Local Hybrid Approximation for Scattered Data Fitting with Bivariate Splines

Oleg Davydov
Department of Mathematics, University of Strathclyde,
26 Richmond Street, Glasgow G1 1XH, Scotland, UK

Rossana Morandi
Dipartimento di Energetica, Università di Firenze,
Via Lombroso 6/17, 50134 Firenze, ITALY

Alessandra Sestini
Dipartimento di Energetica, Università di Firenze,
Via Lombroso 6/17, 50134 Firenze, ITALY

Abstract

We suggest a local hybrid approximation scheme based on polynomials and radial basis functions, and use it to improve the scattered data fitting algorithm of [7]. Similar to that algorithm, the new method has linear computational complexity and is therefore suitable for large real world data. Numerical examples suggest that it can produce high quality artifact–free approximations that are more accurate than those given by the original method where pure polynomial local approximations are used.

Keywords: Scattered Data Fitting, Bivariate Splines, Radial Basis Functions

e-mail addresses: oleg.davydov@strath.ac.uk
morandi@de.unifi.it, sestini@de.unifi.it

1 Introduction

In this paper we suggest an improvement of the scattered data fitting method of [7] based on the extension of local polynomial approximations to bivariate splines. Instead of computing the local polynomials directly from data, we first compute *hybrid* local approximations consisting of a polynomial term and a linear combination of radial basis functions. This is done by the least squares method, where the approximation power is controlled with the help of the minimal singular value of the local collocation matrix [5, 7], and the knots are selected by an ascending greedy algorithm. These hybrid approximations are subsequently converted to a polynomial of higher degree and finally extended to a spline, as in [7]. Clearly, the overall method retains its *linear* computational complexity which is very important for dealing with large real world data.

This approach is motivated by the well known excellent quality of the radial basis function approximation of scattered data contrasted with the difficulties in its computation and use for large data sets (the need for sophisticated numerical techniques to compute the approximation or interpolation and to evaluate it at a low computational cost), see [2]. Since in our approach radial basis approximations are only computed for very small subsets of the data, these difficulties are irrelevant. Note that similar ideas have been explored previously in [11] in the context of the two–stage scattered data fitting methods, where, however, the use of radial basis functions has not apparently led to a substantial improvement comparing to a pure polynomial local approximation method. A different approach for obtaining a smooth overall approximation by using radial basis functions only locally is the partition of unity method (see e.g. [13, 20, 28]).

There are obvious advantages in constructing scattered data approximations as spline surfaces: Apart from the fact that such surfaces can be directly used for the CAGD applications, we mention the prospects of efficient post-processing (e.g. compression) by using spline wavelets and other non–linear methods (see [3, 6]).

In addition, we note that our scheme includes adaptation of the scaling parameters of the local radial basis functions to the local data. The related problem of locally adapted scaling for the radial basis methods has been considered e.g. in [1, 19, 23].

In this paper several different sets of data are considered to illustrate the performance of the proposed approach. The results are systematically compared with those presented in [7]. Our results for Franke test function are also compared to [12]. For each test, figures showing the data distribution and the approximating surface are given, as well as tables reporting the parameters used, the approximation errors and, for larger sets, the computational time. The tests confirm that (almost) artifact–free surfaces can be obtained with our hybrid scheme even using very fine grids, so improving the accuracy of the approximation.

The computational cost of the method depends heavily on the average number of knots used for constructing the local approximations, besides on the prescribed minimum number of local data. So, as the experiments confirm, it is higher than that related to the original method if the averaged number of knots used for the local hybrid approximations is significant.

This behaviour is an obvious consequence of our choice of using an ascending knot insertion strategy to determine the knot distribution for each local hybrid approximation (see Section 3). On the other hand, we remark that it remains linear with respect to the number of data points because the maximum number of local data is controlled by an input parameter.

The paper is organized as follows: After summarizing in Section 2 the bivariate spline approximation method based on direct extension of local polynomials, we introduce in Section 3 the local hybrid approximation scheme and provide the pseudocode (Procedure LHA) to explain its implementation details. The remaining details of the improved scattered data fitting method are clarified in Section 4. Finally, Section 5 is devoted to a detailed presentation of the numerical results with synthetic data (Franke test function) and the real world data already used in [7].

2 Bivariate Spline Approximation Based on Direct Extension of Local Polynomials

In this section we summarize the main lines of the scattered data approximation method introduced in [7, 16] which is based on direct extension of local polynomials to bivariate splines.

First of all, it is a robust scheme because, as shown in [7], for many different data distributions it gives good results, in the sense of both shape recovery and errors at the data or against the exact function for synthetic examples. It is also efficient as its computational complexity grows linearly with the number N of data points $\{\, (\mathbf{X}_i, f_i),\, i = 1, \ldots, N\,, \mathbf{X}_i \in \Omega \subset \mathbb{R}^2\, \}$. Moreover, it does not require construction of a good (e.g. shape regular) data triangulation, which may be challenging for unevenly distributed data, for example contour or track data. Finally, it relies on a numerically stable two–stage procedure that separates from each other the tasks of computing high quality local approximations and obtaining an artifact-free global surface model.

We first briefly describe the second stage of the method of [7] since our approach is not different at this point. Full details may be found in [7].

Let $\triangle^2$ be the so-called *uniform type-2 triangulation* or the *four-directional mesh* [4], which is obtained by drawing vertical and horizontal straight lines at equal distances h_x and h_y, respectively, and adding the two diagonals to each rectangular cell, see Figure 1(a).

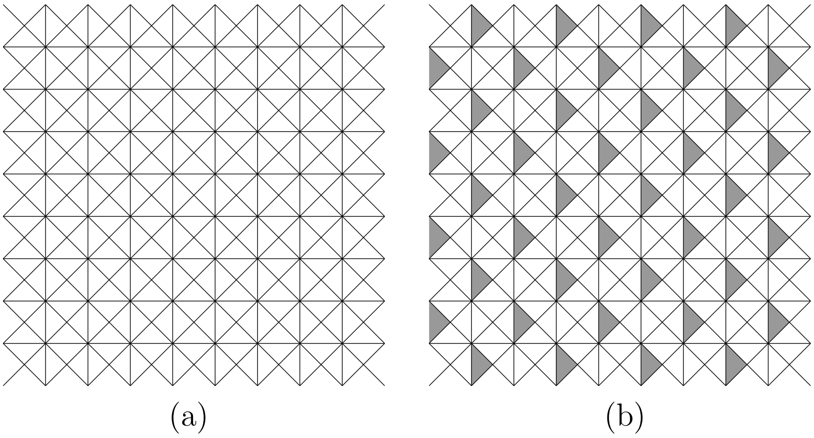

(a) (b)

Figure 1: (a) The four-directional mesh. (b) The triangles in $\mathcal{T}$.

Denote by $\mathcal{S}_d^r(\triangle^2)$ the space of bivariate piecewise polynomial splines of smoothness $r \geq 0$ and degree $d \geq r + 1$,

$$\mathcal{S}_d^r(\triangle^2) \;:=\; \{\, s \in C^r(\mathbb{R}^2) : \, s|_T \in \mathcal{P}_d \text{ for all triangles } T \in \triangle^2\},$$

where $\mathcal{P}_d$ is the space of bivariate polynomials of total degree d.

Every polynomial patch $s|_T$, $T \in \triangle^2$, of a spline $s \in \mathcal{S}_d^r(\triangle^2)$ is determined by its Bernstein-Bézier coefficients $c_{ijk}^{(T)}$, $i + j + k = d$, $i, j, k \in \mathbb{N}$, which can be associated with the *domain points* $\eta_{ijk}^{(T)} \in T$,

$$\eta_{ijk}^{(T)} \;:=\; \frac{i\mathbf{a} \,+\, j\mathbf{b} \,+\, k\mathbf{c}}{d},$$

where $\mathbf{a}$, $\mathbf{b}$ and $\mathbf{c}$ are the vertices of T. We denote by $\mathcal{D}_{d,T} := \{\, \eta_{ijk}^{(T)} : i+j+k=d\,,\, i,j,k \in \mathrm{I\!N}\,\}$ the set of all domain points in T, and by

$$\mathcal{D}_{d,\triangle^2} = \bigcup_{T\in\triangle^2} \mathcal{D}_{d,T}$$

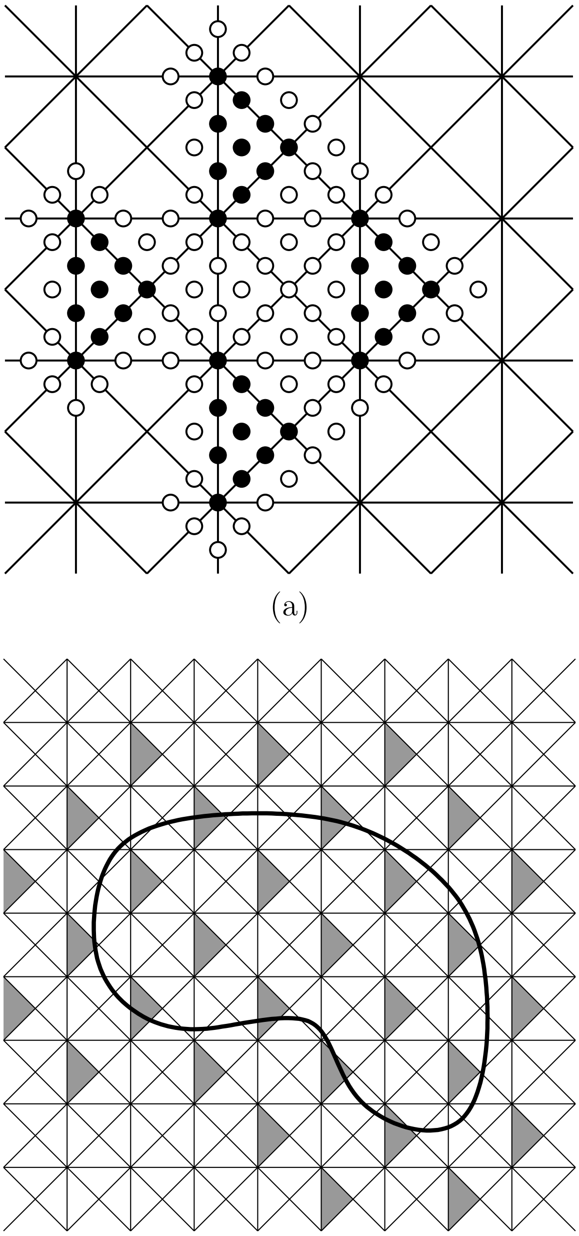
the set of the domain points related to all triangles of $\triangle^2$. If two triangles $T_1, T_2 \in \triangle^2$ share an edge or a vertex, then $\mathcal{D}_{d,T_1} \cap \mathcal{D}_{d,T_2} \neq \emptyset$. However, the continuity of $s \in \mathcal{S}_d^r(\triangle^2)$ ensures that its Bernstein-Bézier coefficients $c_{ijk}^{(T_1)}$ and $c_{ijk}^{(T_2)}$ coincide if $\eta_{ijk}^{(T_1)} = \eta_{ijk}^{(T_1)}$. Therefore, for each $\eta \in \mathcal{D}_{d,\triangle^2}$, where $\eta = \eta_{ijk}^{(T)}$ for some T, the coefficient $c_\eta = c_{ijk}^{(T)}$ of s associated with it is uniquely determined, and the set of all coefficients c_η, $\eta \in \mathcal{D}_{d,\triangle^2}$, completely describes the spline s. Conversely, given arbitrary real numbers c_η, $\eta \in \mathcal{D}_{d,\triangle^2}$, there is a unique spline $s \in \mathcal{S}_d^0(\triangle^2)$ whose Bernstein-Bézier coefficients are c_η.

