Scattered data reconstruction by regularization in B-spline and associated wavelet spaces

Michael J. Johnson, Zuowei Shen and Yuhong Xu

Abstract

The problem of fitting a nice curve or surface to scattered, possibly noisy, data arises in many applications in science and engineering. In this paper, we solve the problem, using a standard regularized least square framework, in an approximation space spanned by the shifts and dilates of a single compactly supported function ϕ. We first provide an error analysis to our approach which, roughly speaking, states that the error between the exact (probably unknown) data function and the obtained fitting function is small whenever the scattered samples have a high sampling density and a low noise level. We then give a computational formulation in the univariate case when ϕ is a uniform B-spline and in the bivariate case when ϕ is the tensor product of uniform B-splines. Though sparse, the arising system of linear equations is ill-conditioned; however, when written in terms of a short support wavelet basis with a well-chosen normalization, the resulting system, which is symmetric positive definite, appears to be well-conditioned, as evidenced by the fast convergence of the conjugate gradient iteration. Finally, our method is compared with the classical cubic/thin-plate smoothing spline methods via numerical experiments, where it is seen that the quality of the obtained fitting function is very much equivalent to that of the classical methods, but our method offers advantages in terms of numerical efficiency. We expect that our method remains numerically feasible even when the number of samples in the given data is very large.

Keywords: scattered data reconstruction, regularized least square, principal shift invariant spaces, B-splines, wavelets, conjugate gradient method

Michael J. Johnson, Department of Mathematics and Computer Science, Kuwait University, P.O. Box 5969, 13060 Safat, Kuwait. Email: johnson@mcc.sci.kuniv.edu.kw.

Zuowei Shen and Yuhong Xu, Department of Mathematics, National University of Singapore, 2 Science Drive 2, Singapore 117543. Emails: matzuows@nus.edu.sg, xuyuhong@nus.edu.sg.

1 Introduction

The concern of this paper is the reconstruction of a curve or surface from given scattered data via a principal shift invariant system and its dilations. Scattered data reconstruction (also known as scattered data fitting) problems arise in many fields and applications, such as signal processing, computer graphics and neural networks. In a typical scattered data reconstruction problem, we are given a set of scattered data sites $\Xi = \{x_1, x_2, \cdots, x_n\} \subset \mathbb{R}^d$ and associated function values $f|_\Xi = \{f_1, f_2, \cdots, f_n\}$, and we seek a function g, belonging to a prescribed function space $\mathcal{H}$ (e.g. C^2 or the Sobolev space W_2^m), which fits the given data $\{(x_i, f_i)\}_{i=1}^n$ well. In contrast to *gridded data*, whose data sites are regularly spaced or latticed, scattered data makes no assumptions on the locations of the data sites, and this is what makes scattered data reconstruction a difficult problem.

The two basic approaches to scattered data reconstruction are interpolation and approximation: Interpolation requires the fitting function g to exactly reconstruct (or interpolate) the given data (i.e. $g(x_i) = f_i$), while approximation (i.e. $g(x_i) \approx f_i$) allows g to deviate from the given data. Interpolation is usually applied to noise-free data, while approximation is suitable when the given data is contaminated by noise or contains more detail than actually required by the application.

A classical approach to scattered data approximation is the cubic smoothing spline (for 1D data), or the surface smoothing spline (for multi-dimensional data), which is posed as the solution of the following regularized least square problem:

$$\text{minimize } \sum_{i=1}^{n} (g(x_i) - f_i)^2 + \alpha |g|_{H^m}^2, \tag{1.1}$$

where the minimization is taken over all functions g belonging to the Beppo-Levi space H^m (defined in section 2). Minimization problem (1.1) is a standard regularization problem — the first least square term measures the fitting error, while the second (regularization) term measures the roughness of g. The parameter $\alpha > 0$ is called the regularization (or smoothing) parameter, which serves as a weight to adjust the balance between the two terms. Large values of α will lead to a very smooth function g, at the cost of a potentially large fitting error, while small values of α will lead to a small fitting error, but with a potentially rough fitting function g (i.e. one with $|g|_{H^m}$ large). When $d = 1$ and $m = 2$, the solution to (1.1) is the cubic smoothing spline (see e.g. [24]), while if $2m > d \geq 2$, and under a mild condition on the location of the data sites, the solution to (1.1) is a surface spline (called a thin-plate spline when $m = d = 2$) of order m (see [18, 37]). The smoothing spline is a popular method for scattered data approximation in a wide range of applications (see e.g. [10, 42]). However, in the multivariate setting ($d \geq 2$) the method becomes computationally expensive as the number of data sites n grows large. One reason for this is the lack of a compactly supported basis for the finite dimensional space in which g is found. Representing the solution g in terms of a globally supported basis causes expensive evaluations of g, full $n \times n$ matrices to be stored and inverted, and on top of all that, these full matrices tend to be seriously ill-conditioned (see e.g. [20, 40]). Although significant progress has been made in the direction of reducing these computational difficulties (see [2] and the references therein), we believe alternative approaches to the problem are worth pursuing, not only for the sake of more efficient methods, but also for the insight they may provide in constructing a preconditioner for smoothing splines.

Our approach is to solve minimization problem (1.1), not over the Beppo-Levi space, but rather, over the principal shift invariant (PSI) space generated by a single, carefully chosen, compactly supported function ϕ. Denoting this subspace by $S^h(\phi)$, where h is the scale parameter that controls refinement of the subspace, we thus arrive at the following minimization problem:

$$\text{minimize } \sum_{i=1}^{n} (g(x_i) - f_i)^2 + \alpha |g|_{H^m(\Omega)}^2, \quad g \in S^h(\phi), \tag{1.2}$$

where Ω is a domain of interest which contains Ξ. Here we choose a proper PSI space as the approximation space since it enjoys several desirable properties for data fitting. It has a simple structure and provides good approximation to smooth functions, which leads to simple and accurate algorithms. Furthermore, the PSI space can be associated to a wavelet system and one can then solve the data fitting problem in the wavelet domain with an efficient algorithm as well as a sparse approximation to the data function.

The solution of (1.2) can be viewed as an approximation to the solution of (1.1), since $S^h(\phi)$ will be a subspace of H^m. This, of course, raises the issue as to whether the solution of (1.2) will be as useful, in applications, as the solution of (1.1). We address this issue in two ways. First, we provide an error analysis for our approach, in section 2, which estimates the $L_p(\Omega)$-norm of the error $f - g$ in terms of the data site density in Ω and the noise level in the given data. We then implement an algorithm to solve (1.2), by choosing ϕ as a uniform B-spline or its 2D tensor product, and apply it to a few examples of curve/surface fitting. The numerical experiments demonstrate that the solutions to (1.1) and (1.2) are very close. Moreover, the computation can be very efficiently performed if one solves the problem in the wavelet domain, as evidenced by the numerical experiments in section 4. Compared to the surface smoothing spline, we expect that our approach will remain feasible on larger data sets and hence will extend the scope of applications.

For scattered data fitting, there are a lot of existing methods and algorithms in the literature; a survey on scattered data interpolation, in which various methods are extensively tested and compared can be found in [22]. An error analysis for an interpolation method in PSI spaces which inspired us to consider the approximation approach in this paper is given in [34]. For approximation methods, as we discussed above, cubic and surface smoothing splines have a solid mathematical foundation and have been found to be effective in practice (see e.g. [24, 44]). These are special cases of a more general class of interpolation techniques, called radial basis function interpolation, for which a rich theory on its approximation power is available (see e.g. [9]), and on which many successful applications are built, e.g., neural network [21] and 3D object reconstruction [10]. Several approximation methods employ a multilevel structure to approximate data efficiently. In particular, a multilevel scheme based on B-splines is proposed in [35] to approximate scattered data; a wavelet-based smoothing method which operates in a coarse-to-fine manner to get the fitting function efficiently is suggested in [12]. We mention that the use of uniform B-splines as basis functions for scattered data approximation is not new. The approaches taken in [1, 29, 41, 43] are examples of such approaches; however, in the present contribution, we provide an analysis of the approximation power and conduct numerical experiments in both B-spline and wavelet domains.

The remainder of this paper is organized as follows. In section 2, a brief introduction to PSI spaces is given and is followed by an error analysis for the solution of (1.2). In section 3, we consider computational formulations in one dimensional and two dimensional settings, where ϕ is chosen to be a uniform B-spline in 1D and a tensor product of uniform B-splines (or a particular box spline) in 2D. We briefly review the basics of B-spline (box spline) and wavelets, and then present the computational procedures in both B-spline (box spline) and wavelet domain. Finally, in section 4, several examples from curve/surface fitting are used to examine the effectiveness of the proposed method, and numerical experiments are used to demonstrate the computational efficiency of solving data fitting problems in the wavelet domain.

2 PSI approach to scattered data approximation

This section is devoted to an error analysis of the solution to (1.2). We start with an introduction to PSI spaces, and then give two problem formulations and prove corresponding error estimates.

2.1 PSI spaces

We first introduce some notation that will be used throughout the paper. In $\mathbb{R}^d$, we use the standard multi-index notations. For multi-indices $\alpha = \{\alpha_1, \alpha_2, \cdots, \alpha_d\}$, define $|\alpha| := \alpha_1 + \alpha_2 + \cdots + \alpha_d$, $D^\alpha := (\partial^{\alpha_1}/\partial x_1^{\alpha_1})(\partial^{\alpha_2}/\partial x_2^{\alpha_2}) \cdots (\partial^{\alpha_d}/\partial x_d^{\alpha_d})$; for $x \in \mathbb{R}^d$, define $|x| := \sqrt{x_1^2 + x_2^2 + \cdots + x_d^2}$; for $x, y \in \mathbb{R}^d$, let $x \cdot y$ denote the inner product between them. One often employed set in $\mathbb{R}^d$ is the open unit ball $B := \{|x| < 1, x \in \mathbb{R}^d\}$.

Let m be a positive integer, and let H^m denote the Beppo-Levi space of tempered distributions f for which $D^\alpha f \in L_2(\mathbb{R}^d)$ for all $|\alpha| = m$. For measurable $\Omega \subset \mathbb{R}^d$ and $f \in H^m$, we define the seminorm

$$|f|_{H^m(\Omega)} := (2\pi)^{d/2} \sqrt{\sum_{|\alpha|=m} \tau_\alpha \|D^\alpha f\|^2_{L_2(\Omega)}},$$

where the τ_α's are the positive integers determined by the equation $|x|^{2m} = \sum_{|\alpha|=m} \tau_\alpha x^{2\alpha}, x \in \mathbb{R}^d$. If $\Omega = \mathbb{R}^d$, we write simply $|f|_{H^m}$. It can be easily shown that $|f|_{H^m}$ has the representation in the Fourier domain as $\||\cdot|^m \widehat{f}\|_{L_2(\mathbb{R}^d \backslash 0)}$ for all $f \in H^m$, where $\widehat{f}(\xi) := \int_{\mathbb{R}^d} f(x) e^{-i\xi \cdot x} \mathrm{d}x$ denotes the Fourier transform of f. With this representation, it easily follows that $|f(h\cdot)|_{H^m} = h^{m-d/2}|f|_{H^m}$.

