Detection of Discontinuities in Scattered Data Approximation

Tim Gutzmer and Armin Iske

Abstract. A Detection Algorithm for the localisation of unknown fault lines of a surface from scattered data is given. The method is based on a local approximation scheme using thin plate splines, and we show that this yields approximation of second order accuracy instead of first order as in the global case. Furthermore, the Detection Algorithm works with triangulation methods, and we show their utility for the approximation of the fault lines. The output of our method provides polygonal curves which can be used for the purpose of constrained surface approximation.

1 Introduction

Feature recognition has become an attractive field for research especially within industrial applications. A feature of a surface $f : \mathbf{R}^2 \to \mathbf{R}$ typically reflects characteristic properties of f, such as discontinuities across planar curves. In geophysical sciences these discontinuities are referred to as *fault lines* [1], [20], [9]. The motivation for this work is given by applications from oil industry where methods are required for the detection of fault lines from scattered data, as far as possible automatically. For the purpose of this paper we assume we are given finitely many scattered samples $f(x_1), \ldots, f(x_N)$ of f, and in addition we do not make any assumptions on the distribution of the points from the set $X = \{x_1, \ldots, x_N\}$ except for that they must not all lie on a straight line.
We suggest a *Detection Algorithm* for the approximation of the unknown fault lines of f in an open, bounded domain $\Omega \subset \mathbf{R}^2$ which is assumed to be a superset of the convex hull $\text{conv}(X)$ of X. Additionally, for reasons of simplicity, Ω is required to be simply connected. The method generates polygonal curves which approximate the fault lines.
This work is to be considered in the context of constrained surface approximation (see [1], [20]). The output of the Detection Algorithm can directly be used as input for the method of Arge & Floater [1] where the fault lines are required to be known beforehand.
As to related work, in [12] fault lines are determined by using a divergence property of sequences measuring gradients near discontinuities. Recovering functions with discontinuities sampled on a cartesian grid are studied in [2], [3]; an error minimising model is used to construct an approximation which is allowed to be discontinuous across some prescribed jump lines in one of the coordinate directions. A more general treatment on modelling fault lines for surfaces in a variety of situations can be found in [8].

The Detection Algorithm given by this paper works locally and is divided into two steps:

(1) The localisation of *fault points* from X which are near a fault line. Here we use a *local* interpolation scheme involving thin plate splines which are already well-known for their good approximation behaviour for the global interpolation of scattered bivariate data. The decision for the detection is based on the Gibbs phenomenon (cf. [21]), according to which smooth approximating functions overshoot jumps of the function to be approximated. Details about the thin plate spline interpolation scheme as well as further analysis of its local approximation behaviour is the subject of Section 2.

(2) For each fault point x, we compute a local approximation of the (expected) fault line in the neighbourhood of x. The local approximation is given by a polygonal curve.

Within the above two steps we will work with triangulation methods. The required material on this will be provided by Section 3, and a more detailed discussion of the Detection Algorithm can be found in Section 4. Finally, Section 5 contains selected examples which illustrate the efficacy of our method.

2 Local Approximation

For interpolation of multivariate scattered data radial basis function techniques are well established; for a general survey on this subject we refer the reader to [13], [16], [17]. In this paper we work with thin plate splines that are optimal in a Beppo Levi space equipped with a seminorm that represents the bending energy of a thin plate of infinite extent. We briefly review the basic features of this particular interpolation scheme, before we discuss its local approximation behaviour.
Let f be an unknown function $\mathbf{R}^2 \to \mathbf{R}$ and $X = \{x_1, \ldots, x_N\} \subset \mathbf{R}^2$ a set of $N \geq 3$ pairwise distinct points which are not collinear. Moreover, assume we are given a data vector $T_X f = (f(x_1), \ldots, f(x_N))^T \in \mathbf{R}^N$ containing function values sampled from f at the data points.
By the thin plate spline interpolation scheme, the solution to the interpolation problem

$$f(x_i) = s(x_i) \quad 1 \leq i \leq N \tag{1}$$

is given by a linear combination of translates of the basis function $\Phi(x) = \phi(\|x\|_2)$ with $\phi(r) = r^2 \log(r)$ for $r \geq 0$ exceeded by a linear polynomial, such that the *thin plate spline interpolant* $s : \mathbf{R}^2 \to \mathbf{R}$ has the form

$$s(x) = \sum_{i=1}^{N} \lambda_i \phi(\|x - x_i\|_2) + \lambda_{N+1} + \lambda_{N+2}\xi + \lambda_{N+3}\eta \tag{2}$$

where $x = (\xi, \eta)^T$. In addition, the coefficients $\lambda_1, \ldots, \lambda_{N+3}$ are subject to constraints

$$\sum_{i=1}^{N} \lambda_i = 0 \quad \text{and} \quad \sum_{i=1}^{N} \lambda_i x_i = 0. \tag{3}$$