For $r \geq 1$, the coefficients c_η of $s \in \mathcal{S}_d^r(\triangle^2)$ satisfy the well–known *smoothness conditions* [10], which are explicit equations involving c_η for domain points η from neighboring triangles. A set $\mathcal{M} \subset \mathcal{D}_{d,\triangle^2}$ is called a *minimal determining set (MDS)* for a linear subspace $\mathcal{S} \subset \mathcal{S}_d^0(\triangle^2)$ if setting the coefficients of $s \in \mathcal{S}$ associated with the domain points in $\mathcal{M}$ to zero implies that all the coefficients of s vanish and no proper subset of $\mathcal{M}$ exists with the same property. Thus, for any given values of the coefficients c_η, $\eta \in \mathcal{M}$, there is a unique spline $s \in \mathcal{S}$, whose Bernstein-Bézier coefficients corresponding to the domain points in $\mathcal{M}$ are c_η.

There are three spline spaces considered in [7], namely $\mathcal{S}_3^1(\triangle^2)$, $\mathcal{SS}_6^2(\triangle^2)$ and $\mathcal{RS}_6^2(\triangle^2)$, the latter two being certain subspaces of $\mathcal{S}_6^2(\triangle^2)$. Their common feature is an MDS defined as

$$\mathcal{M} = \mathcal{D}_{d,\triangle^2} \cap \bigcup_{T\in\mathcal{T}} T,$$

where $d=3$ for $\mathcal{S}_3^1(\triangle^2)$, $d=6$ for $\mathcal{SS}_6^2(\triangle^2)$, $\mathcal{RS}_6^2(\triangle^2)$, and $\mathcal{T}$ denotes a subset of $\triangle^2$ defined according to the pattern of gray triangles in Figure 1(b). An important point is that the spline s in any subdomain $\Omega \subset \mathrm{I\!R}^2$ is determined by the coefficients associated with domain points in $\mathcal{M}$ that lie in or near Ω. To illustrate this in the case of $\mathcal{S}_3^1(\triangle^2)$, we provide Figure 2 that shows (a) all domain points that are completely determined by the part of $\mathcal{M}$ contained in four neighboring triangles of $\mathcal{T}$, and (b) the part $\mathcal{T}_\Omega$ of $\mathcal{T}$ needed to determine the spline on a subdomain Ω. For $\mathcal{SS}_6^2(\triangle^2)$ and $\mathcal{RS}_6^2(\triangle^2)$ the situation is similar, see [7].



(a)

(b)

Figure 2: (a) Domain points shown as dots are completely determined by the part of $\mathcal{M}$ contained in four neighboring triangles of $\mathcal{T}$ (black dots). (b) The gray triangles shown here completely determine the spline on this subdomain.

Thus, the second ("extension") stage of the method amounts to computing, using the smoothness conditions, the entire Bézier net of the spline s provided that its coefficients associated with domain points in each $T \in \mathcal{T}$ are known.

The goal of the first stage is to compute the Bézier nets of the polynomial pieces $p_T = s|_T$ of s on all triangles $T \in \mathcal{T}$, using the data $(\mathbf{X}_i, f_i)$, $i = 1, \ldots, N$. To this end, for each $T \in \mathcal{T}$, the data points corresponding to locations $\mathbf{X}_i$ inside a circle Ω_T centered at the barycenter of T and with radius equal to its diameter are used. If, however, these local data are too few, the radius is suitably increased and, in the opposite case, a grid–type thinning algorithm is used to reduce their number N_T. This process of local data selection is controlled by two user specified parameters, M_{min} and M_{max}, see [7] for further details.

The local approximation method employed in [7] is the discrete least–squares polynomial fit to the local data using the Bernstein–Bézier basis for the computations, where the degree of the polynomial (not exceeding the spline degree d) is selected by an adaptive algorithm. Here, the crucial role is played by the minimal singular value $\sigma_{min,q,T}$ of the collocation matrix $M_{q,T}$ related to the evaluations of the Bernstein-Bézier basis polynomials of degree q at the locations $\mathbf{X}_i$ of the local data. As shown in [5], keeping $\sigma^{-1}_{min,q,T}$ within a reasonably small bound is important not only for the numerical stability of the computational process but also for guaranteeing the approximation power of the least-squares approximation method. Therefore, a user specified tolerance $\kappa_P \geq 1$ is introduced, such that a local polynomial approximation of a certain degree $q \leq d$ is accepted only if $\sigma^{-1}_{min,q,T} \leq \kappa_P$. Thus, for each $T \in \mathcal{T}$, the highest degree $0 \leq q \leq d$ is selected such that $\sigma^{-1}_{min,q,T}$ is acceptably small. (Note that $\sigma_{min,0,T} = 1$, such that $q = 0$ may always be accepted.)

3 Local Hybrid Approximation Scheme

The idea of the hybrid method is to enhance the approximation quality of the local approximations by using linear combinations of polynomials and radial basis functions.

The local hybrid approximation scheme employs the space

$$\Pi_q^2 = \text{span}\,\{p_1^T, \ldots, p_m^T\}, \qquad m = \binom{q+2}{2},$$

of bivariate polynomials of degree $q \geq 0$ and a function $\phi_T : \mathbb{R}_{\geq 0} \to \mathbb{R}$. The latter can be any suitably smooth *positive definite* function or a *conditionally positive definite* function of order at most $q+1$ on $\mathbb{R}^2$, see [2].

Let $\Xi_T = \{\mathbf{X}_1, \ldots, \mathbf{X}_{N_T}\}$ denote the local scattered locations related to the triangle T with vertices $\mathbf{v}_1^T, \mathbf{v}_2^T, \mathbf{v}_3^T$, and let us assume that $N_T \geq m$. The *local hybrid approximation* g_T of the form

$$g_T(\cdot) = \sum_{j=1}^{m} a_j^T \, p_j^T(\cdot) \; + \; \sum_{j=1}^{n_T} b_j^T \, \phi_T(\| \cdot - \mathbf{Y}_j^T \|_2) \tag{1}$$

is constructed via the minimization of the ℓ_2-norm of the residual on Ξ_T,

$$\Big(\sum_{i=1}^{N_T} (f_i - g_T(\mathbf{X}_i))^2 \Big)^{1/2}, \tag{2}$$

where $0 \leq n_T \leq N_T - m$, and the set of *knots* $Y_T = \{\mathbf{Y}_j\,, j = 1, \ldots, n_T\}$ is a subset of Ξ_T.

A priori there is no guarantee that this least-squares minimization problem has a unique solution, nor that g_T has good approximation properties. Usually, the unique solvability of the interpolation with conditionally positive definite radial basis functions is enforced by applying additional *orthogonality* constraints

$$\sum_{j=1}^{n_T} b_j^T \, p(\mathbf{Y}_j^T) = 0, \qquad \text{all } p \in \Pi_q^2, \tag{3}$$

on the coefficients b_j^T in (1) and assuming that Y_T contains a Π_q^2–unisolvent subset, see [2]. This approach also extends to the least-squares [18, 21].

We however take a different route in this paper. In order to retain all $n_T + \binom{q+2}{2}$ degrees of freedom, we skip the constraints (3), and consider the full hybrid space

$$\mathcal{H}_T := \text{span} \left\{ p_1^T, \ldots, p_m^T, \phi_T(\| \cdot - \mathbf{Y}_1^T \|_2), \ldots, \phi_T(\| \cdot - \mathbf{Y}_{n_T}^T \|_2) \right\}$$

instead of its subspace

$$\mathcal{R}_T := \left\{ \sum_{j=1}^{m} a_j \, p_j^T(\cdot) \; + \; \sum_{j=1}^{n_T} b_j \, \phi_T(\| \cdot - \mathbf{Y}_j^T \|_2) \; : \; (b_1, \ldots, b_{n_T}) \text{ satisfies (3)} \right\}.$$

Clearly, the approximation power of $\mathcal{H}_T$ is at least as good as that of $\mathcal{R}_T$ since

$$E(f, \mathcal{H}_T)_{C(T)} \leq E(f, \mathcal{R}_T)_{C(T)},$$

where $E(f,\mathcal{S})_{C(T)}$ is the error of the best approximation of f from a linear space $\mathcal{S}$,

$$E(f,\mathcal{S})_{C(T)} := \inf_{g\in\mathcal{S}} \|f-g\|_{C(T)}.$$

To ensure that the quality our local discrete least squares approximations is comparable with that of the local *best* approximation from $\mathcal{H}_T$, we compute the minimal singular value $\sigma_{min}(C_T)$ of the collocation matrix C_T defined by

$$\begin{bmatrix} p_1^T(\mathbf{X}_1) & \dots & p_m^T(\mathbf{X}_1) & \phi_T(\|\mathbf{X}_1-\mathbf{Y}_1^T\|_2) & \dots & \phi_T(\|\mathbf{X}_1-\mathbf{Y}_{n_T}^T\|_2) \\ \vdots & & \vdots & \vdots & & \vdots \\ p_1^T(\mathbf{X}_{N_T}) & \dots & p_m^T(\mathbf{X}_{N_T}) & \phi_T(\|\mathbf{X}_{N_T}-\mathbf{Y}_1^T\|_2) & \dots & \phi_T(\|\mathbf{X}_{N_T}-\mathbf{Y}_{n_T}^T\|_2) \end{bmatrix}$$

and proceed to compute g_T only if

$$\sigma_{min}^{-1}(C_T) \le \kappa_H, \tag{4}$$

where κ_H is a user specified tolerance. This also ensures that the considered least squares problem is uniquely solvable. Note that we prefer avoiding rank deficient least squares problems since the minimum norm solutions available in that case do not necessarily maintain the exact reproduction of the elements of the space (in particular, local polynomial reproduction) vital for the estimation of the approximation power of the method [5].