Let W_2^m denote the Sobolev space of all tempered distributions f for which $D^\alpha f \in L_2(\mathbb{R}^d)$ for all $|\alpha| \le m$. In the Fourier domain, the Sobolev norm can be defined as follows

$$\|f\|_{W_2^m} := \|(1 + |\cdot|^2)^{m/2} \widehat{f}\|_{L_2(\mathbb{R}^d)}.$$

We now define a principal shift-invariant (PSI) space. Let $\phi : \mathbb{R}^d \to \mathbb{R}$ be a continuous and compactly supported function, and let $c : \mathbb{Z}^d \to \mathbb{R}$ be a sequence. The semi-discrete convolution between ϕ and c is defined by

$$\phi *' c := \sum_{j \in \mathbb{Z}^d} c(j) \phi(\cdot - j).$$

The principal shift-invariant space $S(\phi)$ generated by ϕ is the smallest closed subspace of $L_2(\mathbb{R}^d)$ that contains all functions $\phi *' c$, where c is a finitely supported sequence; that is,

$$S(\phi) = closure\{\phi *' c : c \in \ell_0(\mathbb{Z}^d)\},$$

where $\ell_0(\mathbb{Z}^d)$ denotes the set of all finitely supported sequences on $\mathbb{Z}^d$. The space $S(\phi)$ can be refined by dilation, and we define for $h > 0$

$$S^h(\phi) = \{f(\cdot/h) : f \in S(\phi)\}.$$

PSI spaces are particularly important in the field of approximation theory, due to the following appealing properties. The structure of a PSI space is simple, as the space can be generated by only one function ϕ (which is called the generator). A PSI space provides good approximation to W_2^m if ϕ satisfies the Strang-Fix conditions. Recall that a function ϕ is said to satisfy the Strang-Fix conditions of order m if

$$\widehat{\phi}(0) \neq 0, \quad \text{and} \quad D^\alpha \widehat{\phi}(2\pi j) = 0, \ \forall j \in \mathbb{Z}^d \backslash 0, |\alpha| < m.$$

It is well-known (see [7, 31]) that ϕ satisfies the Strang-Fix conditions of order m if and only if for all $f \in W_2^m$,

$$\inf_{s \in S^h(\phi)} \|f - s\|_{L_2(\mathbb{R}^d)} = O(h^m) \text{ as } h \to 0.$$

Further, PSI spaces also have an associated wavelet system, provided the generator ϕ satisfies some conditions, e.g. refinability, which will be discussed in the next section. The interested reader is referred to [5, 7] for more discussions on PSI spaces.

In most applications, data to be processed comes from a bounded subset of $\mathbb{R}^d$. For a bounded domain $\Omega \subset \mathbb{R}^d$, we work with a space spanned by those shifts of ϕ whose support intersects the interior of Ω, namely,

$$S(\phi, \Omega) := \{\phi *' c : \ c(j) = 0 \text{ whenever } supp\ \phi(\cdot - j) \cap \Omega^o = \emptyset\}.$$

The space $S(\phi, \Omega)$ can also be refined by dilation, and so we define

$$S^h(\phi, \Omega) = \left\{ \sum_{j \in \mathbb{Z}^d} c(j)\phi(\cdot/h - j) : c(j) = 0 \text{ whenever } supp\ \phi(\cdot/h - j) \cap \Omega^o = \emptyset \right\}.$$

This space $S^h(\phi, \Omega)$ is the approximation space in which we will formulate our regularized least square schemes and look for numerical solutions.

At this stage, there is no guarantee that the above-mentioned functions $\phi(\cdot/h - j)$, which span $S^h(\phi, \Omega)$, are linearly independent over Ω. Although that is of no concern at the theoretical level, it is an important consideration when one begins to make numerical computations. The concept of *local linear independence* is precisely the one needed: The shifts of ϕ are *locally linearly independent* if for every bounded open set G, all shifts of ϕ (ie $\phi(\cdot - j)$, $j \in \mathbb{Z}^d$) having some support in G are linearly independent over G. For the sake of generality, we will not assume, at this early stage, that the shifts of ϕ are locally linearly independent; however, we will explicitly state this assumption later on as needed.

2.2 Approximation in PSI spaces

We shall propose two regularization schemes and then pursue an estimate on the error in terms of data site density and noise level. Let $m > d/2$ be an integer, and assume that $\phi \in W_2^m$ is compactly supported and satisfies the Strang-Fix conditions of order m. Let Ω be a bounded subset in $\mathbb{R}^d$ and let $f \in W_2^m$, but assume that we are given a noisy sample $\widetilde{f}|_\Xi$ at scattered data sites $\Xi \subset \Omega$, with the noise level satisfying

$$\|f - \widetilde{f}\|_{\ell_2(\Xi)} \le \epsilon.$$

In the first scheme we seek an $s \in S^h(\phi, \Omega)$ which nearly minimizes

$$e_\alpha(s, \widetilde{f}, \Xi) := \alpha |s|^2_{H^m(\Omega)} + \|s - \widetilde{f}\|^2_{\ell_2(\Xi)}, \tag{2.1}$$

while in the second we seek an $s \in S^h(\phi, \Omega)$ which nearly minimizes

$$E_\alpha(s, \widetilde{f}, \Xi) := \alpha \|s\|^2_{W_2^m(\Omega)} + \|s - \widetilde{f}\|^2_{\ell_2(\Xi)}. \tag{2.2}$$

Here the phrase "nearly minimize" means to bring to within a constant of it's minimal value. For example, to choose $g \in G$ to nearly minimize $\|g\|$ means to choose g so that $\|g\| \le const\ \inf\{\|\widetilde{g}\| : \widetilde{g} \in G\}$.

We say that Ω has the cone property if there exist positive constants $\epsilon_\Omega, r_\Omega$ such that for all $x \in \Omega$ there exists $y \in \Omega$ such that $|x - y| = \epsilon_\Omega$ and

$$x + t(y - x + r_\Omega B) \subset \Omega, \quad \forall t \in [0, 1].$$

The separation distance in Ξ is defined by

$$sep(\Xi) := \inf\{|\xi - \xi'| : \xi, \xi' \in \Xi,\ \xi \ne \xi'\}.$$

The fill distance from Ξ to Ω is given by

$$\delta := \delta(\Xi, \Omega) := \sup_{x \in \Omega} \inf_{\xi \in \Xi} |x - \xi|.$$

And the accumulation index of Ξ in Ω is defined by

$$\gamma := \gamma(\Xi, \Omega) := \max_{x \in \Omega} \#\{\xi \in \Xi : |x - \xi| \le \delta\}.$$

To derive error estimates, we start with some lemmas and propositions.

Lemma 2.1. *Let $\Xi \subset \Omega$, where Ω is a bounded subset of $\mathbb{R}^d$ having the cone property with parameters ϵ_Ω and r_Ω. With $\delta := \delta(\Xi, \Omega)$, the following hold:*

(i) There exists $\delta_0 > 0$ (depending only on ϵ_Ω and r_Ω) such that if $\delta \le \delta_0$, then there exists $\Xi_0 \subset \Xi$ such that $\delta(\Xi_0, \Omega) \sim \delta$ and $sep(\Xi_0) \sim \delta$, where the equivalency constants depend only on ϵ_Ω and r_Ω;

(ii) There exists a partition of Ξ, $\Xi = \bigcup_{i=1}^n \Xi_i$, such that $n \le const(d)\gamma$ and $sep(\Xi_i) \ge \delta$ for $i = 1, 2, \cdots, n$.

Proof. Put $\delta_0 := r_\Omega/(5\sqrt{d} + 2)$ and assume $\delta := \delta(\Xi, \Omega) \le \delta_0$.

(i) Define a set of lattice nodes as follows

$$P = \{5\delta j \in \Omega, j \in \mathbb{Z}^d\}.$$

The cone property implies that there is a ball with radius r_Ω lying inside Ω. It is easy to see, by the choice of δ, that this ball contains at least two points of the form $5\delta j$ ($j \in \mathbb{Z}^d$). Hence P is not empty and has at least two nodes. For any $p \in P$, by the definition of δ, $\inf_{\xi \in \Xi} |p - \xi| \le \delta$, and hence there exists a $\xi_p \in \Xi$ such that $|p - \xi_p| < 2\delta$. Define Ξ_0 by picking one such ξ_p for each $p \in P$ and collecting them together, i.e.,

$$\Xi_0 = \{\xi_p : \xi_p \in \Xi, |p - \xi_p| < 2\delta, p \in P\}.$$

By the triangle inequality, it follows from the construction of P and Ξ_0 that $|\xi_p - \xi_q| \ge \delta$ for any pair $p, q \in P$, and that $|\xi_p - \xi_q| \le 9\delta$ for any two neighboring nodes $p, q \in P$. Hence $sep(\Xi_0) \sim \delta$.

For any $x \in \Omega$, by the cone property, there exists y such that $|x - y| = \epsilon_\Omega$ and $x + t(y - x + r_\Omega B) \subset \Omega,\ \forall t \in [0, 1]$. Let $t = \delta/\delta_0$, the ball $B_1 := x + t(y - x + r_\Omega B) \subset \Omega$, and its radius is $(5\sqrt{d} + 2)\delta$. By the construction of P, there exists a $p \in P$ such that $p + 2\delta B \subset B_1$. By the definition of Ξ_0, there exists a $\xi_p \in \Xi_0$ such that $\xi_p \in B_1$. Then the triangle inequality gives

$$\big|x - \xi_p\big| \le \big|x - (x + t(x - y))\big| + \big|(x + t(x - y)) - \xi_p\big| \le t(\epsilon_\Omega + r_\Omega),$$

from which we have $\delta(\Xi_0, \Omega) \le const(r_\Omega, \epsilon_\Omega)\delta$. On the other hand, $\delta(\Xi_0, \Omega) \ge \delta := \delta(\Xi, \Omega)$ since $\Xi_0 \subset \Xi$. Hence $\delta(\Xi_0, \Omega) \sim \delta$.

(ii) Since Ω is bounded, there is a "bounding box" BD of the form $[l_1, r_1] \times [l_2, r_2] \cdots \times [l_d, r_d]$ which covers Ω. Define a set of lattice nodes

$$Q = \{3\delta j \in BD, j \in \mathbb{Z}^d\}.$$

Associate each node $p \in Q$ with a closed ball $B_p := p + \delta\bar{B}$. By the definition of γ, in B_p there are at most γ points in Ξ. A subset of Ξ can be formed by picking one point from Ξ in each ball if it contains such a point, and grouping them together. Thus, for all the points in Ξ that lie in the balls, we can group them into at most γ such subsets which do not intersect with each other. By the construction of the subsets, the separation distance of each subset is not less than δ.

Let U be the union of all the balls defined above, and consider the translates of U with translation distance of a multiple of $\delta/\sqrt{d}$ on all d directions. We can easily see that a finite number (depending only on d) of such translates cover BD (hence cover Ω). Similarly, we can group the points of Ξ in each translate of U into at most γ subsets, each subset having separation distance no less than δ. This grouping gives us at most $const(d)\gamma$ subsets of Ξ that cover Ξ, and the separation distance of each subset is not less than δ.

□

Our proofs will draw heavily on the ideas and conclusions from [34]. The next inequality plays an essential role in [34]. It is first proved in [18], and has been generalized recently in [39]. In the remainder of this section, it is assumed that Ω is a compact subset of $\mathbb{R}^d$ having the cone property and a Lipschitz boundary.

Lemma 2.2 (Duchon's inequality)**.** *There exists $\delta^* > 0$ (depending only on ϵ_Ω and r_Ω) such that if $\Xi \subset \Omega$ satisfies $\delta := \delta(\Xi, \Omega) \leq \delta^*$, then for all $2 \leq p \leq \infty$*

$$\|g\|_{L_p(\Omega)} \leq const(m, \Omega)\delta^{m-d/2+d/p}|g|_{H^m(\Omega)}, \qquad \forall g \in H^m(\Omega) \text{ with } g|_\Xi = 0.$$

Duchon's inequality is proved for the case of scattered zeros. Here we generalize this inequality as follows to cope with scattered non-zeros for our regularization approaches.