Combining (1) with (2) and (3), the unknown vector $\lambda = (\lambda_1, \ldots, \lambda_{N+3})^T \in \mathbf{R}^{N+3}$ is to be computed by solving the linear system

$$\begin{pmatrix} A & P \\ P^T & 0 \end{pmatrix} \cdot \lambda = \begin{pmatrix} T_X f \\ 0 \end{pmatrix}.$$

where $A = (\phi(\|x_i - x_j\|_2))_{1 \leq i,j \leq N}$, $P = (p_k(x_i))_{\substack{1 \leq i \leq N \\ 1 \leq k \leq 3}}$ with $p_1(x) \equiv 1, p_2(x) = \xi, p_3(x) = \eta$.

As it has been proven by Micchelli in his fundamental paper [11], the thin plate spline interpolation scheme provides for all possible choices of non-collinear pairwise distinct data points $x_1, \ldots, x_N$ and real function values $f(x_1), \ldots, f(x_N)$ a unique solution of the above form. In particular, there is a Lagrange-type representation

$$s(x) = \sum_{i=1}^{N} u_i(x) f(x_i) \tag{4}$$

of the thin plate spline interpolant in terms of *cardinal functions* $u_i : \mathbf{R}^2 \to \mathbf{R}$, $1 \leq i \leq N$, whose pointwise evaluation at a fixed $x \in \mathbf{R}^2$ is given by the solution of the linear system

$$\begin{pmatrix} A & P \\ P^T & 0 \end{pmatrix} \cdot \begin{pmatrix} (u_i(x))_{1 \leq i \leq N} \\ (v_k(x))_{1 \leq k \leq 3} \end{pmatrix} = \begin{pmatrix} (\phi(\|x - x_j\|_2))_{1 \leq j \leq N} \\ (p_k(x))_{1 \leq k \leq 3} \end{pmatrix}. \tag{5}$$

Indeed, by inserting $x = x_j$, $1 \le j \le N$, into the above system (5), its right hand side will coincide with the j-th column of its collocation matrix and thus we obtain $u_i(x_j) = \delta_{i,j}$ for $1 \le i, j \le N$ as required. As it follows directly from (5),

$$p_k(x) = \sum_{i=1}^{N} u_i(x) p_k(x_i) \quad 1 \le k \le 3 \tag{6}$$

holds true, i.e. the thin plate spline interpolation scheme reproduces linear polynomials. In the context of this paper, the functions v_1, v_2, v_3 from the above system (5) are not significant, and therefore we omit further details here. Instead of this, we refer the reader to the work [22] of Wu & Schaback (see also the survey [17]) where among other things the variational formulation of the interpolation problem is discussed.

In the subsequent of this section, we are interested in analysing the local approximation behaviour of the thin plate spline interpolation scheme. For this purpose, we fix one point $x_0 \in \mathbf{R}^2$, and consider interpolation problems of the form

$$f(x_0 + hx_i) = s^h(x_0 + hx_i) \quad 1 \le i \le N \tag{7}$$

where $h > 0$ is a positive scaling factor and s^h is the thin plate spline interpolant to (7). Since (7) and thus its unique solution s^h varies with the quantity of the scale h, we obtain a whole family $\{s^h\}_{h>0}$ of thin plate spline interpolants. This enables us to measure the approximation quality of the interpolation scheme locally around x_0 by analysing the effect of the scaling according to the following definition.

Definition 2.1: *Let $X = \{x_1, \ldots, x_N\} \subset \mathbf{R}^2$ denote a set of N pairwise distinct points which are not collinear. For a fixed $x_0 \in \mathbf{R}^2$ and $h > 0$, let s^h denote the thin plate spline interpolant satisfying (7) for a sufficiently smooth function $f : \mathbf{R}^2 \to \mathbf{R}$. We say that the* `local approximation order` *of the thin plate spline interpolation scheme at x_0 is p, iff*

$$|f(x_0 + hx) - s^h(x_0 + hx)| = \mathcal{O}(h^p), \; h \to 0$$

holds for every $x = \sum_{i=1}^N \alpha_i x_i$ with $\sum_{i=1}^N \alpha_i = 1$.

