta\cdot)\|_{H^\tau(\mathbb{R}^d)}
\end{aligned}$$

using the fact that $\widetilde{s} = I_{X/\delta,\Phi} f(\delta\cdot)$ is the $\|\cdot\|_{H^\tau(\mathbb{R}^d)}$-norm minimal interpolant to $f(\delta\cdot)$. Finally, the latter norm can, up to a constant, be bounded by

$$\begin{aligned}
\|f(\delta\cdot)\|^2_{H^\tau(\mathbb{R}^d)} &= \int_{\mathbb{R}^d} |\widehat{f(\delta\cdot)}(\omega)|^2 (1+\|\omega\|_2^2)^\tau d\omega \\
&= \delta^{-2d} \int_{\mathbb{R}^d} |\widehat{f}(\omega/\delta)|^2 (1+\|\omega\|_2^2)^\tau d\omega. \\
&= \delta^{-d} \int_{\mathbb{R}^d} |\widehat{f}(\omega)|^2 (1+\|\omega\|_2^2\delta^2)^\tau d\omega \\
&\le \delta^{-d} \|f\|^2_{H^\tau(\mathbb{R}^d)}
\end{aligned}$$

provided that $\delta \le 1$. Putting things together leads to

$$|f-s|_{H^\tau(\Omega)} \le |f|_{H^\tau(\Omega)} + |s|_{H^\tau(\Omega)} \le (1 + C\delta^{-\tau}) \|f\|_{H^\tau(\Omega)} \le C\delta^{-\tau} \|f\|_{H^\tau(\Omega)},$$

which finally gives the stated result. □

Thus, the choice $\delta = ch_{X,\Omega}$ leads only to an acceptable error if c is sufficiently large, which stands in contrast to choosing c as small as possible. This is also reflected in the so called *uncertainty* or *trade-off principle*, see [15].

However, in practical applications generally only *one* data set is given and it is often possible to find a compromise in choosing an appropriate support radius. Nonetheless, our goal now is to describe a possible approach so that the application benefits from both a stable interpolation matrix and a good approximation property.

The idea of a multilevel method is again based on a decomposition of the set of data sites X, but this time in a nested sequence of subsets,

$$X_1 \subseteq X_2 \subseteq \ldots \subseteq X_k = X. \tag{9}$$

If X is quasi-uniform meaning that q_X has comparable size to $h_{X,\Omega}$, then the subsets X_j should also be quasi-uniform. Moreover, they should satisfy $q_{X_{j+1}} \approx c_a q_{X_j}$ and $h_{X_{j+1},\Omega} \approx c_a h_{X_j,\Omega}$ with a fixed constant c_a. A good choice for c_a would be $c_a = 1/2$.

Now, the *multilevel method*, introduced in [40], is simply one cycle of the domain decomposition method discussed in Section 3.2. But this time we use compactly supported basis functions with a different support radius at each level. We could even use different basis functions at different levels. Hence, a general formulation goes as follows. For every $1 \leq j \leq k$ we choose a basis function Φ_j and form the interpolant

$$I_j f := I_{X,\Phi_j} f = \sum_{x_j \in X_j} c_{x_j}(f)\Phi_j(\cdot - x_j),$$

but using now the basis function Φ_j on level j. We have in mind to take Φ_j as $\Phi(\cdot/\delta_j)$ with a compactly supported basis function Φ and scaling parameter δ_j proportional to $h_{X_j,\Omega}$. The idea behind this algorithm is that one starts with a very thin, widely spread set of points and uses a smooth basis function to recover the global behavior of the function f. In the next level a finer set of points is used and a less smooth function possibly with a smaller support is employed to resolve more details and so on.

As said before, the algorithm performs one cycle of the domain decomposition algorithm. This means

(i) set $f_0 = f$ and $s_0 = 0$.
(ii) for $1 \leq j \leq k$:
$s_j = s_{j-1} + I_j f_{j-1}$,
$f_j = f_{j-1} - I_j f_{j-1}$.

Even if the multilevel algorithm resembles the alternating projection algorithm, the idea behind it is completely different. The most obvious differences are that we use different basis functions at each level and that we perform only one cycle. The latter is reasonable since any further cycle would not change our interpolant because the data sets are nested.

The results of our test data sets are represented in Figure 5. The data set has been subdivided into a sequence of 5 nested subsets. The first row of Figure 5 shows the accumulated interpolants while the second row contains the residuals.

