
Advances in Computational Mathematics **14:** 329–359, 2001.

Computational aspects of approximation to scattered data by using 'shifted' thin-plate splines

Jungho Yoon

Department of Applied Mathematics, School of Mathematical Sciences, Tel-Aviv University, Ramat Aviv, Tel-Aviv 69978, Israel
E-mail: yoon@post.tau.ac.il

Received 26 January 2000; accepted 20 February 2001
Communicated by R. Schaback

A new multivariate approximation scheme to scattered data on arbitrary bounded domains in $\mathbb{R}^d$ is developed. The approximant is selected from a space spanned (essentially) by corresponding translates of the 'shifted' thin-plate spline ('essentially,' since the space is augmented by certain functions in order to eliminate boundary effects). This scheme applies to noisy data as well as to noiseless data, but its main advantage seems to be in the former case. We suggest an algorithm for the new approximation scheme with a detailed description (in a MATLAB-like program). Some numerical examples are presented along with comparisons with thin-plate spline interpolation and Wahba's thin-plate smoothing spline approximation.

Keywords: radial basis function, approximation order, 'shifted' thin-plate spline, scattered data approximation, Gauss elimination by degree

AMS subject classification: 41A15, 41A25, 41A30, 41A63, 41A65, 65D10, 65D15

1. Introduction

Given a set of scattered points Ξ in $\Omega \subset \mathbb{R}^d$ and values $f|_\Xi$ (possibly contaminated) of some function f, our objective is to construct a function $s : \Omega \to \mathbb{R}^d$ such that, in some sense, s approximates f:

$$s \approx f \quad \text{on } \Omega.$$

This problem is usually referred to *scattered data approximation* and has many important applications. There are cases where the domain Ω is a rectangle and the points Ξ are uniformly gridded. There are, however, many other practical instances where Ω is of irregular shape and/or where the points Ξ are irregularly distributed on Ω. A large number of ideas have been proposed for the solution of this problem. In order for an approximation scheme to scattered data to be useful in practice, there are some important issues to be addressed:

(A) *Approximation power*. It is basic to require that the approximant s approximates f better as the point set Ξ becomes dense in Ω. In most cases, the approximation power is quantified by the asymptotic rate at which the error decays. The problem is how we can get the largest possible asymptotic rate when the function f is smooth.

(B) *Boundary effects*. Approximation near the boundary is a difficult problem. Because the data are usually given only inside the domain, the boundary effect is very serious, and eventually, one must lose some order of accuracy. Hence some special techniques are necessary for the approximation near the boundary.

(C) *Numerical stability*. An approximation scheme should be local in the sense that the contribution to the approximant's value at a point x by the data value at $\xi \in \Xi$ decreases (fast!) as the distance between x and ξ increases. At the same time, many of the approximation methods are using (for good reasons) basis functions that are neither compactly supported, nor decay at ∞. Hence, in order to circumvent this initial instability, a 'localization process' is necessary. The localness of the scheme also ensures that 'boundary effects' do not spill over into the interior of the domain.

(D) *Noise*. Noisy data arise in many scientific applications according to the model

$$y_\xi = f(\xi) + \varepsilon_\xi, \quad \xi \in \Xi,$$

where, for example, ε_ξ's are independent noise with mean 0 and with (known or unknown) variance σ^2. In this case, the approximation scheme should have smoothing effects.

One of the well-known approaches for the scattered data approximation is the use of piecewise-polynomials. In this case Ω has to be partitioned into suitable regions, different polynomials are employed on the different regions separately, and usually the pieces have to be joined in a smooth way. In the multivariate case, however, this problem is computationally expensive. For example, the evaluation of an approximant at a given point requires one to identify the polynomial piece relevant to the point.

Other techniques are based on forming suitable linear combinations of certain radially symmetric basis functions. In particular, one may employ the translates along Ξ of one such fixed function ϕ. This approximation method has the general form

$$s(x) := \sum_{\xi \in \Xi} c_\xi \phi(x - \xi), \quad x \in \Omega. \tag{1.1}$$

The set of scattered points Ξ in Ω by which a radial basis function ϕ is shifted is referred to as a set of 'centers.' The use of a radially symmetric basis function ϕ is particularly useful because this facilitates the evaluation of the approximant, while still leaving enough flexibility in the choice of ϕ. The common choices of ϕ include: $\phi(x) = |x|^\lambda \log|x|$, d, λ even, (thin-plate spline), $\phi(x) = (|x|^2 + c^2)^{\lambda/2}$, λ, d odd, (multiquadric), and $\phi(x) = \exp(-c|x|^2)$, $c > 0$ (Gaussian). One must choose the basis function ϕ very carefully so that approximation schemes of the form (1.1) are effective (i.e., error decay) in some sense. In view of the discussion so far, one might wonder why compactly supported functions are not in this list (e.g., box splines). The answer

is that, in general, because such basis functions do not respect the geometrical aspect of points, it can not yield a good approximation scheme. We refer the reader to [17] for more details.

Some basis functions $\phi : \mathbb{R}^d \to \mathbb{R}$ that are well known (e.g., multiquadric, thin-plate spline) are not suitable to be used directly for the purposes of approximation since they increase polynomially fast around ∞. However, a suitable bell-shaped

$$\psi(x) = \sum_{\alpha \in \mathbb{Z}^d} \mu(\alpha)\phi(x - \alpha) \tag{1.2}$$

can be found by applying a difference operator to ϕ appropriately. There are several reviews about the state-of-the-art in the investigation of radial basis function methods. The reader is referred to the surveys [6,15] for the description of these works.

The initial approach to scattered data using radial basis functions has been focused on interpolation at the scattered points $\Xi \subset \mathbb{R}^d$. The general conditions on ϕ that ensure the existence and the uniqueness of a solution of the interpolation problem have been given by Micchelli [12]. The reader is also referred to the work of Madych and Nelson [13,14]: there, the approach of reproducing kernel Hilbert spaces is used. That approach is suitable when we approximate functions that lie in the underlying Hilbert space (see also [14,18]). More recently, Johnson [10] established an asymptotic upper bound on the approximation order on the unit ball Ω in $\mathbb{R}^d$ for the basis function ϕ of the form $\phi = |\cdot|^{\lambda/2}$ for d, λ odd, and $\phi = |\cdot|^{\lambda/2} \log |\cdot|$ for d, λ even.

Interpolation by translates of suitable radial basis functions is certainly an important approach towards solving the scattered data problem. However, it carries its own disadvantages. For example, for a large class of basis functions (including multiquadric and Gaussian), the existing theory guarantees the interpolant to approximate well only for a very small class of approximands (see [14]). The approximands need to be extremely smooth for an efficient error analysis. Another disadvantage of the interpolation method is that, with the increase in the number of centers, one needs to solve a large linear system which is very ill-conditioned. Most importantly, when the given data are contaminated, the interpolation method should not be used. All in all, there is an overwhelming need for approximation methods other than interpolation. Hence, our main concern in this study is to provide an approximation scheme on bounded domains that addresses all the issues (A)–(D) discussed in the beginning of this section.

Approximations of the type (1.1) that are *not* interpolatory are also discussed intensively in the literature. However, most of the results there deal with the case when the center set Ξ is infinite and uniform, i.e., a scale $h\mathbb{Z}^d$ of the integer lattice $\mathbb{Z}^d$. There has been much less investigation into the study of approximation for the general case of Ξ in $\mathbb{R}^d$. Buhmann et al. [5] were among the first to construct a non-interpolatory approximation scheme for infinitely many scattered centers and to analyze its approximation power. Dyn and Ron [8] showed that the scheme in [5] can be understood as 'an approximation to a uniform mesh approximation scheme'. In both papers, quasi-interpolations from radial basis function space with infinitely many centers Ξ were studied and both papers showed that the approximation orders obtained in the scattered case are identical

to those that had been known on uniform grids. In particular, Dyn and Ron provide a general tool that allows us to convert *any* known approximation scheme on uniform grids to non-uniform grid, while preserving (to the extent that this is possible) the approximation orders known in the former case. The approach of [8] can be described as follows. Suppose that an approximation scheme of the form

$$f \mapsto \sum_{\alpha\in\mathbb{Z}^d} \lambda_\alpha(f)\phi(\cdot-\alpha)$$

is given. Then, we replace each $\phi(\cdot-\alpha)$ by a suitable combination

$$\sum_{\xi\in\Xi} C(\alpha,\xi)\phi(\cdot-\xi), \tag{1.3}$$

with Ξ a set of scattered centers we wish to use. This result, however, requires us to select the density of the uniform grid that is associated with the given scattered set Ξ.

In [20], a new approximation scheme on $\mathbb{R}^d$ was developed on the general idea of [8]. This scheme is intrinsically 'scattered', i.e., employs directly the scattered shifts of the basis function ϕ. This study chooses the basis function to be the 'shifted' thin-plate splines

$$\phi_c(x) := \begin{cases} \big(|x|^2+c^2\big)^{\lambda/2}, & \lambda\in\mathbb{Z}^d_+,\ \lambda, d \text{ odd},\\ \big(|x|^2+c^2\big)^{\lambda/2}\log\big(|x|^2+c^2\big)^{1/2}, & \lambda\in\mathbb{Z}^d_+,\ \lambda, d \text{ even}. \end{cases} \tag{1.4}$$

The properties of these basis functions are quite well understood, both theoretically and practically. One of the reasons for choosing this particular function ϕ_c is the desire to use the parameter c as a 'tension' parameter. Note that we stress this tension parameter by using the notation ϕ_c. In particular, letting ω be a parameter depending on h, i.e.,

$$\omega := \omega(h),$$

the parameter c is selected to satisfy the relation

$$\rho := \frac{c}{\omega} \tag{1.5}$$

for the purpose of constructing a numerically stable scheme. Then this scheme provides spectral approximation orders (i.e., approximation order that depends only on the smoothness of the function f we approximate).

Our study in this paper treats approximation on the arbitrary bounded domain in $\mathbb{R}^d$ by using the same function ϕ_c, whereas the paper [20] treats the case over $\mathbb{R}^d$. Throughout this paper, we assume that the parameter c satisfies the relation (1.5).

A typical form of approximation scheme considered in [20] can be written as

$$Rf := \int_{\mathbb{R}^d} K_{c,\omega}(\cdot,t)(\Lambda f)(\omega t)\,dt, \tag{1.6}$$

where Λ is a bounded operator of the form $\Lambda f = m_{c,\omega} * f$ with $m_{c,\omega}*$ a mollifier, and the kernel $K_{c,\omega}(\cdot,\cdot)$ depends on tension parameters c (that appear in ϕ_c) and ω. Here and in the sequel, for the sake of simplicity, we use the notation

$$K(\cdot,\cdot) := K_{c,\omega}(\cdot,\cdot).$$

The fundamental properties of $K(\cdot,\cdot)$ are as follows:

(a) for a fixed $t \in \mathbb{R}^d$, $K(\cdot,t) \in S_\Xi(\phi_c)$,
(b) for a fixed $x \in \mathbb{R}^d$, $K(x,\cdot) \in L_1(\mathbb{R}^d)$.