Let us explain our motivation for using (4), as opposite to any cheaper technique avoiding the computation of $\sigma_{min}(C_T)$ (see e.g. [26]). According to the estimate given in [5], under assumption that $f_i = f(\mathbf{X}_i)$, $i=1,\dots,N_T$, for a continuous function f, we have

$$\|f-g_T\|_{C(T)} \le \left(1+\frac{K_T\sqrt{N_T}}{\sigma_{min}(C_T)}\right) E(f,\mathcal{H}_T)_{C(T)}, \tag{5}$$

where

$$K_T := \max_{\{a_j\},\{b_j\}} \frac{\left\|\sum_{j=1}^m a_j\, p_j^T(\cdot) + \sum_{j=1}^{n_T} b_j\, \phi_T(\|\cdot - \mathbf{Y}_j^T\|_2)\right\|_{C(T)}}{\left(\sum_{j=1}^m |a_j|^2 + \sum_{j=1}^{n_T} |b_j|^2\right)^{1/2}}.$$

Now, for a properly scaled polynomial basis $\{p_1^T,\dots,p_m^T\}$ (for example, the Bernstein–Bézier basis with respect to T used in our implementation), and appropriate ϕ_T, K_T is bounded independently of T. Since also $N_T \le$

M_{max}, the estimate (5) shows that the size of $\|f - g_T\|_{C(T)}$ is comparable with that of the best approximation $E(f, \mathcal{H}_T)_{C(T)}$ if (4) holds.

Thus, (4) provides a criterion to accept or reject a given knot set Y_T (which uniquely determines the matrix C_T). In order to find a suitable Y_T, it is possible, for example, to start by removing m points from Ξ_T, check whether (4) is satisfied for such Y_T, and, if not, successively remove further knots until (4) holds. In this paper we take another approach, namely we use an ascending greedy procedure, i.e. we start with very few knots and keep adding them at the positions of the highest current error while (4) holds. If our initial small Y_T violates (4), then we simply switch to the pure polynomial local approximation by applying the procedure described in [7] with starting degree q. Note that a different greedy algorithm of knot selection for global approximation with radial basis functions has been proposed in [15].

We now formulate our hybrid approximation scheme as *Procedure LHA*. Besides the local data set Ξ_T and the related function values $f_1, \ldots, f_{N_T}$, its *input parameters* are the three vertices $\{\mathbf{v}_1^T, \mathbf{v}_2^T, \mathbf{v}_3^T\}$ of the triangle T, the upper bound $n_{max} \geq 3$ of $n_T = \# Y_T$, the degree q and tolerance κ_P for the polynomial part and the tolerance κ_H from (4). Its *output* is the set of knots Y_T with $n_T = \#Y_T \leq \min\{n_{max}, N_T - m\}$ and the related hybrid local approximation g_T, i.e. the coefficients $a_1^T, \ldots, a_m^T$ and $b_1^T, \ldots, b_{n_T}^T$ that define g_T according to (1).

As mentioned above, the procedure is *ascending*, and we start with just three knots if N_T is big enough. Namely, if $N_T \geq m + 3$, we select by symmetry reasons the first three knots to be the three distinct points in Ξ_T nearest to the vertices of the triangle. (If else $N_T < m + 3$, then we compute the local polynomial approximation p_T of degree at most q using the algorithm of [7], and terminate.) We keep adding knots one at a time while $n_T < \min\{n_{max}, N_T - m\}$ and (4) holds. (Should the condition (4) fail at a step of this while loop, we remove the last inserted knot and go back to the previous hybrid approximation.) At each step the new candidate knot is determined by choosing a location $\mathbf{X}_i \in \Xi_T \setminus Y_T$ where the current approximation error is the worst.

Procedure LHA

```
Input: $N_T$ , $\Xi_T = \{\mathbf{X}_1, \ldots, \mathbf{X}_{N_T}\}$ , $\{f_1, \ldots, f_{N_T}\}$ , $\{\mathbf{v}_1^T , \mathbf{v}_2^T , \mathbf{v}_3^T\}$ , $n_{max}$ , $q$ ,
       $\kappa_P$ , $\kappa_H$
Output: $Y_T$ , $g_T$
1.   Set $n_T = 0$ , $Y_T = \emptyset$
2.   If $N_T < \binom{q+2}{2} + 3$
         2.1a Set $g_T = p_T$ (polynomial approximation as in Section 2)
     else
         2.1b Set $n_T = 3$
         2.2b For i=1,2,3
                 2.2b.1 Set $Y_T = Y_T \cup \{\mathbf{X}_{k_i}\}$ , where
                        $\|\mathbf{X}_{k_i} - \mathbf{v}_i^T\|_2 = \min_{\mathbf{X} \in \Xi_T \setminus Y_T} \|\mathbf{X} - \mathbf{v}_i^T\|_2$
             end
         2.3b Initialize the current collocation matrix $C_T$
         2.3c If $\sigma_{min}^{-1}(C_T) > \kappa_H$
                 2.3c.1a Set $n_T = 0$ , $Y_T = \emptyset$
                 2.3c.2a Set $g_T = p_T$
             else
                 2.3c.1b Set $g_T =$ current hybrid approximation
                 2.3c.2b While $n_T < \min\{n_{max} , N_T - \binom{q+2}{2}\}$
                             Set $err_j = |f_j - g_T(\mathbf{X}_j)|$ , $j = 1, \ldots, N_T$
                             Set $i = \arg\max_{j \in \Xi_T \setminus Y_T} err_j$
                             Set $n_T = n_T + 1$ , $Y_T = Y_T \cup \{\mathbf{X}_i\}$
                             Update $C_T$
                             If $\sigma_{min}^{-1}(C_T) > \kappa_H$
                                 Set $n_T = n_T - 1$ , $Y_T = Y_T \setminus \{\mathbf{X}_i\}$
                                 Break
                             end
                             Set $g_T =$ current hybrid approximation
                         end
             end
     end
```

4 Scattered Data Fitting Method

The above local hybrid approximation scheme is incorporated into the two–stage scattered data fitting algorithm of [7], replacing the pure polynomial local approximations originally employed there. We now clarify the remaining details of the improved scattered data fitting method and give some specific information related to the numerical tests of Section 5.

By the nature of the direct extension method, the output of the first stage on each triangle $T \in \mathcal{T}$ must be a polynomial p_T of degree d in Bernstein-Bézier form (see Section 2). Since the local hybrid approximation g_T is given differently, namely as a sum of the form (1), it has to be *converted* into a polynomial by an approximation method.

Specifically, we use for this purpose the discrete least squares with respect to the evaluations of g_T at $\binom{D+2}{2}$ domain points on T related to the space polynomials of a certain degree $D \geq d$. In our numerical experiments we take $D = 2d$. This delivers slightly better results than those with $D = d$ (that corresponds to interpolation). Note that $\sigma^{-1}_{min,d,T}$ for the corresponding collocation matrix is small (2.87 for $D = 6$ and 21.74 for $D = 12$), which ensures that the least squares polynomial of degree d is a good approximation of g_T [5]. Moreover, this matrix is the same for all triangles $T \in \mathcal{T}$. Therefore, the computational cost of the conversion is negligible comparing to the total cost of the scheme.

Obviously, *all* triangles $T \in \triangle^2$ may be split into eight groups, each of which is obtained from $\mathcal{T}$ by shifting and/or varying the orientation of the coordinate system (see [7, Figure 6]). Therefore, a way to remove some unsymmetry pertinent to $\mathcal{T}$ is to average the eight different approximations. Indeed, the numerical results in [7] have shown that such *averaging* leads to substantially lower approximation errors and better visual quality approximations, at the expense of nearly eight times higher computational cost. Since the goal of our numerical tests is to obtain the best possible reconstructions of the surfaces represented by the given scattered data, we always apply this averaging procedure in the examples below.

More precisely, in all the experiments reported in Sections 5.1–5.3, the final approximation is computed using the averaged operator RQ_2^{av} that produces a C^2 continuous spline of total degree $d = 6$ with respect to the four–directional mesh. Following the choice made in [7], we use the operator Q_1^{av}, leading to a C^1 spline of total degree 3, only for the experiments related to the data set considered in Section 5.4. (See [7, Section 5] for more details about RQ_2^{av} and Q_1^{av}.) Thus, the comparison with the original pure polynomial method is done relating to the results given in [7] always using the same method in the second stage.

For the local hybrid approximation method we still have to specify the local polynomial basis $\{p_1^T, \ldots, p_m^T\}$ and the function ϕ_T, see Section 3. As $\{p_1^T, \ldots, p_m^T\}$ we always take the basis of bivariate Bernstein polynomials with respect to the triangle T notable for its superior numerical stability [9].

Among several standard radial basis functions [2] we choose Hardy's *multiquadric* $\phi_{MQ}(r) = -\sqrt{1+r^2}$ [17], which is C^∞ and conditionally positive definite of order 1, for the majority of our tests, including all tests with real-world data. Indeed, multiquadric is one of the best known and most often used radial basis functions. (However, in Section 5.1 we present a set of experiments with synthetic data using 11 different functions.)

An obvious advantage of the two–stage approach is that the local problems solved in the first stage are independent from each other, which makes it easy to adapt the method to the local data distribution. A major example of such adaptation is the above procedure LHA, where the selection of knots $\mathbf{Y}_T$ for each local portion of data depends on the local errors. We add another degree of adaptivity to the method in what we use on each T a scaled version ϕ_T of the same radial basis function ϕ. Thus, we introduce a *scaling parameter* δ [23] and determine the function ϕ_T to be used in (1) by

$$\phi_T(r) = c_T \phi\Big(\frac{r}{\delta d_T}\Big), \tag{6}$$

where

$$d_T := \max_{1 \le i,j \le N_T} \|\mathbf{X}_i - \mathbf{X}_j\|_2$$

is the diameter of the local data set Ξ_T, and c_T is a constant. (Clearly, the size of d_T will vary with the local density of the data.) For instance, for $\phi = \phi_{MQ}$ we set

$$\phi_T(r) = \phi_{T,MQ}(r) := -\delta d_T\, \phi_{MQ}\Big(\frac{r}{\delta d_T}\Big) = \sqrt{(\delta d_T)^2 + r^2}, \tag{7}$$

which shows that δ is related to the classical parameter c of the multiquadric in its form $\sqrt{c^2 + r^2}$.

Altogether, in addition to the radial basis function ϕ, our method depends on several *parameters*: the grid width h_x and height h_y, the degree q of the polynomial part, the inverse minimal singular value tolerances κ_H and κ_P, local data selection parameters M_{min}, M_{max} (see [7, Section 6.1]), the scaling coefficient δ used in (6), the upper bound n_{max} on the number of knots n_T in (1), and the parameter D of the uniform grid used for the evaluation of the local hybrid approximations g_T.

Instead of h_x and h_y we actually use as parameters in our numerical tests the *grid size* $n_x \times n_y$. That is, we compute the minimal rectangle $[a, b] \times [c, d]$ containing all data sites $\mathbf{X}_i$, $i = 1, \dots, N$, and split it into $n_x n_y$ equal rectangles by $n_x - 1$ equispaced vertical and $n_y - 1$ horizontal straight lines. (Thus, $h_x = (b-a)/n_x$, $h_y = (d-c)/n_y$.) The four–directional mesh is obtained by drawing in the diagonals of the rectangles. Note that the number of parameters needed to store the resulting spline is $5n_x n_y + 4(n_x + n_y) + 3$ in the C^1 case, and $14n_x n_y + 12(n_x + n_y) + 6$ in the C^2 case, see [7, Section 6.1].

In all tests we take $D = 2d$, as explained above, and $n_{max} = 400$. With this high value for n_{max}, the number of local knots is in fact restricted only by the influence of the tolerance κ_H. Therefore, D and n_{max} are no more reported below.