3.5. *Which Method to Choose?*

As mentioned earlier, the right choice of the basis function depends mainly upon additional information on the target function such as smoothness. The right choice of the particular reconstruction method depends also on the application. For example, if exact interpolation is necessary, any method that is based on a far field

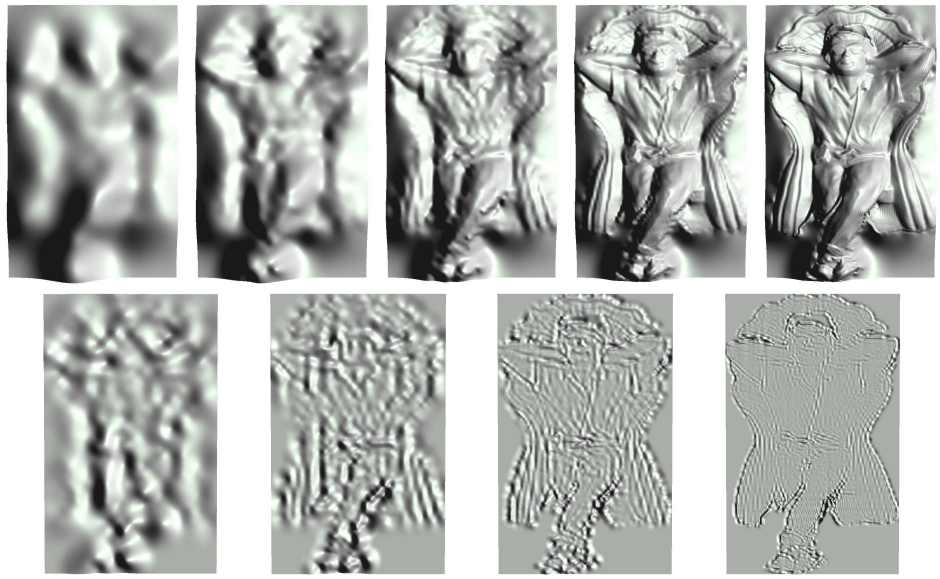

Fig. 5. Multilevel-reconstruction.

expansion is only of limited use, since such a method lives on the approximate character in the far field. On the other hand, if the data set has holes which have to be filled (e.g. for mesh repair) any purely local method is the wrong choice, even the partition of unity approach has to be handled with care here. Better suited is a global method such as domain decomposition or the multilevel method. In all other cases, it seems that, in particular for huge data sets, the partition of unity method is the fastest one.

4. Stabilizing the Interpolation Process and Dealing with Noise

4.1. *Error Estimates and Condition Numbers*

Our discussion of error estimates and condition numbers in Section 2.3 has in particular shown that, when working in $H^\tau(\Omega)$ or $\mathrm{BL}_\tau(\Omega)$, we are confronted with an error and a smallest eigenvalue behaving like

$$\begin{aligned} \|f - I_X f\|_{L_\infty(\Omega)} &\leq C h_{X,\Omega}^{\tau-d/2} \|f\|_{W_2^\tau(\Omega)} &&=: C_f F(h_{X,\Omega}) \\ \lambda_{\min}(A) &\geq c q_X^{2\tau-d} &&=: cG(q_X) \end{aligned}$$

and the reader should be aware of the fact that the exponent $2\tau - d$ of q_X in the lower bound of $\lambda_{\min}$ is precisely twice the exponent of $h_{X,\Omega}$ in the error estimate, or, with other words, $F(h^2) = G(h)$, using the above introduced functions. This is

indeed an intrinsic relation between error estimates and smallest eigenvalues. To be more precise, the *uncertainty relation* (cf. [15]) states roughly that

$$cG(q_X) \leq \lambda_{\min}(A) \leq CF(h_{X,\Omega}).$$

This has the unfortunate side-effect that in the case of quasi-uniform data sets, i.e. data sets where q_X and $h_{X,\Omega}$ are of comparable size, the smallest eigenvalue indeed behaves like

$$ch_{X,\Omega}^{2\tau-d} \leq \lambda_{\min}(A) \leq Ch_{X,\Omega}^{2\tau-d},$$

so that this is the best rate that we can expect for deriving an prescribed error of size $h_{X,\Omega}^{\tau-d/2}$.