The natural way to pass from this approximant to one which is suitable for arbitrary bounded domain Ω is to truncate as follows:

$$Lf := \int_{\Omega/\omega} K(\cdot,t)(\Lambda f)(\omega t)\,dt. \tag{1.7}$$

In practical points of view, we make two observations on this form. First, in most cases, the function f is unknown, but the (only known) data are $f|_\Xi$ or the possibly contaminated values $(y_\xi)_{\xi\in\Xi}$. Hence, by using the given data, we need to find a function, say F, which can replace the underlying function f. Second, such truncation is known to be too naive to approximate near the boundary. Thus, in order to eliminate the boundary effects, we select our approximant from a space spanned 'essentially' by the corresponding translates $\phi_c(\cdot-\xi)$, $\xi \in \Xi$, of the basis function ϕ_c. Specifically, we augment the set of shifts $\phi_c(\cdot-\xi)$ by adding to Ξ some more centers around Ω and then look for the approximant from the extended space. For convenience, we use the same notation Ξ to indicate the extended center set.

Letting $\Omega \subset \Omega_\delta \subset \Omega_{2\delta}$, our approximation scheme on the domain Ω is of the form

$$L : f \mapsto \int_{\Omega_\delta/\omega} K(\cdot,t)\Lambda(\chi_{\Omega_{2\delta}} F)(\omega t)\,dt, \tag{1.8}$$

where F is a replacement of f (if the data are known only at Ξ). The advantages of this scheme are as follows:

(i) The scheme has a 'smoothing' effect when the given data are contaminated. In fact, our scheme L consists of two smoothing steps (see section 4). Hence, the major advantage of the scheme L seems to be in noisy data approximation.

(ii) The potential numerical instability in the scheme is overcome by deriving a computationally stable 'local' algorithm for the computation and evaluation of the approximant.

(iii) The scheme is adjusted to deal properly with bounded domains. This is done by adding a 'predictor step' to the algorithm. It seems that this scheme has another issue of numerical integration but there is a discretization which converges to the above scheme fast(!) enough as the mesh size of the discretization tends to zero. We will discuss this point later.

By imposing some conditions on the kernel $K(\cdot,\cdot)$, for every sufficiently smooth function f, we obtain the error estimate of the form

$$\|f - Lf\|_{L_\infty(\Omega)} = O\big(h^k\big), \quad \text{as } h \to 0,$$

with h defined as

$$h := \sup_{x\in\Omega} \inf_{\xi\in\Xi} |x - \xi|, \tag{1.9}$$

and where k depends on the conditions satisfied by $K(\cdot,\cdot)$ and the smoothness of f. The initial numerical tests reveal that the algorithm gives results comparable to, or better than, the state-of-art method for both noiseless data and noisy data.

The structure of this paper is as follows. Section 2 is devoted to the constructions of the kernel $K(\cdot,\cdot)$ and a function F by using the given data $f|_\Xi$ or noisy data $(y_\xi)_{\xi\in\Xi}$. Section 3 provides the specific form of our scheme L and observes its approximation behavior on the bounded domain Ω. In particular, to avoid the boundary effect, an extrapolation method will be adopted. We derive an error estimate that is expressed in terms of h and L_∞-Sobolev semi-norm of f. Indeed, it depends on the conditions of the localized kernel K and smoothness of f. Section 4 explores the smoothing effect of the scheme L. We will see how the scheme L works in case the given data set is contaminated. A numerical example is provided with smoothing steps by L. Finally, in section 5, we provide an algorithm for the computation of Lf. The crucial part of this algorithm is a method for constructing a suitable coefficient sequence $(C(t,\xi))_{\xi\in\Xi}$ for the pseudo-shift $\phi_c(\cdot,t)$ in (1.3). Specifically, 'Gauss elimination by degree', which was introduced by de Boor and Ron [4], is applied to a linear system generated by some basis of a polynomial space. We give a detailed description (in a MATLAB-like program). In addition, some numerical examples are provided in section 6.

Throughout this paper we use the following notations. For $x = (x_1, \dots, x_d)$ in $\mathbb{R}^d$, $|x| := (x_1^2 + x_2^2 + \cdots + x_d^2)^{1/2}$ stands for its Euclidean norm and, for $\alpha \in \mathbb{Z}_+^d := \{\beta \in \mathbb{Z}^d\colon\ \beta \geqslant 0\}$, we set $|\alpha|_1 := \sum_{k=1}^d \alpha_k$. Let Π_k denote the space of all polynomials of degree $< k$ in d variables. For the given function ϕ_c and a finite set $\Xi \subset \Omega$, we define

$$S_\Xi(\phi_c) := \operatorname{span}\big\{\phi_c(\cdot - \xi)\colon\ \xi \in \Xi\big\}.$$

The Fourier transform of $f \in L_1(\mathbb{R}^d)$ is defined as

$$\hat{f}(\theta) := \int_{\mathbb{R}^d} f(t)\mathrm{e}^{-\mathrm{i}\theta\cdot t}\,dt.$$

Also, for a function $f \in L_1(\mathbb{R}^d)$, we use the notation $f^\vee$ for the inverse Fourier transform.

In particular, we are interested in approximating smooth functions in the space

$$W_\infty^k\big(\mathbb{R}^d\big), \quad k \in \mathbb{Z}_+,$$

of all functions whose derivatives of orders $\leqslant k$ are bounded. By $|\cdot|_{k,\infty}$, we shall denote the homogeneous kth order L_∞-Sobolev semi-norm, i.e.,

$$|f|_{k,\infty} = \sum_{|\alpha|_1=k} \left\| D^\alpha f \right\|_\infty.$$

2. Preliminary

We note from (1.4) that since the basis function ϕ_c grows at a certain polynomial degree away from zero, a 'localization process' is necessary to circumvent the initial instabilities. Usually, localization is done by applying a difference operator to ϕ_c, which constructs a bell-shaped function

$$\psi_c = \sum_{\alpha\in N} \mu(\alpha)\phi_c(\cdot - \alpha),$$

where N is a finite subset of $\mathbb{Z}^d$ (generally a milder condition is imposed on μ) and the localized function ψ_c is assumed to satisfy the conditions

$$\sup_x \big(1 + |x|\big)^{d+q} \psi_c(x) < \infty, \quad \hat{\psi}_c(0) \neq 0 \quad \text{and} \quad \hat{\psi}_c \in C^{d+q}\big(\mathbb{R}^d\big) \tag{2.1}$$

for a positive integer $q > 0$. Then, for reasons of numerical stability, we need to employ the scaled localization

$$\psi_\rho(\cdot/\omega - t) = \psi_{c/\omega}(\cdot/\omega - t) = \sum_{\alpha\in N} \mu(\alpha) \frac{\phi_c(\cdot - (t+\alpha)\omega)}{\omega^\lambda}, \tag{2.2}$$

for every $t \in \mathbb{R}^d$.

The kernel $K(\cdot, t)$ is considered as a counterpart of $\psi_\rho(\cdot/\omega - t)$ in the space $S_{\phi_c}(\Xi)$. The construction of $K(\cdot, t)$ is done as follows. We first approximate each shift $\phi_c(\cdot - t)$ by

$$\phi_c(\cdot, t) := \sum_{\xi\in\Xi} C(t, \xi)\phi_c(\cdot - \xi) \in S_\Xi(\phi_c), \tag{2.3}$$

which is referred as a 'pseudo-shift' of ϕ_c. Next, by replacing each shift $\phi_c(\cdot - t)$ in (2.2) by pseudo-shift $\phi_c(\cdot, t)$, we define $K(\cdot, t)$ by

$$K(\cdot, t) := \sum_{\alpha\in N} \mu(\alpha) \frac{\phi_c(\cdot, (t+\alpha)\omega)}{\omega^\lambda}. \tag{2.4}$$

It is obvious from the above discussion that $K(\cdot, t) \in S_\Xi(\phi_c)$ for each $t \in \mathbb{R}^d$. Note also that K depends on the locations of the centers Ξ.

We now introduce the notion of 'admissible coefficients' $(C(\cdot, \xi))_{\xi\in\Xi}$.

Definition 2.1. The coefficients $(C(\cdot, \xi))_{\xi\in\Xi}$ are termed *admissible* for Π_n on Ω if they satisfy the following three conditions:

(a) There exists $c_1 > 0$ such that, for any $t \in \Omega$, $C(t, \xi) = 0$ whenever $|t - \xi| > c_1 h$, with h the density of Ξ as in (1.9).

(b) The set $\{(C(t, \xi))_{\xi \in \Xi} : t \in \Omega\}$ is bounded in $\ell_1(\Xi)$.

(c) For every $t \in \Omega$, $\sum_{\xi \in \Xi} C(t, \xi)\delta_\xi = \delta_t$ on Π_n, i.e.,

$$\sum_{\xi \in \Xi} C(t, \xi) p(\xi) = p(t), \quad \forall p \in \Pi_n. \tag{2.5}$$

Remark. In the above definition, the centers

$$\Xi_t := \big\{\xi \in \Xi\colon\ C(t, \xi) \neq 0\big\}, \quad t \in \Omega,$$

are assumed to be some 'close neighbors' of t. Of course, the set Ξ_t is required to have the nondegeneracy property for Π_n

$$(p|_\Xi = 0,\ p \in \Pi_n) \quad \text{implies} \quad p \equiv 0. \tag{2.6}$$

This implies that $\#X_t$ should be no smaller than $\dim \Pi_n(\mathbb{R}^d)$ where $\#S$ denotes the number of elements in S.

Remark. It is shown (see [8,20]) that if $(C(t, \xi))_{\xi \in \Xi}$ is admissible for Π_n with $n > \lambda + d$, the difference $\phi_c(\cdot - t) - \phi_c(\cdot, t)$ satisfies the relation

$$\big|\phi_c(x - t) - \phi_c(x, t)\big| \leqslant c_1\big(1 + |x - t|\big)^{-m_1}, \quad m_1 > d,$$

with c_1 independent of x and t. It is immediate from this relation that

$$\left|\psi_\rho\left(\frac{x}{\omega} - t\right) - K(x, t)\right| \leqslant c_2\left(1 + \left|\frac{x}{\omega} - t\right|\right)^{-m_2}, \quad m_2 > d, \tag{2.7}$$

where c_2 is independent of x, t and ω but depends on ρ.

On the other hand, assume that f is known only at a set Ξ with the form

$$y_\xi = f(\xi) + \varepsilon_\xi,$$

where all the ε_ξ's are zero or ε_ξ's, for example, are independent noise with mean 0 and with variance σ^2. Then we approximate other function values by

$$F(t) := \sum_{\xi \in \Xi} C_f(t, \xi) y_\xi, \tag{2.8}$$

where $C_f(t, \xi)$ is admissible coefficients for Π_n with some $n > 0$. We use the notation C_f instead of C since they may be implemented by (computationally) different algorithms.

3. Approximation on bounded domains

In this section, we provide the specific form of the scheme L and then look for an approximation power of L on Ω under the assumption that the given data are noiseless, i.e., are of the form $(\Xi, f|_\Xi)$. In particular, approximation near the boundary is a difficult problem. Since there is no information available outside the domain, deterioration in fidelity of the approximation near the boundary is unavoidable. For example, an asymptotic upper bound on the approximation order of thin-plate interpolation on the unit ball in $\mathbb{R}^2$ is $O(h^{5/2})$ while $O(h^4)$ is available inside the domain (see [10]). Some special care is necessary in order to eliminate boundary effects. To make this specific, we approach this problem by adding additional new centers to Ξ around Ω and use an extended set of shifts $\phi_c(\cdot - \xi)$ to find approximants. For this, we define a superset of Ω by

$$\Omega_\delta := \{y = x + z\colon\ x \in \Omega \text{ and } |z| \leqslant \delta\} = \Omega + B_\delta.$$

We think of δ as either being fixed, or decreasing to 0 as the density of Ξ increases.

