TECHNICAL UNIVERSITY OF LIBEREC

FACULTY OF MECHATRONICS AND INTERDISCIPLINARY ENGINEERING STUDIES

RESUMÉ OF PH.D. THESIS

INTERPOLATION METHODS FOR CONSTRUCTION OF SURFACES

MIROSLAV DRESSLER

KUNŠTÁT, MAY 2007

Technická univerzita v Liberci

Fakulta mechatroniky a mezioborových inženýrských studií

Obor: Přírodovědné inženýrství

Autoreferát disertační práce:

Interpolační metody pro konstrukci ploch

Mgr. Miroslav Dressler

Školitel: RNDr. Vladimír Onderka, CSc.

Kunštát, Květen 2007

Content

List of symbols

Symbol	Meaning
$\mathfrak{R}^3$	three dimensional Euclidean space
XYZ	sequence $\{(X_i, Y_i, Z_i) \in \mathfrak{R}^3, \text{i=1,\dots n}\}$ of points in 3D space
$x1, x2$	minimum and maximum of x-coordinates of points *XYZ*
$y1, y2$	minimum and maximum of y-coordinates of points *XYZ*
$z1, z2$	minimum and maximum of z-coordinates of points *XYZ*
$f(x, y)$	interpolation / approximation function
D	*planar domain of $f(x, y)$*
$\gamma(h)$	variogram function
P	matrix representing grid values
$i1, j1$	size of the grid = number of columns and rows of the matrix *P*
Dx	step of the grid in the x-direction
Dy	the step of the grid in the y-direction
NB	matrix of the nearest points
K	matrix of distances
K_{max}	maximal element of the matrix *K*
3D	three-dimensional
RS	resolution of map
Filter	parameter of the ABOS method used for setting resolution
Dmc	the minimal Chebyshev distance
$A \rightarrow B$	copy of the matrix (or vector or number) *A* into the matrix (or vector or number) *B*

1. Introduction

The presented work deals with methods for the interpolation of points in three-dimensional Euclidean space using a continuous function of two independent variables. In the first part of the work, mathematical basics of commonly used methods are described. Each method is evaluated for its advantages, limitations, applicability in individual disciplines such as geology, reservoir engineering or geophysics and for its capability to solve large problems in an acceptable time and in an acceptable quality.

The goal of the second part of the work is to design an interpolation / approximation method, which would eliminate disadvantages and limitations of existing methods and which would be robust and flexible enough for solving a broad range of interpolation / approximation problems. This method was named ABOS (**A**pproximation **B**ased **O**n **S**moothing) and despite the fact that it should be used as an approximation method, according to its name, it can also be used for solving interpolation problems.

In contrast to the sophisticated interpolation methods such as Radial basis functions, Kriging or Minimum curvature, the ABOS method does not fulfil any mathematical formulation such as linear combination of basis functions, statistical formulation of the best linear unbiased estimate or requirement of minimal curvature of resulting surface. Instead, it provides tools for modelling the resulting surface based on numerical tensioning and smoothing enabling to create a broad range of surface shapes and to accommodate the resulting surface according to the user's conception. Moreover, the ABOS method makes it possible to solve large problems without the necessity to search points in a specified surrounding of the grid nodes and without the necessity to solve a system of equations.

The motivation for the development of a new method was not only to provide sufficient flexibility and to eliminate limitations of existing methods, but also to decrease the computational time of solving large problems, when the number of input points reaches tens or hundreds of thousands – the ABOS method is in such cases up to twenty times faster than the Kriging method, which in the fifties of the last century became the ground of a new science field called geostatistics.

The computer implementation of the ABOS method consists of two basic programmes. The first one, written in FORTRAN77, implements the interpolating algorithm itself and the second one, written in C language, provides graphical user interface for displaying and comfortable manipulation both with input map objects (points, boundaries, faults) and with resulting surfaces. The graphical user interface and the whole computer system was named SurGe (**Sur**face **Ge**nerator).

After more than three-years publication through the medium of Internet WWW pages and through the medium of more than 300 software distributors, the SurGe software has proven wide applicability. Now thousands of people worldwide use the software not only in traditional fields such as reservoir engineering, geology and geophysics, but also in hydrology, geodesy, archeology, bathymetry, gravimetry, genetics, neurology, electronics, aerodynamics or nuclear physics.

2. Formulation of the interpolation / approximation problem

Let us denote *XYZ* as a sequence $\{(X_i, Y_i, Z_i) \in \Re^3, i=1,...n\}$ of points in 3D space and *D* as a rectangular planar domain containing points $\{(X_i, Y_i), i=1,...n\}$. In this work, the *solving an interpolation or approximation problem* will mean the finding such continuous function of two independent variables $f(x, y)$, for which $f(X_i, Y_i) = Z_i$ or $|f(X_i, Y_i) - Z_i| < \delta \quad \forall i = 1,...,n$ respectively.

Except for trivial cases (for example approximation by a plane or by a polynomial function of two independent variables of higher degree) it is usually not possible to express the interpolation / approximation function by a simple analytic formula. That is why the following procedure is used:

The domain *D* containing the points *XYZ* is covered by a regular rectangular grid. At each node of the grid the z-value is calculated / estimated by the method solving the above-mentioned problem using all points *XYZ* or using only the points *XYZ* belonging to the certain surrounding of the node. This procedure is called *gridding*.

The value of the function *f* can then be computed, for example, using the bilinear equation $f(x, y) = a \cdot xy + b \cdot x + c \cdot y + d$, where the coefficients *a*, *b*, *c* and *d* are determined by the corner points of the grid rectangle containing the point (x, y).

3. Commonly used approaches to solution

The goal of this section is to present commonly used techniques for solving interpolation / approximation problems and to evaluate their applicability for the solving practical tasks. The below presented interpolation / approximation methods are:

- *Triangulation with linear interpolation*
- *Natural neighbour*
- *Inverse distance*
- *Minimum curvature*
- *Regression by plane with weights*
- *Radial basis functions*
- *Kriging*

Modification of these methods for solving large problems is described in the last paragraph of this section.

3.1 Triangulation with linear interpolation

The method of triangulation with linear interpolation is historically one of the first methods used before the intensive development of computers. It is based on the division of the domain *D* into triangles. Each triangle then defines, by its three vertices, a plane – that is why the resulting surface is per partes linear.

Advantages:

- very fast algorithm

- resulting surface is interpolative

Disadvantages:

- the domain of the function f is limited to the convex envelope of the points *XYZ*.

- resulting surface is not smooth and isolines consists of line segments

- the division into triangles may be ambiguous, as the following simple example of alternative division of rectangle shows – in the first case a valley was created, in the second case a ridge was created.
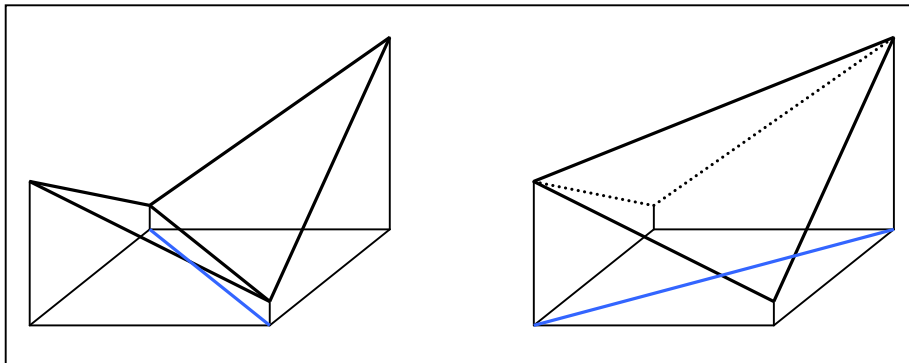

Fig. 1.2.1: Alternative division of rectangle into triangles.

Application:

This method is still used in geodesy and digital models of terrain. As a rule, characteristic points of terrain are measured – it means that the person performing terrain measurements surveys only points where the slope of terrain changes (tops, edges, valleys and so on) and thus avoids the above-mentioned ambiguity. For interpretation of such data, the Triangulation with linear interpolation is quite suitable.

3.2 Natural neighbour

The Natural neighbour is an interpolation method based on Voronoi tessellation. Voronoi tessellation can be defined as "the partitioning of a plane with n points into n convex polygons such that each polygon contains exactly one point and every point in a given polygon is closer to its central point than to any other" (see [S11]). A description of the natural neighbour interpolation technique follows:

Given a set of data points distributed on a plane, natural neighbour interpolation computes the interpolated value for a given point X as the weighted sum of the points which are natural neighbours of X. The natural neighbours can be intuitively understood as those points which would be adjacent to X in a Voronoi tessellation of the point set including X. Figure 3.2 depicts in black a Voronoi tessellation of the points A through E. The grey region marks the new Voronoi cell which would be present if the point X were included in the tessellation. The weights of points A through E used to compute the interpolated value of X are respectively the areas of the grey region intersecting each original cell of A through E and are also known as the natural neighbour coordinates of X.
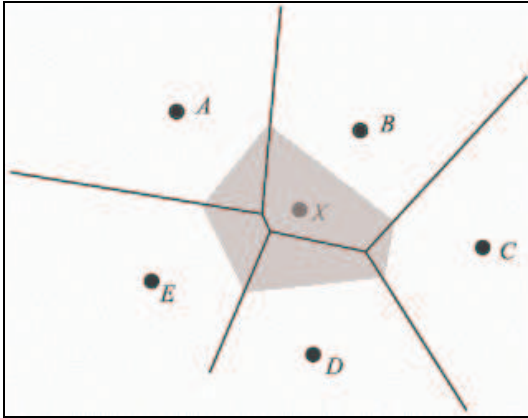
Fig. 3.2: New Voronoi cell and areas for computation of neighbour points weights.