Proposition 9. *The interpolation process derives an error of size $\mathcal{O}(h_{X,\Omega}^{\tau-d/2})$ with a smallest eigenvalue of size $\mathcal{O}(q_X^{2\tau-d})$. Moreover, the most stable interpolation process, where q_X is comparable to $h_{X,\Omega}$ still has a smallest eigenvalue of size $\mathcal{O}(h_{X,\Omega}^{2\tau-d})$.*

This proposition has direct consequences for any numerical strategy for solving the interpolation equations. In particular, we have to deal, at least, with the following two situations:

(i) The data set has a moderate sized $h_{X,\Omega}$ but a small q_X.
(ii) Both quantities $h_{X,\Omega}$ and q_X are small, but of comparable size.

If $h_{X,\Omega}$ is rather small, it might still happen that q_X is still essentially smaller than $h_{X,\Omega}$. However, the solution for the first case will show that we can avoid such a situation, and that it suffices to restrict ourselves then to the situation described in the second case.

In the first case, our point set is actually sparsely distributed over Ω with some almost identical points. Proposition 9 tells us that we achieve the accuracy at a cost that is too high, namely $\mathcal{O}(q_X^{2\tau-d})$, while a more "intelligent" chosen data set would lead to the same accuracy but with a smallest eigenvalue behaving like $\mathcal{O}(h_{X,\Omega}^{2\tau-d})$, which is, for a not too small $h_{X,\Omega}$ acceptable.

One strategy to avoid the situation described in the first case is to sort out nearly identical points in a preprocessing step. This, assuming a good data structure, can be done efficiently, as long as the number of coalescing points is small when compared to the number of all points.

However, sorting out coalescing points only makes sense if the information carried by two such points is also almost identical. But if, for example, the points x_j and x_k are almost identical and if the associated function values f_j and f_k differ significantly, additional information from the user is required to specify that point that is to be sorted out. Unfortunately, such a situation can and will quite often appear in real-world applications, where the data represents measured quantities that contain noise.

Fortunately, the solution of (2) gives another possibility for stabilizing the interpolation process by regularization. We will discuss this in the next subsection, while the final subsection deals with the situation in the second case.

4.2. *Regularization and Stabilization by Smoothing*

Suppose we still work in the spaces $H^\tau(\Omega)$ or $\mathrm{BL}_\tau(\Omega)$. Let s_λ denote the solution of the smoothing problem (2). From Theorems 2 and 4 we see that s_λ can be computed by inverting a slightly modified interpolation matrix. The modification simply consists in adding a λ to the diagonal of the main part A of the interpolation matrix. This already gives the first part of the following theorem. The second part has recently been proved in [41].

Theorem 10. *Suppose the data are generated by a function $f \in H^\tau(\Omega)$ or $f \in \mathrm{BL}_\tau(\Omega)$ respectively. Let s_λ be the solution of (2). Then, the smallest eigenvalue of the associated system matrices behaves like*

$$\lambda_{\min}(A + \lambda I) = \lambda_{\min}(A) + \lambda \geq C q_X^{2\tau - d} + \lambda,$$

where I denotes the $N \times N$ identity matrix. Moreover, the error between f and s_λ can be bounded by

$$\|D^\alpha (f - s_\lambda)\|_{L_p(\Omega)} \leq C \left(h_{X,\Omega}^{\tau - |\alpha| - d(1/2 - 1/p)_+} + h_{X,\Omega}^{-|\alpha|} \sqrt{\lambda} \right) \|f\|_{H^\tau(\Omega)},$$

provided that Ω is a bounded Lipschitz domain satisfying an interior cone condition, and $\tau = k + \sigma$ with $k > |\alpha| + d/2$.

If we look at the particular situation of pure $L_\infty(\Omega)$ error estimates we see that

$$\begin{aligned} \|f - s_\lambda\|_{L_\infty(\Omega)} &\leq C \left(h_{X,\Omega}^{\tau - d/2} + \sqrt{\lambda} \right) \|f\|_{H^\tau(\Omega)} \\ \lambda_{\min}(A + \lambda I) &\geq c q_X^{2\tau - d} + \lambda. \end{aligned}$$

Hence, choosing $\lambda = \widetilde{c} h_{X,\Omega}^{2\tau - d}$ leads to

$$\begin{aligned} \|f - s_\lambda\|_{L_\infty(\Omega)} &\leq C h_{X,\Omega}^{\tau - d/2} \|f\|_{H^\tau(\Omega)} \\ \lambda_{\min}(A + \lambda I) &\geq c q_X^{2\tau - d} + c h_X^{2\tau - d} = c h_X^{2\tau - d}. \end{aligned}$$

Corollary 11. *If the smoothing parameter $\lambda > 0$ is determined in this way then the best possible approximation order is achieved in the most stable way, meaning that the "interpolation matrix" has a largest possible smallest eigenvalue.*

Here, we have restricted ourselves to the most important case of L_∞-error estimates. But Theorem 10 indicates also how the smoothing parameter has to be

chosen if one is interested in estimates employing different L_p-norms and derivatives.