With F and K in (2.8) and (2.4) respectively, our approximation scheme L is defined by

$$L : f \mapsto \int_{\Omega_\delta/\omega} K(\cdot, t)\Lambda(\chi_{\Omega_{2\delta}} F)(\omega t)\, dt,$$

where χ_Ω is the characteristic function of Ω, and Λ is an operator on $L_\infty(\mathbb{R}^d)$ defined by

$$\Lambda : g \mapsto \left(\frac{\sigma_\omega}{\hat{\psi}_\rho(\omega\cdot)}\right)^\vee * g \tag{3.1}$$

with $\sigma_\omega : x \mapsto \sigma(\omega x)$, and $\sigma : \mathbb{R}^d \to [0, 1]$ a nonnegative C^∞-cutoff function such that support σ lies in the ball $B_\eta := \{x \in \mathbb{R}^d\colon\ |x| < \eta\} \subset [-2\pi, 2\pi]^d$ with $\sigma = 1$ on $B_{\eta/2}$ and $\| \sigma \| = 1$.

Remark. Invoking the definition of $K(\cdot, t)$ in (2.4), the approximant Lf has the explicit form

$$Lf(x) = \sum_{\xi\in\Xi} \phi_c(x - \xi) \sum_{\alpha\in N} \mu(\alpha) c_{\xi,\alpha}(f)$$

with

$$c_{\xi,\alpha}(f) := \int_{\Omega_\delta} \frac{C((\omega(t + \alpha), \xi)\Lambda(\chi_{\Omega_{2\delta}} F)(\omega t))\, dt}{\omega^\lambda}.$$

It ensures that the approximant Lf belongs to $S_\Xi(\phi_c)$.

Remark. From (3.1), we observe that

$$\Lambda(\chi_{\Omega_{2\delta}} F) = \int_{\Omega_{2\delta}} \left(\frac{\sigma}{\hat{\psi}_\rho}\right)^\vee \left(\frac{\cdot - s}{\omega}\right) F(s)\, ds. \tag{3.2}$$

Here, we find that $\sigma/\hat{\psi}_\rho \in C^{2\lambda+2d-1}(\mathbb{R}^d)$ (see [7]) and $(\sigma/\hat{\psi}_\rho)^\vee$ decays at some polynomial rate. Thus, we realize that the localization property of L is due to the decaying properties of the kernel K and $(\sigma/\hat{\psi}_\rho)^\vee$ in the sense that the contribution to the approximant's value at a point x by the data value at $\xi \in \Xi$ decreases as the distance between x and ξ increases.

Now, for any $x \in \Omega$, the error analysis is based on the estimate

$$\begin{aligned} \big|f(x) - Lf(x)\big| \leqslant{}& \left|\sigma_\omega^\vee * f(x) - \int_{\Omega_\delta/\omega} \psi_\rho\left(\frac{x}{\omega} - t\right) \Lambda f(\omega t)\, dt\right| \\ &+ \left|\int_{\Omega_\delta/\omega} \left(\psi_\rho\left(\frac{x}{\omega} - t\right) - K(x,t)\right) \Lambda f(\omega t)\, dt\right| \\ &+ \left|\int_{\Omega_\delta/\omega} K(x,t)\Lambda(\chi_{\Omega'_{2\delta}} f)(\omega t)\, dt\right| \\ &+ \left|\int_{\Omega_\delta/\omega} K(x,t)\Lambda\big(\chi_{\Omega_{2\delta}}(f - F)\big)(\omega t)\, dt\right| + \big|f(x) - \sigma_\omega^\vee * f(x)\big|, \end{aligned} \tag{3.3}$$

where Ω' indicates the complement set of Ω in $\mathbb{R}^d$.

The next lemma treats the first term in (3.3).

Lemma 3.1. Let ψ_c satisfy the conditions in (2.1) and Λ be as in (3.1). Assume that $c = \rho\omega$ for some $\rho > 0$. Then, for every $f \in L_\infty(\mathbb{R}^d)$,

$$\left\|\sigma_\omega^\vee * f - \int_{\Omega_\delta/\omega} \psi_\rho(\cdot/\omega - t)\Lambda f(\omega t)\, dt\right\|_{L_\infty(\Omega)} = O\big(\omega^q\big),$$

where q is as in (2.1).

Proof. Using the representation

$$\sigma_\omega^\vee * f = \int_{\mathbb{R}^d} \psi_\rho(\cdot/\omega - t)\Lambda f(\omega t)\, dt,$$

we have the expression

$$\sigma_\omega^\vee * f - \int_{\Omega_\delta/\omega} \psi_\rho(\cdot/\omega - t)\Lambda f(\omega t)\, dt = \int_{\Omega'_\delta/\omega} \psi_\rho(\cdot/\omega - t)\Lambda f(\omega t)\, dt. \tag{3.4}$$

By estimating

$$\left|\Lambda(f)(t)\right| = \left|\int_{\mathbb{R}^d} \left(\frac{\sigma}{\hat{\psi}_\rho}\right)^\vee (s) f(t-\omega s)\, d\theta\right| \leqslant \|f\|_{L_\infty(\Omega)} \left\|\left(\frac{\sigma}{\hat{\psi}_\rho}\right)^\vee\right\|_{L_1(\mathbb{R}^d)},$$

we can bound (3.4) by a constant multiple of

$$\int_{\Omega'_\delta/\omega} \left|\psi_\rho(\cdot/\omega - t)\right| dt.$$

Moreover, we find from the expression (2.1) that $\psi_\rho \leqslant c(1+|\cdot|)^{-d-q}$, where q is in (2.1) and c depends on ρ. Since $|x-t| > \delta$ for $x \in \Omega$ and $t \in \Omega'_\delta$, a direct calculation yields

$$\int_{\Omega'_\delta/\omega} \left|\psi_\rho\left(\frac{x}{\omega} - t\right)\right| dt \leqslant c\omega^q \int_{B'_\delta} \frac{1}{(\omega+|t|)^{d+q}}\, dt = O\big(\omega^q\big),$$

which completes our proof. □

A bound of the third term in (3.3) is obtained in next lemma.

Lemma 3.2. Let K be as above. Assume that ψ_c satisfies the conditions in (2.1), and let the operator Λ be as in (3.1). Assume that the relation $c = \rho\omega$ holds for some $\rho > 0$. Then, for every $f \in L_\infty(\mathbb{R}^d)$, we have

$$\left\|\int_{\Omega_\delta/\omega} K(\cdot,t)\Lambda(\chi_{\Omega'_{2\delta}} f)(\omega t)\, dt\right\|_{L_\infty(\Omega)} = o\big(\omega^q\big),$$

with q in (2.1).

Proof. From (2.1) and (2.7), it is clear that

$$\left\|\int_{\Omega_\delta/\omega} K(\cdot,t)\Lambda(\chi_{\Omega'_{2\delta}} f)(\omega t)\, dt\right\|_{L_\infty(\Omega)} \leqslant c\left\|\Lambda(\chi_{\Omega'_{2\delta}} f)\right\|_{L_\infty(\Omega_\delta)}.$$

For every $t \in \Omega_\delta$, it is immediate from (3.1) that

$$\begin{aligned}\Lambda(\chi_{\Omega'_{2\delta}} f)(t) &= \omega^{-d} \int_{t-\Omega'_{2\delta}} f(t-s)\left(\frac{\sigma}{\hat{\psi}_\rho}\right)^\vee \left(s/\omega\right) ds \\ &\leqslant \omega^q \int_{t-\Omega'_{2\delta}} f(t-s)\frac{g(s/\omega)}{|s|^{q+d}}\, ds \leqslant \omega^q \|f\|_{L_\infty(\mathbb{R}^d)} \int_{B'_\delta} \frac{g(s/\omega)}{|s|^{q+d}}\, ds, \quad (3.5)\end{aligned}$$

with

$$g = |\cdot|^{q+d}\left(\frac{\sigma}{\hat{\psi}_\rho}\right)^\vee.$$

Since $\sigma\hat{\psi}_\rho^{-1} \in C^{d+q}(\mathbb{R}^d)$ (see (2.1)), the Riemann–Lebesgue lemma shows that $|g(s/\omega)| \to 0$ for any $s \in B'_\delta$ as $\omega \to 0$. Hence, the last integral in expression (3.5) tends to 0. □

A bound of the fourth term in (3.3) is provided in next lemma.

Lemma 3.3. Let ψ_c and Λ be defined as above, and let the relation $c = \rho\omega$ hold for some $\rho > 0$. Assume that the coefficient sequence $(C_f(\cdot,\xi))_{\xi\in\Xi}$ for F in (2.8) is admissible for Π_k on Ω. Then, for every $f \in W_\infty^k(\mathbb{R}^d)$, we have the relation

$$\left\| \int_{\Omega_\delta/\omega} K(\cdot - t)\Lambda\big(\chi_{\Omega_{2\delta}}(f - F)\big)(\omega t)\,dt \right\|_{L_\infty(\Omega)} \leqslant O\big(\omega^{\min(k,q)}\big).$$

Proof. Due to the fact that $K(x,\cdot) \in L_1(\mathbb{R}^d)$ for any $x \in \mathbb{R}^d$, it is obvious that

$$\left\| \int_{\Omega_\delta/\omega} K(\cdot, t)\Lambda\big(\chi_{\Omega_{2\delta}}(f - F)\big)(\omega t)\,dt \right\|_{L_\infty(\Omega)} \leqslant c\big\|\Lambda\big(\chi_{\Omega_{2\delta}}(f - F)\big)\big\|_{L_\infty(\Omega_\delta)}.$$

Since $\sum_{\xi\in\Xi} C_f(\cdot,\xi) = 1$ by definition, it is clear that

$$(f - F)(s) = \sum_{\xi\in\Xi} C_f(s,\xi)\big(f(s) - f(\xi)\big), \quad s \in \Omega.$$

Letting $T_s f$ be the Taylor polynomial of degree k of f about $t \in \Omega$, the function $\{f(s) - T_s f(y)\colon\ y \in \mathbb{R}^d\}$ is also a polynomial in Π_k. Then we have

$$\sum_{\xi\in\Xi} C(s,\xi)\big(f(s) - T_s f(\xi)\big) = 0$$

by the definition of admissible coefficients $(C_f(s,\xi))_{\xi\in\Xi}$. It induces the expression

$$\big|(f - F)(s)\big| \leqslant |f|_{k,\infty} \sum_{|\alpha|_1=k} \sum_{\xi\in\Xi} C_f(s,\xi)(s - \xi)^\alpha.$$

If $s \in \Omega$, it is clear that $|(f - F)(s)| \leqslant ch^k$ with $c > 0$ independent of s. Next, let $s \in \overline{\Omega}' \cap \Omega_{2\delta}$. Then, since the set Ξ_s is to be some 'close neighbors' of s, without loss of generality, we can assume that for any $s \in \Omega_{2\delta}$,

$$|s - \xi| \leqslant c_k|s - t|, \quad \xi \in X_s,\ s \in \Omega_{2\delta}. \tag{3.6}$$