The surface formed by natural neighbour interpolation has the useful properties of being continuous (C0) everywhere and passing exactly through z values of all data points. Moreover, the interpolated surface is continuously differentiable (C1) everywhere except at the data points, providing smooth interpolation in contrast to the Triangulation with linear interpolation method.

Advantages:

- fast algorithm

- resulting surface is interpolative and smooth expect at the data points.

Disadvantages:

- the domain of the function f is limited to the convex envelope of the points *XYZ*

- the shape of resulting surface is not acceptable in some fields such as in geology or hydrogeology.

Application:

The Natural neighbour method is mainly used in GIS systems as a digital model of terrain and fast interpolation of terrain data providing smooth surface.

3.3 Inverse distance

This method computes a value of function f at an arbitrary point $(x, y) \in D$ as a weighted average of values Z_i:

$$f(x, y) = \sum_{i=1}^{n} Z_i w_i , \quad \text{where} \quad w_i = h_i \bigg/ \sum_{i=1}^{n} h_i \quad \text{and} \quad h_i = 1 \bigg/ \sqrt{(x - X_i)^2 + (y - Y_i)^2 + \sigma^2}$$

σ^2 is a smoothing parameter.

If the number of points n is too great, the value of $f(x, y)$ is calculated only from points belonging to the specified circle surrounding the point (x, y). The method was frequently implemented in the first stages of computer development.

Advantages:

- simple computer implementation; for its simplicity, the method is implemented in almost all gridding software packages

- if $\sigma^2 = 0$, the method provides interpolation.

Disadvantages:

- high computer time consumption if the number of points n is large (due to computation of distances)

- typical generation of "bull's-eyes" surrounding the position of point locations within the domain D – that is why the resulting function is not acceptable for most applications.

3.4 Minimum curvature method

This method and namely its computer implementation was developed by W.H.F. Smith and P.Wessel (see [1]) in 1990. The interpolated surface generated by the Minimum curvature method is analogous to a thin, linearly elastic plate passing through each of the data values with a minimum amount of bending. The algorithm of the Minimum curvature method is based on the solution of the modified biharmonic differential equation $(1-T)\nabla^4 f(x,y) - (T)\nabla^2 f(x,y) = 0$, where

$T \in \langle 0,1 \rangle$ is a tensioning parameter,

∇^2 is the Laplacian operator $\nabla^2 f = \partial^2 f / \partial x^2 + \partial^2 f / \partial y^2$ and

$\nabla^2 \nabla^4 = (\nabla^2)^2$ is the biharmonic operator $\nabla^4 f = \partial^4 f / \partial x^4 + \partial^4 f / \partial y^4 + 2\partial^4 f / \partial x^2 \partial y^2$.

If $T=0$, the biharmonic differential equation is solved; if $T=1$, the Laplace differential equation is solved – in this case the resulting surface may have local extremes only at points *XYZ*.

Advantages:

- speed of computation is high and an increasing number of points *XYZ* has small influence on decreasing the computational speed.

- suitable method for a large number of points *XYZ*.

Disadvantages:

- complicated algorithm and computer implementation

- if the parameter T is near zero, the resulting surface may have local extremes out of the points location

- bad ability to conserve extrapolation trends.

Application:

- universal method suitable for smooth approximation and interpolation (for example distribution of temperature, water heads, potential fields and so on).

3.5 Regression by plane with weights

This method is based on regression by plane $f(x,y) = ax + by + c$ using a weighted least square fit. The weight w_i assigned to the point (X_i, Y_i, Z_i) is computed as an inverse distance from point (x,y) to the point (X_i, Y_i, Z_i). Then the minimum of the following function of three independent variables has to be found:

$F(a,b,c) = \sum_{i=1}^{n} w_i (z(X_i, Y_i) - Z_i)^2 = \sum_{i=1}^{n} w_i (aX_i + bY_i + c - Z_i)^2$, which leads to a solution of three linear equations:

$$\partial F/\partial a = 0 = 2\sum_{i=1}^{n} w_i X_i (aX_i + bY_i + c - Z_i)$$

$$\partial F/\partial b = 0 = 2\sum_{i=1}^{n} w_i Y_i (aX_i + bY_i + c - Z_i)$$

$$\partial F/\partial c = 0 = 2\sum_{i=1}^{n} w_i (aX_i + bY_i + c - Z_i)$$

After rearrangement the following equations are obtained:

$$a\sum_{i=1}^{n} w_i X_i^2 + b\sum_{i=1}^{n} w_i X_i Y_i + c\sum_{i=1}^{n} w_i X_i = \sum_{i=1}^{n} w_i X_i Z_i$$

$$a\sum_{i=1}^{n} w_i X_i Y_i + b\sum_{i=1}^{n} w_i Y_i^2 + c\sum_{i=1}^{n} w_i Y_i = \sum_{i=1}^{n} w_i Y_i Z_i$$

$$a\sum_{i=1}^{n} w_i X_i + b\sum_{i=1}^{n} w_i Y_i + c\sum_{i=1}^{n} w_i = \sum_{i=1}^{n} w_i Z_i$$

Advantages:

- simple algorithm, good extrapolation properties

Disadvantages:

- resulting function is only approximative

- slow speed of computation if n is great (due to computation of distances)

Application:

- surface reconstruction from digitised contour lines

3.6 Radial basis functions

The method of Radial basis functions uses the interpolation function in the form:

$$f(x, y) = p(x, y) + \sum_{i=1}^{n} w_i \cdot \phi(|(x, y) - (X_i, Y_i)|) \qquad (3.6)$$

where

$p(x, y)$	is a polynomial
$w_i \in \Re$	are real weights
$|(x, y) - (X_i, Y_i)|$	is the Euclidean distance between points (x, y) and (X_i, Y_i)
$\phi(r)$	is a radial basis function

Commonly used radial basis functions are (c^2 is the smoothing parameter):

Multiquadric: $\qquad \phi(r) = \sqrt{r^2 + c^2}$

Multilog: $\qquad \phi(r) = \log(r^2 + c^2)$

Natural cubic spline: $\phi(r) = (r^2 + c^2)^{3/2}$

Natural plate spline: $\phi(r) = (r^2 + c^2) \cdot \log(r^2 + c^2)$

The interpolation process starts with polynomial regression using the polynomial $p(x, y)$. Then the following system of n linear equations is solved for unknown weights w_i, $i = 1,...,n$:

$$Z_j - p(X_j, Y_j) = \sum_{i=1}^{n} w_i \cdot \phi(|(X_j, Y_j) - (X_i, Y_i)|), \quad j = 1,...,n$$

As soon as the weights w_i are determined, the z-value of the surface can be directly computed from equation (3.6) at any point $(x, y) \in D$.

Advantages:

- simple computer implementation; the system of linear equations has to be solved only once (in contrast to the Kriging method, where a system of linear equations must be solved for each grid node – see the next section)

- the resulting function is interpolative

- easy implementation of smoothing

Disadvantages:

- if the number of points n is large, the number of linear equations is also large; moreover the matrix of the system is not sparse, which leads to long computational time. That is why this method, as presented, is used for solving small problems with up to a few thousand points. Solving large problems is also possible, but requires an additional algorithm for searching points in the specified surrounding of each grid node – see paragraph ***3.8 Modification for solving large problems***.

Application:

- universal method suitable for use in any field

3.7 Kriging

Kriging is an interpolation method that was originally developed for use in geostatistics by D. G. Krige (see [2]), a professor at the University of Witwatersrand, South Africa, in the 1950s. In fact, the work of professor Krige is the base of science field called geostatistics.

Kriging is probably the most often used method for solving interpolation / approximation problems, namely because it is based on the statistical formulation of the *best linear unbiased estimate*. An important concept for deriving this method is *empirical* or *experimental variogram* $\gamma(h)$:

$$\gamma(h) = \frac{1}{2} \cdot \frac{1}{C(N(h))} \sum_{N(h)} (Z_i - Z_j)^2 \text{ , where}$$

$N(h) = \left\{ i, j : \left| (X_i, Y_i) - (X_j, Y_j) \right| = h \right\}$ a $C(N(h))$ is the number of elements of set $N(h)$.