Theorem 10 can also be interpreted in a slightly different, more practicable way. To guarantee stability, the user can simply add an a priori λ to the diagonal entries of A, for example something like $\lambda = 10^{-6}$. Then, Theorem 10 implies stability and an error which is of size $\mathcal{O}(10^{-3})$, which, in many applications, is sufficient. As a matter of fact, since this is a worst case upper bound, the user can, in general, expect a much better behavior.

4.3. *Change of Basis*

So far, we have learned that smoothing is an adequate choice in the situation of highly non-uniform data sets. It also helps in the case of quasi-uniform data sets and infinitely smooth basis functions, like Gaussians and (inverse) multiquadrics, since their associated interpolation matrices are already highly ill-conditioned in that particular situation for moderate separation distances. Unfortunately there exists no theoretical coverage of error estimates in that situation, even if numerical test (cf. [41]) show promising results.

Our final task, for basis functions of finite smoothness, is to deal with the case of really dense data sets. Due to the last section we can restrict ourselves to quasi-uniform data sets.

In the first place, the bad conditioning in this situation is due to the naturally chosen basis, namely $\Phi(\cdot, x_1), \ldots, \Phi(\cdot, x_N)$ (plus a basis for $\mathcal{P}$), provided Φ is conditionally positive definite with respect to $\mathcal{P}$. Thus, it seems to be natural to investigate and to search for better suited bases for the subspace

$$V_X := \left\{ \sum_{j=1}^{N} \alpha_j \Phi(\cdot, x_j) : P^T \alpha = 0 \right\} + \mathcal{P},$$

where $P \in \mathbb{R}^{N \times Q}$ is the usual matrix with entries $p_j(x_i)$. The ideas we are going to describe now are in particular fitted to the situation of thin-plate splines, while they might also work for other conditionally positive definite kernels. They mainly come from [37].

To introduce a different basis of V_X, we first select a $\mathcal{P}$-unisolvent subset $\Xi = \{\xi_1, \ldots, \xi_Q\}$ of $X \subset \Omega$ and a cardinal basis $p_1, \ldots, p_Q$ for $\mathcal{P}$ satisfying $p_\ell(\xi_k) = \delta_{\ell,k}$. Then we define the kernels

$$\begin{aligned} \kappa(x, y) := \Phi(x, y) &- \sum_{k=1}^{Q} p_k(x) \Phi(\xi_k, y) - \sum_{\ell=1}^{Q} p_\ell(y) \Phi(x, \xi_\ell) \\ &+ \sum_{\ell=1}^{Q} \sum_{k=1}^{Q} p_k(x) p_\ell(y) \Phi(\xi_k, \xi_\ell) \end{aligned} \tag{10}$$

and

$$K(x,y) = \kappa(x,y) + \sum_{\ell=1}^{Q} p_\ell(x) p_\ell(y).$$

It is easily verified that the resulting functions $\kappa(\cdot, x_j)$ and $K(\cdot, x_j)$ belong to the space V_X defined above. However, κ has the additional feature $\kappa(\cdot, \xi_\ell) = 0$, $1 \le \ell \le Q$. The following results can be found in [37], see also [2].

Theorem 12. *The kernel $K : \Omega \times \Omega \to \mathbb{R}$ is positive definite on Ω. Moreover, if $\widetilde{\Omega} = \Omega \setminus \Xi$ then $\kappa : \widetilde{\Omega} \times \widetilde{\Omega} \to \mathbb{R}$ is positive definite on $\widetilde{\Omega}$.*

Thus we can restate our initial interpolation problem in two new ways. Let us start with the simpler one.