Although ϕ_{MQ}, according to [22], does not require in principle any polynomial term to ensure the non–singularity of the collocation matrix with respect to any knots, the degree $q = 0$ polynomial part in (1), that is constant, has proved useful in our tests with this function, especially when almost flat subareas have to be reconstructed. Therefore we always include a low degree polynomial term, as described in Section 3. Some other radial basis functions, such as thin plate spline require q to be at least 1, see Section 5.1. Our experience is that in most cases q should be kept as small as possible. One advantage of $q = 0$ is that the tolerance κ_P is irrelevant in this case since the constant polynomial collocation matrix has the minimal singular value equal to 1 independently of the knot locations. (Note that κ_P may be also irrelevant when $q > 0$ in case that there are no pure polynomial local approximations, which are only obtained in items *2.1a* and *2.3c.2a* of Procedure LHA.)

The grid size $n_x \times n_y$ as well as the parameters M_{min}, M_{max}, δ and κ_H significantly influence the performance of the method. Moreover, our experience shows that there are no universally best values for them, but rather they should be always adjusted to the type of data in question. (Similar behaviour has been observed for the method of [7] with pure polynomial local stage, where parameter κ_P played the role of our δ and κ_H.) On the other hand, since the same parameter values successfully work for all local approximations computed for a large data set (e.g. the 621624 points of the Rotterdam

Port data in Section 5.4), it is clear that these values may be found in practice by a calibration procedure which would involve experimenting with a small subset of the data and cross–validation. In this paper, similarly to [7], we have not tried to formalize such procedures but were always able to find good parameter values for our tests by a straightforward try–and–error approach.

5 Numerical Results

In what follows, our scattered data fitting method is denoted by ($\mathrm{H_{MQ}}$) when the multiquadric $\phi_T = \phi_{T,MQ}$ defined in (7) is used in (1). In Section 5.1 (H) appears also with different indices when we test our method with other radial basis functions. The scattered data fitting method of [7] relying on the pure polynomial local approximations is denoted by (P).

We consider below four data sets which were also used in [7]. The locations $\{\mathbf{X}_i\}$ of the data sites are shown in Figure 3. The aim of the experiments is the accurate and artifact–free recovery of the underlying shape. The numerical results of each test are reported in a table which contains the *maximum*, the *mean* and the *root mean square error* calculated either on a suitable uniform grid for synthetic data ("*maxg*", "*meang*" and "*rmsg*", respectively), or at the data sites for the real world data ("*max*", "*mean*" and "*rms*", respectively). The tables also report the average number of knots used for local radial basis approximation, denoted by n_T^{aver}. Since the real world data sets tested in Sections 5.2–5.4 do not come from a known function, figures that allow the evaluation of the results by the visual inspection of the surfaces are also provided.

The numerical tests have been performed on a Sun Ultra 60 workstation with a 450 MHz processor using our C implementation of the method. Our program also invokes the routines from the two–stage scattered data fitting toolbox [8], and, for the computation of the singular value decomposition of the collocation matrices, from "Numerical Recipes in C" [24]. For the visualization we have used Matlab 6.

5.1 Franke test function

Our first test deals with a small set of $N = 100$ data points obtained by sampling the Franke test function (Figure 4a) at the locations shown in Figure 3a. This data set is available from [14] as `ds3`. It is interesting for testing the shape recovery capability of the method because the underlying function is available,

$$
\begin{aligned}
f(x,y) = \ & \tfrac{3}{4}\exp\Big[-\tfrac{(9x-2)^2+(9y-2)^2}{4}\Big] + \tfrac{3}{4}\exp\Big[-\tfrac{(9x+1)^2}{49}-\tfrac{(9y+1)}{10}\Big] \\
& + \tfrac{1}{2}\exp\Big[-\tfrac{(9x-7)^2+(9y-3)^2}{4}\Big] - \tfrac{1}{5}\exp\Big[-(9x-4)^2-(9y-7)^2\Big].
\end{aligned}
$$

For this data set we consider, in addition to ϕ_{MQ}, several other well-known radial basis functions, see [2]:

- Inverse multiquadric $\phi_{IMQ}(r) = \frac{1}{\sqrt{1+r^2}}$.
- Gaussian $\phi_G(r) = e^{-r^2}$.
- Thin plate splines (also called polyharmonic splines) $\phi_{TP}(r) = r^2 \log r$, $\phi_{TP3}(r) = r^3$, $\phi_{TP4}(r) = -r^4 \log r$, $\phi_{TP5}(r) = -r^5$.
- Wendland's functions $\phi_{W2}(r) = (1-r)_+^4(4r+1)$, $\phi_{W4}(r) = (1-r)_+^6(35r^2+18r+1)$, $\phi_{W6}(r) = (1-r)_+^8(32r^3+25r^2+8r+1)$. These functions are C^2, C^4 and C^6, respectively.
- Buhmann's C^3 function

$$
\phi_{B3}(r) = \begin{cases} \frac{112}{45}r^{9/2} + \frac{16}{3}r^{7/2} - 7r^4 - \frac{14}{15}r^2 + \frac{1}{9}, & \text{if } 0 \le r \le 1, \\ 0, & \text{if } r > 1. \end{cases}
$$

The factor c_T in (6) is $c_T = 1/\delta d_T$ for ϕ_{IMQ}, $c_T = 2$ for ϕ_{TP} and ϕ_{TP4}, and $c_T = 1$ for the other functions. Thus, in addition to the method ($\mathrm{H_{MQ}}$) based on the multiquadric, we consider ($\mathrm{H_{IMQ}}$), ($\mathrm{H_G}$), ($\mathrm{H_{TP}}$), ($\mathrm{H_{TP3}}$), ($\mathrm{H_{TP4}}$), ($\mathrm{H_{TP5}}$), ($\mathrm{H_{W2}}$), ($\mathrm{H_{W4}}$), ($\mathrm{H_{W6}}$), and ($\mathrm{H_{B3}}$).

The results are reported in Table 1. To facilitate the comparison, we use the same C^2 method RQ_2^{av} and take $n_x = n_y = 5$, $M_{max} = N = 100$ and $M_{min} = 16$ for all tests. (These are the parameters of the best result with `ds3` in [7]). The other parameters q, κ_H and δ are chosen individually for the basis functions and are reported in the table. Note that ϕ_{TP}, ϕ_{TP3} and ϕ_{TP4}, ϕ_{TP5} are conditionally positive definite of order 2 and 3, respectively. Therefore, the degree of the polynomial term should be at least 1, respectively 2 for the methods based on these functions. The parameter κ_P is not shown since there have been no pure polynomial local approximations in these tests. The errors are calculated on the uniform 101 × 101 grid, which is also used for the visualization in Figure 4b showing the spline surface obtained with the method (H_{MQ}). For comparison we included in the table the grid errors for the same data obtained with the method (P) of [7], see Table 3 in that paper. Since global methods based on five of the above radial basis functions have been tested on `ds3` in [12], we also included in Table 1 the corresponding grid errors reported there. The global methods are denoted by (G) with appropriate indices. Note that (G_{MQ}), (G_{IMQ}) and (G_G) do not make use of any polynomial term, whereas (G_{TP}) and (G_{TP3}) do add a linear polynomial, and that the errors are computed in [12] on a 33 × 33 grid. Clearly, with the global methods there is only one radial basis approximation involved, and it simply uses all $N = 100$ data sites of `ds3` as knots $\{\mathbf{Y}_j\}$ to interpolate the data by a sum of the form $\sum_{j=1}^N b_j\, \phi(\|\cdot - \mathbf{Y}_j\|_2)$ or $\tilde{p} + \sum_{j=1}^N b_j\, \phi(\|\cdot - \mathbf{Y}_j\|_2)$, $\tilde{p} \in \Pi_q^2$, with the orthogonality conditions $\sum_{j=1}^N b_j\, p(\mathbf{Y}_j) = 0$ for all $p \in \Pi_q^2$.

method	q	κ_H	δ	*maxg*	*meang*	*rmsg*	n_T^{aver}
(H_{MQ})	0	10^5	0.4	$1.6\ 10^{-2}$	$1.9\ 10^{-3}$	$3.0\ 10^{-3}$	21.1
(H_{IMQ})	0	10^4	0.5	$1.5\ 10^{-2}$	$2.0\ 10^{-3}$	$3.1\ 10^{-3}$	21.3
(H_G)	0	10^4	0.4	$1.9\ 10^{-2}$	$2.2\ 10^{-3}$	$3.5\ 10^{-3}$	19.3
(H_{TP})	1	10^5	2.0	$5.7\ 10^{-2}$	$7.8\ 10^{-3}$	$1.3\ 10^{-2}$	20.7
(H_{TP3})	1	10^5	2.0	$4.7\ 10^{-2}$	$4.5\ 10^{-3}$	$7.5\ 10^{-3}$	20.3
(H_{TP4})	2	10^5	2.0	$3.0\ 10^{-2}$	$3.4\ 10^{-3}$	$5.5\ 10^{-3}$	14.9
(H_{TP5})	2	10^6	2.0	$2.8\ 10^{-2}$	$3.4\ 10^{-3}$	$5.2\ 10^{-3}$	13.4
(H_{W2})	0	10^4	2.0	$3.9\ 10^{-2}$	$4.1\ 10^{-3}$	$7.1\ 10^{-3}$	22.7
(H_{B3})	0	10^5	2.0	$3.3\ 10^{-2}$	$3.6\ 10^{-3}$	$6.0\ 10^{-3}$	22.4
(H_{W4})	0	10^4	2.0	$2.1\ 10^{-2}$	$2.1\ 10^{-3}$	$3.6\ 10^{-3}$	21.2
(H_{W6})	0	10^5	2.0	$1.6\ 10^{-2}$	$1.9\ 10^{-3}$	$3.0\ 10^{-3}$	20.9
(P)	6			$3.8\ 10^{-2}$	$5.2\ 10^{-3}$	$7.6\ 10^{-3}$	
(G_{MQ})				$2.3\ 10^{-2}$	$1.8\ 10^{-3}$	$3.6\ 10^{-3}$	
(G_{IMQ})				$2.5\ 10^{-2}$	$2.8\ 10^{-3}$	$5.2\ 10^{-3}$	
(G_G)				$6.2\ 10^{-2}$	$6.0\ 10^{-3}$	$1.1\ 10^{-2}$	
(G_{TP})	1			$5.2\ 10^{-2}$	$5.3\ 10^{-3}$	$9.5\ 10^{-3}$	
(G_{TP3})	1			$2.5\ 10^{-2}$	$3.1\ 10^{-3}$	$5.8\ 10^{-3}$	

Table 1: Franke function test (`ds3` data set).