Proposition 2.3. *There exists $\delta_0 > 0$ (depending only on ϵ_Ω and r_Ω) such that if $\Xi \subset \Omega$ satisfies $\delta := \delta(\Xi, \Omega) \leq \delta_0$, then*

$$\|g\|_{L_p(\Omega)} \leq const(m, \Omega)\big(\delta^{m-d/2+d/p}|g|_{H^m(\Omega)} + \delta^{d/p}\|g\|_{\ell_2(\Xi)}\big), \qquad \forall g \in H^m,\ 2 \leq p \leq \infty.$$

Proof. Let $\sigma \in C_c^\infty(\mathbb{R}^d)$ be such that $\sigma(0) = 1$ and $supp\ \sigma \subset B$. Let δ_0 be as in Lemma 2.1, and assume that $\delta \leq \delta_0$. Then, by Lemma 2.1, there exists $\Xi_0 \subset \Xi$ such that $\delta_1 := \delta(\Xi_0, \Omega) \sim \delta$ and $sep(\Xi_0) \sim \delta$. There exists $\tau \sim \delta$ (e.g., $\tau = sep(\Xi_0)/3$) such that the support of the functions $\{\sigma((\cdot - \xi)/\tau)\}_{\xi \in \Xi_0}$ are pairwise disjoint. It then follows that the function

$$\widetilde{g} := g - \sum_{\xi \in \Xi_0} g(\xi)\sigma((\cdot - \xi)/\tau).$$

satisfies

$$\begin{aligned} |\widetilde{g}|_{H^m(\Omega)} &\leq |g|_{H^m(\Omega)} + \Big|\sum_{\xi \in \Xi_0} g(\xi)\sigma((\cdot - \xi)/\tau)\Big|_{H^m} \\ &= |g|_{H^m(\Omega)} + \|g\|_{\ell_2(\Xi_0)}|\sigma(\cdot/\tau)|_{H^m} = |g|_{H^m(\Omega)} + \tau^{-m+d/2}|\sigma|_{H^m}\|g\|_{\ell_2(\Xi_0)}. \end{aligned}$$

Assume that $\delta_1 \leq \delta^*$ as required in Duchon's inequality (otherwise, this condition can be satisfied by scaling δ_0). Noting that $\widetilde{g}|_{\Xi_0} = 0$ and applying Duchon's inequality to $\widetilde{g}$ yields

$$\|\widetilde{g}\|_{L_p(\Omega)} \leq const(m, \Omega)\delta_1^{m-d/2+d/p}\big(|g|_{H^m(\Omega)} + \tau^{-m+d/2}|\sigma|_{H^m}\|g\|_{\ell_2(\Xi_0)}\big).$$

Again, by the pairwise disjoint property, we have

$$\|g - \widetilde{g}\|_{L_p(\Omega)} = \Big\|\sum_{\xi \in \Xi_0} g(\xi)\sigma((\cdot - \xi)/\tau)\Big\|_{L_p} = \|g\|_{\ell_p(\Xi_0)}\|\sigma\|_{L_p}\tau^{d/p} \leq const\,\delta^{d/p}\|g\|_{\ell_2(\Xi)},$$

where the inequality holds since $\delta \sim \tau$ and $\|g\|_{\ell_p(\Xi_0)} \leq \|g\|_{\ell_p(\Xi)} \leq \|g\|_{\ell_2(\Xi)}$ for $2 \leq p \leq \infty$. The proof is finally completed by the triangle inequality $\|g\|_{L_p(\Omega)} \leq \|\widetilde{g}\|_{L_p(\Omega)} + \|g - \widetilde{g}\|_{L_p(\Omega)}$ in conjunction with the equivalencies $\tau \sim \delta_1 \sim \delta$.

□

The generator $\phi \in W_2^m$ is assumed to be compactly supported and to satisfy the Strang-Fix conditions of order m. This assumption on ϕ ensures (see [32, Lemma 2.6]) that there exists a finitely supported sequence $a : \mathbb{Z}^d \to \mathbb{R}$ such that $\psi := \phi *' a$ satisfies the Strang-Fix conditions of order m and the condition $\psi *' q = q$ for all $q \in \pi_{m-1}^d$, where π_{m-1}^d denotes the set of polynomials of degree $\leq (m-1)$. In the following, we will make crucial use of the function $s \in S^h(\phi)$ defined by

$$s := \sum_{j \in Z^d} f(hj)\psi(\cdot/h - j).$$

Proposition 2.4. *For s defined above, and with $h \le 1$, the following hold:*

(i) $|s|_{H^m} \le const(\psi, m)|f|_{H^m}, \quad \forall f \in H^m$;

(ii) $\|s\|_{W_2^m} \le const(\psi, m)\|f\|_{W_2^m}, \quad \forall f \in W_2^m$;

(iii) $\|f - s\|_{\ell_2(\Xi)} \le const(\psi, m)h^m\delta^{-d/2}\sqrt{\gamma}|f|_{H^m}, \quad \forall f \in H^m$.

Proof. (i) Put $s_h := s(h\cdot)$ and $f_h := f(h\cdot)$ and note that $s_h = \psi *' f_h$. By [34, Proposition 5.2],

$$|s_h|_{H^m} \le const(\psi, m)|f_h|_{H^m},$$

and hence (i) follows from the equalities $|s_h|_{H^m} = h^{m-d/2}|s|_{H^m}$ and $|f_h|_{H^m} = h^{m-d/2}|f|_{H^m}$.

(ii) It follows from the proof of [34, Proposition 5.7] that

$$\sum_{j\in Z^d} \|s_h - f_h\|^2_{L_\infty(j+C)} \le const(\psi, m)|f_h|^2_{H^m},$$

where $C := [-1/2, 1/2)^d$ denotes the unit cube. Employing the inequality $\|s_h - f_h\|^2_{L_2} \le \sum_{j\in Z^d} \|s_h - f_h\|^2_{L_\infty(j+C)}$ yields

$$\|s - f\|_{L_2} = h^{d/2}\|s_h - f_h\|_{L_2} \le const(\psi, m)h^{d/2}|f_h|_{H^m} = const(\psi, m)h^m|f|_{H^m}.$$

Hence

$$\|s\|_{L_2} \le \|s - f\|_{L_2} + \|f\|_{L_2} \le const(\psi, m)\|f\|_{W_2^m},$$

which, in view of (i), proves (ii).

(iii) The proof of [34, Proposition 5.7] can be easily modified to show that if $A \subset R^d$ satisfies $\epsilon := sep(A) > 0$, then

$$\|s_h - f_h\|^2_{\ell_2(A)} \le const(\psi, m)\epsilon^{-d}|f_h|^2_{H^m}.$$

By (ii) of Lemma 2.1, it is possible to partition Ξ as $\Xi = \bigcup_{i=1}^n \Xi_i$ such that $n \le const(d)\gamma$ and $sep(\Xi_i) \ge \delta$. With $\widetilde{\Xi}_i := h^{-1}\Xi_i$ we see that

$$\begin{aligned}\|s - f\|^2_{\ell_2(\Xi)} = \|s_h - f_h\|^2_{\ell_2(h^{-1}\Xi)} &= \sum_{i=1}^n \|s_h - f_h\|^2_{\ell_2(\widetilde{\Xi}_i)} \le \sum_{i=1}^n const(\psi, m)sep(\widetilde{\Xi}_i)^{-d}|f_h|^2_{H^m} \\ &\le \gamma const(\psi, m)(h^{-1}\delta)^{-d}|f_h|^2_{H^m} = \gamma const(\psi, m)(\delta)^{-d}h^{2m}|f|^2_{H^m}\end{aligned}$$

which proves (iii). □

With Proposition 2.3 and Proposition 2.4 in hand, we are now ready to give our error estimates for the two schemes proposed at the beginning of this subsection. We assume, without further mention, that $h \le 1$ and $\delta := \delta(\Xi, \Omega) \le \delta_0$, so that we can invoke Proposition 2.3 and Proposition 2.4 in the following.

Theorem 2.5. *If $f \in H^m$ and $S_f \in S^h(\phi, \Omega)$ nearly minimizes $e_\alpha(s, \widetilde{f}, \Xi)$, defined in* (2.1)*, then*

$$\|f - S_f\|_{L_p(\Omega)} \le const(\phi, m, \Omega)\left(\frac{\delta^{m-d/2+d/p}}{\sqrt{\alpha}} + \delta^{d/p}\right)\left(\sqrt{\alpha + h^{2m}\delta^{-d}\gamma}|f|_{H^m} + \epsilon\right)$$

for $2 \le p \le \infty$.

Proof. For any $s_1 \in S^h(\phi,\Omega)$ and for any $s_2 \in S^h(\phi)$ whose support lies outside of Ω, by the definition of $e_\alpha(s,\widetilde{f},\Xi)$, we have

$$e_\alpha(s_1,\widetilde{f},\Xi) = e_\alpha(s_1+s_2,\widetilde{f},\Xi).$$

This implies that a minimizer in $S^h(\phi,\Omega)$ is also a minimizer in $S^h(\phi)$. Hence S_f nearly minimizes $e_\alpha(s,\widetilde{f},\Xi)$ over $S^h(\phi)$. In particular, for $s \in S^h(\phi)$ in Proposition 2.4, we have

$$e_\alpha(S_f,\widetilde{f},\Xi) \le const\cdot e_\alpha(s,\widetilde{f},\Xi).$$

By Proposition 2.4 and triangle inequality, it follows that

$$\begin{aligned}
\alpha|S_f|^2_{H^m(\Omega)} + \|\widetilde{f}-S_f\|^2_{\ell_2(\Xi)} &= e_\alpha(S_f,\widetilde{f},\Xi) \le const\cdot e_\alpha(s,\widetilde{f},\Xi)\\
&= const\big(\alpha|s|^2_{H^m(\Omega)} + \|\widetilde{f}-s\|^2_{\ell_2(\Xi)}\big)\\
&\le const\big(\alpha|s|^2_{H^m} + 2\|f-s\|^2_{\ell_2(\Xi)} + 2\|\widetilde{f}-f\|^2_{\ell_2(\Xi)}\big)\\
&\le const\big((\alpha+h^{2m}\delta^{-d}\gamma)|f|^2_{H^m}+\epsilon^2\big).
\end{aligned}$$

Applying Proposition 2.3 to $(f-S_f)$, we have

$$\begin{aligned}
\|f-S_f\|_{L_p(\Omega)} &\le const\big(\delta^{m-d/2+d/p}|f-S_f|_{H^m(\Omega)} + \delta^{d/p}\|f-S_f\|_{\ell_2(\Xi)}\big)\\
&\le const\big(\delta^{m-d/2+d/p}\big(|f|_{H^m}+|S_f|_{H^m(\Omega)}\big) + \delta^{d/p}\big(\epsilon+\|\widetilde{f}-S_f\|_{\ell_2(\Xi)}\big)\big).
\end{aligned}$$

Using the above estimate on $e_\alpha(S_f,\widetilde{f},\Xi)$ to bound $|S_f|_{H^m(\Omega)}$ and $\|\widetilde{f}-S_f\|_{\ell_2(\Xi)}$ completes the proof. □

The above proof can be easily modified to prove

Theorem 2.6. *If $f \in W_2^m$ and $S_f \in S^h(\phi,\Omega)$ nearly minimizes $E_\alpha(s,\widetilde{f},\Xi)$, defined in (2.2), then*

$$\|f-S_f\|_{L_p(\Omega)} \le const(\phi,m,\Omega)\left(\frac{\delta^{m-d/2+d/p}}{\sqrt{\alpha}}+\delta^{d/p}\right)\big(\sqrt{\alpha+h^{2m}\delta^{-d}\gamma}\|f\|_{W_2^m}+\epsilon\big)$$

for $2 \le p \le \infty$.

When the noise level is very low but not zero, one may want to fit the data closely. In this case, since the smoothing becomes less important, one may choose the smoothing parameter to be small to improve the approximation. For example, if we assume that $h \sim \delta$ and $\alpha \sim \delta^{2m-d}$, so that $\frac{\delta^{m-d/2+d/p}}{\sqrt{\alpha}} \sim \delta^{d/p}$ and $\sqrt{\alpha+h^{2m}\delta^{-d}\gamma} \sim \delta^{m-d/2}$, then the above error bounds can be simplified as follows.

Corollary 2.7. *Suppose $h \sim \delta$ and $\alpha \sim \delta^{2m-d}$. If $f \in H^m$ and $S_f \in S^h(\phi,\Omega)$ nearly minimizes $e_\alpha(s,\widetilde{f},\Xi)$, then*

$$\|f-S_f\|_{L_p(\Omega)} \le const(\phi,m,\Omega,\alpha)\delta^{d/p}(\delta^{m-d/2}|f|_{H^m}+\epsilon).$$

If $f \in W_2^m$ and $S_f \in S^h(\phi,\Omega)$ nearly minimizes $E_\alpha(s,\widetilde{f},\Xi)$, then

$$\|f-S_f\|_{L_p(\Omega)} \le const(\phi,m,\Omega,\alpha)\delta^{d/p}(\delta^{m-d/2}\|f\|_{W_2^m}+\epsilon).$$

3 Approximation in B-spline and wavelet domain

The purpose of this section is to formulate the computational task of solving the minimization problem (3.3) in both the B-spline and wavelet domains, in one and two dimensions.

The first scheme (2.1) differs from the second (2.2) only in the regularization term: the first employs the penalty $|\cdot|_{H^m(\Omega)}$, while the second employs $\|\cdot\|_{W_2^m(\Omega)}$. This means that the first scheme penalizes only function derivatives, while the second penalizes function derivatives as well as function values, which artificially dampens function values. For this reason, we focus on the first scheme and we are interested to find a numerical procedure to

$$\text{minimize} \quad \|s - \widetilde{f}\|^2_{\ell_2(\Xi)} + \alpha |s|^2_{H^m(\Omega)}, \ s \in S^h(\phi, \Omega). \tag{3.3}$$

Uniform B-splines in one dimension or a tensor product of uniform B-splines in two dimensions are good candidates for the function ϕ, since they have explicit form and thus one can efficiently compute their values at scattered sites. Furthermore, they satisfy the desired Strang-Fix conditions and they are associated with wavelets with short support.