Corollary 13. *If $\Xi \subseteq X$ then the interpolant $I_X f$ can be written as*

$$I_X f = \sum_{j=1}^{N} \alpha_j K(\cdot, x_j),$$

where the coefficients are determined by $I_X f(x_j) = f_j$, $1 \le j \le N$.

When using κ we have to be more careful, since $\Xi \subseteq X$ does not lead to linearly independent functions $\kappa(\cdot, x_j)$. But we need $\Xi \subseteq X$ to ensure that we get the same interpolant. So assume that $x_j = \xi_j$ for $1 \le j \le Q$. Then we know at least that the matrix

$$C = (\kappa(x_i, x_j))_{Q+1 \le i,j \le N}$$

is positive definite. Or with other words the family $\{\kappa(\cdot, x_j) : Q+1 \le j \le N\}$ is linearly independent. Since $\kappa(\cdot, x_j) = 0$ for $1 \le j \le Q$, we can immediately conclude that $\{\kappa(\cdot, x_j) : Q+1 \le j \le N\} \cup \{p_k : 1 \le k \le Q\}$ is a basis for V_X.

Thus we can restate the interpolation problem using this basis.

Corollary 14. *If $\Xi \subseteq X$ satisfies $x_j = \xi_j$ for $1 \le j \le Q$ then the interpolant can be written as*

$$I_X f(x) = \sum_{j=1}^{Q} \beta_k p_k(x) + \sum_{j=Q+1}^{N} \alpha_j \kappa(x, x_j)$$

and the coefficients are again determined by $I_X f(x_j) = f(x_j)$, $1 \le j \le N$.

Since the $\{p_\ell\}$ form a Lagrangian basis for Ξ and since κ vanishes if one of its arguments is an element from Ξ the interpolation conditions lead to the matrix equation

$$\begin{pmatrix} I & O \\ \widetilde{P} & C \end{pmatrix} \begin{pmatrix} \beta \\ \alpha \end{pmatrix} = f|X$$

with the $Q \times Q$ identity matrix I and the $(N-Q) \times Q$ matrix $\widetilde{P} = (p_j(x_i))$ where i runs over the last indices starting with $i = Q+1$. The first Q rows in this system indicate $\beta = (f(x_1), \ldots, f(x_Q))^T$, so that we can reduce the system to solving

$$C\alpha = \widetilde{f} - \widetilde{P}\beta$$

where $\widetilde{f} = (f(x_{Q+1}), \ldots, f(x_N))$.

Hence, we now have three possibilities to solve the interpolation equations. We can either use the conventual approach using the $(N+Q) \times (N+Q)$-interpolation matrix $\widetilde{A}$ from Theorem 4, or one of our new approaches.

Table 1 shows the results for all three approaches using a regular grid of 25 points with grid size h. This reflects in particular the situation in the partition of unity approach, where the number of data sites is small but the fill and separation distance can become arbitrarily small. The results show that both new methods are superior to the conventional one, and that the system involving the matrix C even leads to a condition number which is independent of the parameter h.

Table 1
Condition numbers for a fixed number of centers.

Fill distance h	Conventional matrix $\widetilde{A}$	Reproducing kernel matrix K	Homogeneous matrix C
0.001	2.4349×10^8	8.4635×10^8	5.4938×10^3
0.01	2.4364×10^6	8.4640×10^6	5.4938×10^3
0.1	2.5179×10^4	8.5134×10^4	5.4938×10^3
1.0	3.6458×10^2	1.3660×10^3	5.4938×10^3
10	1.8742×10^6	1.2609×10^3	5.4938×10^3
100	1.1520×10^{11}	1.1396×10^5	5.4938×10^3
1000	3.5478×10^{15}	$1.1386e \times 10^7$	5.4938×10^3

The theoretical background for such a behavior is given in the next theorem. It states that for a specific class of basis functions, which includes thin-plate splines, the associated kernel κ is invariant under scaling. This means in particular, that the eigenvalues of the matrix C scale in the same way showing that the condition number of C is independent of the scaling parameter.