Using an earlier approximation result given in [14] by Powell, we deduce a lower bound for the local approximation order according to Definition 2.1. Powell proves in [14], Lemma 2.3

Proposition 2.2: *Let $f : \mathbf{R}^2 \to \mathbf{R}$ be a function that has square integrable second derivatives. Let*

$$I(f) = \int_{\mathbf{R}^2} \left(\frac{\partial^2 f}{\partial \xi^2}\right)^2 + 2\left(\frac{\partial^2 f}{\partial \xi \partial \eta}\right)^2 + \left(\frac{\partial^2 f}{\partial \eta^2}\right)^2 < \infty$$

and $x = \sum_{i=1}^N \alpha_i x_i$ where the multipliers α_i sum to 1.
Then the error of the thin plate spline interpolant at x is bounded above by the expression

$$|f(x) - s(x)| \le \left[(8\pi)^{-1} I(f) \Phi(\alpha)\right]^{1/2},$$

where $\Phi(\alpha)$ is the number

$$\Phi(\alpha) = \sum_{i,j=1}^{N} \alpha_i \alpha_j \phi(\|x_i - x_j\|_2) - 2\sum_{i=1}^{N} \alpha_i \phi(\|x - x_i\|_2). \tag{8}$$

Note that if Ω is any bounded set in $\mathbf{R}^2$ and $X \subset \Omega$, the result of Proposition 2.2 gives us essentially an $\mathcal{O}(h_\delta)$ error bound where in the situation of [14]

$$h_\delta = \sup_{y \in \Omega} \min\{\|y - x_i\|_2 \,:\, 1 \le i \le N\}$$

measures the density of the interpolation points. As already pointed out in [14], numerical experiments have shown that edge effects usually prevent the achievement of a higher order.

By combining the following two observations, we conclude that from the result in Proposition 2.2 the local approximation order according to Definition 2.1 is at any $x_0 \in \mathbf{R}^2$ bounded below by one.

- The expression $\Phi(\alpha)$ in (8) is invariant under translations, i.e. if we replace x, x_i, $i = 1, \ldots, N$ by $x_0 + x$, $x_0 + x_i$, $i = 1, \ldots, N$ for some $x_0 \in \mathbf{R}^2$, but do not change the parameters α_i, $i = 1, \ldots, N$, then the value of $\Phi(\alpha)$ does not change.
- If we replace for a positive scaling factor h the points x, x_i, $i = 1, \ldots, N$ by hx, hx_i, $i = 1, \ldots, N$ but do not change the parameters α_i, $i = 1, \ldots, N$, then $\Phi(\alpha)$ is scaled with factor h^2 (see [14], Lemma 2.4).

By using the cardinal representation (4) of the thin plate spline interpolant we can even prove order two.

Proposition 2.3: *Let X be as in Definition 2.1, and let $f \in \mathcal{C}^2(\mathbf{R}^2, \mathbf{R})$. Then, for arbitrary $x_0 \in \mathbf{R}^2$*

$$|f(x_0 + hx) - s^h(x_0 + hx)| = \mathcal{O}(h^2), \; h \to 0$$

holds for every $x = \sum_{i=1}^N \alpha_i x_i$ with $\sum_{i=1}^N \alpha_i = 1$, i.e. the local approximation order of the thin plate spline interpolation scheme is two at every $x_0 \in \mathbf{R}^2$.

Proof: Due to the invariance of the thin plate spline interpolation scheme under translations we assume $x_0 = 0$ without loss of generality. Therefore, by rewriting (7) accordingly we restrict ourselves to interpolation problems of the form

$$s^h(hx_i) = f(hx_i) \quad 1 \le i \le N. \tag{9}$$

Following along the lines of the argumentation from the beginning of this section we see that for a fixed $h > 0$ the thin plate spline interpolant s^h satisfying (9) can be expressed in terms of cardinal functions $u_i^h(hx)$, $1 \le i \le N$, such that

$$s^h(hx) = \sum_{i=1}^N u_i^h(hx) f(hx_i)$$

with $u_i^h(hx_j) = \delta_{i,j}$ for $1 \le i, j \le N$. Furthermore, for $x \in \mathbf{R}^2$ the evaluation of each function u_i^h at hx is given by the solution of the linear system

$$\begin{pmatrix} A_h & P_h \\ P_h^T & 0 \end{pmatrix} \begin{pmatrix} (u_i^h(hx))_{i=1,\ldots,N} \\ (v_k^h(hx))_{k=1,2,3} \end{pmatrix} = \begin{pmatrix} (\phi(\|hx - hx_j\|_2))_{j=1,\ldots,N} \\ (p_k(hx))_{k=1,2,3} \end{pmatrix}, \tag{10}$$

where $A_h = (\phi(\|hx_i - hx_j\|_2))_{1 \le i,j \le N}$, $P_h = (p_k(hx_i))_{\substack{1 \le i \le N \\ 1 \le k \le 3}}$.