Then, we deduce that

$$\big|(f - F)(s)\big| \leqslant c_k|s - t|^k.$$

Now, with this bound in hand, we return to the estimate

$$\begin{aligned}
\Lambda\big(\chi_{\Omega_{2\delta}}(f - F)\big)(t) &= \int_{\Omega_{2\delta}} \left(\frac{\sigma_\omega}{\hat{\psi}_\rho(\omega)}\right)^\vee (t - s)(f - F)(s)\,ds \\
&\leqslant \omega^{\min(k,d+q)} \int_{\Omega_{2\delta}/\omega} \left(\frac{\sigma}{\hat{\psi}_\rho}\right)^\vee \left(\frac{t}{\omega} - s\right)\left|\frac{t}{\omega} - s\right|^{\min(k,d+q)} ds \\
&= O\big(\omega^{\min(k,q)}\big),
\end{aligned}$$

where the last inequality is true by condition (2.1). □

Finding the approximation power of the second term in (3.3) was a focal point in the study [20] in the case $\Omega = \mathbb{R}^d$. By using the same idea as in [20], we can prove the exactly same approximation property on bounded domains. Specifically, we prove that:

Lemma 3.4. Let ψ_c, K, and Λ be as above, and let the coefficients $(C(t,\xi))_{\xi\in\Xi}$ for $\phi_c(\cdot,t)$ in (2.3) be admissible for Π_n with $n > \lambda + d$ on Ω_δ. Assume that $f \in W_\infty^k(\mathbb{R}^d)$ and $c = \rho\omega$ for some $\rho > 0$. Then, we estimate that if $k < \lambda + d$ and $\omega(h) = h$,

$$\left\| \int_{\Omega_\delta/\omega} \big(\psi_\rho(\cdot/\omega - t) - K(\cdot,t)\big)\Lambda f(\omega t)\,dt \right\|_{L_\infty(\Omega)} = o\big(h^k\big).$$

Furthermore, if $k \geqslant \lambda + d$ and $\omega = h^r$ with $0 < r \leqslant 1$, we have

$$\left\| \int_{\Omega_\delta/\omega} \big(\psi_\rho(\cdot/\omega - t) - K(\cdot,t)\big)\Lambda f(\omega t)\,dt \right\|_{L_\infty(\Omega)} = O\big(h^{(1-r)n+r(\lambda+d)}\big).$$

We have observed in [20] that the last term in (3.3) is of the magnitude

$$\big\| f - \sigma_\omega^\vee * f \big\|_{L_\infty(\Omega)} = o\big(\omega^k\big)$$

for every function $f \in W_\infty^k(\mathbb{R}^d)$. Now, we summarize all the results in this section as follows.

Theorem 3.5. Let ϕ_c, K and L be as above, and assume that ψ_c satisfies condition (2.1). Assume that $c = \rho\omega$ for some $\rho > 0$ and assume further that:

(a) The function F is defined as in (2.8), and the coefficients $(C_f(\cdot,\xi))_{\xi\in\Xi}$ for F are admissible for Π_k on $\Omega_{2\delta}$.

(b) The coefficients $(C(t,\xi))_{\xi\in\Xi}$, $t \in \Omega$, for $\phi_c(\cdot,t)$ are admissible for Π_n with $n > \lambda + d$ on Ω_δ.

If $k < \lambda + d$ and $\omega(h) = h$, then, for every $f \in W_\infty^k(\mathbb{R}^d)$,

$$\|f - Lf\|_{L_\infty(\Omega)} = O\big(h^k\big).$$

Furthermore, if $k \geqslant \lambda + d$ and $\omega(h) = h^r$ with $0 < r \leqslant 1$, for f as above,

$$\|f - Lf\|_{L_\infty(\Omega)} \leqslant O\big(h^{rk}\big) + O\big(h^{rq}\big) + O\big(h^{(1-r)n+r(\lambda+d)}\big),$$

where $q(> \lambda + d)$ is in (2.1).

Remark. The approximation power of L depends on the decaying property of ψ_ρ and the smoothness of f. It is well known from the literature (see [7,15]) that a possible value of q is $\lambda + 2d$.

Corollary 3.6. Under the same conditions and notations of theorem 3.5, assume that if $k > \lambda + d$, the number n $(> \lambda + d)$ is chosen to be $(1 - r)n + r(\lambda + d) > r \min(k, q)$ for the given $r \in (0, 1)$. Then, for every $f \in W_\infty^k(\mathbb{R}^d)$, we have

$$\|f - Lf\|_{L_\infty(\Omega)} = O\big(h^{r \min(k,q)}\big).$$

4. Smoothing noisy data

In many scientific phenomena and technologies, the data can be contaminated. Hence, in this section, we will observe the smoothing effects of the scheme L. For this, we assume that the data $(y_\xi)_{\xi \in \Xi}$ arise according to the model

$$y_\xi = f(\xi) + \varepsilon_\xi,$$

where the ξ's belong to $\Omega \subset \mathbb{R}^d$ and, for example, ε_ξ's are independent normally distributed random variables with mean 0 and (known or unknown) variance σ^2. Here the underlying function $f : \mathbb{R}^d \to \mathbb{R}$ is assumed to be a smooth function. Our smoothing procedure with L can be interpreted as following three steps:

Step 1. Construct a function $F = \sum_{\xi \in \Xi} C_f(\cdot, \xi) y_\xi$.

Step 2. Take the convolution $\Lambda(\chi_{\Omega_{2\delta}} F) = (\sigma_\omega / \hat{\psi}_\rho(\omega \cdot))^\vee * (\chi_{\Omega_{2\delta}} F)$.

Step 3. Find the final approximant $Lf = \int_{\Omega_\delta / \omega} K(\cdot, t) \Lambda(\chi_{\Omega_{2\delta}} F)(\omega t)\, dt$.

In the following example, we illustrate these three steps.

Example 4.1. The given data are of the form

$$y_\xi = f(\xi) + \varepsilon_\xi, \quad \xi \in [-1, 1]^2$$

with ε_ξ independent random variables normally distributed with mean 0 and variance $\sigma = 0.05$. The underlying function is

$$f(x, y) = \big[B_2\big(1.5(x - 0.5)\big) - B_2\big(1.5(x + 0.5)\big)\big] \exp\big(-y^2\big)$$

with B_2 a tensor-product of quadratic splines. In this example, the sets Ξ and $(\varepsilon_\xi)_{\xi \in \Xi}$ come from a random number generator in MATLAB. Figures 1(B)–(D) shows the surfaces obtained in each of the steps described above. Figure 1(A) displays the surface of the underlying function f, and figure 1(B) the surface F obtained by interpolating the noisy data. Figure 1(C) presents the surface after smoothing the noise. We finally obtain the surface displayed in figure 1(D).

Remark. As we see in the form of Lf, we have the parameters c and ω which are being adjusted according to the density of centers and noise $(\varepsilon_\xi)_{\xi \in \Xi}$. As $c, \omega \to 0$, the function $\Lambda(\chi_{\Omega_{2\delta}} F)$ tends to the local interpolant F, which makes the approximant lose some smoothness. Also, as c is getting bigger, the approximant becomes smoother, hence it

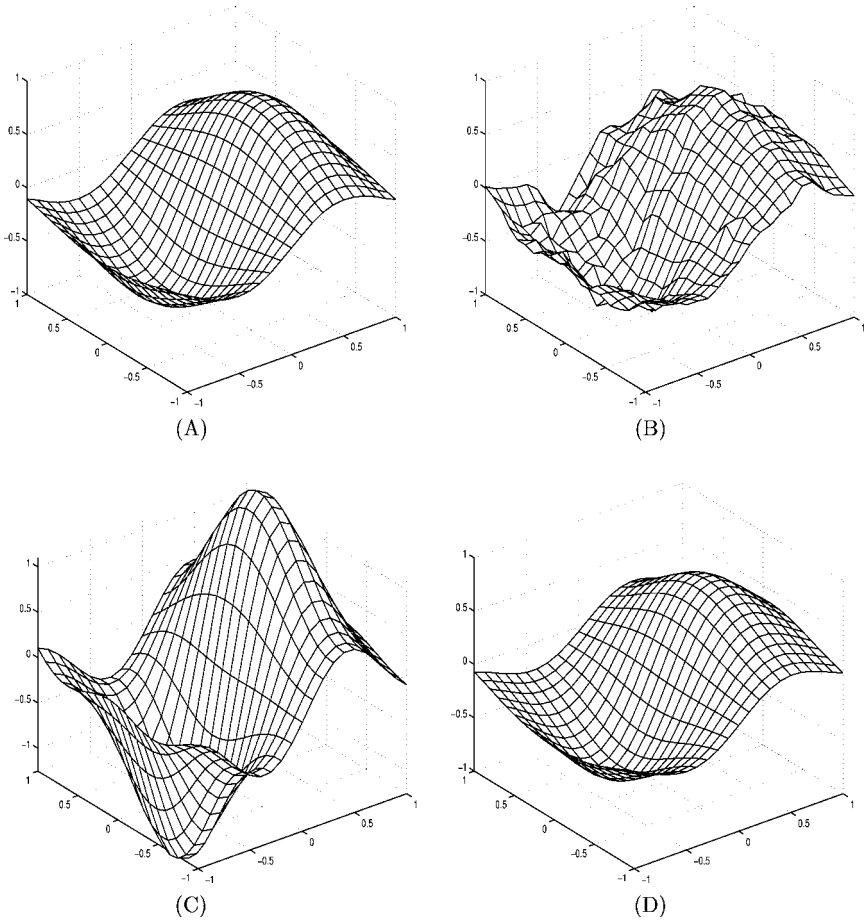

Figure 1. Procedures of noisy data approximation by the scheme L. Figure (A) shows the surface of the underlying function f and (B) is the function F (step 1). Figure (C) indicates $\Lambda(\chi_{\Omega_{2\delta}} F)$ (step 2) and (D) is the final approximant Lf (step 3).

may lose some 'details'. In fact, a good choice for the parameters c and ω can be interpreted as a balanced compromise between smoothness and fidelity of the approximant to the data.

5. Algorithm

5.1. *On the coefficients of pseudo-shift* $\phi(\cdot, t)$

In this section we provide an algorithm for the construction of the admissible coefficients $(C(t, \xi))_{\xi \in \Xi}$ for the pseudo-shift

$$\phi_c(\cdot, t) = \sum_{\xi \in \Xi} C(t, \xi) \phi_c(\cdot - \xi).$$

The same algorithm is also applied to find the coefficients $(C_f(t,\xi))_{\xi\in\Xi}$ for the function F in (2.8). To each $t \in \Omega$, we associate a subset

$$\Xi_t := \{\xi \in \Xi\colon\ C(t,\xi) \neq 0\}.$$

The admissible coefficients $(C(t,\xi))_{\xi\in\Xi_t}$ for $\phi_c(\cdot,t)$ are required to satisfy the linear system

$$\sum_{\xi\in\Xi_t} C(t,\xi)p(\xi) = p(t) \tag{5.1}$$

for $p \in \Pi_n$ with n greater than $\lambda + d$, the order of singularity of $\hat{\phi}_c$ at the origin. The choice of polynomial space Π_n requires that

$$\#\Xi_t > \dim\Pi_n\big(\mathbb{R}^d\big) = \binom{n+d}{d} =: n_d.$$

Letting $q_1,\ldots,q_{n_d}$ be a basis of a polynomial space Π_n, this condition (5.1) holds if and only if the coefficient matrix

$$\bar{c} := \big(C(t,\xi)\big)_{\xi\in\Xi_t}$$

solves the linear system

$$E\bar{c} = \bar{b} \tag{5.2}$$

with

$$E = \big(q_j(\xi)\colon\ j = 1,\ldots,n_d,\ \xi \in \Xi_t\big)$$

and

$$\bar{b}^{\mathrm{T}} := \big(q_j(t)\colon\ j = 1,\ldots,n_d\big).$$

Here, in order to find the coefficient matrix $(C(t,\xi))_{\xi\in\Xi_t}$ for $\phi_c(\cdot,t)$, we suggest using a minimization problem

$$\begin{aligned} &\text{minimize} \sum_{\xi\in\Xi_t}\eta(t,\xi)A^2(t,\xi) \\ &\text{subject to}\quad E\bar{c} = \bar{b} \end{aligned} \tag{5.3}$$

with a penalty function $\eta(t,\cdot)$. In the univariate case, we can make this problem computationally simple by choosing some good basis functions of Π_n (e.g., Hermite or Lagrange polynomials), which makes the matrix E banded or triangular. However, in the multivariate case, since we do not know of a basis for Π_n that results in a matrix E with simple structure, we confront numerical difficulties caused by the conditioning of the matrix E. Eventually, the system (5.2) becomes ill-conditioned with the increase in the number of constraints. The well-known standard method to solve this problem is via Gauss elimination. However, Gauss elimination has to deal with another numerical difficulty in providing a solution to the linear system in (5.3) when all the pivots available

for the current step in the current column are all zero. In this case, we need to interchange some centers and to recompute a part of the elimination. So the actual location and configuration of Ξ_t need to be taken into account.