3.1 Uniform B-splines and Wavelets

We first introduce B-splines and wavelets in the univariate setting and then extend them to the bivariate setting by tensor product. The uniform B-spline function of order p, denoted by B_p, can be obtained via the following recursive formula: $B_1 = \chi_{[0,1]}$, the characteristic function of the interval $[0, 1]$, and

$$B_p(x) := \int_0^1 B_{p-1}(x - t)\mathrm{d}t, \ x \in \mathbb{R}, \ p = 2, 3, \ldots.$$

It is well known that B_p is a compactly supported piecewise polynomial which satisfies the Strang-Fix conditions of order p. Another property of B_p, which plays an important role in designing the corresponding wavelet system, is refinability, i.e.,

$$B_p(x) = \frac{1}{2^p} \sum_{k=0}^{p} \binom{p}{k} B_p(2x - k).$$

Recall that a function $\phi : \mathbb{R} \mapsto \mathbb{R}$ is refinable if it satisfies the refinement equation

$$\phi = 2 \sum_{k \in \mathbb{Z}} a(k) \phi(2 \cdot -k), \tag{3.4}$$

where $a : \mathbb{Z} \mapsto \mathbb{R}$ is a sequence on $\mathbb{Z}$, called the refinement mask for ϕ.

Next we introduce a key concept in the wavelet theory, multiresolution analysis, which provides the framework for most wavelet constructions, see e.g. [11, 14, 36]. For a compactly supported refinable function $\phi \in L_2(\mathbb{R})$, define $\phi_{j,k} = 2^{j/2}\phi(2^j \cdot -k)$, and let V_j be the PSI space generated by $\phi_{j,0}$:

$$V_j := S(\phi_{j,0}), \ j \in \mathbb{Z}.$$

Then the sequence of PSI spaces V_j ($j \in \mathbb{Z}$) forms a multiresolution analysis (MRA) generated by ϕ, i.e., (i)$V_j \subset V_{j+1}$; (ii)$\overline{\cup_{j\in\mathbb{Z}} V_j} = L_2(\mathbb{R})$; and (iii)$\cap_{j\in\mathbb{Z}} V_j = 0$ (see e.g. [33]).

Based on the theory of MRA, a family of short supported wavelets are induced from uniform B-splines in [26]. Let ϕ be B_p and a its refinement mask, and define

$$\psi(x) = 2 \sum_{k \in \mathbb{Z}} (-1)^{k-1} \overline{a(1 - k)} \phi(2x - k). \tag{3.5}$$

This function ψ is called a *Riesz* wavelet function, since its dyadic system

$$X(\psi) := \{\psi_{j,k} := 2^{j/2}\psi(2^j \cdot -k), j \in \mathbb{Z}, k \in \mathbb{Z}\}$$

is a Riesz basis of $L_2(\mathbb{R})$ (see [26]), meaning there exist $C_1, C_2 > 0$ such that

$$C_1\|\{c_{j,k}\}\|_{\ell_2(\mathbb{Z}^2)} \le \left\| \sum_{j\in\mathbb{Z}} \sum_{k\in\mathbb{Z}} c_{j,k}\psi_{j,k} \right\|_{\ell_2(\mathbb{Z}^2)} \le C_2\|\{c_{j,k}\}\|_{\ell_2(\mathbb{Z}^2)} \tag{3.6}$$

for all $\{c_{j,k}\} \in \ell_2(\mathbb{Z}^2)$ and the span of $X(\psi)$ is dense in $L_2(\mathbb{R})$.

We say that a function ψ has regularity α if $\psi \in W_2^\beta$ for all $\beta < \alpha$, and we say that ψ has p vanishing moments if $D^j\widehat{\psi}(0) = 0$ for $j = 0, \cdots, p-1$. An interesting property of the above-defined ψ is that it has the shortest support among all Riesz wavelets having regularity $p - 1/2$ and p vanishing moments, and for this reason, it is called the "short support" wavelet. We remark that although ϕ and ψ have short support, the corresponding generators of the dual system, i.e. $\widetilde{\phi}$ and $\widetilde{\psi}$, are not compactly supported (see [26]). However, it is shown in [26] that $\widetilde{\phi}$ and $\widetilde{\psi}$ are in $L_2(\mathbb{R})$ and $\widetilde{\psi}$ has p vanishing moments.

With the above ϕ and ψ, we can define the MRA, V_j ($j \in \mathbb{Z}$), and a sequence of wavelet spaces

$$W_j := closure\left\{ \sum_{k\in\mathbb{Z}} c_k\psi_{j,k} : \{c_k\} \in \ell_0(\mathbb{Z}) \right\},\ j \in \mathbb{Z}.$$

We have the following complement relation

$$V_j = V_{j-1} \oplus W_{j-1},\ j \in \mathbb{Z},$$

where $\oplus$ represents the direct sum of subspaces. This complement relation is the basis of the wavelet decomposition and reconstruction algorithm. Let J be an integer, then any function $s \in V_J$ can be represented by

$$s = \sum_{k\in\mathbb{Z}} \langle s, \widetilde{\phi}_{J,k}\rangle \phi_{J,k}. \tag{3.7}$$

On the other hand, the function s has a wavelet representation as follows

$$s = \sum_{k\in\mathbb{Z}} \langle s, \widetilde{\phi}_{J_0,k}\rangle \phi_{J_0,k} + \sum_{j=J_0}^{J-1} \sum_{k\in\mathbb{Z}} \langle s, \widetilde{\psi}_{j,k}\rangle \psi_{j,k}, \tag{3.8}$$

where J_0 is the coarsest level. In order to solve (3.3), we need to find either $\langle s, \widetilde{\phi}_{J,k}\rangle$, that means we solve (3.3) in B-spline domain, or $\langle s, \widetilde{\phi}_{J_0,k}\rangle$ and $\langle s, \widetilde{\psi}_{j,k}\rangle$ in the corresponding wavelet domain, numerically. As we will see, the efficient reconstruction algorithm derived from the short support of the B-spline and the corresponding wavelet will play an important role in the numerical computation. Furthermore, since we never use the wavelet decomposition algorithm in the computation, the infinite support of the dual basis does not get into the picture.

Sobolev spaces can be characterized by the means of wavelets; Sobolev norm of a function is equivalent to the ℓ_2 norm of weighted wavelet coefficients of the function (see [16, 27, 38]). In particular, Sobolev spaces $W_2^m(\mathbb{R})$, $m \in \mathbb{N}$, can be characterized by the short supported wavelet. For the $s \in V_J$ defined in (3.8), if the regularity of ψ and the vanishing moment of $\widetilde{\psi}$ are larger than m, it is shown in [27] that there exist two positive constants C_1 and C_2 such that

$$C_1\|s\|^2_{W_2^m(\mathbb{R})} \le \sum_{k\in\mathbb{Z}} |\langle s, \widetilde{\phi}_{J_0,k}\rangle|^2 + \sum_{j=J_0}^{J} \sum_{k\in\mathbb{Z}} 2^{2m(j-J_0)} |\langle s, \widetilde{\psi}_{j,k}\rangle|^2 \le C_2\|s\|^2_{W_2^m(\mathbb{R})}. \tag{3.9}$$

As we shall see, this norm equivalence plays a key role in accelerating the conjugate gradient method in the wavelet domain.

The above discussions are restricted to the univariate setting. Uniform B-splines and short supported wavelets can be extended to the bivariate setting by the tensor product construction.

Since the tensor product construction is standard, we omit here the details and only remark that the bivariate tensor product of uniform B-splines of order p also satisfies the Strang-Fix conditions of order p, and the characterization of Sobolev spaces by bivariate tensor product wavelets, which is similar to (3.9), is still valid.

In addition to tensor product B-splines, box splines are an alternative way of generalizing uniform B-splines to multi-dimensions (see [6]). In particular, we are interested in a bivariate box spline (denoted $M_{2,2,2}$ in the sequel), which is the basis function for the Loop scheme in computer graphics, since it induces compactly supported Riesz wavelet functions (see [25]) and consequently make fast computation in the wavelet domain feasible. Although we have implemented the algorithm to solve (3.3) with ϕ being $M_{2,2,2}$, we do not discuss in the following the implementation details for the box spline based algorithm. We also skip the discussion of the box spline and its corresponding wavelets, and instead refer the interested readers to [25].

3.2 Computation in B-spline domain

In this subsection, we will investigate the computational components of solving the minimization problem (3.3). We first consider the computation in the general setting (i.e. with ϕ, d and m unspecified), and then discuss the specific issues of the computation in B-spline domain.

The approximation space $S^h(\phi, \Omega)$ is spanned by all shifted and dilated ϕ whose support intersect with the interior of Ω. In other words, any function $s \in S^h(\phi, \Omega)$ has the form

$$s = \sum_{j=1}^{M} u_j \phi(\cdot/h - k_j), \quad \{k_1, k_2, \ldots, k_M\} := \{k \in \mathbb{Z}^d : supp\ \phi(\cdot/h - k) \cap \Omega^o \neq \emptyset\}. \tag{3.10}$$

With s represented as in (3.10), the regularization term $|s|^2_{H^m(\Omega)}$ can be written as a quadratic term, as shown in the following lemma.

Lemma 3.1. *Let s be defined in* (3.10). *Denote by $G = [g_{ij}]$ an $M \times M$ matrix with (i,j)-entry*

$$g_{ij} = \big\langle \phi(\cdot/h - k_i),\ \phi(\cdot/h - k_j) \big\rangle_{H^m(\Omega)},$$

where $\langle \cdot,\ \cdot \rangle_{H^m(\Omega)}$ is the semi-inner product associated with $|\cdot|_{H^m(\Omega)}$. Let $\mathbf{u}$ be the column vector with components u_j ($1 \le j \le M$), then one has

$$|s|^2_{H^m(\Omega)} = \mathbf{u}^T G \mathbf{u}.$$

Proof. Since $|s|^2_{H^m(\Omega)} = \langle s,\ s \rangle_{H^m(\Omega)}$, one has

$$\begin{aligned} |s|^2_{H^m(\Omega)} &= \left\langle \sum_{j=1}^{M} u_j \phi(\cdot/h - k_j),\ \sum_{j=1}^{M} u_j \phi(\cdot/h - k_j) \right\rangle_{H^m(\Omega)} \\ &= \sum_{i=1}^{M} \sum_{j=1}^{M} u_i u_j \big\langle \phi(\cdot/h - k_i),\ \phi(\cdot/h - k_j) \big\rangle_{H^m(\Omega)} \\ &= \mathbf{u}^T G \mathbf{u} \end{aligned}$$

□

In addition to G, another matrix involved in (3.3), called the observation matrix A, is obtained by evaluating each basis function at each site in Ξ. If $\Xi = \{x_i\}_{i=1}^n$, then A is defined by $A(i,j) = \phi(\frac{x_i}{h} - k_j)$,$i = 1, \cdots, n$, $j = 1, \cdots, M$. Let $\mathbf{f}$ denote the column vector consisting of the functional

data $\{\widetilde{f}_i\}_{i=1}^n$, then the second term in (3.3) becomes $\|A\mathbf{u}-\mathbf{f}\|^2$. Thus the regularized least square problem (3.3) becomes a standard unconstrained minimization problem

$$\text{minimize} \quad \alpha \mathbf{u}^T G\mathbf{u} + \|A\mathbf{u}-\mathbf{f}\|^2. \tag{3.11}$$

Although A^TA and G can only be guaranteed to be positive semi-definite, we can show that $(A^TA+\alpha G)$, the Hessian of (3.11), is always positive definite provided that the shifts of ϕ are locally linearly independent and that a mild condition on the data sites holds. We say that the data sites $\{x_i\}_{i=1}^n$ are unisolvent for π_{m-1}^d, if there does not exist a nontrivial polynomial $p(x)$ of degree $(m-1)$ such that $p(x_i)=0$ $(i=1,\cdots,n)$. We mention that this unisolvency condition is often employed in radial basis function interpolation to guarantee the uniqueness of the interpolant (see [9]).