If we let the polynomial $p^h : \mathbf{R}^2 \to \mathbf{R}$ be given by

$$p^h(y) = f(hx) + \nabla f(hx)^T (y - hx),$$

we obtain

$$f(hx) - s^h(hx) = \sum_{i=1}^{N} u_i^h(hx) \underbrace{\left[p^h(hx_i) - f(hx_i)\right]}_{\mathcal{O}(h^2),\ h\to 0} \tag{11}$$

for the error of the interpolation at hx.

In the subsequent, we will show that

$$u_i^h(hx) = u_i(x) \quad 1 \le i \le N \tag{12}$$

holds true for every $h > 0$, i.e. the Lebesque constants are bounded. By using (11) this implies

$$|f(hx) - s^h(hx)| = \mathcal{O}(h^2),\ h \to 0$$

which will then complete our proof.
For proving (12), we let $h = 1$ for the moment and expand for an index $1 \le j \le N$ the j-th row of the above system (10):

$$\sum_{i=1}^{N} u_i^1(x)\phi(\|x_i - x_j\|_2) + \sum_{k=1}^{3} v_k^1(x)p_k(x_j) = \phi(\|x - x_j\|_2). \tag{13}$$

Using (6) and by observing $u_i^1 = u_i$ for all $1 \le i \le N$ we find

$$h^2 \log(h) \sum_{i=1}^{N} u_i^1(x)(\|x_j\|_2^2 - 2x_j^T x_i) = h^2 \log(h)(\|x_j\|_2^2 - 2x_j^T x). \tag{14}$$

Combining (13) and (14) gives us

$$\begin{aligned}
&\sum_{i=1}^{N} u_i^1(x) \left(h^2\phi(\|x_i - x_j\|_2) + h^2 \log(h)(\|x_j\|_2^2 - 2x_j^T x_i + \|x_i\|_2^2)\right) \\
&\qquad -h^2 \log(h) \sum_{i=1}^{N} u_i^1(x)\|x_i\|_2^2 + h^2 \sum_{k=1}^{3} v_k^1(x)p_k(x_j) \\
&\qquad = \quad h^2\phi(\|x - x_j\|_2) + h^2 \log(h)(\|x_j\|_2^2 - 2x_j^T x).
\end{aligned}$$

Adding $h^2 \log(h)\|x\|_2^2$ on both sides of this equation, we obtain

$$\sum_{i=1}^{N} u_i^1(x)\phi(\|hx_i - hx_j\|_2) + h^2 \log(h) \left(\|x\|_2 - \sum_{i=1}^{N} u_i^1(x)\|x_i\|_2^2\right) + h^2 \sum_{k=1}^{3} v_k^1(x)p_k(x_j) = \phi(\|hx - hx_j\|_2).$$

If we let the functions $\tilde{v}_k^h : \mathbf{R}^2 \to \mathbf{R}$, $1 \le k \le 3$, be defined by

$$\begin{aligned}
\tilde{v}_1^h(x) &= h^2 v_1^1(x) + h^2 \log(h) \left(\|x\|_2^2 - \sum_{i=1}^{N} u_i^1(x)\|x_i\|_2^2\right) \\
\tilde{v}_k^h(x) &= h v_k^1(x), \quad k = 2, 3
\end{aligned}$$

we find that for each $x \in \mathbf{R}^2$ the vector

$$\begin{pmatrix} \left(u_i^1(x)\right)_{1\le i\le N} \\ \left(\tilde{v}_k^h(x)\right)_{1\le k\le 3} \end{pmatrix}$$

solves the linear system (10) for arbitrary $h > 0$. Since the solution of (10) is unique, we conclude (12) as desired which completes our proof. □

Note that for $f \in \mathcal{C}^1(\mathbf{R}^2, \mathbf{R})$ we achieve order one for the local approximation at any $x_0 \in \mathbf{R}^2$ using the same arguments as in the proof above. If f is continuous the local approximation is at least consistent, i.e. $|f(x_0 + hx) - s^h(x_0 + hx)| \to 0,\ h \to 0$.

3 Triangulations

The Detection Algorithm of Section 4 is based on triangulation methods. These incorporate well-known techniques, which are useful especially for applications dealing with planar scattered data. We do not intend to give a review on triangulation methods but rather refer the reader to the works [15], [18], [19] and references therein for an overview. In order to keep this paper self-contained, we will touch in this section a selected excerpt from the material in [18] required for the design of the Detection Algorithm. Furthermore, we will introduce some notation.
For the following we assume that $X = \{x_1, \ldots, x_N\} \subset \mathbf{R}^2$ is a fixed set of finitely many pairwise distinct points in the plane.