The first conclusion we can made from the table is that our method performs very well for this standard data set, whatever the function ϕ is. In particular, the errors are in most cases better than those with (P), and comparable to the best results reported in [12]. Note that apart from ($\mathrm{G_{MQ}}$)–($\mathrm{G_{TP3}}$) there are 24 other scattered data fitting methods tested on `ds3` in [12]. The results range from $maxg = 4.9\ 10^{-2}$, $meang = 5.4\ 10^{-3}$, $rmsg = 9.4\ 10^{-3}$ for the minimum norm network method by Nielson to $maxg = 2.7\ 10^{-1}$, $meang = 4.2\ 10^{-2}$, $rmsg = 6.2\ 10^{-2}$ for the Shepard method. Another comparative study of scattered data fitting methods has been performed recently in [25]. Here, 10 methods published as ACM algorithms are tested on various synthetic data sets, including `ds3`. The best error on this data reported in [25] corresponds to $rmsg = 6.8\ 10^{-3}$, which confirms once again that the results shown in Table 1 belong to the best ones available for `ds3`.

Clearly, the differences in errors obtained with various radial basis functions in Table 1 cannot be considered as sufficient evidence to prefer one function over another. Nevertheless, based on these results and motivated by the practical need to keep the number of experiments reasonable, we have picked *multiquadric* ϕ_{MQ} for our subsequent tests. Figure 4 confirms the high visual quality of the spline surface obtained with ($\mathrm{H_{MQ}}$).

Table 2 presents the results of tests with random data. More precisely, for each $N = 10^2, 10^3, 10^4$, N points in $[0,1]^2$ are selected randomly (with uniform distribution), and the data is created by evaluating the Franke function at these sites. Then the spline approximation is computed using the method ($\mathrm{H_{MQ}}$) with $q = 0$. This test is repeated 10 times with different random data, and the average errors and knot numbers are reported in the table. For $N = 10^2$ we use exactly the same parameters as above for the 100 point set `ds3`. For the larger data sets we increase $n_x = n_y$ up to the closest integer to $\sqrt{N}/2$ and take $M_{max} = 400$. Interestingly, M_{min}, κ_H and δ had to be adjusted, too, to achieve a better performance, see the table for the values that we use.

N	n_x	M_{min}	κ_H	δ	$maxg$	$meang$	$rmsg$	n_T^{aver}
10^2	5	16	10^5	0.4	$4.60\ 10^{-2}$	$3.98\ 10^{-3}$	$7.46\ 10^{-3}$	19
10^3	16	40	10^{12}	1.0	$1.69\ 10^{-4}$	$1.53\ 10^{-6}$	$6.47\ 10^{-6}$	47
10^4	50	40	10^{15}	1.6	$4.64\ 10^{-7}$	$5.62\ 10^{-9}$	$1.51\ 10^{-8}$	48

Table 2: Franke function test (random data).

The results in Table 2 compare favorably to analogous tests with (P) reported in [7, Table 1] and three other methods reported in [20, Table 4].

5.2 Black Forest

The second test relates to a set of 15885 points (the "black forest" data set of [7]) representing a terrain in the neighborhood of Freiburg, Germany. The difference between the highest and the lowest point is $\approx 1214\,m$. The respective locations are shown in Figure 3b where we also indicate with a box a difficult area with varying point density used to check the visual quality of the surface.

Figure 5 shows the C^2 approximating surface obtained with the hybrid scheme ($\mathrm{H_{MQ}}$) based on multiquadrics using $n_x = n_y = 80$ (i.e. the same grid size as used in [7, Fig. 16c]). Figures 5a and 5b (both related to the subregion indicated in Figure 3b) correspond to the tests reported in the first and second rows of Table 3, respectively.

Figure 5 has been obtained using the evaluations of the spline on the 1201×1201 grid (this is the same grid used to evaluate the errors in Table 3), while Figures 6a and 6b rely on the 421×421 evaluation grid in the subregion.

Table 3 presents the parameters of each test, as well as the data errors *max, mean* and *rms*, average number of knots n_T^{aver} and the computation CPU time t in seconds. As for the Franke function test, we take $n_y = n_x$, $M_{max} = 100$ and do not provide κ_P since there is no need in this tolerance in the case $q = 0$. Moreover, there have been no pure polynomial (i.e., constant) local approximations in these tests. For the ease of comparison, Table 3 also includes the errors, the computation time and the values of n_x and M_{min} for the test with the method (P) reported in [7].

	n_x	κ_H	M_{min}	δ	*max*	*mean*	*rms*	n_T^{aver}	t (sec)
($\mathrm{H_{MQ}}$)	80	10^4	12	0.3	$32.0\,m$	$1.39\,m$	$2.17\,m$	12.2	137
($\mathrm{H_{MQ}}$)	160	10^5	16	0.3	$15.7\,m$	$0.29\,m$	$0.60\,m$	20.0	2050
(P)	80		3		$30.6\,m$	$2.56\,m$	$3.57\,m$		9.8

Table 3: Black forest test.

According to these results, by using the local hybrid approximation scheme for the same spline grid ($n_x = n_y = 80$), we achieve essential reduction of the mean and root mean square errors at the data, while maintaining the high quality of the surface (Figure 5a). Moreover, the errors reduce significantly if we double n_x and n_y, obtaining very moderate oscillations in the difficult areas of low data density (Figure 5b). On the other hand, the last column of the table shows clearly that this significant improvement is associated with a higher computational cost. In the following subsection we will see however that this is not always the case.

5.3 Glacier

We consider next the "glacier" data set of 8345 points representing digitized contours of a glacier, available from [14] as `vol87`. The locations of the data sites are shown in Figure 3c.

The results of our tests are presented in Table 4 and Figures 7–9. For the sake of comparison, Table 4 and Figures 8–9 also include results obtained with the method (P) in [7].

Since the domain is not square, we use non–equal grid parameters in x and y direction such that $n_y = 1.2\,n_x$. (Thus, only n_x is reported in the table). The visualization is done by using evaluations of the C^2 spline surfaces on a 301×361 grid. Following [7], we take $M_{max} = 160$ in this test. Since we use $q = 0$, parameter κ_P is not relevant.

We compute three C^2 spline approximations using the method ($\mathrm{H_{MQ}}$), as usual. We use the same value of scaling parameter $\delta = 0.4$ in all three cases. For the first of these tests we take $n_x = 20$ and $M_{min} = 60$, as in [7] and achieve about 20% improvement in the data errors comparing to the method (P), while the visual quality remains high, see the first row of Table 4, and Figures 7, 8a and 9a. The computation cost is however about 4 times higher than for (P). Therefore, in our second test we change the parameters M_{min} and κ_H to lower values and are still able to obtain a good approximation, with the computation cost reduced to that of the polynomial method, see the second row of Table 4, and Figures 8b and 9b. This reduction of the computation cost is achieved obviously because the number of the knots used in local approximations (n_T^{aver} in Table 4) significantly drops. As in the case of "black forest," the surface remains almost artifact–free when we double the spline grid to $n_x = 40$, thus significantly reducing the errors at the data, see the third row of Table 4, and Figures 8c and 9c.

5.4 Rotterdam Port

Finally, we consider the "Rotterdam Port" data set provided to the authors of [7] by Quality Positioning Services (Zeist, The Netherlands). The original data consists of 634604 raw measurements from the bottom of the Rotterdam harbor produced by the density multibeam echosounder and recorded using

	n_x	κ_H	M_{min}	δ	*max*	*mean*	*rms*	n_T^{aver}	t (sec)
(H$_{\mathrm{MQ}}$)	20	10^5	60	0.4	$15.6\,m$	$1.57\,m$	$2.26\,m$	14.9	65.9
(H$_{\mathrm{MQ}}$)	20	10^4	20	0.4	$17.9\,m$	$1.71\,m$	$2.47\,m$	8.9	15.1
(H$_{\mathrm{MQ}}$)	40	10^5	20	0.4	$9.9\,m$	$0.57\,m$	$0.92\,m$	12.4	61.6
(P)	20		60		$18.7\,m$	$1.95\,m$	$2.78\,m$		15.9

Table 4: Glacier test.

the QINSy software. The processing of this data reported in [7, Section 6.6] involves two steps, where the method (P) is applied twice with different parameters values. The goal of the first step is data cleaning, where outliers are removed and noise reduced. In this paper we skip this first step and work directly with the "cleaned" data obtained in [7], which is a subset of the original data and consists of 621624 points.

As in [7], we use for this data the C^1 spline method Q_1^{av} in the second stage. The parameters of both (H$_{\mathrm{MQ}}$) used here and (P) used in [7] are reported in Table 5, whereas the results are shown in Table 6 and Figures 10 and 11. Note that for this data the value $q = 1$ turned out to be advantageous, which was not the case in any of the previous tests, where we always used $q = 0$ with (H$_{\mathrm{MQ}}$). Moreover, some 9.9% of local approximations, namely 13579 out of 136840, are pure polynomial, of whose 10935 are linear and 2644 are constant.

method	$n_x \times n_y$	M_{min}	M_{max}	q	κ_P	κ_H	δ
(H$_{\mathrm{MQ}}$)	100×281	3	100	1	100	$2 \cdot 10^4$	0.4
(P)	100×281	3	49	3	5		

Table 5: Rotterdam Port test: parameters.

method	*max*	*mean*	*rms*	n_T^{aver}	t (sec)
(H$_{\mathrm{MQ}}$)	$92.5\,cm$	$5.46\,cm$	$7.46\,cm$	5.4	525
(P)	$107.7\,cm$	$5.76\,cm$	$7.84\,cm$		115

Table 6: Rotterdam Port test: results.

Figure 10 presents a view of the full spline surface obtained with (H$_{\mathrm{MQ}}$). Since it is hardly distinguishable from the corresponding figure [7, Figure 17c] for the method (P), we also include Figure 11, where zooms into identical subareas of both surfaces are shown for comparison. The grid used for the visualization is 601 × 1630.

The presented results show that, even if the computational time is almost multiplied by five, the hybrid approach is capable to reduce the errors even in this test case, while keeping the same visual quality obtained with the original approach considered in [7]. Note that the accuracy of the measurements is estimated at between 5 and 10 *cm*, and so there is not much room for the improvement of, say, the *rms* error of 7.46 *cm* that we have for (H_{MQ}).

Acknowledgements

The authors are grateful to CNR which partially supported this work and to the referees for their useful comments on an earlier version of this paper.

References

[1] M. Bozzini, L. Lenarduzzi, 2003, Bivariate knot selection procedure and interpolation perturbed in scale and shape, in Curve and Surface Design: Saint Malo 2002, T. Lyche, M. L. Mazure, L. L. Schumaker (eds), 1–10.

[2] M. D. Buhmann, 2003, Radial Basis Functions, Cambridge Univ. Press.

[3] M. D. Buhmann, O. Davydov, T. N. T. Goodman, 2003, Cubic spline prewavelets on the four–directional mesh, Found. Comp. Math. 3, 113–133.

[4] C. K. Chui, Multivariate Splines, 1988, CBMS-NSF Reg. Conf. Series in Appl. Math., vol. 54, SIAM, Philadelphia.

[5] O. Davydov, 2002, On the approximation power of local least squares polynomials, in Algorithms for Approximation IV, J. Levesley, I. J. Anderson and J. C. Mason (eds), 346–353.

[6] O. Davydov, P. Petrushev, 2003, Nonlinear approximation from differentiable piecewise polynomials, SIAM J. Math. Anal. 35, 708–758.