For real data it is not probable that a pair of points will satisfy the condition $\left| (X_i, Y_i) - (X_j, Y_j) \right| = h$ and therefore for practical computation the set $N(h)$ is specified as $N(h) = \left\{ i, j : \left| (X_i, Y_i) - (X_i, Y_i) \right| \in [h - \delta h, h + \delta h] \right\}$

The empirical variogram is approximated using the *theoretical variogram* or *model*. The commonly used models are:

Linear model:
$\gamma(h) = C_0 + Sh$, $h \neq 0$, where C_0 is the so called nugget effect and S is the unknown slope.

Gaussian model:
$\gamma(h) = C_0 + (C - C_0) \cdot \left\{ 1 - \exp(-h^2 / a^2) \right\}$

Exponential model:
$\gamma(h) = C_0 + (C - C_0) \cdot \left\{ 1 - \exp(-h / a) \right\}$

The Kriging method is intended for estimating the interpolation / approximation function $f(x, y)$ at point (X, Y) under the following assumptions:

a) The estimate Z of function $f(x, y)$ at an arbitrary selected point $(X, Y) \in D$ can be expressed as a weighted average $Z = \sum_{i=1}^{n} w_i \cdot Z_i$

b) Sum of weights is 1: $\sum_{i=1}^{n} w_i = 1$

c) The estimate of value Z is unbiased i.e. the mean $E[f(x, y) - Z] = 0$.

The weights are to be computed so that the dispersion variance $D(f(x, y) - Z)$ is minimal. In the next derivation we use f instead of $f(x, y)$. Taking into consideration the definition of dispersion variance $D(X) = E(X^2) + (E(X))^2$ and assumption c) it is obvious that we have to minimize the expression

$$E([f - \sum_{i=1}^{n} w_i \cdot Z_i]^2) = E([f \cdot (\sum_{i=1}^{n} w_i) - \sum_{i=1}^{n} w_i \cdot Z_i]^2) =$$

$$= E([\sum_{i=1}^{n} w_i \cdot f - \sum_{i=1}^{n} w_i \cdot Z_i]^2) = E([\sum_{i=1}^{n} w_i \cdot (f - Z_i)]^2) = \sum_{i=1}^{n} \sum_{j=1}^{n} w_i w_j E((f - Z_i)(f - Zj))$$

From equation

$$E([Z_i - Z_j]^2) = E([Z_i - f + f - Z_j]^2) = E([Z_i - f]^2) - 2E([(f - Z_i)(f - Z_j)] + E([f - Z_j]^2))$$

it follows that

$$E([(f - Z_i)(f - Z_j)]) = \frac{1}{2} \cdot \left[E([f - Z_i]^2) + E([f - Z_j]^2) - E([Z_i - Z_j]^2) \right].$$

This term can be substituted into the expression which has to be minimized:

$$\sum_{i=1}^{n} \sum_{j=1}^{n} w_i w_j E((f - Z_i)(f - Z_j)) =$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} w_i w_j \frac{1}{2} \left[E([f - Z_i]^2) + E([f - Z_j]^2) - E([Z_i - Z_j]^2) \right] =$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} w_i w_j \frac{1}{2} E([f - Z_j]^2) + \sum_{i=1}^{n} \sum_{j=1}^{n} w_i w_j \frac{1}{2} E([f - Z_i]^2) - \sum_{i=1}^{n} \sum_{j=1}^{n} w_i w_j \frac{1}{2} E([Z_i - Z_j]^2)$$

The first two terms can be simplified:

$$\sum_{i=1}^{n} \sum_{j=1}^{n} w_i w_j \frac{1}{2} E([f - Z_j]^2) = \sum_{j=1}^{n} \frac{1}{2} w_j E([f - Z_j]) \sum_{i=1}^{n} w_i = \sum_{j=1}^{n} \frac{1}{2} E([f - Z_j]^2)$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} w_i w_j \frac{1}{2} E([f - Z_i]^2) = \sum_{i=1}^{n} \frac{1}{2} w_i E([f - Z_i]) \sum_{j=1}^{n} w_j = \sum_{i=1}^{n} \frac{1}{2} E([f - Z_i]^2)$$

The final form of the expression, which has to be minimized is:

$$2 \sum_{j=1}^{n} \frac{1}{2} E([f - Z_j]^2) - \sum_{i=1}^{n} \sum_{j=1}^{n} w_i w_j \frac{1}{2} E([Z_i - Z_j]^2) = F(w_i, ..., w_n)$$

If the function $F(w_1,...,w_n)$ has to be minimal, the partial derivatives according to w_k must be zero:

$$\frac{\partial F}{\partial w_k} = 0 \quad \forall k = 1,...,n$$

It means that the following system of linear equations is obtained:

$$\sum_{i=1}^{n} \frac{1}{2} E([Z_k - Z_i]^2) w_i = \frac{1}{2} E([f - Z_k]^2) \quad k = 1,...,n$$

The halves of means $\frac{1}{2} E([Z_k - Z_i]^2)$ and $\frac{1}{2} E([f - Z_i]^2)$ can be substituted by values from the theoretical variogram. The system of equations must be completed by condition b), i.e. the sum of weights is 1 (that is why a fictive variable μ has to be added). In the form of matrix notation the system of linear equations can be expressed as:

$$\begin{bmatrix} \Gamma & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} w \\ \mu \end{bmatrix} = \begin{bmatrix} \gamma^x \\ 1 \end{bmatrix}$$

where $\Gamma_{ij} = \gamma(\|(X_i,Y_i) - (X_j,Y_j)\|)$ and $\gamma_i^x = \gamma(\|(x,y) - (X_i,Y_i)\|)$, $i, j = 1,...,n$

Advantages:
- the algorithm is based on statistical formulation of the best linear unbiased estimate which means, there is no better interpolation method from the statistical point of view.

Disadvantages:
- the weights w_i must be computed (i.e. the system of linear equations must be solved) for each node of the grid.

- if the number of points n is large, a large number of linear equations must be solved; that is why this method is used for solving small problems with up to a few thousand points. For solving large problems an additional algorithm for searching points in the specified surrounding of each grid node must be implemented, as described in the next paragraph.

- the method produces undesirable "pits" and "circular" isolines in the generated surface – see examples in paragraphs *4.4.2 Shape of generated surface* and *4.4.1 Smoothness of interpolation and oscillations*.

Application:
- universal method designed especially for geology and geophysics applications.

3.8 Modification for solving large problems

Except for the Triangulation with linear interpolation and Minimum curvature method, presented techniques need to compute horizontal distances from estimated point (grid node) to all points *XYZ*. For large problems (thousands or more points) it means increasing of computational time, which brings necessity to reduce number of points involved into the estimation. This is solved by specification of the surrounding which determines the set of points included into the estimation. Of course, additional algorithm for searching of points falling into the surrounding must be implemented.

On one hand the method can be used for large problems, but on the other hand new complications are involved:

- The surrounding is usually circular (or elliptical, if an anisotropy has to be modelled) with specified radius. In addition, implementations of gridding procedures requires specification of maximal and minimal number of points, minimal and maximal number of points in each quadrant (octant) and so on. It means that additional parameters influencing the quality of resulting function has to be entered.
- The resulting surface may contain discontinuities (see [4]) as the set of selected points may change while moving the estimated point from one grid node to the next. This phenomenon is illustrated in the next chapter in paragraph *4.4.3 Conservation of extrapolation trend*.
- In the case of the Radial basis functions method the system of linear equation must be completed and solved for each node of grid.

4. The ABOS method

The goal of this chapter is to design an interpolation / approximation method which is sufficiently flexible and robust enough for solving large problems, provides results comparable with the Kriging method (respectively with the Radial basis function method or Minimum curvature method) and which does not have disadvantages and limitations of these methods presented in the first chapter.

The method was called ABOS (**A**pproximation **B**ased **O**n **S**moothing) and despite the fact that it should be used as an approximation method, according to its name, it can also be used for solving interpolation problems, as it will be explained in this chapter.

4.1 The definition of the interpolation function and notations

The interpolation function is determined by a matrix P of real numbers, whose elements (z-coordinates) are assigned to nodes of a regular rectangular grid covering the domain D (see the next figure).
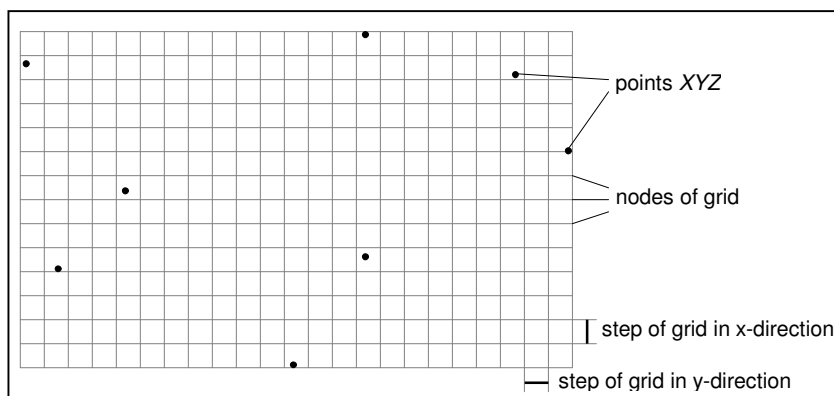

Figure 4.1: Regular rectangular grid for defining the interpolation function.