Theorem 3.2. *Suppose that the shifts of ϕ are locally linearly independent. If the given scattered sites $\{x_i\}_{i=1}^n$ are unisolvent for π_{m-1}^d, then $(A^TA+\alpha G)$ is positive definite.*

Proof. To prove that $(A^TA+\alpha G)$ is positive definite, we need to show that $\mathbf{u}^T(A^TA+\alpha G)\mathbf{u}=0$ holds only for $\mathbf{u}=\mathbf{0}$. Since both A^TA and G are positive semi-definite, $\mathbf{u}^T(A^TA+\alpha G)\mathbf{u}=0$ implies that $A\mathbf{u}=\mathbf{0}$ and $\mathbf{u}^TG\mathbf{u}=0$. Since

$$|s|^2_{H^m(\Omega)} = \mathbf{u}^TG\mathbf{u},$$

we have $|s|_{H^m(\Omega)}=0$, which implies that $s|_\Omega$ is a polynomial of degree $(m-1)$. On the other hand, by $A\mathbf{u}=\mathbf{0}$, we know that

$$\mathbf{0}=A\mathbf{u}=\big(s(x_1),s(x_2),\cdots,s(x_n)\big)^T.$$

Since there does not exist a non-trivial polynomial of degree $(m-1)$ which vanishes at all data sites, $s|_\Omega$ must be trivial, i.e., $s|_\Omega\equiv 0$. Since s is a linear combination of the basis functions $\phi(\cdot/h-k)$, which are linearly independent over Ω, the coefficient vector $\mathbf{u}$ must be the zero vector.

□

General d, Ω, ϕ and m are assumed in the above. For the purpose of curve/surface fitting, we implement the algorithm in one and two dimensions (i.e. $d=1,2$). Furthermore, our implementation is designed for a special domain Ω, an interval or a rectangle, which is the domain of interest for many applications. Without loss of generality, we assume that Ω is $[0,1]$ or $[0,1]^2$.

In order to compare our scheme to cubic (thin-plate) smoothing spline, we choose ϕ to be cubic uniform B-spline in one dimension and the tensor product of cubic uniform B-spline in two dimensions, and $m=2$ (other choices of ϕ and m can be made as long as the degree of B-spline is larger than m). Under these assumptions, in 1D case, G can be computed in the following way:

$$g_{ij}=\int_0^1 \phi''(\frac{x}{h}-i)\phi''(\frac{x}{h}-j)dx,$$

where ϕ'' denotes the second derivative of ϕ. It is easy to show that G is symmetric band diagonal of width seven (without considering boundary effects, it is a Toeplitz matrix). In the 2D case, G can be obtained similarly; it is band block diagonal of width seven with each block being band diagonal of width seven.

We need to check the conditions in Theorem 3.2 for the positive definiteness of $(A^TA+\alpha G)$. When $d=1,2$, $m=2$ (which concern our implementations), the unisolvency condition on the data sites reduces to the requirement that there exist two distinct data sites, if $d=1$, and three non-collinear data sites, if $d=2$. Note that these are mild conditions which are normally satisfied in practice. The functions ϕ used in our implementations, namely the uniform cubic B-spline when $d=1$, and a tensor product of cubic B-spline when $d=2$, have locally linearly independent shifts (see [6], p. 38).

Assuming that $(A^TA + \alpha G)$ is positive definite, finding the solution of (3.11) is equivalent to solving the following linear system

$$(A^TA + \alpha G)\mathbf{u} = A^T\mathbf{f}, \tag{3.12}$$

where G is a banded matrix (or block banded in 2D) of width seven. Moreover, since the support of ϕ is compact, A^TA is also a banded matrix (or block banded in 2D) of width seven, and therefore, the matrix $(A^TA + \alpha G)$ is banded. In 1D problems, as the size of the linear system is usually small, the direct methods, e.g., LU factorization or Cholesky factorization, will be efficient enough to solve (3.12). However, for most 2D problems, if a fine resolution (i.e. small h) is required for good approximation, the size of the linear system could be very large so that the direct solvers become inefficient and even impractical due to storage constraints. In that case we have to resort to iterative solvers. Since $(A^TA + \alpha G)$ is sparse, symmetric and positive definite, the conjugate gradient method is a method of choice.

3.3 Computation in wavelet domain

Assume that we have a reconstruction formula $\mathbf{u} = R\mathbf{c}$, where R is derived from the wavelet reconstruction algorithm and $\mathbf{c}$ is the coefficient vector in terms of a wavelet basis. Then replacing $\mathbf{u}$ with $R\mathbf{c}$ in (3.11) leads to

$$R^T(A^TA + \alpha G)R\mathbf{c} = R^TA^T\mathbf{f}, \tag{3.13}$$

where the left multiplication by R^T is employed to make the resulting system symmetric and positive definite. Although the two linear systems, (3.12) and (3.13), are equivalent, solving in the wavelet domain brings several benefits. First, the wavelet-based solution is sparse, i.e, most of the entries in $\mathbf{c}$ are very small and can be set to zero without losing accuracy. As demonstrated in our numerical experiments, this sparseness property can be used to adaptively choose wavelet basis functions to improve the quality of fitting results especially in curve fitting. Second, although the linear systems (3.12) and (3.13) produce the same fitting function, the latter can be solved more efficiently. Generally speaking, the conjugate gradient method is one of the most efficient iterative methods for symmetric and positive definite linear systems. However, for the linear system (3.12), its convergence is quite slow because of ill-conditioning, and consequently a large number of iterations are required for convergence. However, if a proper normalization factor of the wavelet basis is encoded in the reconstruction matrix R, (3.13) can be solved efficiently by the conjugate gradient method. We will quantify the speed improvement in the last section.

Next we discuss how to construct the matrix R based on the reconstruction algorithm. Though our implementation is based on cubic uniform B-spline ($m = 2$), we assume general uniform B-spline and general m in the following discussion. Here we restrict ourselves to the univariate case; the bivariate case can be discussed similarly. Since wavelets are defined at dyadic scales, we assume that $h = 1/2^J$ for some $J \in \mathbb{N}$. Further, we assume that $\Omega = [0, 1]$ and ϕ is a uniform B-spline. Then each function in $S^h(\phi, [0, 1])$ can be expanded in the basis

$$\Phi_J := \{\phi_{J,k} : supp\,\phi_{J,k} \cap (0, 1) \neq \emptyset, k \in \mathbb{Z}\},$$

and let the basis Φ_{J-1} be defined similarly. In order to obtain a one-level reconstruction matrix, we need to find a set of wavelet basis functions on the level $(J - 1)$, denoted as Ψ_{J-1}, which is adjoined to Φ_{J-1} to produce a basis which is equivalent to Φ_J. A simple choice

$$\Psi_{J-1} = \{\psi_{J-1,k} : supp\,\psi_{J-1,k} \cap (0, 1) \neq \emptyset, k \in \mathbb{Z}\}$$

does not work directly since generally $\#\Phi_J < \#\Phi_{J-1} + \#\Psi_{J-1}$, where $\#$ denotes the cardinality of a set. To have $\#\Phi_J = \#\Phi_{J-1} + \#\Psi_{J-1}$, we can modify Ψ_{J-1} by discarding a number of basis functions whose support intersect with boundaries. Once the set Ψ_{J-1} is determined, by

using the two-scale relations (3.4) and (3.5), a $\phi_{J-1,k} \in \Phi_{J-1}$ or $\psi_{J-1,k} \in \Psi_{J-1}$ can be written in terms of $\phi_{J,k}$; it should be noted that when applying the two-scale relations, if $\phi_{J,k} \notin \Phi_J$, set the corresponding coefficient to be zero since it has no contribution to the data fitting term and the regularization term. In matrix notation, there exist matrices G_J and H_J such that

$$\Phi_{J-1} = \Phi_J G_J, \ \Psi_{J-1} = \Phi_J H_J,$$

where we use the same notations $\Phi_J, \Phi_{J-1}, \Psi_{J-1}$ to denote the row vectors of the basis functions in the corresponding sets. Thus we have the one-level reconstruction formula

$$\begin{bmatrix} \Phi_{J-1} & \Psi_{J-1} \end{bmatrix} = \Phi_J R_J,$$

where the one-level reconstruction matrix, R_J, is defined by

$$R_J = \begin{bmatrix} G_J & H_J \end{bmatrix}.$$

We note that R_J is nonsingular, since $\#\Phi_J = \#\Phi_{J-1} + \#\Psi_{J-1}$ and since both Φ_J and $\Phi_{J-1} \cup \Psi_{J-1}$ are bases. This reconstruction can be applied in a similar fashion to each level $j = J_0, \cdots, J-2$, where J_0 denotes the coarsest level. Ultimately, we arrive at the following reconstruction formula:

$$\begin{bmatrix} \Phi_j & \Psi_j \end{bmatrix} = \Phi_{j+1} R_{j+1}, \quad j = J_0, \cdots, J-1.$$

We remark that the matrices R_{j+1}, $j = J_0, \cdots, J-1$, are very sparse, due to the short support of the refinement and wavelet masks for ϕ and ψ.

The Riesz property (3.6) shows that the basis functions $\psi_{j,k}$ are normalized such that the L_2 norm of a function is equivalent to the ℓ_2 norm of the corresponding wavelet coefficients $c_{j,k}$. For the purpose of solving the linear system (3.12), this normalization is undesirable since it leads to an ill-conditioned linear system (probably because the regularization quantity $u^T G u$ represents the Sobolev seminorm of s rather than its L_2-norm). In order to avoid this ill-conditioning in (3.12), we renormalize the wavelet basis functions so that the Sobolev norm is equivalent to the ℓ_2 norm of the corresponding wavelet coefficients. In view of the norm equivalence for Sobolev spaces, given by (3.9), this goal can be achieved by the following renormalization scheme

$$\phi_{j,k}^{-m} = 2^{-mj}\phi_{j,k}, \ \psi_{j,k}^{-m} = 2^{-mj}\psi_{j,k}, \ \widetilde{\phi}_{j,k}^{-m} = 2^{mj}\widetilde{\phi}_{j,k}, \ \widetilde{\psi}_{j,k}^{-m} = 2^{mj}\widetilde{\psi}_{j,k}.$$

Under this new normalization, Φ_j and Ψ_j will be replaced by Φ_j^{-m} and Ψ_j^{-m}, and the reconstruction formula on each level becomes

$$\begin{bmatrix} \Phi_j^{-m} & \Psi_j^{-m} \end{bmatrix} = \Phi_{j+1}^{-m} 2^m R_{j+1}, \quad j = J_0, \cdots, J-1. \tag{3.14}$$

This renormalization scheme is motivated by the Sobolev norm equivalency, and it also can be viewed as a diagonal preconditioning technique, see [13, 30].

Combining the equations in (3.14) yields

$$\begin{bmatrix} \Phi_{J_0}^{-m} & \Psi_{J_0}^{-m} & \cdots & \Psi_{J-1}^{-m} \end{bmatrix} = \Phi_J^{-m} 2^m R_J \begin{bmatrix} 2^m R_{J-1} & 0 \\ 0 & I \end{bmatrix} \cdots \begin{bmatrix} 2^m R_{J_0+1} & 0 \\ 0 & I \end{bmatrix},$$

where I denotes an identity matrix whose (varying) size is determined by the requirements of matrix multiplication. Considering that each basis function in Φ_J^{-m} has normalization factor $2^{(1/2-m)J}$, but the basis functions in (3.10) have no such factor, we define

$$R = 2^{(1/2-m)J} 2^m R_J \begin{bmatrix} 2^m R_{J-1} & 0 \\ 0 & I \end{bmatrix} \cdots \begin{bmatrix} 2^m R_{J_0+1} & 0 \\ 0 & I \end{bmatrix},$$

and subsequently obtain

$$\begin{bmatrix} \Phi_{J_0}^{-m} & \Psi_{J_0}^{-m} & \cdots & \Psi_{J-1}^{-m} \end{bmatrix} = 2^{(m-1/2)J} \Phi_J^{-m} R. \tag{3.15}$$

Note that $2^{(m-1/2)J}\Phi_J^{-m}$ is the basis, employed in (3.10), with which the B-spline coefficient vector $\mathbf{u}$ is associated. If $\mathbf{c}$ is the coefficient vector in terms of the basis $\begin{bmatrix} \Phi_{J_0}^{-m} & \Psi_{J_0}^{-m} & \cdots & \Psi_{J-1}^{-m} \end{bmatrix}$, then equation (3.15) gives the reconstruction formula $\mathbf{u} = R\mathbf{c}$.