[7] O. Davydov, F. Zeilfelder, 2004, Scattered data fitting by direct extension of local polynomials to bivariate splines, Adv. Comp. Math.21, 223–271.

[8] O. Davydov, F. Zeilfelder, Toolbox for Two-Stage Scattered Data Fitting, in preparation.

[9] R. T. Farouki, T. N. T. Goodman, 1996, On the optimal stability of the Bernstein bases, Math. Comp.65, 1553–1566.

[10] G. Farin, 1993, Curves and Surfaces for Computer Aided Geometric Design, Academic Press, San Diego.

[11] T. A. Foley, 1987, Interpolation and approximation of 3–D and 4–D scattered data, Comput. & Math. with Appls. 13, 711-740.

[12] R. Franke, 1979, A critical comparison of some methods for interpolation of scattered data, Report NPS-53-79-003, Naval Postgraduate School.

[13] R. Franke, 1982, Smooth interpolation of scattered data by local thin plate splines, Comp. & Maths. with Appls. 8, 273–281.

[14] R. Franke, Homepage, `http://www.math.nps.navy.mil/∼rfranke/`, Naval Postgraduate School.

[15] R. Franke, H. Hagen, G. M. Nielson, 1994, Least squares surface approximation to scattered data using multiquadric functions, Adv. Comp. Math.2, 81–99.

[16] J. Haber, F. Zeilfelder, O. Davydov, H.-P. Seidel, 2001, Smooth approximation and rendering of large scattered data sets, in Proceedings of IEEE Visualisation 2001, T. Ertl, K. Joy and A. Varshney (eds), 341–347, 571.

[17] R. L. Hardy, 1990, Theory and applications of the multiquadric–biharmonic method: 20 years of discovery 1968–1988, Comp. & Maths. with Appls. 19, 163–208.

[18] A. Iske, 1999, Reconstruction of smooth signals from irregular samples by using radial basis function approximation, in Proceedings of the 1999 International Workshop on Sampling Theory and Applications, Y. Lyubarskii (ed), 82–87.

[19] E. J. Kansa, R. E. Carlson, 1992, Improved accuracy of multiquadric interpolation using variable shape parameters, Comp. & Maths. with Appls. 24, 99–120.

[20] D. Lazzaro, L. B. Montefusco, 2002, Radial basis functions for the multivariate interpolation of large scattered data sets, J. Comput. Appl. Math. 140, 521–536.

[21] J. R. McMahon, R. Franke, 1992, Knot selection for least squares thin plate splines, SIAM J. Sci. Stat. Comput. 13, 484–498.

[22] C. A. Micchelli, 1986, Interpolation of scattered data: distance matrices and conditionally positive definite functions, Constr. Approx.2, 11–22.

[23] R. Morandi, A. Sestini, 2002, Geometric knot selection for radial scattered data approximation, in Algorithms for Approximation IV, J. Levesley, I. J. Anderson and J. C. Mason (eds), 244–251.

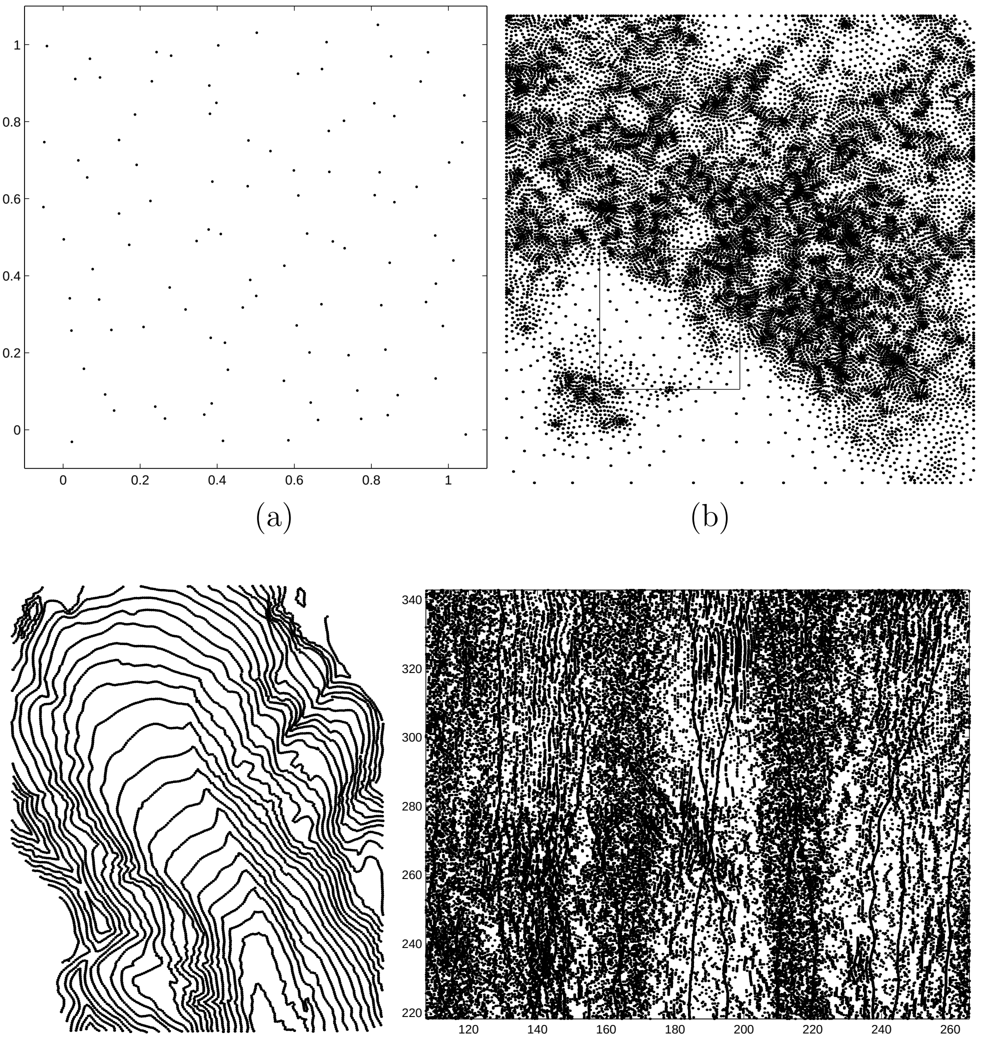
[24] W. Press, S. Teukolsky, W. Vetterling, B. Flannery, 1992, Numerical Recipes in C, 2nd edn. Cambridge University Press, Cambridge.

[25] R. J. Renka, R. Brown, 1999, Algorithm 792: accuracy tests of ACM algorithms for interpolation of scattered data in the plane, ACM Trans. Math. Softw.25, 78–94.

[26] G. W. Stewart, 1998, Matrix Algorithms. Volume I: Basic Decompositions, SIAM, Philadelphia.

[27] H. Wendland, 1995, Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree, Adv. Comp. Math.4, 389–396.

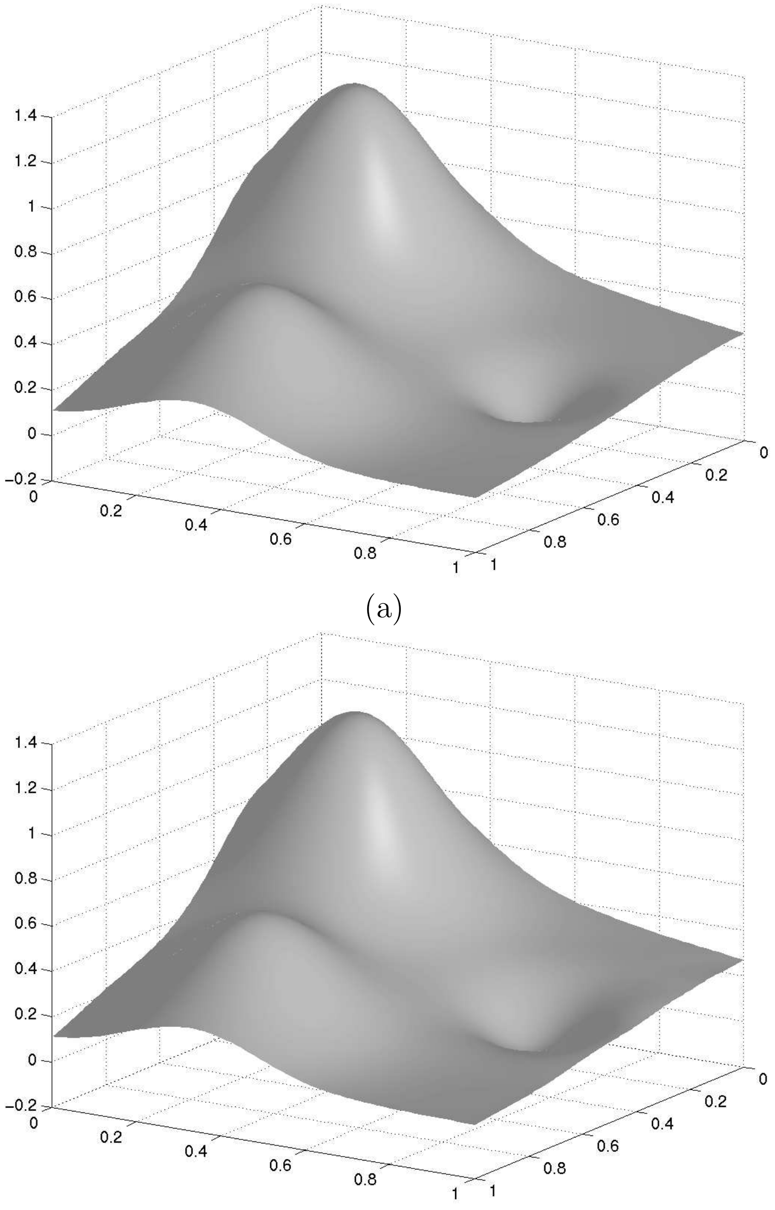
[28] H. Wendland, 2002, Fast evaluation of radial basis functions: methods based on partition of unity, in Approximation Theory X, C. K. Chui, L. L. Schumaker, J. Stöckler (eds), Vanderbilt Univ. Press, Nashville, TN, 473–483.



(a) (b)

(c) (d)

Figure 3: (a) The locations of the 100 data of `ds3` used for the Franke function test. (b) The locations of the 15885 data used for the "black forest" test. (c) The locations of the 8345 data used for the "glacier" test. (d) Typical distribution of the locations of the 621624 data used for the "Rotterdam Port" test. (About 6% of the domain covered by the data is shown.)



(a)

(b)

Figure 4: Franke function test: (a) The exact Franke function. (b) The C^2 spline appoximation obtained with $(\mathrm{H}_{\mathrm{MQ}})$.