Theorem 15. *Suppose the symmetric kernel $\Phi \in C(\mathbb{R}^d \times \mathbb{R}^d)$ satisfies $\Phi(hx, hy) = h^\lambda \Phi(x,y) + q_h(x-y)$ for all $h > 0$ and $x, y \in \mathbb{R}^d$, where $\lambda \in \mathbb{R}$ and $q_h \in \pi_{2m-1}(\mathbb{R}^d)$. Let $\Xi = \{\xi_1, \ldots, \xi_Q\}$ be unisolvent for $\pi_{m-1}(\mathbb{R}^d)$ with associated Lagrange basis $p_1, \ldots p_Q$. Let κ be the kernel (10) and κ^h for $h > 0$ be the kernel κ for the set*

$h\Xi = \{h\xi_1, \dots, h\xi_Q\}$ and the Lagrange functions $p_1^h, \dots, p_Q^h$ associated to this set. Then $\kappa^h(hx, hy) = h^\lambda \kappa(x, y)$ for all $x, y \in \mathbb{R}$.

References

1. M. D. Buhmann, Radial Basis Functions, Cambridge Monographs on Applied and Computational Mathematics, Cambridge University Press, Cambridge, 2003.
2. H. Wendland, Scattered Data Approximation, Cambridge Monographs on Applied and Computational Mathematics, Cambridge University Press, Cambridge, UK, 2005.
3. N. Aronszajn, Theory of reproducing kernels, Trans. Am. Math. Soc. 68 (1950) 337–404.
4. G. Wahba, Spline Models for Observational Data, CBMS-NSF, Regional Conference Series in Applied Mathematics, Siam, Philadelphia, 1990.
5. R. Schaback, Native Hilbert spaces for radial basis functions I, in: M. W. Müller, et al. (Eds.), New developments in approximation theory. 2nd international Dortmund meeting (IDoMAT) '98, Germany, February 23-27, 1998, Vol. 132 of Int. Ser. Numer. Math., Birkhäuser Verlag, Basel, 1999, pp. 255–282.
6. R. Schaback, A unified theory of radial basis functions: Native Hilbert spaces for radial basis functions II, J. Comput. Appl. Math. 121 (2000) 165–177.
7. W. A. Light, H. Wayne, Error estimates for approximation by radial basis functions, in: S. P. Singh, A. Carbone (Eds.), Wavelet Analysis and Approximation Theory, Kluver Academic, Dordrech, 1995, pp. 215–246.
8. W. A. Light, H. Wayne, On power functions and error estimates for radial basis function interpolation, J. Approx. Theory 92 (1998) 245–266.
9. W. R. Madych, Error estimates for interpolation by generalized splines, in: P.-J. Laurent, A. L. Méhauté, L. L. Schumaker (Eds.), Curves and Surfaces, Academic Press, Boston, 1991, pp. 297–306.
10. W. R. Madych, S. A. Nelson, Error bounds for multiquadric interpolation, in: C. K. Chui, L. L. Schumaker, J. D. Ward (Eds.), Approximation Theory VI, Vol. 2, Academic Press, New York, 1989, pp. 413–416.
11. W. R. Madych, S. A. Nelson, Bounds on multivariate polynomials and exponential error estimates for multiquadric interpolation, J. Approx. Theory 70 (1992) 94–114.
12. F. J. Narcowich, J. D. Ward, Scattered-data interpolation on $\mathbb{R}^n$: Error estimates for radial basis and band-limited functions, SIAM J. Math. Anal. 36 (2004) 284–300.
13. F. J. Narcowich, J. D. Ward, H. Wendland, Refined error estimates for radial basis function interpolation, Constr. Approx. 19 (2003) 541–564.
14. F. J. Narcowich, J. D. Ward, H. Wendland, Sobolev bounds on functions with scattered zeros, with applications to radial basis function surface fitting, Math. Comput. 74 (2005) 643–763.