Definition 3.1: *A set $\mathcal{T}(X) = \{T_k\}_{k=1}^K$ of nondegenerate closed triangles is a* `triangulation` *of X, if the following conditions hold:*

(1) X is the set of all vertices of the triangles in $\mathcal{T}(X)$;

(2) for two distinct triangles $T_1, T_2 \in \mathcal{T}(X)$ their intersection $T_1 \cap T_2$ is either empty or they share a common edge or they share a common vertex;

(3) the union of $\{T_k\}_{k=1}^K$ is equal to the convex hull of X.

An edge $E = T_1 \cap T_2$ which is shared by two adjacent triangles T_1, T_2 is referred to as `interior edge`*. A vertex x in $\mathcal{T}(X)$ which is shared by interior edges only will be referred to as* `interior vertex` *or* `interior node` *of $\mathcal{T}(X)$.*

The following definition will be used in the next section.

Definition 3.2: *For an interior vertex $x \in X$ of a triangulation $\mathcal{T}(X)$ we denote by*

$$\mathcal{N}_x = \{y \in X \setminus \{x\} \ :\ \textit{there is an edge in } \mathcal{T}(X) \textit{ whose endpoints are } x \textit{ and } y\}$$

the set of `neighbours` *of x, and by*

$$\mathcal{C}_x = \{T \in \mathcal{T}(X) \ :\ \ x \textit{ is one vertex of } T\}$$

the `cell` *of x. Furthermore, $\mathcal{V}_x = \mathcal{N}_x \cup \{x\}$ stands for the set of vertices in $\mathcal{C}_x$.*

Note that in general there exist many triangulations of X, one of which is the *Delaunay triangulation* $\mathcal{D}(X)$ which maximizes the minimum angle in the triangulation (cf. [18]).

4 Detection Algorithm

Assume we are given data points $X = \{x_1, \ldots, x_N\}$ and real function values $f(x_1), \ldots, f(x_N)$ of an unknown function $f : \Omega \to \mathbf{R}$ which is discontinuous across a non-empty set $\Gamma = \{\Gamma_k\}_{k=1}^M$ of *fault lines*

$$\Gamma_k = \{\gamma_k(t) \, : \, t \in [0,1]\} \subset \Omega$$

for unknown curves $\gamma_k \in \mathcal{C}([0,1],\Omega)$, $1 \leq k \leq M$, $M \geq 1$. The function f is supposed to be smooth on $\Omega \setminus \Gamma$, and $\Omega \subset \mathbf{R}^2$ denotes a bounded, open, and simply connected domain which contains the convex hull of X. For reasons of simplicity we require the fault lines to be pairwise disjoint. Moreover, for each curve γ_k we assume that it has no self-intersections, i.e. there is no pair $0 \leq t_1 < t_2 < 1$ such that $\gamma_k(t_1) = \gamma_k(t_2)$. Furthermore, let $\mathcal{T}(X)$ be a triangulation of X.

In this Section we provide a *Detection Algorithm* which approximates the unknown fault lines. The algorithm is divided into two steps:

- **Classification:** Mark all interior nodes of $\mathcal{T}(X)$ which are classified as to be near a fault line and collect these in a set $\mathcal{R} \subset X$.
- **Separation:** For each $x \in \mathcal{R}$ split selected triangles of its cell $\mathcal{C}_x$.

The remainder of this section elaborates these two steps.

4.1 Classification

Recall from the previous section the definition of the set $\mathcal{N}_x$ containing all neighbours of an interior node of a triangulation of X, and note that $\#\mathcal{N}_x \geq 3$ holds. In this step of the Detection Algorithm, we will interpolate f at $\mathcal{N}_x$ for each interior node x by a thin plate spline interpolant $s_{\mathcal{N}_x,f}$ subject to

$$s_{\mathcal{N}_x,f}(y) = f(y) \qquad \text{for all } y \in \mathcal{N}_x$$

and measure the smoothness of f around x by

$$\sigma(x) = |f(x) - s_{\mathcal{N}_x,f}(x)|.$$

Note that σ also depends on f and the triangulation, but for reasons of notational simplicity we drop this here. According to our results of Section 2, the quantity $\sigma(x)$ is expected to be small for those interior nodes x in $\mathcal{T}(X)$, around whose cell $\mathcal{C}_x$ the function f is smooth, whereas tremendous variations of f at $\mathcal{C}_x$ will be captured by a high value for $\sigma(x)$. Indeed, for the special case where the restriction of f to $\mathcal{V}_x$ coincides with a linear function, $\sigma(x)$ will be zero. Moreover, σ can be considered as a function on all interior nodes whose extension to Ω will be zero for a smooth function f. For the purpose of measuring the smoothness around an interior node x by $\sigma(x)$, we prefer to work with the Delaunay triangulation $\mathcal{D}(X)$, since this yields an isotropic distribution of the points in $\mathcal{C}_x$ around x.