De Boor and Ron proposed a particular elimination method, the so-called Gauss elimination by degree, which is more efficient for this problem (cf. [4]). It was originally designed for the construction of multivariate polynomial interpolation. Actually, it applies Gauss elimination degree-by-degree (not monomial-by-monomial) with partial pivoting to the Vandermonde matrix

$$V := \left(\xi^k\right)$$

by treating all the entries of a given degree as one entry. Hence, this method can be applied to the matrix E^{T} with E in (5.2) to obtain the factorization $E^{\mathrm{T}} = LU$ with L^{T} in row echelon form and U a block upper triangular non-singular matrix, and it induces the linear system $L^{\mathrm{T}}\bar{c} = \bar{b}'$ with $\bar{b}' = (U^{\mathrm{T}})^{-1}\bar{b}$ from (5.2). In the following, we give the algorithm in detail.

5.2. *Factorization*

Assume that $\#\Xi_t = m(> n_d)$ for all $t \in \Omega$ and let

$$\Xi_t =: \{\xi_1, \xi_2, \ldots, \xi_m\}.$$

Choosing a set of functions $\{(t - \cdot)^\alpha\}_{|\alpha|_1 \leqslant n}$ as a basis of Π_n, we denote the matrix E^{T} as

$$E^{\mathrm{T}} =: \left((t - \xi_p)^\alpha\colon\ p = 1, \ldots, m,\ 0 \leqslant |\alpha|_1 \leqslant n\right),$$

and correspondingly, the matrix $\bar{b}$ in (5.3) is changed to

$$\bar{b} := [1\ \underbrace{0 \ldots 0}_{(n_d - 1)\text{ terms}}]^{\mathrm{T}}.$$

The strategy of Gauss elimination by degree begins by treating all the entries of a given degree as one entry: the (p, q)-entry of $\mathbf{E}^{\mathrm{T}}$ is to be

$$\mathbf{E}^{\mathrm{T}}(p, q) = \left(\xi_p^\alpha\colon\ |\alpha|_1 = q\right).$$

Here and in the sequel, we shall use the notation $\mathbf{E}$ in lieu of E in order to emphasize the alternative point of view. Thus, the rows and columns of $\mathbf{E}^{\mathrm{T}}$ are indexed by $\xi_p \in \Xi_t$ and $k = 0, \ldots, n$, respectively.

Now since the entries in $\mathbf{E}^{\mathrm{T}}$ are considered not as scalars but as vectors, we make all the entries in the pivot column below the pivot row orthogonal to the pivot entry. In order to eliminate entries in column k of $\mathbf{E}^{\mathrm{T}}$, a scalar product is defined as

$$(a, b)_k := \sum_{|\alpha|_1 = k} a(\alpha) b(\alpha) \overline{\omega}(\alpha)$$

with $\overline{\omega}$ a weight function. Furthermore, since each entry in column k consists of

$$C(k+d-1;k) := \frac{(k+d-1)!}{k!\,(d-1)!} \tag{5.4}$$

monomials, after Gauss elimination by degree in column k, we want to have $C(k+d-1;k)$ nonzero orthogonal entries in the kth column below row

$$k^\circ := \sum_{j<k} C(j+d-1;j) + 1,$$

the first working position of elimination in the kth column. Hence, the ultimate goal is to factorize E^{T} as follows:

$$E^{\mathrm{T}} = LU$$

with L^{T} in row echelon form and U a block upper triangular nonsingular matrix.

The algorithm is summarized as follows: Let $\mathbf{W}$ be the 'working array' which is initially equal to $\mathbf{E}^{\mathrm{T}}$. At each column, say kth, we first put our working position at k°th row and go through below the row. At each step (let us assume that we start at k_jth row in column k), we look for the current row k_j or below the row such that, in order to alleviate the devastating interaction of rounding error, we find a largest nontrivial entry (relative to the size of the corresponding entry or row of $\mathbf{E}^{\mathrm{T}}$) and, if it is not on the pivot position, interchange its row with row k_j of $\mathbf{W}$ to bring it into the pivot position $\mathbf{W}(\xi_{k_j}, k)$. Then we subtract the appropriate multiple of the pivot row $\mathbf{W}(\xi_{k_j}, :)$ from all subsequent rows in order to make $\mathbf{W}(\xi_{k_i}, k)$ orthogonal to $\mathbf{W}(\xi_{k_j}, k)$ for all $k_i > k_j$. Then we proceed with elimination by Gram–Schmidt process. Specifically, if we assume that orthogonal entries $w'_1, \ldots, w'_{j-1}$ are already available in the column k, we can compute

$$w'_j := w_j - \sum_{i<j} w'_i \frac{\langle w_j, w'_i \rangle}{\langle w'_i, w'_i \rangle} \tag{5.5}$$

for a next orthogonal entry, and thereby ensure that

$$\langle w'_i, w'_j \rangle = 0, \quad i < j,$$

while $w'_j \neq 0$. It may, of course, happen that all the pivots available for the current step in the current column k are zero before we obtain $C(k+d-1;k)$ nonzero orthogonal entries. Then we have to replace some of the centers in Ξ_t by other centers in $\Xi \setminus \Xi_t$, and perform the calculation (5.5) again on the corresponding row until we obtain $C(k+d-1;k)$ orthogonal entries in the column k.

On the other hand, Gauss elimination is usually performed to a square matrix, and it factors this matrix into a lower triangular matrix and an upper triangular matrix. However, since our matrix E^{T} is an $m \times n_d$ rectangle matrix with $m > n_d$, each step of elimination by degree is equivalent to factoring E^{T} into an $m \times m$ matrix and an $m \times n_d$ matrix. For example, the first step can be expressed as following

$$\mathbf{E}^{\mathrm{T}} = \widetilde{L}_1 \mathbf{W}$$

with a matrix $\mathbf{W}$ of working array and a lower triangular matrix $\widetilde{L}_1$ which is associated with the process of making the entries in row 2 through row m orthogonal to the first row. Continuing this process, the final output of elimination by degree is a factorization of E^{T} in the form

$$\mathbf{E}^{\mathrm{T}} = \widetilde{L}\mathbf{W} \tag{5.6}$$

with $\widetilde{L}$ an $m \times m$ unit lower triangular matrix. Since E^{T} is an $m \times n_d$ matrix with $m > n_d$, the elimination is performed with n_dth columns, and hence the matrix $\widetilde{L}$ can be written in the form

$$\widetilde{L} = [L\ L_0], \tag{5.7}$$

where the matrix L consists of the first n_d columns of $\widetilde{L}$ which are associated with elimination procedures of matrix $\mathbf{E}^{\mathrm{T}}$. However, L_0 takes the last $m - n_d$ columns of $\widetilde{L}$, and it is not associated with any elimination progress, which means $L_0(i, j) = 0$ for $i \neq j$ with $j > n_d$, and $L_0(i, j) = 1$ for $i = j$. Furthermore, the final output $\mathbf{W}$ is a row echelon matrix in the following sense. If we make ordering $k_1, \ldots, k_{n_d}$ of columns according to the degree of each entry and $\xi_1, \ldots, \xi_m$ of the rows, the last $m - n_d$ rows of matrix $\mathbf{W}$ are completely zero, and the leading entry (the first nonzero entry) in the nonzero row $\mathbf{W}(\xi_j, :)$ is the entry $\mathbf{W}(\xi_j, k_j)$ for all $j \leqslant n_d$. Hence, the matrix $\mathbf{W}$ is of the form

$$\begin{bmatrix} U \\ \bar{0} \end{bmatrix},$$

where U is an $n_d \times n_d$ block upper triangular square matrix, and $\bar{0}$ is an $(m - n_d) \times n_d$ zero matrix. In the actual calculation, the two matrices L_0 and $\bar{0}$ can be ignored. Therefore, the factorization in (5.6) is replaced by

$$E^{\mathrm{T}} = LU.$$

We note that the matrix U does not have to be an upper triangular since $k_1, \ldots, k_{n_d}$ need not to be strictly increasing. But each entry in the diagonal entries is orthogonal to each other, hence U is invertible. With this factorization, we return to the original system (5.2)

$$\bar{b} = E\bar{c} = U^{\mathrm{T}} L^{\mathrm{T}} \bar{c}.$$

By substituting

$$\left(U^{\mathrm{T}}\right)^{-1} \bar{b} =: \bar{b}', \tag{5.8}$$

the linear system (5.2) can be replaced by

$$L^{\mathrm{T}} \bar{c} = \bar{b}'. \tag{5.9}$$

With the matrices L and $\bar{b}'$ at hand, we find the coefficient matrix $\bar{c} = (C(t, \xi))_{\xi \in \Xi_t}$ by minimizing the quadratic form

$$\sum_{\xi \in \Xi_t} \eta(t, \xi) C^2(t, \xi)$$

subject to the constraints

$$L^{\mathrm{T}}\bar{c} = \bar{b}'.$$

Theorem 5.1. For $t \in \Omega$, let η be a weight function and $D = 2\,\mathrm{Diag}(\eta(t, \xi_i)\colon\ i = 1, \ldots, m)$. Let $\bar{c} = (C(t, \xi))_{\xi \in \Xi_t}$ be as above, and let L and $\bar{b}'$ be the matrices defined as above. Then

$$\bar{c} = D^{-1}L\big(L^{\mathrm{T}}D^{-1}L\big)^{-1}\bar{b}'.$$

Proof. The method of Lagrange multipliers induces the following linear system

$$D\bar{c} + L\bar{\Lambda} = 0 \quad \text{and} \quad L^{\mathrm{T}}\bar{c} = \bar{b}' \tag{5.10}$$

with $\bar{\Lambda}^{\mathrm{T}} := [\lambda_1 \ \ldots \ \lambda_{n_d}]$ the matrix of Lagrange multipliers. Then it can be easily verified that the matrix for the linear system (5.10),

$$\begin{pmatrix} D & L \\ L^{\mathrm{T}} & 0 \end{pmatrix},$$

is nonsingular. Therefore, $\bar{c} = D^{-1}L(L^{\mathrm{T}}D^{-1}L)^{-1}\bar{b}'$ and $\bar{\Lambda} = (L^{\mathrm{T}}D^{-1}L)^{-1}\bar{b}'$ solve (5.10). □

We shall be mostly interested in an optimality condition for the minimization problem. Here we suggest some examples.