15. R. Schaback, Error estimates and condition number for radial basis function interpolation, Adv. Comput. Math. 3 (1995) 251–264.
16. R. Schaback, Improved error bounds for scattered data interpolation by radial basis functions, Math. Comput. 68 (1999) 201–216.
17. H. Wendland, Error estimates for interpolation by compactly supported radial basis functions of minimal degree, J. Approx. Theory 93 (1998) 258–272.
18. Z. Wu, R. Schaback, Local error estimates for radial basis function interpolation of scattered data, IMA J. Numer. Anal. 13 (1993) 13–27.
19. K. Ball, Eigenvalues of Euclidean distance matrices, J. Approx. Theory (1992) 74–82.
20. K. Ball, N. Sivakumar, J. D. Ward, On the sensitivity of radial basis interpolation to minimal data separation distance, Constr. Approx. 8 (1992) 401–426.
21. P. Binev, K. Jetter, Estimating the condition number for multivariate interpolation problems, in: D. Braess, et al. (Eds.), Numerical methods in approximation theory. Vol. 9: Proceedings of the conference held in Oberwolfach, Germany, November 24-30, 1991, Vol. 105 of Int. Ser. Numer. Math., Birkhäuser, Basel, 1992, pp. 41–52.
22. F. J. Narcowich, J. D. Ward, Norms of inverses and condition numbers for matrices associated with scattered data, J. Approx. Theory 64 (1991) 69–94.
23. F. J. Narcowich, J. D. Ward, Norms of inverses for matrices associated with scattered data, in: P.-J. Laurent, A. L. Méhauté, L. L. Schumaker (Eds.), Curves and Surfaces, Academic Press, Boston, 1991, pp. 341–348.
24. F. J. Narcowich, J. D. Ward, Norm estimates for the inverse of a general class of scattered-data radial-function interpolation matrices, J. Approx. Theory 69 (1992) 84–109.
25. F. J. Narcowich, J. D. Ward, On condition numbers associated with radial-function interpolation, J. Math. Anal. Appl. 186 (1994) 457–485.
26. R. Brownlee, W. Light, Approximation orders for interpolation by surface splines to rough functions, IMA J. Numer. Anal. 24 (2004) 179–192.
27. F. J. Narcowich, J. D. Ward, H. Wendland, Sobolev error estimates and a Bernstein inequality for scattered data interpolation via radial basis functions, college Station (2004).
28. R. K. Beatson, E. Chacko, Fast evaluation of radial basis functions: A multivariate momentary evaluation scheme, in: A. Cohen, C. Rabut, L. L. Schumaker (Eds.), Curve and Surface Fitting: Saint-Malo 1999, Vanderbilt University Press, Nashville, 2000, pp. 37–46.
29. R. K. Beatson, J. B. Cherrie, D. L. Ragozin, Polyharmonic splines in $\mathbb{R}^d$: Tools for fast evaluation, in: A. Cohen, C. Rabut, L. L. Schumaker (Eds.), Curve and Surface Fitting: Saint-Malo 1999, Vanderbilt University Press, Nashville, 2000, pp. 47–56.
30. R. K. Beatson, J. B. Cherrie, D. L. Ragozin, Fast evaluation of radial basis functions: Methods for four-dimensional polyharmonic splines, SIAM J. Math. Anal. 32 (2001) 1272–1310.

31. R. K. Beatson, G. Goodsell, M. J. D. Powell, On multigrid techniques for thin plate spline interpolation in two dimensions, Lect. Appl. Math. 32 (1996) 77–97.
32. R. K. Beatson, L. Greengard, A short course on fast multipole methods, in: M. Ainsworth, et al. (Eds.), Wavelets, multilevel methods and elliptic PDEs. 7th EPSRC numerical analysis summer school, University of Leicester, Leicester, GB, July 8–19, 1996, Clarendon Press, Oxford, 1997, pp. 1–37.
33. R. K. Beatson, W. A. Light, Fast evaluation of radial basis functions: Methods for two-dimensional polyharmonic splines, IMA J. Numer. Anal. 17 (1997) 343–372.
34. R. K. Beatson, G. N. Newsam, Fast evaluation of radial basis functions: I, Comput. Math. Appl. 24 (1992) 7–19.
35. R. K. Beatson, G. N. Newsam, Fast evaluation of radial basis functions: Moment-based methods, SIAM J. Sci. Comput. 19 (1998) 1428–1449.
36. G. Roussos, Computation with radial basis functions, Ph.D. thesis, Imperial College of Science Technology and Medicine, University of London (1999).
37. R. K. Beatson, W. A. Light, S. Billings, Fast solution of the radial basis function interpolation equations: Domain decomposition methods, SIAM J. Sci. Comput. 22 (2000) 1717–1740.
38. R. Franke, Locally determined smooth interpolation at irregularly spaced points in several variables, J. Inst. Maths. Applics. (1977) 471–482.
39. R. Franke, Smooth interpolation of scattered data by local thin plate splines, Comput. Math. Appl. 8 (1982) 273–281.
40. M. S. Floater, A. Iske, Multistep scattered data interpolation using compactly supported radial basis functions, J. Comput. Appl. Math. 73 (1996) 65–78.
41. H. Wendland, C. Rieger, Approximate interpolation, Göttingen (2004).