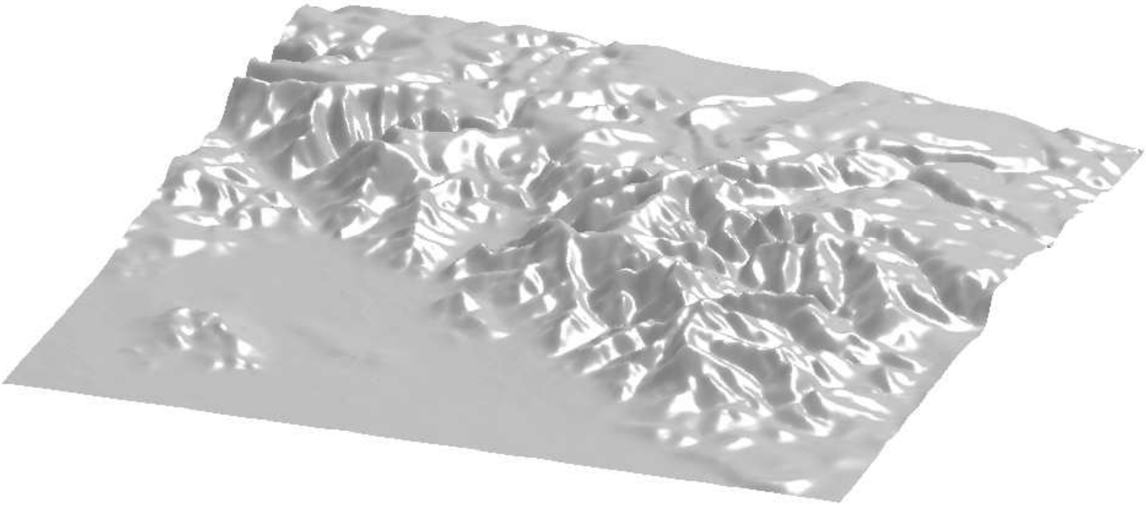
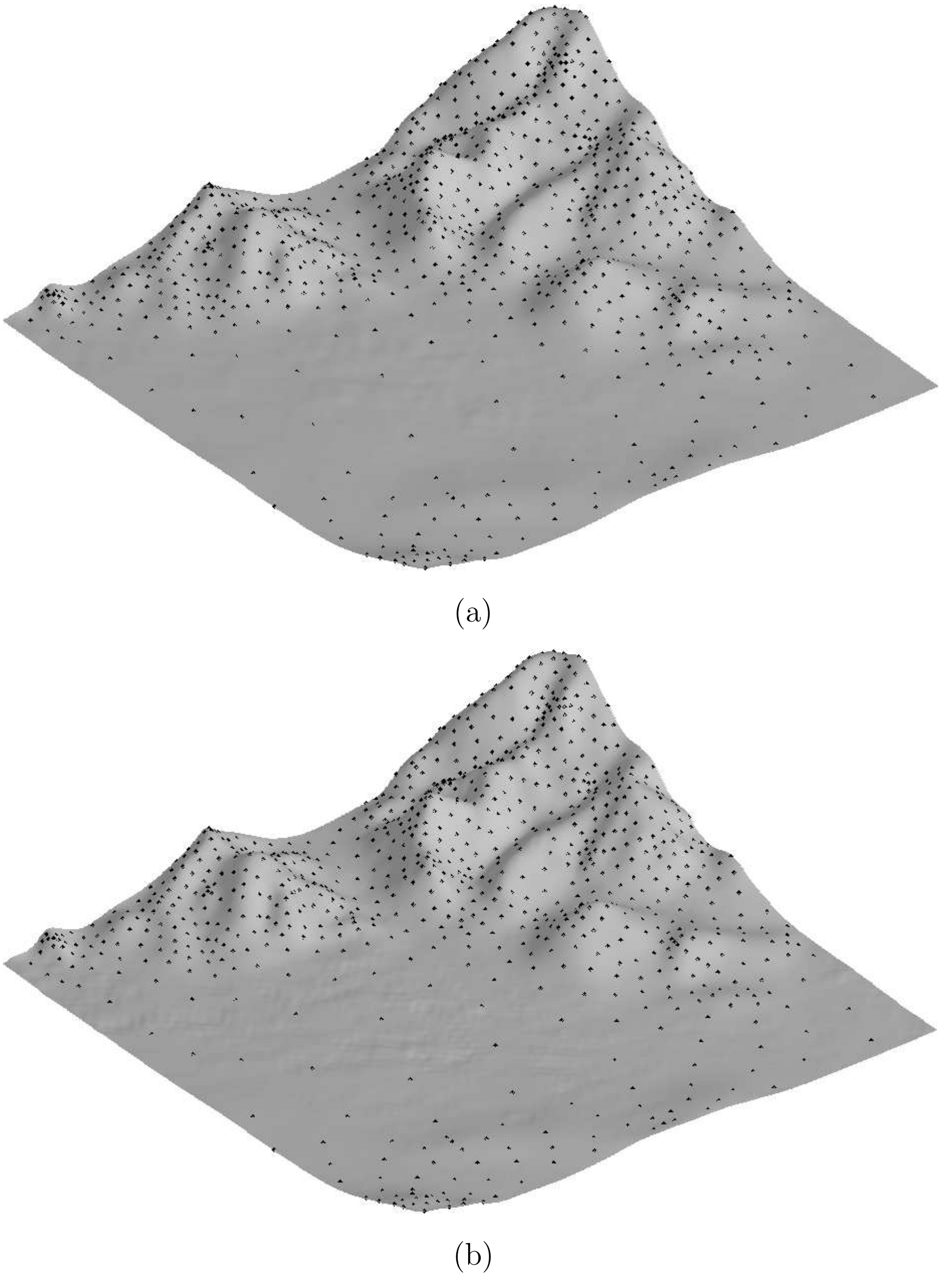


Figure 5: Black forest test: The C^2 spline approximation with $n_x = n_y = 80$ obtained with the hybrid scheme (H_{MQ}).



(a)

(b)

Figure 6: Black forest test: zoom into the subarea indicated with a box in Figure 3b. (a) Approximation with $n_x = n_y = 80$. (b) Approximation with $n_x = n_y = 160$.

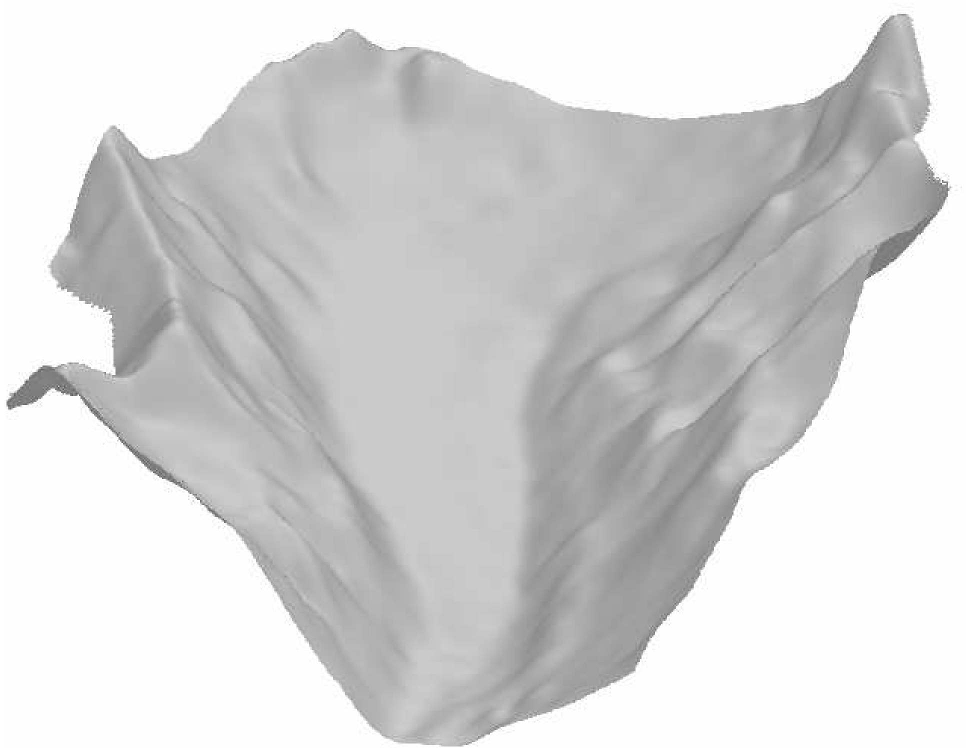
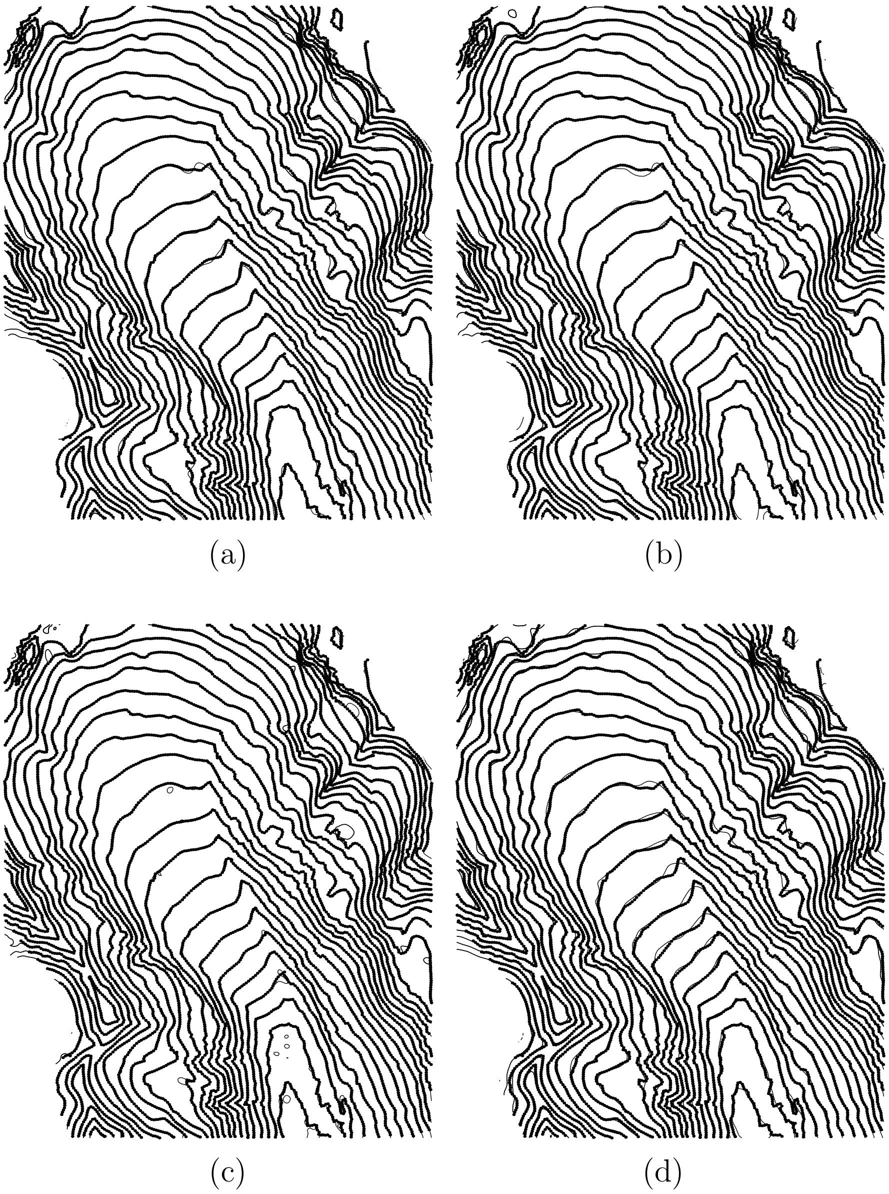


Figure 7: Glacier test: The C^2 spline approximation with $n_x = 20$, $n_y = 24$ obtained with the hybrid scheme (H_{MQ}).