Example 5.1. We adopt a penalty function of the form

$$\eta(t, \xi) = \eta_+\big(|t - \xi|\big),$$

where η_+ is an increasing function on $\mathbb{R}^d+$, and $\eta_+(0) = 0$. As a good choice of η_+, the following function is suggested:

$$\eta_+\big(|t - \xi|\big) = \left[\exp\left(\frac{|t - \xi|^2}{h}\right) - 1\right]|t - \xi|^{2k}$$

with $k \in \mathbb{Z}_+$ and h the density of Ξ. Then it is evident that $\eta_+(|t - \xi|)$ gives high penalty to the coefficient $C(t, \xi)$ as ξ moves far away from t.

Example 5.2. The coefficients $(C(t, \xi))_{\xi \in \Xi_t}$ in the above example do not depend on the basis function ϕ_c. In this example, we look for the coefficient matrix $(C(t, \xi))_{\xi \in \Xi_t}$ minimizing the error $\phi_c(\cdot - t) - \phi_c(\cdot, t)$ in some sense. In order to minimize an upper bound of the error $\phi_c(\cdot - t) - \phi_c(\cdot, t)$, we consider the relation

$$\big|\phi_c(\cdot - t) - \phi_c(\cdot, t)\big|^2 \leqslant \mathrm{const} \sum_{\xi \in \Xi_t} C^2(t, \xi)\big[\phi_c(\cdot - t) - \phi_c(\cdot - \xi)\big]^2.$$

Thus, we choose the penalty function

$$\eta(t,\xi) = \left\|\phi_c(\cdot - t) - \phi_c(\cdot - \xi)\right\|^2_{L_\infty(\Omega)}$$

for our optimization problem in (5.3).

5.3. Algorithm details

Assume that $d = 2$. We give here a MATLAB-like pseudo-program to construct coefficients $(C(t,\xi))_{\xi\in\Xi}$ for the pseudo-shift $\phi_c(t,\xi)$. This algorithm selects a set Ξ_t which has the nondegeneracy property for Π_n (see (2.6)) and then finds admissible coefficients $(C(t,\xi))_{\xi\in\Xi_t}$ for Π_n. In this 'program', we use the following conventions.

The set of scattered centers Ξ is considered as $m \times 2$ matrix. The matrices $\mathbf{E}^{\mathrm{T}}$ and $\mathbf{W}$ are denoted by $\mathtt{E}^{\mathrm{T}}$ and $\mathtt{W}$, respectively. In particular, since $C(k+d-1;k) = k+1$ with $d = 2$ (see (5.4)), we note that $\mathtt{W(j,k)}$ is a vector with $(k+1)$ entries, indexed by $\{\alpha \in \mathbb{Z}^d\colon |\alpha|_1 = k\}$, and $(k+1)$ orthogonal entries will be obtained in column k. All matrices mentioned in the 'program' other than $\mathtt{E}^{\mathrm{T}}$ and $\mathtt{W}$ are proper MATLAB matrices, i.e., have scalar entries. Correspondingly, for two vectors `a` and `b` (such as `W(i,k)`, `W(j,k)`) indexed by $\{\alpha \in \mathbb{Z}^d\colon |\alpha|_1 = k\}$, `<a,b>` denotes a scalar product.

We borrow from MATLAB the notations:

(i) `ones(m,n)` for the matrix of size `m` × `n` with all entries equal to 1;

(ii) `eys(m,m)` for the identity matrix of order `m`;

(iii) `a:b` for the vector with entries `a`, `a + 1`, ..., `a + m`, with `m` the natural number for which `a + m` ⩽ `b` < `a + m + 1`;

(iv) `A*B` for the matrix product of the matrices `A` and `B`;

(v) standard logical constructs like (`for j = 1 : n, ..., end`), and (`if..., ..., end`);

(vi) the construct (`while 1,..., if ..., break, end,..., end`), which is a loop exited only through the break;

(vii) the construct `[p,i] = max(a)` to provide `p = a(i) = max`$_{\mathtt{j}}$`a(j)`;

(viii) the command `function [a,b] = ft-name(x)` defines a new function called `ft-name`. The variables within the body of a function are all by default local;

(ix) the relational operator `a==b` means that `a` is equal to `b` while `a=b` is used for the assignment statement. Furthermore, we use an occasional word to describe an action whose details seem clear.

```
% INPUT: Ξ_t, m, n, tol, penalty function η
% OUTPUT: c̄ = D^{-1}* L*(L^T* D^{-1}*L)^{-1}*b by theorem 5.1
Select Ξ_t (from Ξ) which are closest to t, m = #Ξ_t > n
```

```
nbr_ob=0; k=0
E^T(:,k) = ones(m,1); W(:,k) = E^T(:.k)
L = eye(m,m)
for j=1:m
   while 1
      [p,i] = max_{i>j-1}<W(i,k),W(i,k)>/<E^T(i,k),E^T(i,k)>
      if p > tol, nbr_ob = nbr_ob+1, break, end
      if nbr_ob < k+1
         replace ξ_j by ξ, one of the closest centers to t
         from Ξ \ Ξ_t, i.e., E^T(j,:)=E^T(ξ,:)
         [L_j,W_j] = RE - COMP(j,k,E^T(j,:),W);
         L(j,1:j-1)=L_j; W(j,k)=W_j;
      end
      k = k+1; nbr_ob=0;
      construct E^T(:,k) from E^T(:,k-1) and Ξ_t
      W(:,k) = L^{-1} *E^T(:,k)
   end
   if i > j, interchange i and j, end
   for i=j+1:n
      L(i,j) = <W(i,k),W(j,k)>/<W(j,k),W(j,k)>
      W(i,k) = W(i,k)-L(i,j)*W(j,k)
      end
   end
end
L = L(:,1:n) ; W = W(1:n,:)
b = (W^T)^{-1}b; D = Diag(η(t,ξ))_{ξ∈Ξ_t}
c̄ = D^{-1}*L*(L^T*D^{-1}*L)^{-1}*b
```

function `[`L_j`, W(j,q)] = RE - COMP(j,k,E^T(j,:),W)`

```
q=0
for i=1:j-1
   if W(j,q)==0, q=q+1, end
   L_j(i)=<E^T(j,q),W(i,q)>/<W(i,q),W(i,q)>
   W(j,q) = E^T(j,q)-L_j(i)*W(i,q)
end
```

5.4. *Formulations for the construction of Lf*

We now describe some formulations for the approximant Lf under the assumption that the data $(\xi, y_\xi)_{\xi\in\Xi}$ are arising according to the model

$$y_\xi = f(\xi) \quad \text{or} \quad y_\xi = f(\xi) + \varepsilon_\xi,$$

where $(\varepsilon_\xi)_{\xi\in\Xi}$ is a type of noise (for example, independent normally distributed random variables with mean 0 and variance σ^2). Practically, we focus on the case $d = 2$ and $\lambda = 2$ with the issues of localization sequence, extrapolation and computation of Λ.

Prior to further discussion, we describe the generalized Fourier transform of ϕ_c:

$$\hat{\phi}_c(\theta) = C(\lambda, d)|\theta|^{-\lambda-d}\widetilde{K}_{(d+\lambda)/2}\big(|c\theta|\big), \tag{5.11}$$

see [9], where $C(\lambda, d)$ is a constant depending on λ and d, and $\tilde{K}_\nu(|t|) := |t|^\nu K_\nu(|t|)$ with $K_\nu(|t|)$ the modified Bessel function of order ν, see [1]. Note that despite the similarity in the notations, there is no direct connection between the above $\widetilde{K}$ and the kernel K.

(1) Localization. In section 2, the kernel K in the scheme L can be constructed by associating the localization ψ_c which is obtained by an application of a difference operator, i.e.,

$$\psi_c := \sum_{\alpha\in N}\mu(\alpha)\phi_c(\cdot - \alpha),$$

where N is a finite subset of $\mathbb{Z}^d$. There can be many choices of N. One example of N for this linear combination is illustrated by stencils centered at the origin as shown in figure 2. In fact, since the kernel K consists of the localization sequence $(\mu(\alpha))_{\alpha\in N}$ and the corresponding pseudo-shifts of ϕ_c (see (2.4)), we focus here on the method of finding the sequence $(\mu(\alpha))_{\alpha\in N}$.

Letting

$$\tau(\theta) = \sum_{\alpha\in N}\mu(\alpha)\mathrm{e}^{-\alpha\theta},$$

a sufficient condition for (2.1) is as follows:

$$D^\beta\big(\tau - \hat{\phi}_c^{-1}\big)(0) = 0, \quad |\beta|_1 \leqslant 2\lambda + 2d - 1. \tag{5.12}$$

It is not surprising that the localization involves the behavior of $\hat{\phi}_c$ near the origin. The Fourier transform $\hat{\phi}_c$ of ϕ_c is very smooth off the origin. This means that in order to localize ϕ_c we only need to ensure that the Fourier transform $\hat{\psi}_c$ of the localized function ψ_c is smooth at the origin. Note that we also need to insist that $\hat{\psi}_c(0) \neq 0$. We refer to the paper [7] for more details.

Now, it follows from (5.12) that the localization sequence $(\mu(\alpha))_{\alpha\in N}$ satisfies

$$\sum_{\alpha\in N}\mu(\alpha)\alpha^\beta = D^\beta\hat{\phi}_c^{-1}(0), \quad |\beta|_1 \leqslant 2\lambda + 2d - 1.$$

Let $N := \{\alpha_j \in \mathbb{Z}^d\colon\ j = 1, \ldots, \#N\}$. The sequence $\bar{\mu} := (\mu(\alpha_j)\colon\ j = 1, \ldots, \#N)^{\mathrm{T}}$ is obtained by solving the linear system

$$M\,\bar{\mu} = \bar{g},$$

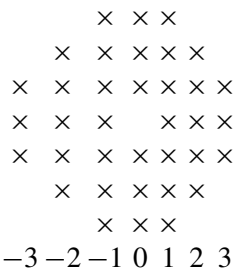

Figure 2. Stencil for N ($d = 2$, $\lambda = 2$).

where

$$M = \left(\alpha_j^{\beta}\colon\ |\beta|_1 \leqslant 2\lambda + 2d - 1,\ \ j = 1, \ldots, \#N\right)$$

and

$$\bar{g} = \left(D^{\beta}\hat{\phi}_c^{-1}(0)\colon\ |\beta|_1 \leqslant 2\lambda + 2d - 1\right)^{\mathrm{T}}.$$

In particular, assuming N is as in figure 2 with $\lambda = 2$ and $d = 2$, the graph of function ψ_c is symmetric about coordinate axes.

(2) Extrapolation. Given a data set $(\xi, y_\xi)_{\xi\in\Xi}$ where y_ξ is a function value $f(\xi)$ or a noisy value $f(\xi) + \varepsilon_\xi$, we define an extrapolation F on a superset of Ω by

$$F = \sum_{\xi\in\Xi} C_f(\cdot, \xi)\, y_\xi.$$

In particular, we suggest finding the coefficients $(C_f(t, \xi))_{\xi\in\Xi}$ for F by a minimizing problem

$$\begin{aligned} &\text{minimize} \sum_{\xi\in\Xi_t} \eta(t, \xi) C_f^2(t, \xi) \\ &\text{subject to} \quad L^{\mathrm{T}}\bar{c} = \bar{b}' \end{aligned}$$

with L and $\bar{b}'$ in (5.8) and (5.9), respectively, and with $\bar{c} := (C_f(t, \xi))_{\xi\in\Xi_t}$. Since the approximation should be local in the sense that its value at t depends on centers which are close to t, we assign a high penalty to centers which are far from x. Having performed some numerical experimentations with several alternatives for the function η, we found out that a good choice is

$$\eta\left(|t - \xi|\right) = \left[\exp\left(\frac{|t - \xi|^2}{h^2}\right) - 1\right]|t - \xi|^{2k}$$

with $k \in \mathbb{Z}_+$.