The value of the interpolation function at any point (x_0, y_0) within the grid can be evaluated from the equation of the bilinear polynomial $f(x, y) = a \cdot xy + b \cdot x + c \cdot y + d$,

which is defined by coordinates of corner points of the grid rectangle containing the point (x_0, y_0).

The following notation is used in the next text:

$x1, x2$ minimum and maximum of x-coordinates of points *XYZ*

$y1, y2$ minimum and maximum of y-coordinates of points *XYZ*

$z1, z2$ minimum and maximum of z-coordinates of points *XYZ*

$i1, j1$ size of the grid = number of columns and rows of the matrix *P*

$P_{i,j}$ elements of the matrix P, $i = 1,...,i1$, $j = 1,...,j1$

DP   auxiliary matrix with the same size as the matrix *P*

Z   vector of z-coordinates of points *XYZ*

DZ   auxiliary vector with the same size as the vector *Z*

NB   matrix of the nearest points - integer matrix with the same size as the matrix P containing for each node of the grid the order index of the nearest point *XYZ*

K   matrix of distances – integer matrix with the same size as the matrix P containing for each node of the grid the distance to the nearest point *XYZ* measured in units of grid

K_{max} maximal element of the matrix *K*

Filter   a parameter of the ABOS method used for setting resolution

RS   resolution of map; $RS = max\{(x2 - x1),(y2 - y1)\} / Filter$

Dx   the step of the grid in the x-direction; $Dx = (x2 - x1)/(i1 - 1)$

Dy   the step of the grid in the y-direction; $Dy = (y2 - y1)/(j1 - 1)$

Dmc   the minimal Chebyshev distance between pairs of points *XYZ*;

$$Dmc = min\{max\{\left|X_i - X_j\right|, \left|Y_i - Y_j\right|\} \ ; \ i \neq j \wedge i, j = 1,...,n\}$$

$A \rightarrow B$ means a copy of the matrix (or vector or number) *A* into the matrix (or vector or number) *B*

4.2 Interpolation algorithm

The algorithm of the ABOS method can be briefly described by the following scheme:

1. Filtering points *XYZ*, specification of the grid, computation of the matrices *NB* and *K*, $Z \rightarrow DZ$, $0 \rightarrow DP$
2. Per partes constant interpolation of values *DZ* into the matrix *P*
3. Tensioning and smoothing of the matrix *P*
4. $P + DP \rightarrow P$
5. $Z_i - f(X_i, Y_i) \rightarrow DZ_i$
6. If the maximal difference $max\{DZ_i, i = 1,...,n\}$ does not exceed defined precision, the algorithm is finished
7. $P \rightarrow DP$, continue from step 2 again (= start the next iteration cycle)

In the following paragraphs the particular steps of the algorithm are explained in full detail.

4.2.1 Filtering of points XYZ

If an interpolation / approximation problem has to be solved, it is necessary to take into consideration the fact that there may be some points *XYZ* with a horizontal distance less than the desired resolution of the resulting surface. That is why the first implemented

algorithm in the ABOS method is the filtering of points *XYZ*. Filtering substitutes every two points (X_i, Y_i, Z_i), (X_j, Y_j, Z_j), such that

$$\left| X_i - X_j \right| < RS \;\land\; \left| Y_i - Y_j \right| < RS, \tag{4.2.1}$$

by one point (X_k, Y_k, Z_k) with average coordinates i.e. $X_k = (X_i + X_j)/2$, $Y_k = (Y_i + Y_j)/2$ and $Z_k = (Z_i + Z_j)/2$.

The resolution *RS* is computed as $max\{(x2 - x1), (y2 - y1)\}/Filter$, where *Filter* is an optional parameter of the ABOS method. It is similar to the resolution of a digital picture – if the distance of two points with different colours is smaller than the pixel size of the digital picture, only one point with "average" colour can be seen.

The formulation of the filtering principle is easy, but computer implementation represents an efficiency problem, which is discussed in the next paragraphs.

To test the condition (2.2.1) means to compare coordinates of all pairs of points *XYZ*. Such a problem is usually solved by nested loops with this pattern:

```
for (i=1;i<n;i++)
  {for (j=i+1;j<=n;j++)
    {
     compare coordinates of point i and j
    }
  }
```

It means that $n \cdot (n-1)/2$ comparisons are performed. If *n* is large, the computational time may be unacceptable. Taking into consideration the efficiency of today's computers, $n \cong 50000 - 100000$ is a critical value from this point of view. For example, filtering 100000 points using this algorithm took 170 seconds on the testing computer.

A significant increase in speed occurs when the points *XYZ* are sorted according to the x-coordinates using a very fast standard C-library function *qsort* (that is why this filtering algorithm is one of the few algorithms written in C language). If the points *XYZ* are sorted according to the x coordinate, the above loops can be changed like this:

```
for (i=1;i<n;i++)
  {j=i+1;
   while ((j<=n)&&fabs(X[i]-X[j])<RS))
     {
      compare y-coordinate of point i and j;
      j++;
     }
  }
```

In other words, if the points *XYZ* are sorted according to the x-coordinates, there is no need to compare point *i* with all points $j = i+1,...,n$ but only with the points *j* having $\left| X_i - X_j \right|$ less than the resolution. Using this approach the time needed for filtering 100000 points was reduced to 5 seconds.

An example of the filtering effect is displayed in the following figure. The input file contains 100000 points laying on a spiral and 50000 points forming a rotated square. These points are displayed in blue while data obtained after filtering are displayed in black. The *Filter* parameter was set to 100.
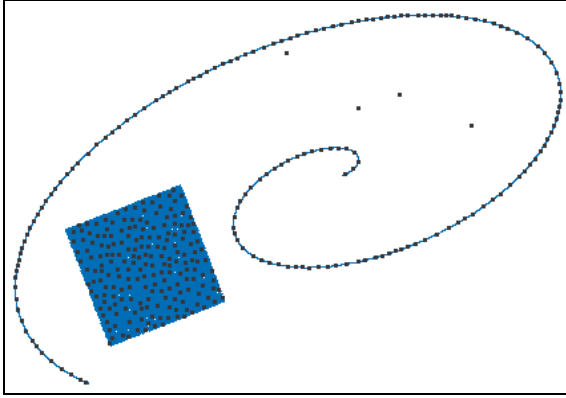
Fig. 3.4.1a: Filtered data.

It is obvious that filtering preserves the shape of clustered data while isolated points remain untouched.

According to performed tests, the above described filtering algorithm is effective for up to 300000 points – in such case the filtering process takes 25 seconds, which is still an acceptable time.

4.2.2 Specification of the grid

The size of the regular rectangular grid is set according to the following points:

1. The greater side of the rectangular domain D is selected, i.e. greater number of $x21 = (x2 - x1)$ and $y21 = (y2 - y1)$. Without loss of generality we can assume that $x21$ is greater.

2. The minimal grid size is computed as $i0 = round(x21 / Dmc)$, where Dmc is the minimal Chebyshev distance between pairs of points XYZ:

$$Dmc = min\{max\{|X_i - X_j|, |Y_i - Y_j|\} \; ; \; i \neq j \wedge i, j = 1,...,n\}$$

3. The optimal grid size is computed as $i1 = max\{k \cdot i0; k = 1,...,5 \wedge k \cdot i0 < Filter\}$

4. The second size of the grid is $j1 = round(y21 / x21 \cdot (i1 - 1)) + 1$

The presented procedure ensures that the difference between Dx and Dy is minimal i.e. the regular rectangular grid is close to a square grid.

4.2.3 Computation of matrices NB and K

The matrices NB and K are computed using the algorithm based on "circulation" around the points XYZ, as the following figure indicates:
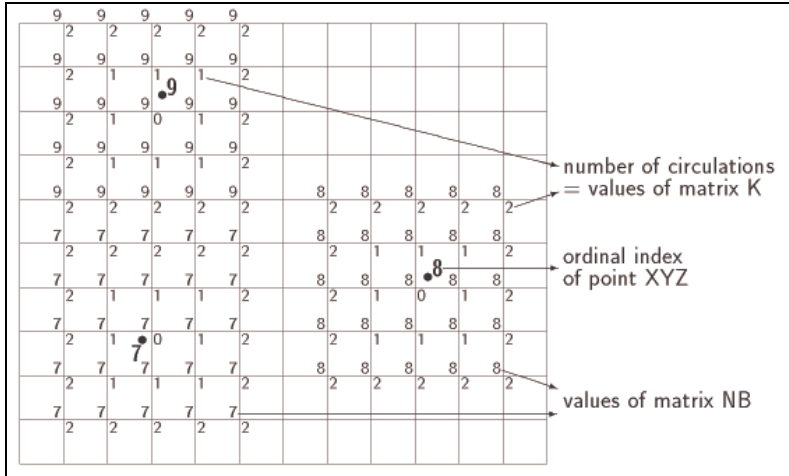
16

Fig. 4.2: Computation of the matrices *NB* and *K*.