4 Numerical experiments

The numerical experiments consist of two parts. In the first part we examine the effectiveness of our method in the context of curve/surface fitting by comparing, on certain test problems, the results of our method with those of the classical smoothing spline. In the second part, we demonstrate the speed improvement obtained by solving the linear system using the wavelet basis.

4.1 Curve and surface fitting

Curve and surface fitting has a wide range of applications in science and engineering. Among various methods for fitting noisy data, the smoothing spline is classical and is a benchmark for comparison with other methods (see [24]). Our numerical experiments intend to compare our method with the smoothing spline method in the context of curve/surface fitting. For the sake of simplicity, in the following presentation, CSSPL stands for cubic smoothing spline (for curve fitting), TPSS stands for thin-plate smoothing spline (for surface fitting), and WAVE stands for our method that gives a solution in the wavelet domain.

We will apply WAVE and CSSPL/TPSS to several synthetic data sets, and then compare their performance in terms of visual quality and numerical errors. In the experiments, a synthetic noisy data set $\{(x_i, f_i) : i = 1, \cdots, n\}$ is generated by adding Gaussian noise to a function f, i.e.,

$$f_i = f(x_i) + \epsilon_i, \ \ i = 1, 2, \cdots, n, \tag{4.1}$$

where x_i's are random data sites over $[0, 1]$ or $[0, 1]^2$, and ϵ_i's are additive Gaussian noise drawn from the normal distribution $N(0, \sigma^2)$. While we assume x_i's are uniformly random in some examples, in others, we allow the data sites to be drawn from some spatially variant distribution and even with large holes, as we want to see how our method performs when given data are rather "scattered". Once such a data set is generated, WAVE and CSSPL/TPSS are applied to obtain the fitting curve/surface. In order to make the results comparable, both methods use the same criterion, generalized cross validation (GCV, see [44]), to choose the smoothing parameter α.

Since the solution given by WAVE expresses the fitting curve/surface in a wavelet basis, it has a sparse representation, in the sense that most of the wavelet coefficients are very small and can be discarded without losing accuracy (to have this property, the wavelet coefficients obtained from solving (3.13) should be renormalized back as they are in the L_2 norm). If we wish to improve the fitting result, a wavelet thresholding technique can be applied to select the most significant basis functions, i.e., those with the largest coefficients. Collecting together the selected basis functions leads to a new (sub) basis that has been adapted to the given data. Once the adapted basis has been constructed, the original data set can be fitted again using this new basis. This new fitting method, WAVE followed by thresholding and refitting, is referred to as TWAVE in the following discussion. We will see that TWAVE can use a much smaller number of basis functions to achieve equivalent, sometimes better, fitting results, especially in curve fitting.

Before we describe the experiments in detail in the next subsection, we provide highlights of the main experimental results and discuss some features of WAVE in comparison with TPSS. Since $S^h(\phi, \Omega)$ is a subspace of the Beppo-Levi space H^m, the solution of (3.3) can be viewed as an approximation to the solution of (1.1) — the smoothing spline. Consequently, we expect that WAVE and CSSPL/TPSS will lead to similar results in terms of visual quality and accuracy of fitting, and this expectation is confirmed in the next subsection.

A prominent difference between WAVE and TPSS (or other RBF methods) is that they employ different strategies to place the centers of basis functions. TPSS associates with each data site a radially symmetric basis function centered at the site, thus leading to a linear system of size $n \times n$. In contrast, WAVE employs a basis that is spanned by the h-dilates and h-shifts of a uniform B-spline, and hence the size of the corresponding linear system depends only on Ω and the dilation parameter

h. For the purpose of fitting large data sets, though some fast RBF methods are now available (see [2, 10]), WAVE offers an alternative solution by providing to the user a flexibility, via choosing the parameter h, to control the size of the problem such that it can be solved efficiently. Moreover, due to its efficiency, WAVE allows for the choice of small h to guarantee good approximation. We emphasize that its efficiency is attributed to two important components in WAVE: compact support of the uniform B-splines resulting in a sparse linear system, and a properly normalized wavelet basis which accelerates the convergence of the conjugate gradient solution to the linear system. We will quantify the speed improvement in the last subsection.

4.2 Numerical experiments

The experiments in this subsection are conducted in the following order: curve fitting, surface fitting and TWAVE method. The curve fitting experiment employs the following two test functions:

$$\begin{aligned} f_1(x) &= 4.26(e^{-\beta x} - 4e^{-2\beta x} + 3e^{-3\beta x}), \quad \text{with } \beta = 3.25; \\ f_2(x) &= \begin{cases} sin(4\pi x), & x < 1/2, \\ sin(16\pi x), & 1/2 \le x \le 1. \end{cases} \end{aligned}$$

The noisy data are generated according to equation (4.1) with the noise level $\sigma = 0.05, 0.1$. We first give an example to illustrate that WAVE and CSSPL generally lead to visually indistinguishable fitting results, and then demonstrate by Monte Carlo experiments that the two methods achieve the same accuracy.

Example 4.1. *We apply CSSPL and WAVE to a noisy data set, generated from f_2, with $n = 150$, $\sigma = 0.1$ and uniformly distributed data sites. The noisy sample is shown in the upper-left subplot of Figure 1. The figure shows the fitting curves obtained by CSSPL and WAVE ($h = 1/2^8$). It also shows the result by TWAVE, which we will elaborate upon later. On each subplot, the number of coefficients used to describe the curve is displayed in parenthesis in the subtitle.*

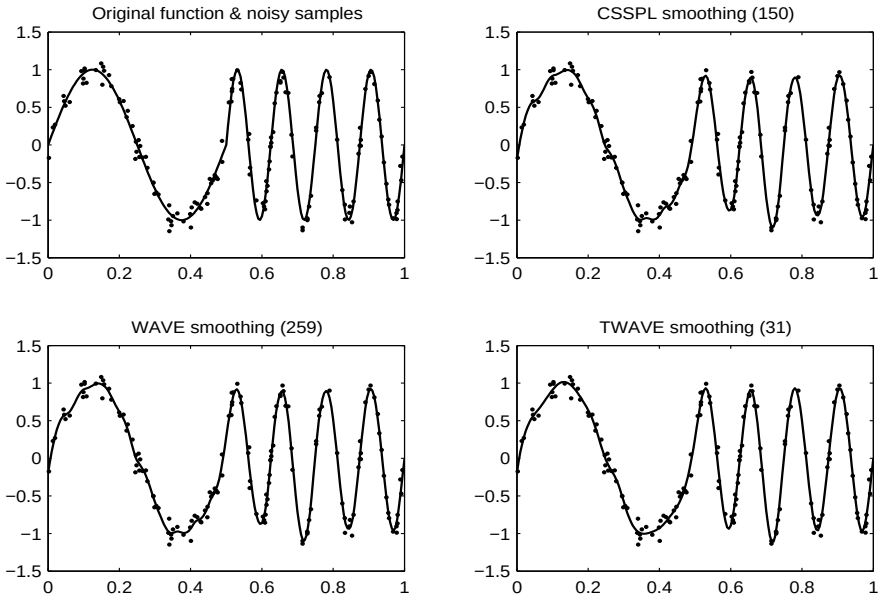

Figure 1: Comparison of CSSPL, WAVE and TWAVE

It can be seen from the figure that CSSPL and WAVE produce very similar fitting curves. This is not surprising, because WAVE can be regarded as an approximation of CSSPL and both methods use the same criteria (i.e. GCV) to determine the smoothing parameter.

We carry out Monte Carlo experiments on $M = 100$ noisy data sets, each consisting of $n = 300$ samples with uniformly distributed data sites. For each data set, let $\widetilde{g}$ be the estimate obtained by one of the methods, then the signal-to-noise ratio (SNR) is computed according to the following definition

$$SNR(DB) = 10\log_{10}\frac{P_s}{P_N} = 10\log_{10}\frac{\sum g_i^2}{\sum(\widetilde{g}_i - g_i)^2},$$

where g_i and $\widetilde{g}_i$ denote respectively the amplitude of the original signal and the estimated signal at site t_i, where $t_i = \frac{i}{m}$ $(i = 0, 1, \cdots, m)$ is a uniform grid in $[0, 1]$ (in the experiment, we take $m = 200$). The experiment aims to compare the average SNR and its standard deviation from M trials. The result is shown in Table 1. It tells that the two methods have very similar performance in terms of numerical error.

		CSSPL		WAVE	
		mean	std	mean	std
f_1	$\sigma = 0.05$	28.50	1.66	28.50	1.66
	$\sigma = 0.1$	24.05	1.51	24.05	1.51
f_2	$\sigma = 0.05$	29.23	2.52	29.23	2.52
	$\sigma = 0.1$	24.66	1.23	24.73	1.18

Table 1: SNR for f_1, f_2: mean and standard deviation.

In the above numerical tests, the data site x_i's are assumed to be uniformly random on $[0, 1]$. Next we give an example to show that these methods are also applicable to the situation when data sites have spatially variant density.

Example 4.2. *A noisy data set is generated from f_1, with $n = 100$ and $\sigma = 0.1$. The distribution of data sites on $[0, 1]$ follows a non-uniform probability density $p(x) = (e - e^x)$; the data density is decreasing from left to right. Figure 2 shows the results obtained by CSSPL, WAVE and TWAVE.*

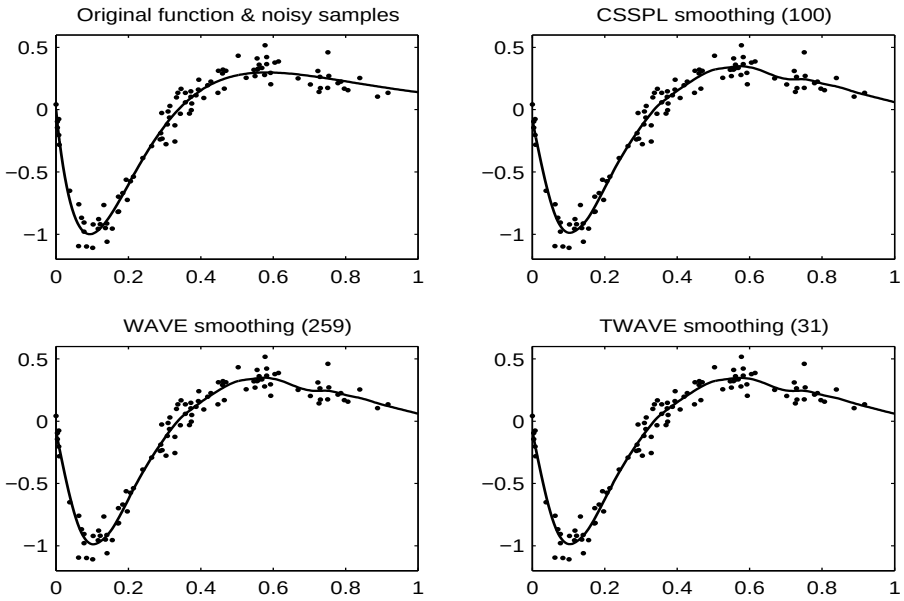

Figure 2: CSSPL, WAVE and TWAVE for the data with non-uniform sites

Next we turn to the experiment on surface fitting problems. Our implementation of TPSS is based on a routine *tpaps* from Spline Toolbox in MATLAB, and we incorporate a GCV procedure

into this routine to determine the smoothing parameter. The surface fitting experiment employs the following test functions

$$
\begin{aligned}
f_3(x,y) &= (-20.25(x-0.5)^2+(y-0.5)^2)/3;\\
f_4(x,y) &= \frac{1.25+\cos(5.4y)}{6(1+(3x-1)^2)};\\
f_5(x,y) &= 0.75\exp\left(-\frac{(9x-2)^2+(9y-2)^2}{4}\right)+0.75\exp\left(-\frac{(9x+1)^2}{49}-\frac{9y+1}{10}\right)\\
&\quad +0.5\exp\left(-\frac{(9x-7)^2+(9y-3)^2}{4}\right)-0.2\exp\left(-(9x-4)^2-(9y-7)^2\right).
\end{aligned}
$$

These test functions, which were used in [23] to test interpolation algorithms, are smooth and present enough shape variations to reveal the quality of a fitting scheme. The standard deviation σ of the Gaussian noise is chosen such that the signal-to-noise ratio (SNR) of the noisy samples is about 20DB, which means that $\sigma = 0.01, 0.015, 0.05$ for f_3, f_4, f_5 respectively.