The smoothness $\sigma(x)$ of f around a single x is to be seen relative to the corresponding values for all interior nodes. Therefore, we first consider the distribution of the function values $\sigma(x)$ on the interior nodes, before we fix a threshold value $\sigma_0 > 0$ which serves to classify each interior node x according to the smoothness of f around its cell $\mathcal{C}_x$. From this we obtain the set

$$\mathcal{R} = \{x \text{ interior node in } \mathcal{D}(X) \, : \, \sigma(x) > \sigma_0\}\,. \tag{15}$$

4.2 Separation

Due to our assumptions on f and the results on the approximation properties of thin plate spline interpolation of Section 2, we expect that each interior node $x \in \mathcal{R}$ is near a fault line of f. To be more precise, for a reasonable choice of σ_0 we expect that a high value $\sigma(x) > \sigma_0$ indicates a discontinuity of f through the cell $\mathcal{C}_x$, i.e. the intersection $\Gamma_x = \mathcal{C}_x \cap \Gamma$ is supposed to be non-empty. Let $\mathcal{N}_x = \{x_{j_\ell}\}_{\ell=1}^{L_x}$ be the set of its neighbours, ordered such that

$$f(x_{j_1}) \leq f(x_{j_2}) \leq \ldots \leq f(x_{j_{L_x}}).$$

By using the forward differences

$$\Delta f(x_{j_\ell}) = f(x_{j_{\ell+1}}) - f(x_{j_\ell}) \quad \text{for } 1 \leq \ell < L_x$$

we localize the expected jumps of f on $\mathcal{C}_x$ by fixing the index $\ell^* \in \{1, \ldots, L_x - 1\}$ satisfying

$$\ell^* = \min\{1 \leq \ell < L_x : \Delta f(x_{j_\ell}) = \max_{1 \leq \ell < L_x} \Delta f(x_{j_\ell})\}.$$

It is reasonable to assume that Γ_x separates the set

$$\mathcal{H}_x = \{y \in \mathcal{V}_x \,:\, f(y) > f(x_{j_{\ell^*}})\}$$

of all nodes from $\mathcal{C}_x$ with high function values from the set

$$\mathcal{L}_x = \{y \in \mathcal{V}_x \,:\, f(y) \leq f(x_{j_{\ell^*}})\}$$

containing those with low function values. This motivates us to approximate Γ_x by splitting selected triangles of $\mathcal{C}_x$.

Definition 4.1: *For $x \in \mathcal{R}$, let $T \in \mathcal{C}_x$ be a triangle whose set of its three vertices is denoted by V_T. We say that T is* `splittable` *with respect to x, iff*

$$1 \leq \#(V_T \cap \mathcal{H}_x), \#(V_T \cap \mathcal{L}_x) \leq 2,$$

i.e. V_T contains nodes from both $\mathcal{H}_x$ and $\mathcal{L}_x$.
By a `split` *of a splittable triangle $T \in \mathcal{C}_x$ we understand the separation of the points in $V_T \cap \mathcal{H}_x$ from those in $V_T \cap \mathcal{L}_x$ by a straight line between the midpoints of two of its three edges.*

Obviously, the split of a splittable triangle T with respect to any $x \in V_T$ is unique. By $\mathcal{S}_x$ we denote the set of all splittable triangles with respect to $x \in \mathcal{R}$. Figure 1 shows an example of splitting splittable triangles from $\mathcal{S}_x$ for a $x \in \mathcal{R}$ according to the values $f(x) = 9$, $f(x_1) = 10$, $f(x_2) = 1$, $f(x_3) = 7$, $f(x_4) = 0$, $f(x_5) = 8$.

Figure 1: $\mathcal{C}_x$ split in $\mathcal{H}_x = \{x, x_1, x_3, x_5\}$ and $\mathcal{L}_x = \{x_2, x_4\}$ by $\mathcal{P}_x = L(p_1^x, p_2^x, p_3^x) \cup L(p_3^x, p_4^x, p_5^x)$

Now let us consider a pair $x, y \in \mathcal{R}$ of neighbouring nodes and a triangle $T \in \mathcal{S}_x \cap \mathcal{S}_y$ which is splittable with respect to both x and y. From the example given in Figure 2 according to the values $f(x) = 1$, $f(y) = 9$, $f(v_1) = 5$, $f(v_2) = 0$, $f(v_3) = f(v_4) = 1$, $f(v_5) = f(v_6) = 11$, $f(v_7) = 10$ we observe that the split of T with respect to x may be different from its split with respect to y.