All elements of the matrices *NB* and *K* are initially set to zero and the process of circulation continues as long as there are zero values in the matrix *NB*. The Euclidean distance is compared only if the element $K_{i,j}$ corresponding to the evaluated node is not zero and $IC/\sqrt{2} \leq K_{i,j}$, where *IC* is the ordinal number of the current circulation. By this way, the number of distance computations is significantly reduced.

The computation of the matrix *NB* defines a natural division of the domain of the interpolation function into polygons (so called Voronoi or Thiessen polygons, see the following figure), inside which interpolation with constant values is performed.
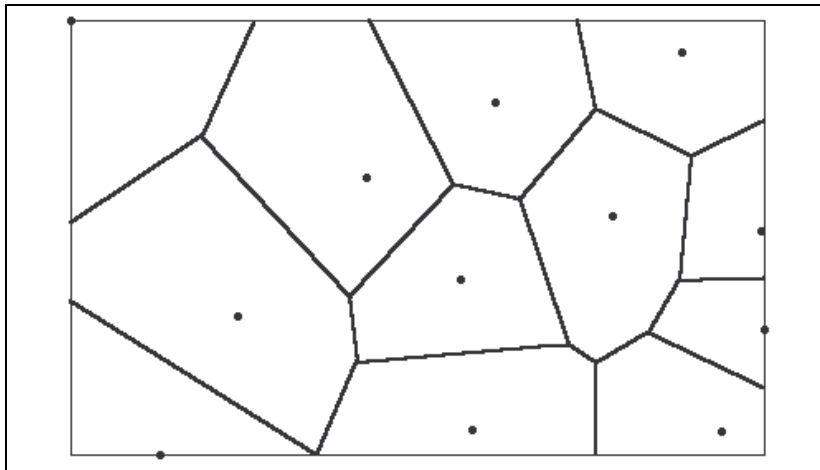
Fig. 4.2.3: Division of the domain of the interpolation function.

4.2.4 Per partes constant interpolation

After computing the nearest points matrix, per partes interpolation (see the figure 4.2.4) is very simple: $P_{i,j} = DZ(NB_{i,j})$.
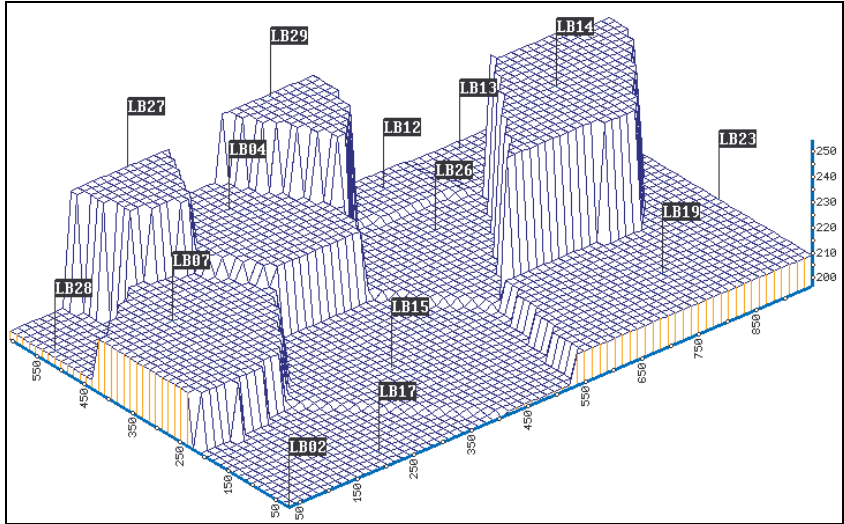
Fig. 4.2.4: Per partes constant interpolation.

4.2.5 Tensioning

Tensioning of the surface (see figure 4.2.5) modifies the matrix P according to the formula:

$$P_{i,j} = (P_{i+k,j} + P_{i,j+k} + P_{i-k,j} + P_{i,j-k})/4,$$ (4.2.5)
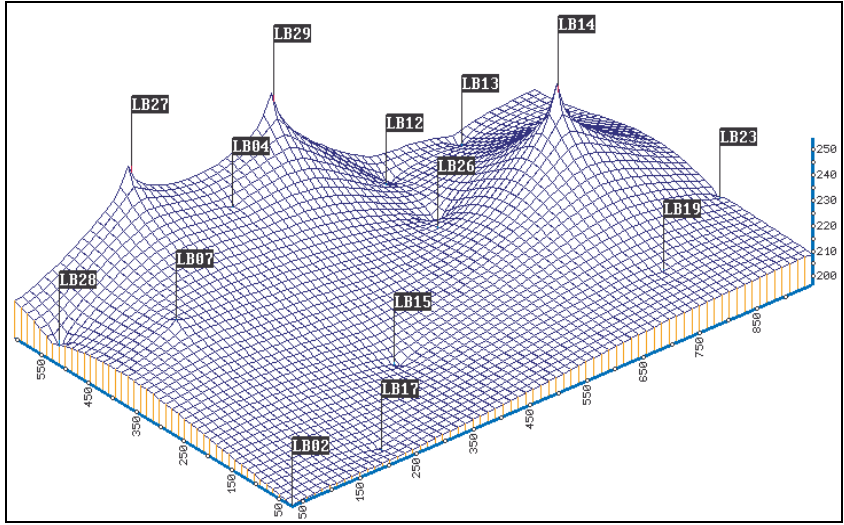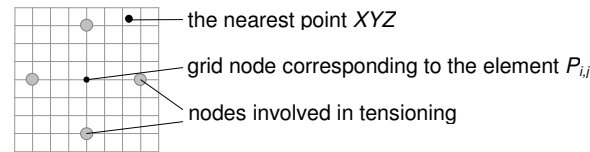
where $k = K_{i,j}$, $i = 1,...,i1$, $j = 1,...,j1$

Fig. 4.2.5: Tensioning of the surface.

The following scheme shows the nodes (marked by gray circles) corresponding to the elements of the matrix P, which are involved in tensioning.

the nearest point *XYZ*

grid node corresponding to the element $P_{i,j}$

nodes involved in tensioning

Tensioning is repeatedly performed in the loop with this pattern:

18

```
DO N = MAX(4,K_max/2+2),1,-1
   …
ENDDO
```

If k is greater than decreasing loop variable N, then $k = N$.

4.2.6 Linear tensioning

Linear tensioning of the surface (see figure 4.2.6) modifies the matrix P according to the formula for weighted average:

$$P_{i,j} = \left(Q \cdot (P_{i+u,j+v} + P_{i-u,j-v}) + P_{i-v,j+u} + P_{i+v,j-u}\right) / (2 \cdot Q + 2) \qquad (4.2.6)$$

$$\forall\ i = 1,...,i1,\ \ j = 1,...,j1\ ;\ \ K_{i,j} > 0$$

where (u, v) is the vector from the node i, j to the nearest grid node of the point $NB_{i,j}$ and the weight $Q = L \cdot (K_{max} - K_{i,j})^2$. The constant $L = 1 / ((0{,}107 \cdot K_{max} - 0.714) \cdot K_{max})$ is an empirical constant suppressing the influence of K_{max}.

In the implementation of the ABOS method there are four degrees of linear tensioning 0, 1, 2 and 3. Here presented formulas are valid for the default degree 1; their modifications for other degrees will be described in chapter dealing with the computer implementation of the ABOS method.
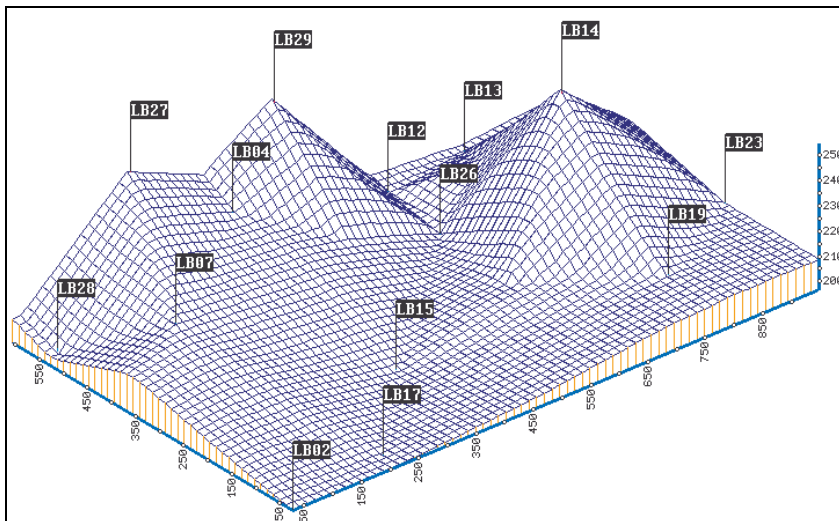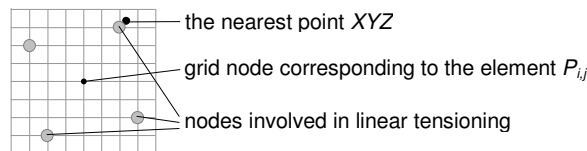

Fig. 4.2.6: Linear tensioning of the surface.