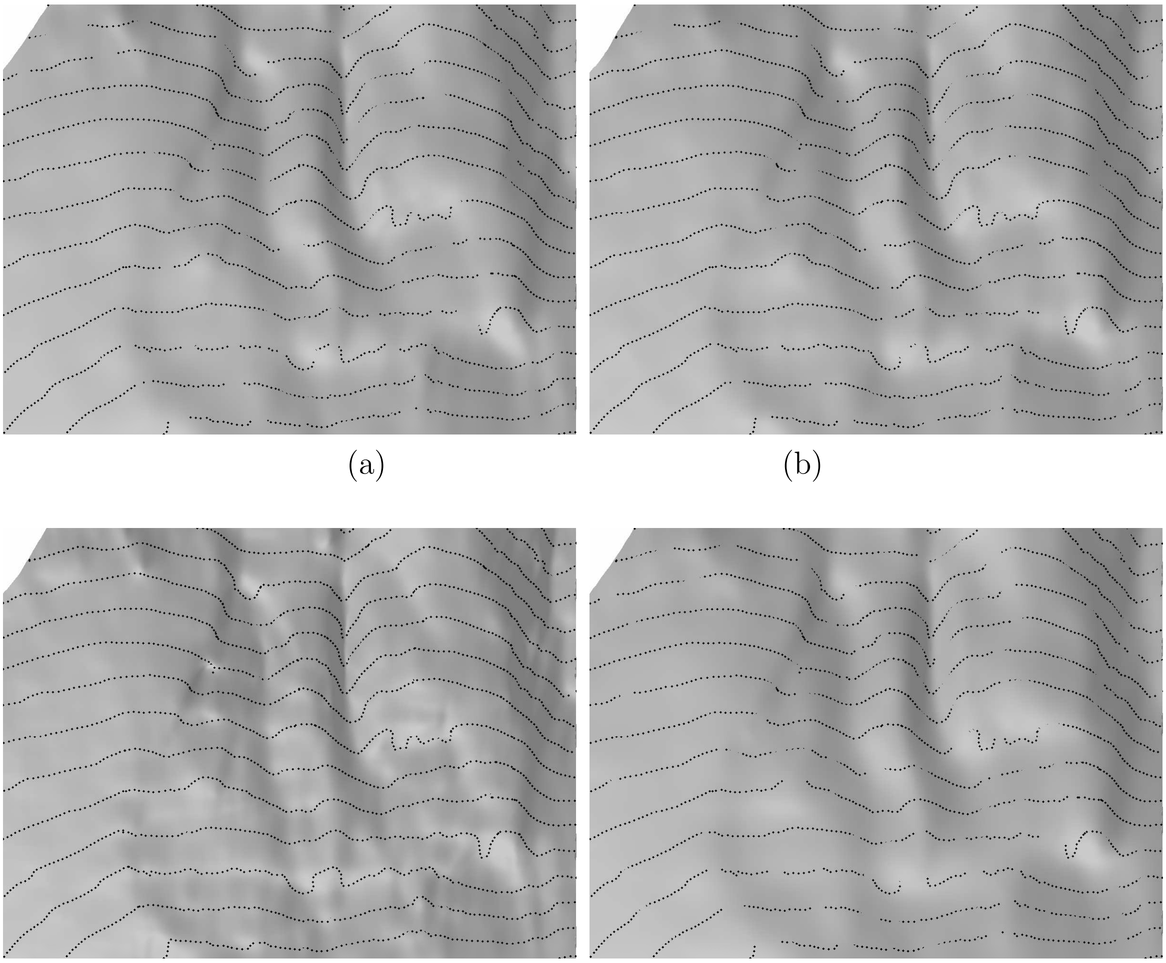
(a)

(b)

(c)

(d)

Figure 8: Glacier test: contours. The thick lines consist of the dots representing the data locations, compare Figure 3c, whereas the thin lines are the contours derived from the approximating surface at precisely the heights of the data. (a) ($\mathrm{H_{MQ}}$) with $n_x = 20, n_y = 24$, $M_{min} = 60$. (b) ($\mathrm{H_{MQ}}$) with $n_x = 20, n_y = 24$, $M_{min} = 20$. (c) ($\mathrm{H_{MQ}}$) with $n_x = 40, n_y = 48$, $M_{min} = 20$. (d) (P) with $n_x = 20, n_y = 24$, $M_{min} = 60$ [7].



(a)

(b)

(c)

(d)

Figure 9: Glacier test: zooms to a subregion. (a) Method ($\mathrm{H_{MQ}}$) with $n_x = 20, n_y = 24$, $M_{min} = 60$. (b) Method ($\mathrm{H_{MQ}}$) with $n_x = 20, n_y = 24$, $M_{min} = 20$. (c) Method ($\mathrm{H_{MQ}}$) with $n_x = 40, n_y = 48$, $M_{min} = 20$. (d) Method (P) with $n_x = 20, n_y = 24$, $M_{min} = 60$ [7].

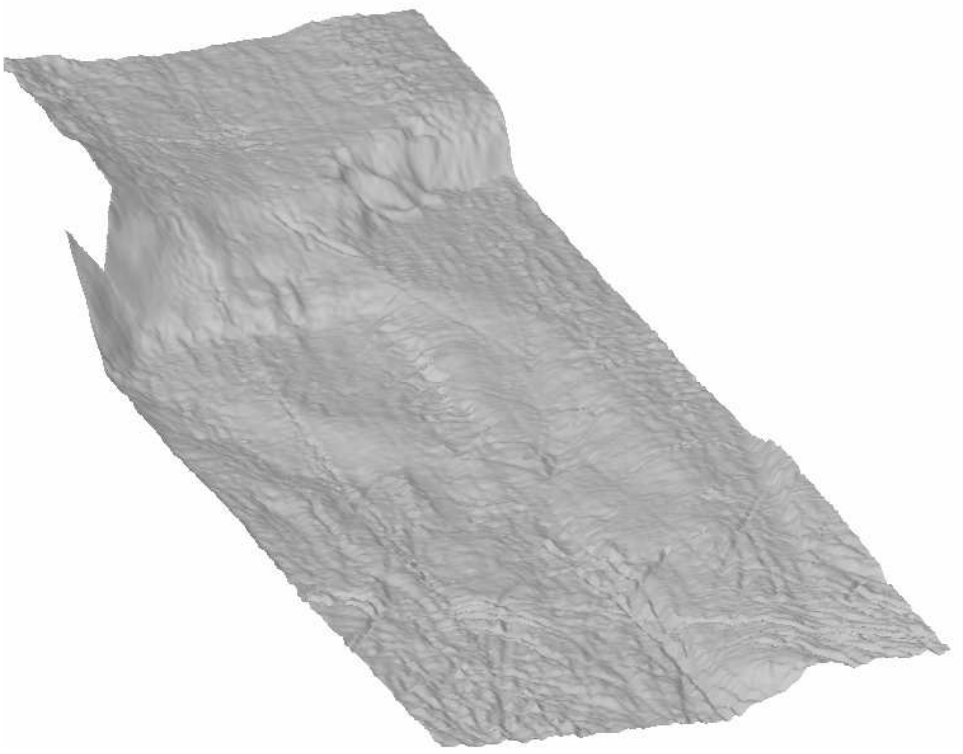
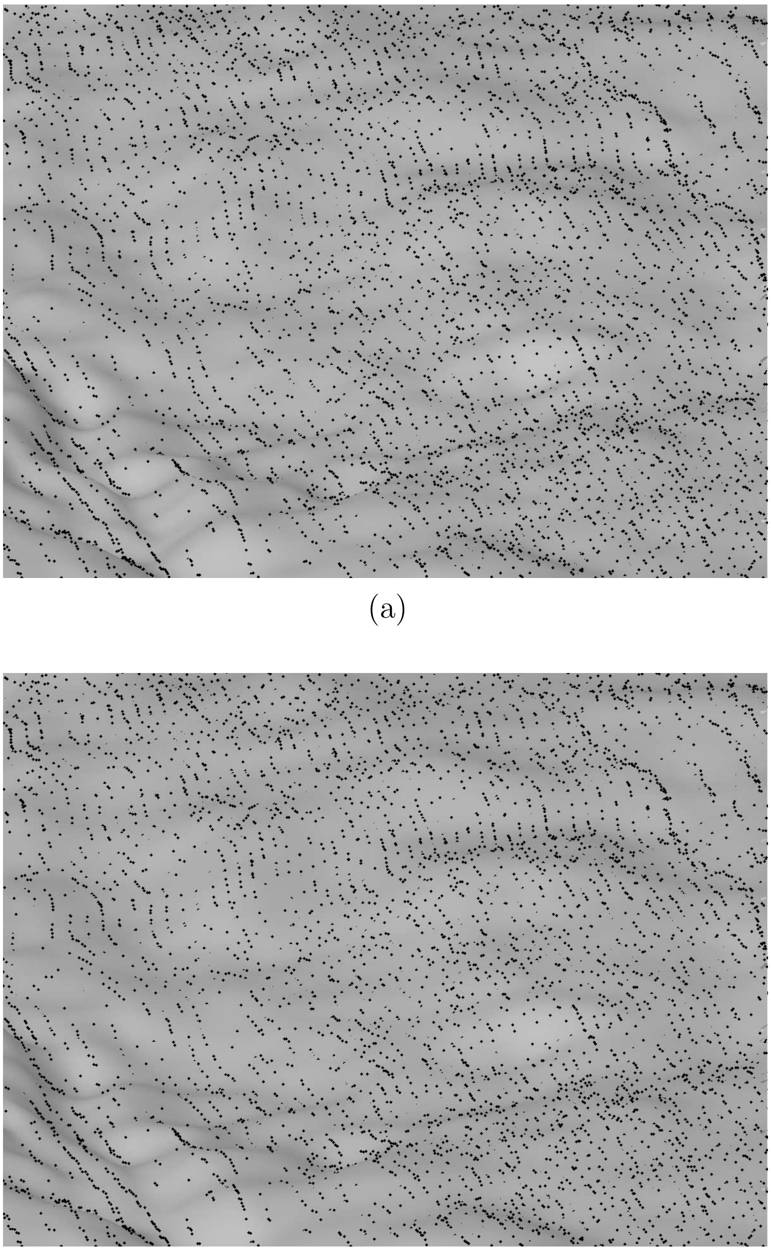


Figure 10: Rotterdam Port test: The C^1 spline approximation obtained with the hybrid method (H_{MQ}).



(a)

(b)

Figure 11: Rotterdam Port test: zoom. (a) Method ($\mathrm{H_{MQ}}$). (b) Method (P).