Figure 2: non-consistent nodes x and y with respect to T

By the following definition we characterize all splittable triangles admitting an unique split with respect to different vertices contained in $\mathcal{R}$.

Definition 4.2: *Let $\mathcal{R}$ denote the set of interior nodes as defined in (15). We say that a pair $x, y \in \mathcal{R}$ is* `consistent` *with respect to T, iff $T \in \mathcal{S}_x \cap \mathcal{S}_y$ and the following condition holds*

$$\text{Either } z \in \mathcal{H}_x \cap \mathcal{H}_y \text{ or } z \in \mathcal{L}_x \cap \mathcal{L}_y \qquad \text{for all } z \in V_T.$$

We say that a splittable triangle T is `consistent`*, iff all possible pairs of neighbouring vertices $x, y \in \mathcal{R} \cap V_T$ are consistent.*

In our numerical examples we found that consistency of the data turns out to be a quite realistic assumption whenever the data sample is sufficiently dense. In the algorithm we decide to remove a triangle T from the set of splittable triangles if T is not consistent.
All splits of splittable triangles give us a collection $\mathcal{P}$ of straight lines whereas each of these corresponds to a split of one triangle. The concatenation of the connected components of $\mathcal{P}$ provides polygonal curves of the form

$$\mathcal{P}_k = L(p_1^{(k)}, \ldots, p_{M_k}^{(k)}),\ k = 1, \ldots, \hat{M},$$

where $\hat{M}$ denotes the number of connected components and $L(p_1^{(k)}, \ldots, p_{M_k}^{(k)})$ is given by the linear connection of subsequent points $p_i^{(k)}, p_{i+1}^{(k)}$ by a straight line for $1 \leq i < M_k$. Hence $\mathcal{P}_k,\ k = 1, \ldots, \hat{M}$ serve to approximate the unknown fault lines $\Gamma_k,\ k = 1, \ldots, M$ of f in Ω.

5 Numerical Results

In this section we present the results yielded by the Detection Algorithm for two examples. In a first example the detection algorithm is applied to f given by

$$f(x,y) = \begin{cases} 1 + 2\lfloor 3.5\, \|(x,y)^T\|_2 \rfloor & : \quad \|(x-0.5, y-0.5)^T\|_2 < 0.4 \\ 0 & : \quad \text{else} \end{cases}$$

which was sampled at 2000 randomly scattered data points in the unit square. In the definition of f, the symbol $\lfloor z \rfloor$ stands for the integer n satisfying $n \leq z < n+1$ for $z \in \mathbf{R}$.

Figure 4 shows the set $\mathcal{R}$ of marked points as computed by the first step of the Detection Algorithm. Figure 5 displays the collection of detected fault lines provided by step 2. The separation of the interior fault lines from the exterior fault line in Figure 5 is due to the treatment of non-consistent triangles by the algorithm described in Section 4.

Figure 5: reconstructed fault lines for f
A second example uses a modified data set of a North Sea ground region. A gridded sample of the surface is displayed in Figure 6. This *Geo* surface was sampled at 2500 scattered locations as input for the Detection Algorithm. The result of both steps of the Detection Algorithm for this example is reflected by Figure 7. The oscillations of the fault lines to be seen in the Figures 7 and 8 are due to boundary effects. At these locations the Delaunay triangulation yields long and thin triangles that are split by the algorithm.

As a final remark, we have also implemented the Detection Algorithm using data-dependent triangulations. In particular, we have been working with both the ABN (angle between normals) and the JND (jump in normal derivatives) criterion for generating locally optimal triangulations according to the *local optimisation procedure* as described in [10]. For more details on data-dependent triangulations we recommend the reader to consult the papers [4], [5], [6], [7].
However, due to our numerical observations from the test examples in this section and several others that do not appear in this paper, this did not significantly improve the approximation behaviour. Therefore, we decided to drop these examples for the presentation of this paper.

6 Conclusion

In this paper, we presented an algorithm for the detection of fault lines of a surface from scattered data. Under reasonable assumptions on the given data, the method generates polygonal curves which approximate the unknown discontinuities. The only user interaction required in the algorithm is the choice for the threshold parameter σ_0 in (15).
Based on a local approximation scheme using thin plate splines, combined with triangulation methods, the algorithm is stable. Our numerical examples illustrate the efficacy of the method, and moreover show the utility of the Delaunay triangulation for the approximation process.

Acknowledgements. The authors would like to thank the anonymous referees for their useful comments which helped to improve the presentation of this paper.

References

[1] E. Arge and M. Floater, Approximating scattered data with discontinuities, *Numerical Algorithms* **8**, 1994, 149-166.