The following scheme shows the nodes (marked by gray circles) corresponding to the elements of the matrix P, which are involved in linear tensioning.

Linear tensioning is repeatedly performed in the loop with this pattern:

```
DO N = MAX(4,K_max/2+2),1,-1
   …
ENDDO
```

If the length $|(u, v)|$ of the vector (u, v) is greater than decreasing loop variable N, then the vector (u, v) is multiplied by constant c so that $c \cdot |(u, v)| = N$.

4.2.7 Smoothing

Smoothing (see figure 4.2.7) replaces elements of the matrix P by the value of weighted average:

$$P_{i,j} = (\sum_{k=i-1}^{i+1} \sum_{l=j-1}^{j+1} P_{k,l} + P_{i,j} \cdot (q \cdot t_{i,j} - 1)) / (q \cdot t_{i,j} + 8) , \quad i = 1,...,i1, \quad j = 1,..., j1 , \qquad (4.2.7)$$

where q is the parameter of the ABOS method controlling smoothness of the interpolation / approximation (its default value is 0.5) and $t_{i,j}$ are weights, which are zero before the first smoothing and afterwards they are computed according to the formula

$$t_{i,j} = (\sum_{k=i-2}^{i+2} \sum_{l=j-2}^{j+2} (P_{i,j} - P_{k,l}))^2 , \quad i = 1,...,i1, \quad j = 1,..., j1$$

and scaled into the interval $<0,100>$. In brief, it can be said, the values of $t_{i,j}$ are higher at nodes where the surface has a local extreme and lower at nodes where the surface is decreasing / increasing.
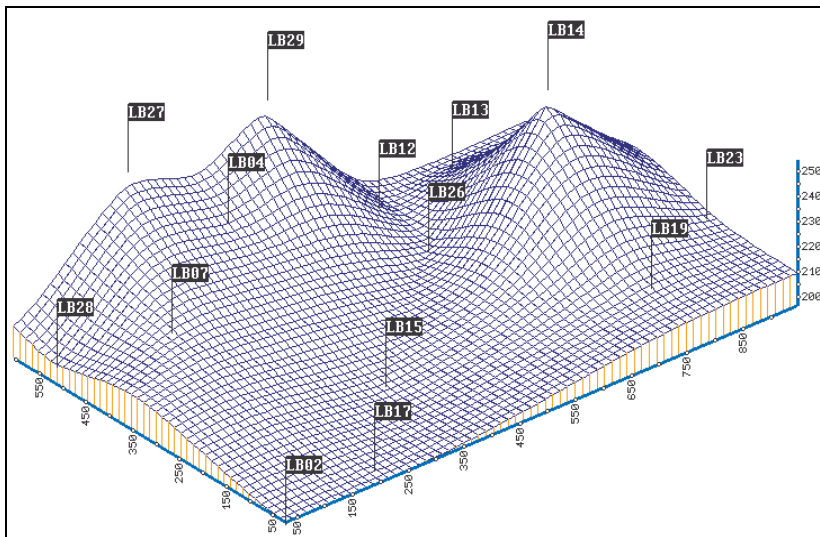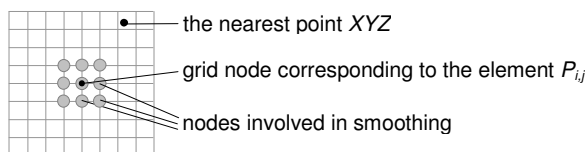

Fig. 4.2.7: Smoothing of the surface.

The following scheme shows the nodes (marked by grey circles) corresponding to the elements of the matrix P, which are involved in smoothing.

Smoothing is repeatedly performed in the loop with this pattern:

```
DO N = MAX(4,K_max*K_max/16),1,-1
   …
ENDDO
```

As an option the ABOS method enables to perform so called LES smoothing – in this case the formula (4.2.7) is not applied if the decreasing loop variable N is greater than $K_{i,j} + 1$. This modification of smoothing suppresses oscillations and exceeding of local extremes, if they occur (see paragraph *4.4.1 Smoothness of interpolation and oscillations*).

4.2.8 Iteration cycle

After smoothing, the matrices *P* and *DP* are added element by element and the result is stored again in the matrix *P*. Note that in the first iteration step the matrix *DP* is zero – that is why the matrix *P* does not change.

The tensioned and smoothed surface does not pass through the z-coordinates of points *XYZ* exactly, so the differences $DZ_i = Z_i - f(X_i, Y_i)$, $i = 1,...,n$ are calculated.

If the maximal difference $max\{DZ_i, i = 1,...,n\}$ is less than the specified accuracy of the interpolation / approximation multiplied by $(z2 - z1)/100$, the algorithm of the ABOS method is finished. In the opposite case the matrix *P* is copied into the matrix *DP* and the algorithm continues from step 2, where a new iteration cycle begins. It differs from the first cycle only in these points:

- the matrix *DP* is not zero
- per partes constant interpolation is applied to the difference values DZ_i and not to the original z-coordinates of the points *XYZ*
- if the maximal difference in the current cycle is not less than the one in the previous cycle, the problem is considered to be non-converging and the algorithm is finished.

After each smoothing of the surface, the inaccuracy of the solution can be decreased by linear transformation of the matrix *P*:

$a \cdot P_{i,j} + b \rightarrow P_{i,j}$, where constants *a* and *b* minimize the term

$$\sum_{i=1}^{n} (a \cdot f(X_i, Y_i) + b - DZ_i)^2$$

In this way, the number of iteration cycles can be reduced by up to 30%.

The *accuracy*, another parameter of the ABOS method, is specified as a percentage value from the difference $z2 - z1$. The default value is 1; if 0 is specified and the iteration process ends with zero maximal difference, then interpolation is achieved.

4.3 Flexibility of the ABOS method

Most of interpolation methods offer some possibility, how to modify constructed surface. For example, the Radial basis function method enables to vary smoothness of surface using smoothing parameter, the Kriging method may use different variogram models with different parameters and the Minimum curvature method uses tension parameter.

The ABOS method can modify the created surface namely through the use of smoothness parameter *q* (see paragraph *4.2.7 Smoothing*) and an approximation can be achieved by setting of an appropriate accuracy parameter. Moreover, the number of smoothing cycles suggested by implementation of the ABOS method SURGEF can be increased so that suitable trend surface is obtained. The cross-section through seven surfaces in the next figure illustrates possible modifications of the surface shape generated by the ABOS method.
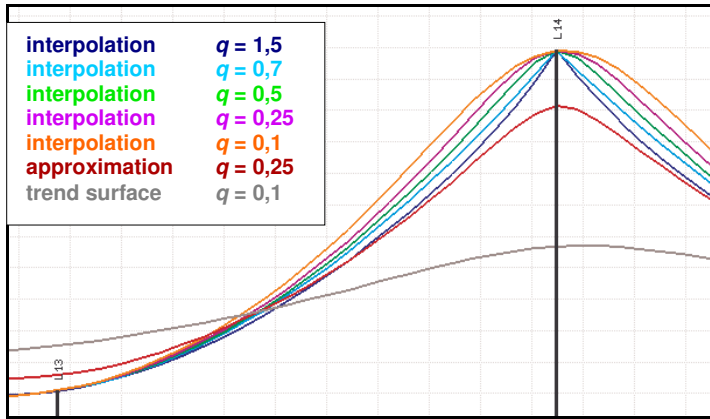
Fig. 4.3: Flexibility of the ABOS method interpolation / approximation.

The graphical user interface SurGe offers a tool for setting of suitable parameters depending on desired interpolation / approximation type.

The usage of a trend surface is demonstrated in paragraph *4.4.3 Conservation of extrapolation trend*.

4.4 Comparison with other interpolation methods

In this section the ABOS method is compared with three methods, which are considered to be the most significant:

- the *Minimum curvature* method with the tension 0.1
- the *Kriging* method using the linear model and zero nugget effect
- the *Radial basis functions* method using the multiquadric basis functions and zero smoothing parameter.

Although the Radial basis functions method was included, its graphical results are not presented because it provides almost the same result as the Kriging method (no differences can be seen in the graphical representation of results).

Interpolations using Minimum curvature were performed using the program SURFACE, which is a part of the GMT system (see [S1]), while Kriging was performed using Surfer software (see [S2]).

To compare interpolation methods, we will use three data sets OSCIL, SHAPE and SIBIR [XIX]. Characteristics of these examples are summarized in the following table:

Data set	Number Of points	Domain range		Grid size	Grid step		Figure
		x	Y		X	Y	
OSCIL	17	0-1	0-1	257x257	3.90625e-3	3.90625e-3	4.4.1a
SHAPE	30	0-1000	0-640	101x65	10	10	4.4.2a
SIBIR	13504	0-73600	0-51200	737x513	100	100	4.4.3a

The first example (see figure 4.4.1a) OSCIL is intended for examining of oscillation phenomena, which may occur especially if a smooth interpolation is used. The example SHAPE was designed for comparison of surface shape. Figure 4.4.2a shows the distribution of points XYZ and contains the horizontal projection of two cross-sections A-A' and B-B' used for detailed illustration of the surface shape. The third example (see figure 4.4.3a) is used for the comparison of trend conservation. The cross-section A-A' was designed for demonstrating extrapolation properties of the tested methods i.e. for evaluating how the tested methods conserve trend in areas where points are missing.