Example 4.3. *Figure 3 illustrates the fitting results when WAVE and TPSS are applied to a noisy data set, generated from f_4, with $n = 400$, $\sigma = 0.015$ and uniformly distributed data sites. An interpolation of the noisy sample, obtained by using a MTALAB routine, griddata, is plotted on the upper-right subplot. On the lower-left is the fitting surface obtained by WAVE, and on the lower-right is the one produced by TPSS.*

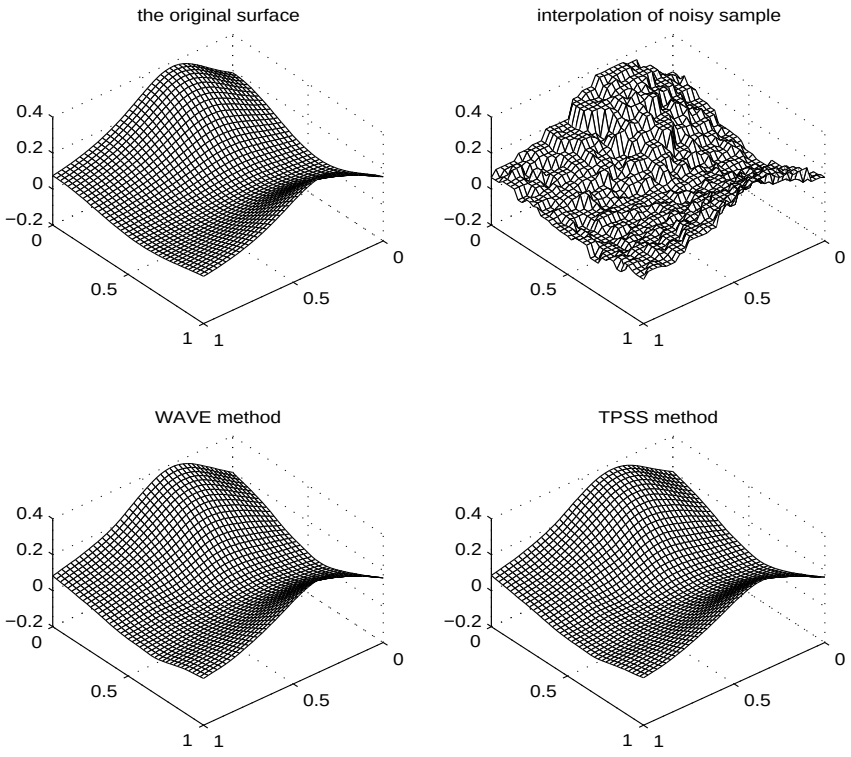

Figure 3: Comparison of WAVE and TPSS

It is obvious that interpolation leads to undesirable solutions since the noise is not reduced. Both regularization approaches, WAVE and TPSS, produce visually pleasing surfaces which are quite close to the original surface. Since both methods use GCV to determine the smoothing parameter, we expect them (and this is confirmed in the experiments) to yield very similar fitting surfaces.

However, minute inspection reveals that the two fitting surfaces differ more near the boundary as compared with the deep interior of the domain $[0,1]^2$. This could be due to the smoothness measure of thin-plate spline being over $\mathbb{R}^2$ while WAVE restricts the smoothness penalty on the unit square.

We next carry out Monte Carlo experiments to compare the SNR attained by WAVE and TPSS. For each test function, $M = 50$ noisy data sets (with $n = 400$) are generated according to (4.1) with uniformly distributed data sites. In WAVE, we choose the scale parameter $h = 1/16$ so that the dimension of the resulting linear system (361) is close to the dimension of the linear system produced by TPSS (400). As in the 1D case, the average SNR and its standard deviation are calculated to obtain Table 2. It is evident from the table that WAVE and TPSS attain very similar SNR. This is consistent with our inspection on the visual appearance of the fitting surfaces produced by the two methods.

		WAVE		TPSS	
		mean	std	mean	std
f_3	$\sigma = 0.01$	26.59	0.72	27.15	0.79
f_4	$\sigma = 0.015$	29.34	0.83	29.14	0.88
f_5	$\sigma = 0.05$	27.70	0.79	28.29	0.76

Table 2: SNR for f_3, f_4, f_5: mean and standard deviation.

The 2D data sites in the above are assumed to be uniformly distributed on the unit square. It is known that TPSS is a good method to smoothly fill large holes where data are missing. We illustrate in the next example that WAVE is also able to do this.

Example 4.4. *Figure 4 illustrates the fitting results when WAVE and TPSS are applied to a noisy data set, generated from f_5, with size $n = 400$ and noise level $\sigma = 0.05$. The interpolation shows a large hole where data are missing. Outside the hole the data are uniformly distributed.*

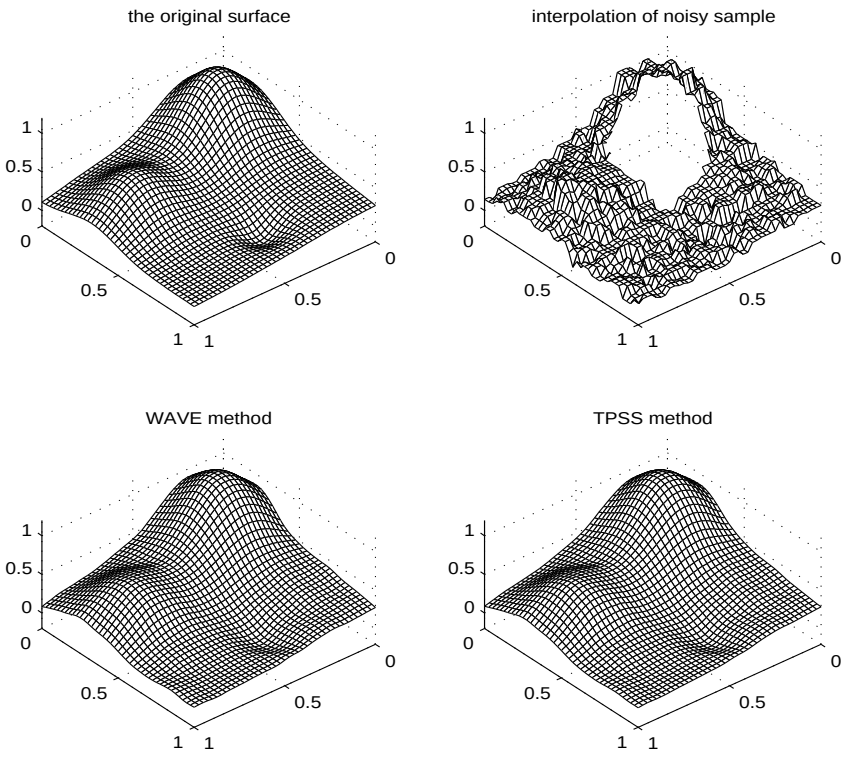

Figure 4: WAVE and TPSS for repairing hole

Both methods, WAVE and TPSS, can fill the hole smoothly. It seems quite natural that TPSS is able to fill holes, since it uses globally supported basis functions. However, why is WAVE, which uses locally supported basis functions, also able to repair large holes. The reason is that whether the chosen basis functions are globally or locally supported, regularization enforces the smoothness of surface and thus holes can be filled smoothly.

In the above experiments on surface fitting, we used a 2D tensor product of uniform cubic B-splines (or its corresponding wavelets) as the basis. We also carried out the numerical experiment with the box spline M_{222} or its corresponding wavelets as the basis, and found that it achieves similar results.

From the above experimental results, we conclude that WAVE achieves the same accuracy and visual quality as the classical CSSPL/TPSS in both curve and surface fitting. Next we turn to examine TWAVE method. It is shown in Figure 1 that TWAVE attained the fitting curve on the lower-right, which is achieved by using the most significant 5% wavelet basis functions. The result is smoother than the ones attained by CSSPL and WAVE; it also achieves smaller error than CSSPL and WAVE. It means that TWAVE uses a smaller number of basis functions (31, compared to 150 in CSSPL and 259 in WAVE) to get a better fitting curve. Here we provide one more example to illustrate the effectiveness of TWAVE. The following titanium heat data set is taken from the Spline Toolbox in MATLAB (see [4]), and it has been used to test spline approximation algorithms ([3, 17]).

Example 4.5. *Let f be defined discretely by titanium heat data, which consist of 49 data points. A noisy sample is produced by adding Gaussian noise with $\sigma = 0.04$ to the value on each data site. By using CSSPL, WAVE and TWAVE, the fitting curves are generated and shown in Figure 5.*

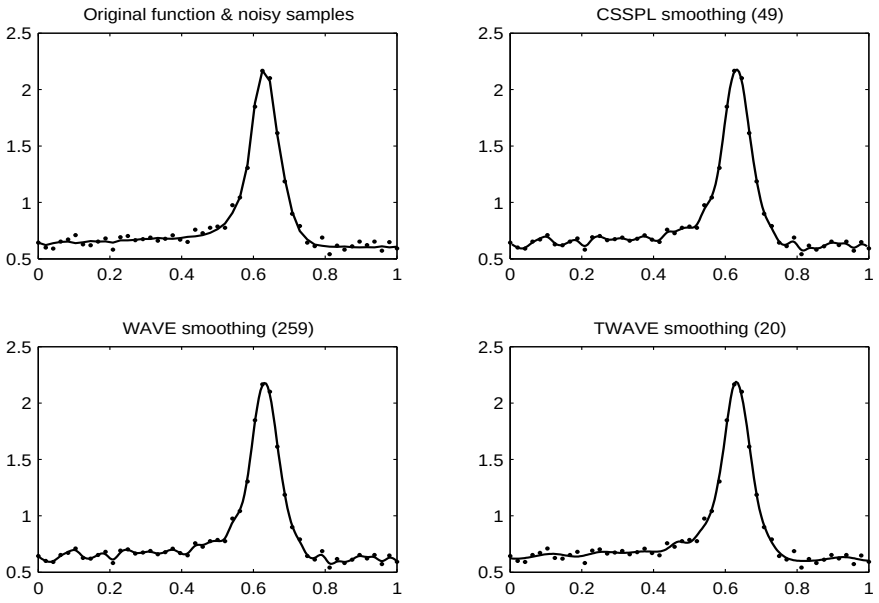

Figure 5: Comparison of CSSPL,WAVE and TWAVE

In this example, WAVE and CSSPL perform equivalently well and both lead to a fitting curve with small oscillations, while TWAVE leads to a smoother curve. It cannot be attributed to a larger amount of smoothing in TWAVE, since both WAVE and TWAVE use the same smoothing parameter. By adaptively choosing the basis functions, via wavelet thresholding, the placement of the chosen basis functions is adapted to the local variation of the function, i.e., the adapted basis gives more attention to highly-varying regions, less attention to flat areas. This results in TWAVE being less sensitive to small local variations, and tends to ignore them and produces more visually

Number of iterations

	N=32	N=64	N=128	N=256
BCG	102	231	915	3052
WCG	38	19	21	17

Computation time

	N=32	N=64	N=128	N=256
BCG	0.27	1.61	30.34	380.95
WCG	0.22	0.53	5.80	17.44

Table 3: Comparison of number of iterations and computation time (seconds)

pleasing curves. In other words, TWAVE is more robust to the outliers introduced by noise. The idea of adaptively choosing basis functions employed in TWAVE is in a similar spirit to the adaptive knot placing strategy in spline approximation (see, [17, 28]).

4.3 Speed improvement in wavelet domain

In this subsection, we intend to quantify the speed improvement obtained, in solving the linear system (3.12) with the conjugate gradient method, when the wavelet domain is used instead of the B-spline domain.

Broadly speaking, the technique of employing the wavelet domain, in place of the B-spline domain, belongs to a large class of multilevel preconditioning methods (see [8, 13, 45]) which have been applied successfully in solving numerical differential equations. Some multilevel methods are wavelet based (see e.g. [13, 30]) in which the norm equivalency is a key element in the design of the preconditioner. In fact, it has been proved (e.g. [13, 30]) that if an appropriate diagonal preconditioner is applied to the linear system that is obtained by discretizing an elliptic partial differential equation in a certain wavelet basis, then the condition number can be bounded by a constant which is independent of the size of the discretization mesh. In our current setting, such diagonal preconditioning has been implicitly encoded into the reconstruction matrix R by the renormalization process.