[2] M. Bozzini and L. Lenarduzzi, Recovering a Function with Discontinuities in Dimension Two, in *Mathematical Methods for Curves and Surfaces*, M. Dæhlen, T. Lyche and L.L. Schumaker (eds.), Vanderbilt University Press, Nashville, 1995, 41-48.

[3] M. Bozzini and L. Lenarduzzi, Recovering a Function with Discontinuities from Correlated Data, in *Advanced Topics in Multivariate Approximation*, F. Fontanella, K. Jetter and P.J. Laurent (eds.), World Scientific Publishing, Singapore, 1996, 1-16.

[4] N. Dyn, D. Levin and S. Rippa, Data Dependent Triangulations for Piecewise Linear Interpolation, *IMA Journal of Numerical Analysis* **10**, 1990, 137-154.

[5] N. Dyn, D. Levin and S. Rippa, Algorithms for the construction of data dependent triangulations, in *Algorithms for Approximation*, J.C. Mason and M.G. Cox (eds.), Chapman and Hall, London, 1990, 185-192.

[6] N. Dyn, D. Levin and S. Rippa, Boundary correction for piecewise linear interpolation defined over data-dependent triangulations, *Journal of Computationol and Applied Mathematics* **39**, 1992, 179-192.

[7] N. Dyn and S. Rippa, Data-dependent triangulations for scattered data interpolation and finite element approximation, *Applied Numerical Mathematics* **12**, 1993, 89-105.

[8] R. Franke and G. Nielson, Surface approximation with imposed conditions, in *Surfaces in Computer Aided Design*, R. Barnhill and W. Boehm (eds.), North-Holland, Amsterdam, 1983, 135-146.

[9] N.P. Fremming, Ø. Hjelle and C. Tarrou, Surface Modelling from Scattered Geological Data, in *Numerical Methods and Software Tools in Industrial Mathematics*, M. Dæhlen and A. Tveito (eds.), Birkhauser, Boston, 1997.

[10] C. Lawson, Software for C^1 interpolation, in *Mathematical Software III*, J. Rice (ed.), Academic Press, New York, 1977, 161-194.

[11] C. A. Micchelli, Interpolation of Scattered Data: Distance Matrices and Conditionally positive definite functions, *Constructive Approximation* **2**, 1986, 11-22.

[12] M.C. Parra, M.C. Lopez de Silanes and J.J Torrens, Vertical Fault Detection from Scattered Data, *Journal of Computational and Applied Mathematics* **73**(5), 1996, 225-239.

[13] M.J.D. Powell, The theory of radial basis function approximation in 1990, in *Advances in Numerical Analysis*, Vol II, W. Light (ed.), Oxford University Press, 1992, 105-210.

[14] M.J.D. Powell, The uniform convergence of thin plate spline interpolation in two dimensions, *Numerische Mathematik* **68**, 1994, 107-128.

[15] F.P. Preparata and M.I. Shamos, *Computational Geometry*, Springer, New York, 1985.

[16] R. Schaback, Creating Surfaces from Scattered Data using Radial Basis Functions, in *Mathematical Methods for Curves and Surfaces*, M. Dæhlen, T. Lyche and L.L. Schumaker (eds.), Vanderbilt University Press, Nashville, 1995, 477-496.

[17] R. Schaback, Multivariate Interpolation and Approximation by Translates of a Basis Function, in *Approximation Theory VIII*, C.K. Chui and L.L. Schumaker (eds.), World Scientific, Singapore, 1995, 491-514.

[18] L.L. Schumaker, Triangulation Methods, in *Topics in Multivariate Approximation*, C.K. Chui, L.L. Schumaker and F.I. Utreras (eds.), Academic Press, New York, 1987, 219-232.

[19] L.L. Schumaker, Triangulations in CAGD, *IEEE Computer Graphics & Applications*, January 1993, 47-52.

[20] J. Springer, Modeling of Geological Surfaces using Finite Elements, in *Wavelets, Images, and Surface Fitting*, P.-J. Laurent, A. Le Mehaute, and L.L. Schumaker (eds.), 1994, 467-474.

[21] G.P. Tolstov, *Fourier Series*, Dover Publications, 1962.

[22] Z. Wu and R. Schaback, Local error estimates for radial basis function interpolation, *IMA Journal of Numerical Analysis* **13**, 1993, 13-27.

Tim Gutzmer
Seminar for Applied Mathematics
ETH Zentrum
Ch-8092 Zürich
Switzerland
Tim.Gutzmer@@sam.math.ethz.ch

Armin Iske
SINTEF Applied Mathematics
Postbox 124, Blindern
N-0314 Oslo
Norway
Armin.Iske@@si.sintef.no