4.4.1 Smoothness of interpolation and oscillations

Depending on the configuration of the points XYZ a smooth interpolation may cause unwanted oscillations in the generated surface, which is a common problem of most smooth interpolation methods. In this paragraph we will examine oscillations on specially designed example (see figure 4.4.1a) containing 13 points distributed along both diagonals of the square. All points have z-coordinate equal to zero except the point in the centre of the square, where the z-coordinate is one. We can expect that there will be oscillations between points having zero z-coordinates – that is why we will focus our attention to the cross-section going through points L09, L05 and L01.
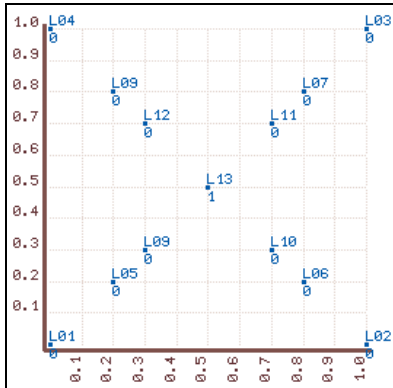

Fig. 4.4.1a: Distribution of points in the example OSCIL.

Firstly let us compare surfaces generated by the Minimum curvature, Kriging and ABOS method. It is obvious (see the next figure 4.4.1b) that the Minimum curvature method with the tensioning 0.1 and the Kriging method with the linear model and zero nugget effect produce very similar surfaces. The ABOS method with the smoothing parameter 9.0 produce similar surface only along diagonals as indicates figure 4.4.1b and 4.4.1c, but otherwise there are apparent differences in the shape of undesirable "circular" contours (for example between points L05 and L06).
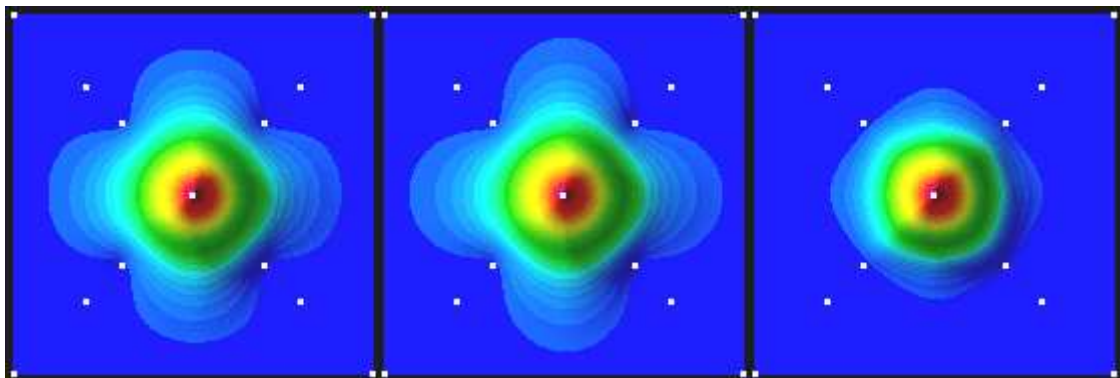

Fig. 4.4.1b: Surface generated by the Minimum curvature, Kriging and ABOS method.

In all three cases tested interpolation methods create a sharp local extreme at the point L13 as follows from the figure 4.4.1c.
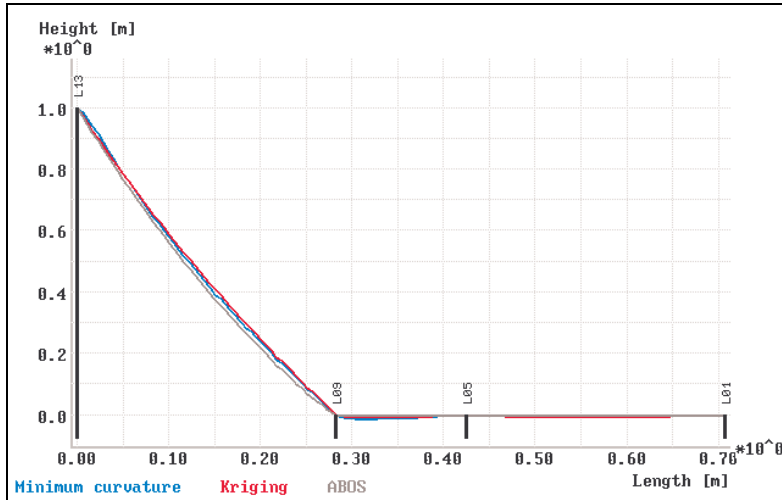
Fig. 4.4.1c: Cross-section through points L13, L09, L05 and L01.

Let's now examine details in the cross-section going through points L09, L05 and L01 where the oscillations were expected. As for size of oscillation between points L09 and L05, the Kriging method provides the best result while the Minimum curvature method provides the worst result. As for oscillation between points L05 and L01, both named methods still produce oscillations but in the opposite order – the oscillation size produced by the Minimum curvature method is slightly smaller than in the case of Kriging method.

The ABOS method produce worse result than the Kriging and better result than Minimum curvature method between points L09 and L05, but oscillations between points L05 and L01 are negligible. Moreover, if the LES smoothing is used during the interpolation process (see the paragraph **4.2.7 Smoothing**), the suppression of oscillations and improper extremes is very effective.
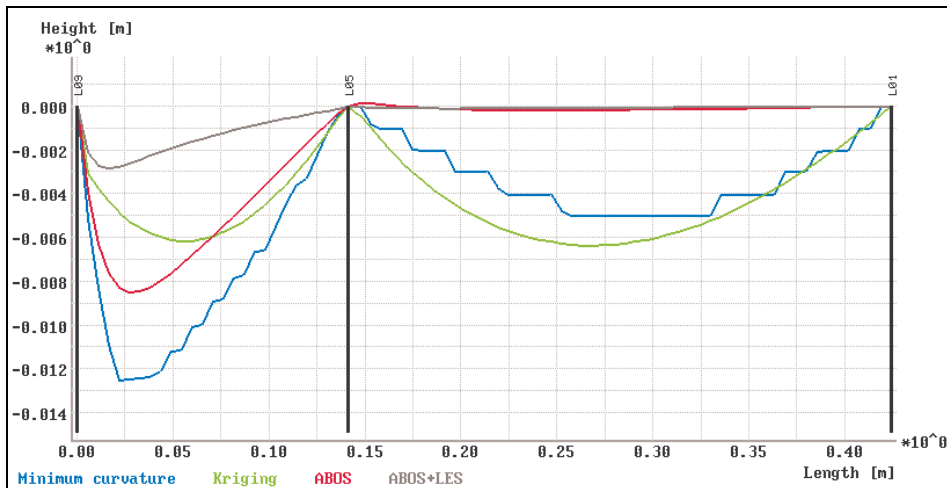

Fig. 4.4.1d: Cross-section through points L09, L05 and L01.

In any case, smooth interpolation may produce unwanted oscillations and improper extremes, which for example means that there are areas containing negative values in the solution of the interpolation problem, while only positive or zero values are possible. This situation is simply solved in the graphical user interface – there is a possibility to replace the grid node values below the specified constant (zero in our case) by this constant

4.4.2 Shape of generated surface

One of the first aspects evaluated by the user of interpolating technique is the shape of the created surface – the surface should be smooth, local extremes should be located at point positions, there should not be unsubstantiated bends on contour lines and so on.

For evaluation of these properties we will use data showed on figure 4.4.2a.
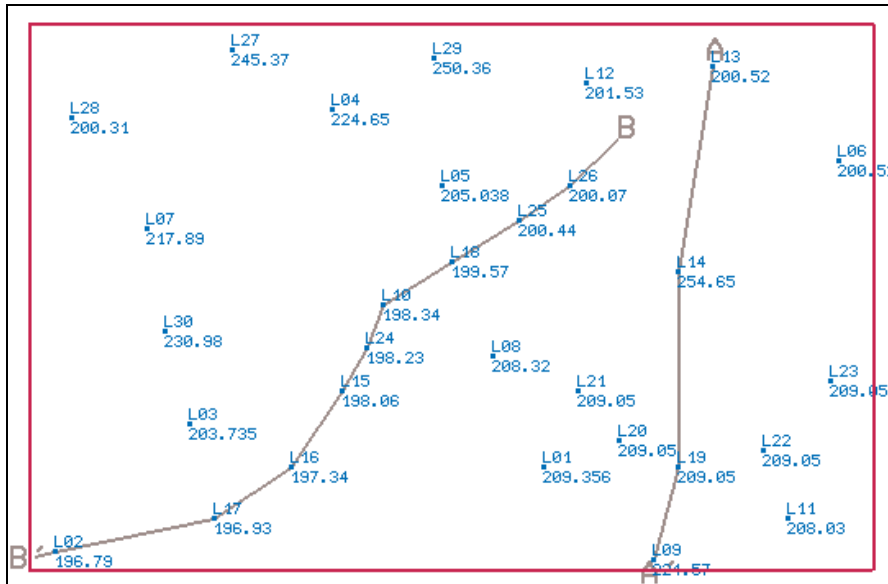

Fig. 4.4.2a: Distribution of points in the example SHAPE.