(3) Computation of Λ. Choosing $\rho = c/\omega$, the function Λg with $g \in L_\infty(\mathbb{R}^d)$ can be written in terms of $(\sigma/\hat{\psi}_\rho)^\vee$ and g. Thus, we construct here some formulations to compute the function $(\sigma/\hat{\psi}_c)^\vee$. First, with the localization sequence $(\mu(\alpha))_{\alpha\in N}$, we find from (5.11) that the Fourier transform of ψ_c is the function

$$\hat{\psi}_c(\theta) = \frac{\mathrm{d}}{\mathrm{d}\beta}\tilde{c}(\lambda)\frac{\widetilde{K}_{(\lambda+d)/2}(\theta)}{|\theta|^{\lambda+d}}\sum_{\alpha\in N}\mu(\alpha)\cos(\alpha\cdot\theta),$$

where $\widetilde{K}_\nu$ is a modified Bessel function of order ν and

$$\tilde{c}(\beta) = \frac{2^{\beta/2+1}(2\pi)}{\Gamma(-\beta/2)}.$$

In the case $\lambda = 2$ and $d = 2$, the constant $(\frac{d\tilde{c}}{d\beta})(2)$ is computed as $(\frac{d\tilde{c}}{d\beta})(2) = 4\pi$ by using tools like MATHEMATICA.

Next, as an example, we present a C^∞-cutoff function σ as the tensor product of a one-variable C^∞-cutoff function σ^1 whose support lies in the ball B_M with $0 < M < 2\pi$, so that $\sigma = 1$ on $B_{M/2}$ and $\|\sigma\| = 1$. For $t \in \mathbb{R}$, let

$$g(t) := \begin{cases} C_0 \exp\left(-\dfrac{1}{1-|t|^2}\right), & t \in [-1,1], \\ 0, & t \in [-1,1]', \end{cases}$$

with

$$C_0 := \left[\int_{[-1,1]} \exp\left(-\frac{1}{1-|t|^2}\right) dt\right]^{-1}.$$

We know that $g \in C^\infty(\mathbb{R}^d)$, $\|g\|_1 = 1$, and it has support in $[-1,1]$. Then, for any $\varepsilon > 0$,

$$g_\varepsilon := \varepsilon^{-1} g(\cdot/\varepsilon)$$

has support in $[-\varepsilon, \varepsilon]$ and $\|g_\varepsilon\|_1 = 1$, and

$$\sigma^1 := \chi_{[-M+\varepsilon, M-\varepsilon]} g_\varepsilon$$

satisfies our requirements. Then the cutoff function σ on $\mathbb{R}^d$ is defined as

$$\sigma(x) = \sigma^1(x_1)\cdots\sigma^1(x_d), \quad x = (x_1, \ldots, x_d).$$

In particular, if $d = 2$ and N is as in figure 2, the function $(\sigma/\hat{\psi}_\rho)^\vee$ is simplified as

$$\left(\frac{\sigma}{\hat{\psi}_\rho}\right)^\vee(\theta) = \frac{1}{\pi^2}\int_{[0,M]^2}\frac{\sigma}{\hat{\psi}_\rho}(s)\cos(\theta\cdot s)\,ds.$$

5.5. Discretization

We are now looking for a discretization of Lf which makes computation easy. For this, we recall the formula of Rf (1.6) which deals with the case when the center set Ξ is infinite in $\mathbb{R}^d$. Then, let us consider its discretization

$$Rf := \sum_{\alpha\in\mathbb{Z}^d} K(\cdot,\alpha)\Lambda f(\omega\alpha).$$

We realize from [21] that Rf is same as a function obtained by applying the conversion tool in (1.3) to a nonstationary uniform scheme developed by de Boor and Ron [3]. Thus, by employing the results in [3] and by applying the same technique of analysis in section 4 and [20], we find that the discretization

$$Lf := \sum_{\alpha\in\mathbb{Z}^d\cap\Omega_\delta/\omega} K(\cdot,\alpha)\Lambda(\chi_{\Omega_{2\delta}}F)(\omega\alpha)$$

preserves the same approximation orders obtained in section 4. The convolution operator Λ also can be discretized in the same way. The reader is referred to the papers [3,8,20,21] for description of these works. When the surfaces of the data are rather complicated and the data are not dense enough, it may be better to use ω as $\omega = \omega(\eta)$ with $\eta = \min_{\xi\neq\xi'\in\Xi} |\xi-\xi'|$. One may choose a compromised a parameter between $\omega(h)$ and $\omega(\delta)$.

6. Numerical results

In this section, we illustrate the accuracy of approximation and smoothing effects (when the data are contaminated) by using the scheme L with two specific examples. Also, some comparisons are given with the thin-plate spline interpolation and Wahba's thin-plate smoothing spline approximation.

In example 6.1, with a given function f, we observe the approximation power of L from the space $S_\Xi(\phi_c)$. We assume that a set Ξ is given in a larger area containing $[-1,1]^2$ so that, for the given function f, we see the approximation behavior of L on $[-1,1]^2$ by using all the given scattered shifts $\phi_c(\cdot-\xi)$ with $\xi\in\Xi$. In this case, since the scheme L is local, the boundary effects do not spill over into the interior of the domain. Currently, one of the most well-known approximation methods to scattered data is the thin-plate spline (TPS) interpolation. Therefore, a comparison is presented between these two approximants.

Next, in example 6.2, we consider approximation of a function f known only at finitely many centers in $[-1,1]^2$ with noise, i.e., the data $(\xi, y_\xi)_{\xi\in\Xi}$ are of the form

$$y_\xi = f(\xi) + \varepsilon_\xi, \quad \xi\in[-1,1]^2,$$

where ε_ξ's are independent noise with mean 0 and variance σ^2. So, in order to eliminate boundary effects, we augment the space $S_\Xi(\phi_c)$ by adding some extra centers around $[-1,1]^2$. Among the other approaches for smoothing noisy data, Wahba's thin-plate

smoothing spline (TPSS) technique is widely used. So we provide a comparison between L and TPSS. In the following examples, all the scattered centers are generated by a random number generator in MATLAB.

Example 6.1. As a first example, we approximate a C^∞-function

$$f(x, y) = -\exp\bigl(-\bigl(x^2 + y^2\bigr)\bigr) + \left[\frac{\sin(x)\sin(y)}{xy}\right]^5.$$

A set of 200 scattered centers are generated in $[-3, 3]$ (see figure 3(A)), and we observe the accuracy of approximation by L over the square $[-1, 1]^2$, the grayed area in figure 3(A). In spite of 200 centers in $[-3, 3]$, these points are very irregularly distributed. We find that there are big holes among the centers, especially on the right-hand side of the grayed area. Figure 3(B) shows the original function. A comparison has been made between L and TPS interpolation. Figures 3(C) and (D) show the approximants by TPS interpolation and by L, respectively. The differences are obvious from the surfaces of approximants and the absolute maximal errors, 0.1682 by TPS interpolation and 0.0397 by L. In fact, the advantage of the method in this paper is due to the 'shifted' thin-plate spline which provides better convergence orders of error estimates for smooth functions. However, the thin-plate spline interpolation provides a stationary approximation order which is determined by the order of singularity of the Fourier transform of the basis function at the origin. Accordingly, we can not expect a higher rate of error convergence (beyond a certain order) when we approximate smoother functions (e.g., C^∞ functions). The stationary case was analyzed in great detail in the literature. Among them, the readers are referred to the paper [3].

Here, in order to compute the TPS interpolation, a MATLAB code written by de Boor was used. This software has been used already in some different sites. Also, to construct the approximant Lf, the values of the parameters c and ω are initialized as $c = 1.2$ and $\omega = 0.6$. The coefficients $(C(t, \xi))_{\xi\in\Xi}$ for $\phi_c(\cdot, t)$ are chosen to be admissible for Π_7 on Ω, i.e., $\sum_{\xi\in\Xi} C(\cdot, \xi)p(\xi) = p$ for $p \in \Pi_7$, and they are computed with penalty function

$$\eta(t, \xi) = \bigl\|\phi_c(\cdot - t) - \phi_c(\cdot - \xi)\bigr\|^2_{L_\infty(\Omega)}$$

by theorem 5.1.

Example 6.2. This example is given to show a noisy data approximation on $[-1, 1]^2$. With given data $(\xi, y_\xi)_{\xi\in\Xi}$ of the form

$$y_\xi = f(\xi) + \varepsilon_\xi,$$

we construct an approximant to the underlying function

$$f(x, y) = \bigl[1.4B_4\bigl(1.2(x - 0.8)\bigr) + B_3\bigl(1.2(x + 0.8)\bigr)\bigr]\exp\bigl(-y^2\bigr),$$

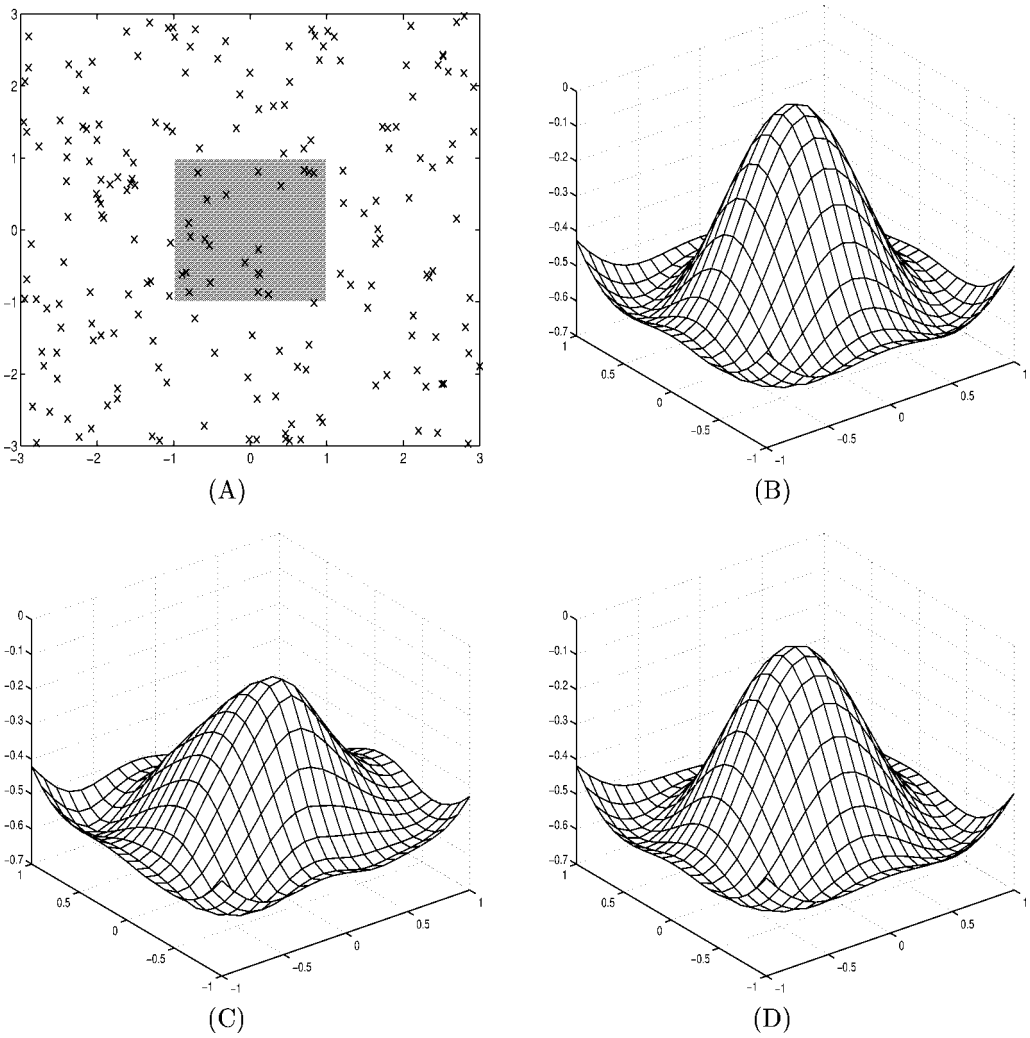

Figure 3. Scattered points (A) and surfaces associated to example 6.1. The dark spot in (A) indicates the area on which we want to approximate. The orginal function f is shown in (B). Figure (C) displays the approximant by the thin-plate spline interpolation, and (D) shows Lf.