To simplify the following presentation, let BCG and WCG denote the conjugate gradient method for (3.12) and (3.13) respectively. The numerical experiment compares the efficiency of BCG and WCG, in terms of iteration numbers and computation time, when they are applied to a surface fitting example (the test function f_5 with $\sigma = 0.05$). The two methods are used to solve the problem at different scales $h = 1/32, 1/64, 1/128, 1/256$ (i.e. $J = 5, 6, 7, 8$). In the experiment, the starting level $J_0 = 3$ (for the wavelet method) and the smoothing parameter $\alpha = 10^{-2}$ are fixed at all scales. The convergence curves in Figure 6 illustrate the decrease of the objective function against the number of iterations at different scales. It is clear that WCG dramatically reduces the number of iterations required to reach the minimum value of the objective function, especially when the resolution is fine. In fact, the convergence rate of WCG is independent of the resolution, while BCG converges slower as the resolution becomes finer. This convergence behavior can be further confirmed in Table 3 which displays the number of iterations and computation time required to achieve the convergence criterion $\|x - x_\infty\|/\|x_\infty\| \leq 10^{-3}$, where x_∞ denotes the "exact" solution obtained by using 4000 conjugate gradient iterations of BCG. However, this is not the case in the wavelet domain, as evidenced by the constant number of required iterations. It is also clear from Table 3 that WCG is much more efficient than BCG in terms of computation time, even though the first uses more time than the latter per iteration. This efficiency is, of course, due to the significantly smaller number of iterations needed in WCG.

It is well known, for a symmetric positive definite linear system $A\mathbf{x} = \mathbf{b}$, that the number

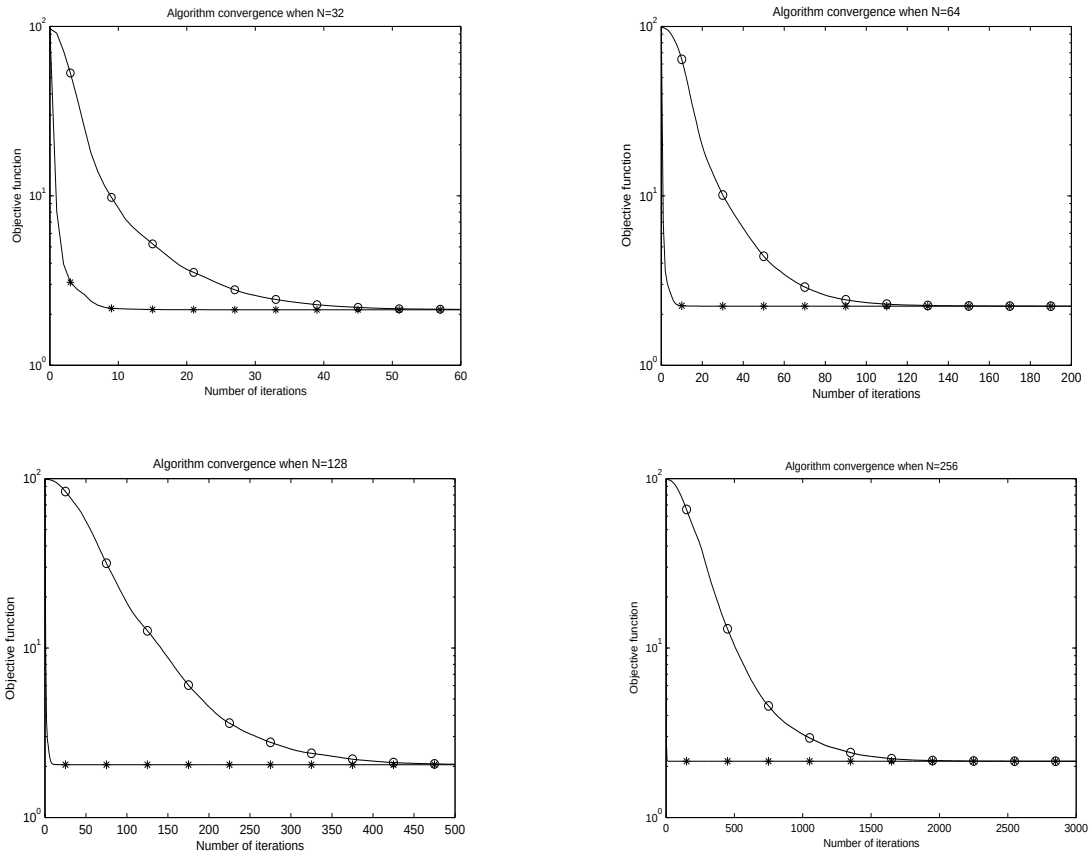

Figure 6: The decrease of objective function value in terms of the iteration number: the curve $-\circ-$ describes the convergence of BCG, while the curve $-*-$ describes the convergence of WCG.

of conjugate gradient iterations required to bring the error to a specified tolerance, in the worst case, is proportional to $\sqrt{k(A)}$ (e.g. [15]), where $k(A)$ is the condition number of A. It can therefore be inferred, from the table, that the conditioning of the linear system in the B-spline domain deteriorates as the resolution becomes finer. In the wavelet domain, however, we see in the experiment that the required number of iterations remains constant, which suggests that the linear system in the wavelet domain (3.13) may have a condition number which is bounded independent of scale. This phenomenon is worthy of further research.

References

[1] M. Arigovindan, M. Suhling, P. Hunziker, and M. Unser. Variational image reconstruction from arbitrarily spaced samples: a fast multiresolution spline solution. *IEEE Trans. Image Processing*, 14(4):450–460, 2005.

[2] 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(5):1717–1740, 2000.

[3] C.de Boor. *A practical guide to splines.* New York, Springer-verlag, 2001.

[4] C.de Boor. *Spline toolbox user's guide.* Mathworks Inc., 2004.

[5] C.de Boor, R.A. Devore, and A. Ron. Approximation from shift-invariant subspaces of $L_2(\mathbb{R}^d)$. *Trans. Amer. Math. Soc.*, 341(2):787–806, 1994.

[6] C.de Boor, K. Hollig, and S. Riemenschneider. *Box splines*. New York, Springer-verlag, 1993.

[7] C.de Boor and A. Ron. Fourier analysis of the approximation power of prinicpal shift-inavriant spaces. *Constr. Approx.*, 8(4):427–462, 1992.

[8] J.H. Bramble, J.E. Pasciak, and J. Xu. Parallel multilevel preconditioners. *Math. Comp.*, 55(191):1–22, 1990.

[9] M.D. Buhmann. *Radial basis functions: theory and implementations*. Cambridge university press, 2003.

[10] J.C. Carr, R.K. Beatson, J.B. Cherrie, T.J. Mitchell, W.R. Fright, B.C. McCallum, and T.R. Evans. Reconstruction and representation of 3D objects with radial basis functions. In *Computer Graphics (SIGGRAPH 2001 Proceedings)*, pages 67–76, 2001.

[11] A. Cohen, I. Daubechies, and J.C. Feauveau. Biorthogonal bases of compactly supported wavelets. *Comm. Pure Appl. Math.*, 45(5):485–560, 1992.

[12] D. Castaño and A. Kunoth. Multilevel regularization of wavelet based fitting of scattered data — some experiments. *Numer. Algor.*, 39(1-3):81–96, 2005.

[13] W. Dahmen and A. Kunoth. Multilevel preconditioning. *Numer. Math.*, 63:315–344, 1992.

[14] I. Daubechies. Orthonormal bases of compactly supported wavelets. *Comm. Pure Appl. Math.*, 41(7):909–996, 1988.

[15] J.W. Demmel. *Applied numerical linear algebra*. SIAM, Philadelphia, 1997.

[16] R.A. Devore. Nonlinear approximation. *Acta Numer.*, 7:51–150, 1998.

[17] P. Dierckx. *Curve and surface fitting with splines*. Oxford University Press, 1993.

[18] J. Duchon. Sur l'erreur d'interpolation des fonctions de plusieur variables par les D^m-splines. *RAIRO Anal. Numer.*, 12(4):325–334, 1978.

[19] N. Dyn, M. S. Floater, and A. Iske. Adaptive thinning for bivariate scattered data. *J. Comput. Appl. Math.*, 145(2):505–517, 2002.

[20] N. Dyn, D. Levin, and S. Rippa. Numerical procedures for surface fitting of scattered data by radial functions. *SIAM J. Sci. Statist. Comput.*, (2):639–659, 1986.

[21] T. Evgeniou, M. Pontil, and T. Poggio. Regularization networks and support vector machines. *Adv. Comput. Math.*, (1):1–50, 2000.

[22] R. Franke. Scattered data interpolation: tests of some methods. *Math. Comp.*, 38(157):181–200, 1982.

[23] R. Franke and G.M. Nielson. Scattered data interpolation and applications: a tutorial and survey. In *Gometric modelling*, pages 131–160. Springer, Berlin, 1991.

[24] P.J. Green and B.W. Silverman. *Nonparametric regression and generalized linear models: a roughness penalty approach*. Chapman and Hall, 1994.

[25] B. Han and Z.W. Shen. Wavelets from the loop scheme. *J. Fourier Anal. Appl.*, 11(6):615–637, 2005.

[26] B. Han and Z.W. Shen. Wavelets with short support. *SIAM J. Math. Anal.*, 38(2):530–556, 2006.

[27] B. Han and Z.W. Shen. Dual wavelet frames and Riesz bases in Sobolev spaces. *Constructive Approximation*, to appear.

[28] X.M. He, L.X. Shen, and Z.W. Shen. A data-adaptive knot selection scheme for fitting splines. *IEEE Signal Processing Letters*, 8(5):137–139, 2001.

[29] H. Inoue. A least-squares smooth fitting for irregularly spaced data: finite-element approach using the cubic B-spline basis. *Geophysics*, 51(11):2051–2066, 1986.

[30] S. Jaffard. Wavelet methods for fast resolution of elliptic problems. *SIAM J. Numer. Anal.*, 29(4):965–986, 1992.

[31] R.Q. Jia. The toeplitz theorem and its applications to approximation theory and linear PDE's. *Trans. Amer. Math. Soc.*, 347(7):2585–2594, 1995.

[32] R.Q. Jia and J. Lei. Approximation by multiinteger translates of functions having global support. *J. Approx. Theory*, 72(1):2–23, 1993.

[33] R.Q. Jia and Z.W. Shen. Multiresolution and wavelets. *Proc. Edinburgh Math. Soc.*, 37(2):271–300, 1994.

[34] M.J. Johnson. Scattered data interpolation from principal shift-invariant spaces. *J. Approx. Theory*, 113(2):172–188, 2001.

[35] S. Lee, G. Wolberg, and S. Shin. Scattered data interpolation with multilevel B-splines. *IEEE Trans. Vis. Comput. Graph.*, 3(3):228–244, 1997.

[36] S. Mallat. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Trans. Pattern Anal. Machine Intell.*, 11(7):674–693, 1989.

[37] J. Meinguet. Mulitvariate interpolation at arbitary points made simple. *Z. Angew. Math. Phys.*, 30:292–304, 1979.

[38] Y. Meyer. *Wavelets and operators*. Cambridge Univ. Press, 1992.

[39] F.J. Norcowich, J.D. Ward, and H. Wendland. Sobolev bounds on functions with scattered zeros with applications to radial basis function surface fitting. *Math. Comp.*, 74(250):743–763, 2005.

[40] R. Sibson and G. Stone. Computation of thin-plate splines. *SIAM J. Sci. Statist. Comput.*, 12(6):1304–1313, 1991.

[41] R. Szeliski. Fast surface interpolation using hierarchical basis functions. *IEEE Trans. Pattern Anal. Machine Intell.*, 12(6):513–528, 1990.

[42] D. Terzopoulos. Regularization of inverse visual problems involving discontinuities. *IEEE Trans. Pattern Anal. Machine Intell.*, 8(4):413–424, 1986.

[43] C. Vazquez, E. Dubois, and J. Konrad. Reconstruction of nonuniformly sampled images in spline spaces. *IEEE. Trans. Image Processing*, 14(6):713–725, 2005.

[44] G. Wahba. *Spline models for observational data*. SIAM, 1990.

[45] H. Yserentant. On the multi-level splitting of finite element spaces. *Numer. Math.*, 49:379–412, 1986.