where B_k ($k = 3, 4$) indicates the kth order standard spline. Here, ε_ξ's are independent normally distributed random variables with mean 0 and variance $\sigma = 0.05$. Figure 4(A) displays a set of 160 scattered centers Ξ. The sets Ξ and $(\varepsilon_\xi)_{\xi\in\Xi}$ also come from a random number generator in MATLAB. As we discussed in section 5, the first step of noisy data approximation is to generate an extrapolation F on a superset of $[-1, 1]^2$ by using $(\xi, y_\xi)_{\xi\in\Xi}$. The coefficients $(C_f(t, \xi))_{\xi\in\Xi}$ for F are computed by theorem 5.1 with

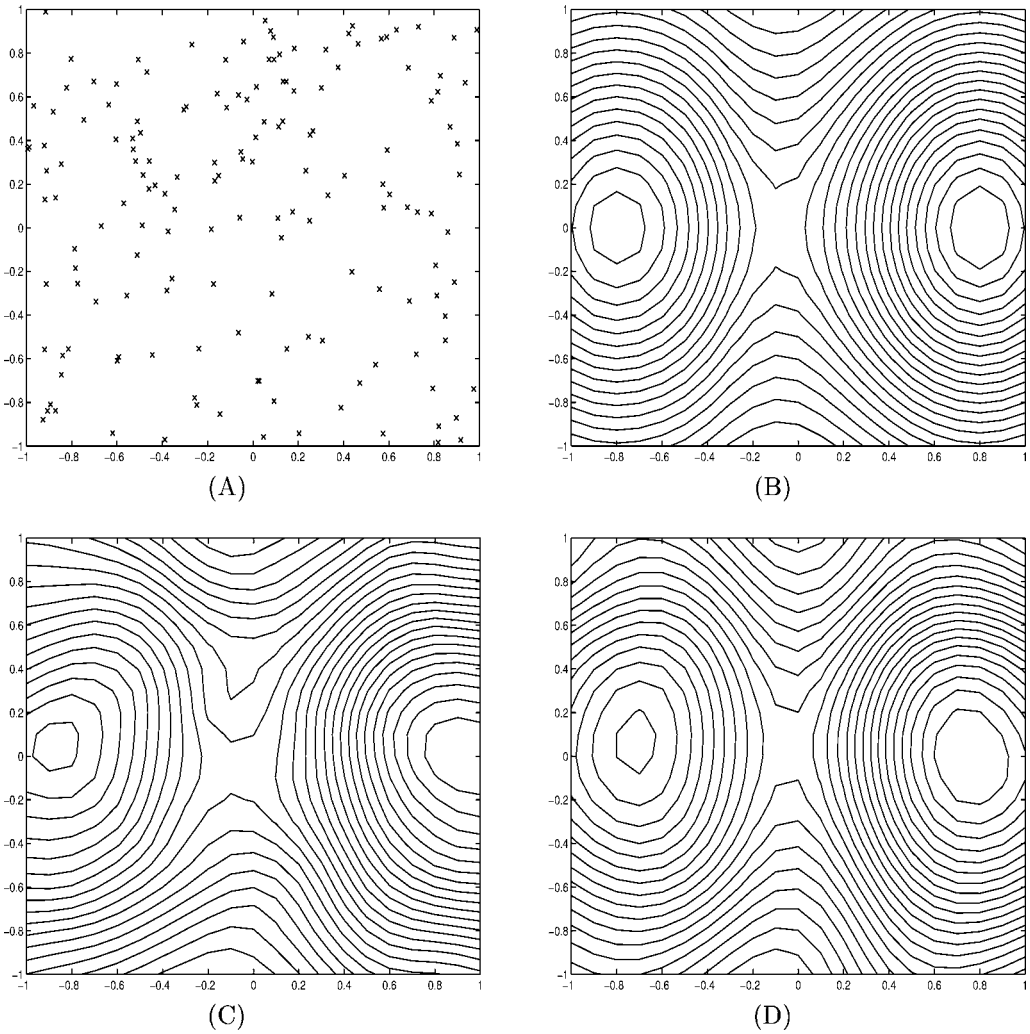

Figure 4. Noisy data approximation of example 6.2. Figure (A) displays the scattered points $[-1, 1]^2$, where the data arise. The underlying function f is shown in (B). Figure (C) displays the approximant by TPSS, and (D) shows Lf.

a penalty function

$$\eta(t, \xi) = \left[\exp\left(\frac{|t-\xi|^2}{h^2}\right) - 1\right] |t-\xi|^2.$$

Since the data are contaminated, we use a low degree of polynomial reproduction $n = 1$ to construct the coefficients $(C_f(t, \xi))_{\xi\in\Xi}$ for F, i.e., $\sum_{\xi\in\Xi} C(\cdot, \xi)p(\xi) = p$ for $p \in \Pi_1$, regardless of the smoothness of the underlying function f. The coefficients $(C(t, \xi))_{\xi\in\Xi}$ for $\phi_c(\cdot, t)$ are computed by the same way as in example 6.1 with $n = 5$, which is the minimal degree of polynomial reproduction, (see section 4). In particular,

because of the uncertainty of extrapolation outside $[-1, 1]^2$, we technically adopt a high tension $\omega = 0.1$ and assign $c = 0.5$. Figure 4(B) gives the contour lines of original function f. A comparison between L and Wahba's thin-plate smoothing spline (TPSS) is made in figures 4(C) and (D). They look similar at first glance, but we find that the approximant by TPSS has wiggles and loses details at some spots. There are no big differences between approximation errors, but Lf provides a slightly smaller error, 0.1136 by TPSS and 0.1010 by Lf. In order to compute the TPSS approximant, the software GCVPACK (which is available from netlib) has been used.

Remark. A well-known drawback of radial basis function (e.g., TPS) interpolation is that, with the increase of the number of centers, it requires the computation with a huge matrix which is very ill-conditioned. However, the scheme L requires us to solve the linear system

$$\sum_{\xi \in \Xi_t} c(t, \xi) p(\xi) = p(t), \quad p \in \Pi_n,$$

which depends only on the degree of polynomials. Furthermore, in case that a huge set of scattered centers is given, it is possible to do parallel computations by dividing the domain into several pieces. Unfortunately, a direct method of selecting the parameters c and ω has not yet been developed. However, though we have the issue of choosing those tuning parameters, the visual appearances of approximants are not sensitive to the choice of c and ω.

Acknowledgements

I am grateful to Prof. Amos Ron for many helpful advices and suggestions which he offered during this work. This work was supported by Research Assistantships of Prof. Ron at University of Wisconsin-Madison (NSF Grant DMS-9626319 and the U.S. Army Research Office Contract DAAH04-95-1-0089). I would also like to express my appreciation to Prof. Carl de Boor for his help for the algorithm and numerical tests.

References

[1] M. Abramowitz and I. Stegun, *A Handbook of Mathematical Functions* (Dover, New York, 1970).
[2] R.K. Beatson and W.A. Light, Quasi-interpolation in the absence of polynomial reproduction, in: *Numerical Methods of Approximation Theory*, eds. D. Braess and L.L. Shumaker (Birkhäuser, Basel, 1992) pp. 21–39.
[3] M.D. Buhmann, N. Dyn and D. Levin, On quasi-interpolation with radial basis functions with scattered centres, Constr. Approx. 11 (1995) 239–254.
[4] C. de Boor and A. Ron, Fourier analysis of the approximation power of principal shift-invariant spaces, Constr. Approx. 8 (1992) 427–462.
[5] C. de Boor and A. Ron, Computational aspects of polynomial interpolation in several variables, Math. Comp. 58 (1992) 705–727.

[6] N. Dyn, Interpolation and approximation by radial and related functions, in: *Approximation Theory VI*, eds. C.K. Chui, L.L. Schumaker and J. Ward (Academic Press, New York, 1989) pp. 211–234.
[7] N. Dyn, I.R.H. Jackson, D. Levin and A. Ron, On multivariate approximation by integer translates of a basis function, Israel J. Math. 78 (1992) 95–130.
[8] N. Dyn and A. Ron, Radial basis function approximation: from gridded centers to scattered centers, Proc. London Math. Soc. 71 (1995) 76–108.
[9] I.M. Gelfand and G.E. Shilov, *Generalized Functions*, Vol. 1 (Academic Press, New York, 1964).
[10] M. Johnson, A bound on the approximation order of surface splines, Constr. Approx. 14 (1998) 429–438.
[11] D. Levin, The approximation power of moving least-squares, Math. Comp. 67 (1998) 1517–1531.
[12] C.A. Micchelli, Interpolation of scattered data: Distance matrices and conditionally positive functions, Constr. Approx. 2 (1986) 11–22.
[13] W.R. Madych and S.A. Nelson, Multivariate interpolation and conditionally positive function I, Approx. Theory Appl. 4(4) (1988) 77–89.
[14] W.R. Madych and S.A. Nelson, Multivariate interpolation and conditionally positive function II, Math. Comp. 54 (1990) 211–230.
[15] M.J.D. Powell, The theory of radial basis functions approximation in 1990, in: *Wavelets, Subdivision Algorithms and Radial Basis Functions*, ed. W.A. Light, Advances in Numerical Analysis, Vol. II (Oxford Univ. Press, Oxford, 1992) pp. 105–210.
[16] M.J.D. Powell, The uniform convergence of thin plate spline interpolation in two dimensions, Numer. Math. 9 (1994) 107–128.
[17] A. Ron, Negative observations concerning approximations from spaces generated by scattered shifts of functions vanishing at ∞, J. Approx. Theory 78 (1994) 364–372.
[18] Z. Wu and R. Shaback, Local error estimates for radial basis function interpolation of scattered data, IMA J. Numer. Anal. 13 (1993) 13–27.
[19] J. Yoon, Approximation to scattered data by radial basis function, Ph.D. thesis, Department of Mathematics, University of Wisconsin, Madison, 1998.
[20] J. Yoon, Approximation in $L^p(\mathbb{R}^d)$ from a space spanned by the scattered shifts of radial basis function, Constr. Approx. 17 (2001) 227–247.
[21] J. Yoon, Nonstationary approximation schemes on scattered centers in $\mathbb{R}^d$, manuscript.