The next three figures show contours of surfaces created by the Minimum curvature method, Kriging and the ABOS method.
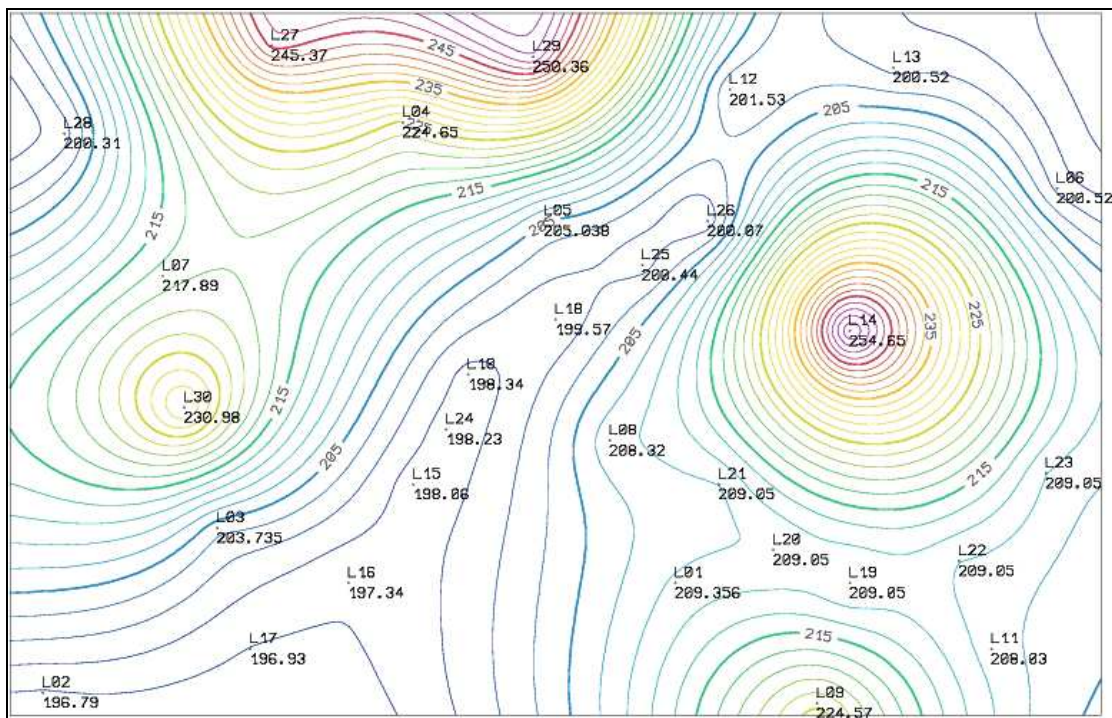

Fig. 4.4.2b: Interpolation of the SHAPE data set using the Minimum curvature method.
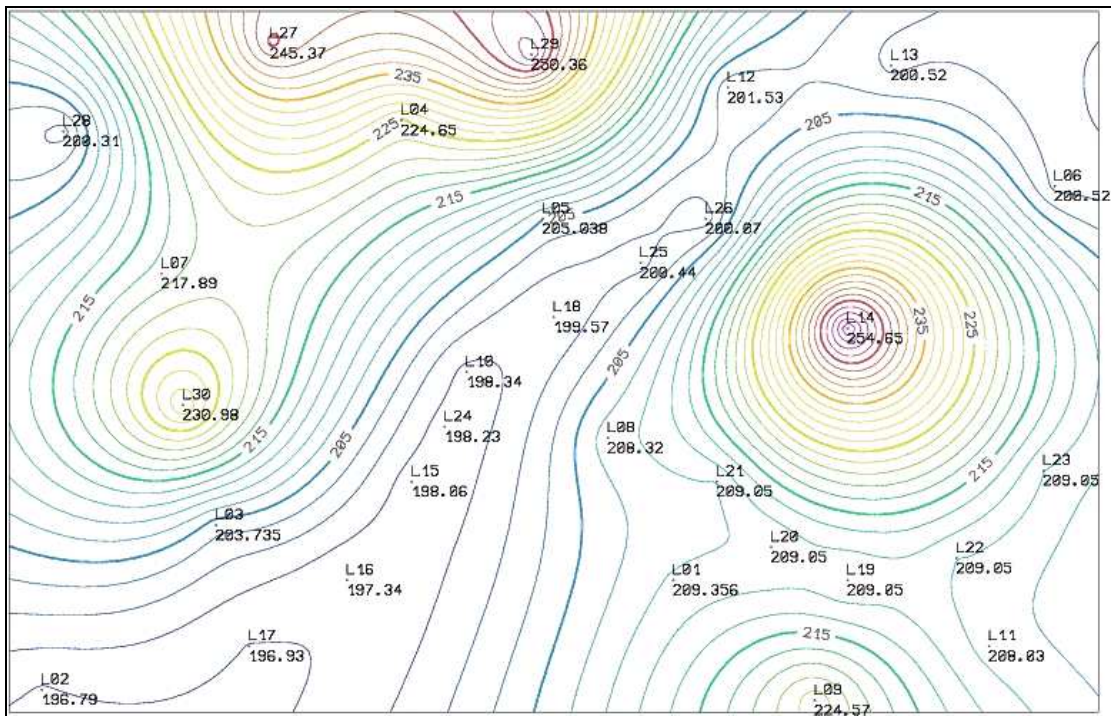
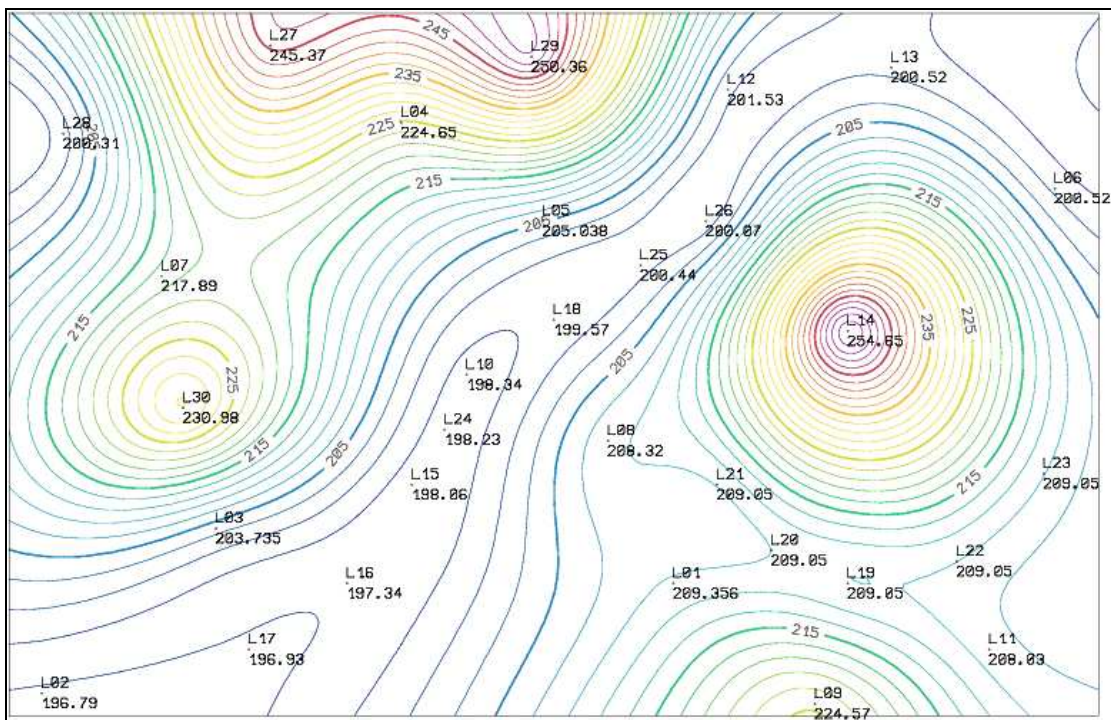Fig. 4.4.2c: Interpolation of the SHAPE data set using the Kriging method.

Fig. 4.4.2d: Interpolation of the SHAPE data set using the ABOS method.

The resulting function $f(x, y)$ can be considered as interpolative in all three cases – the maximal difference $max\{|f(X_i, Y_i) - Z_i|, i = 1,...,30\}$ does not exceed 0,002.

It is apparent that contours generated by the ABOS method are not so curved in the surrounding of some points (L18, L25, L26, L12, L13, L06, L23, L22) as in the case of the Kriging or Minimum curvature method.

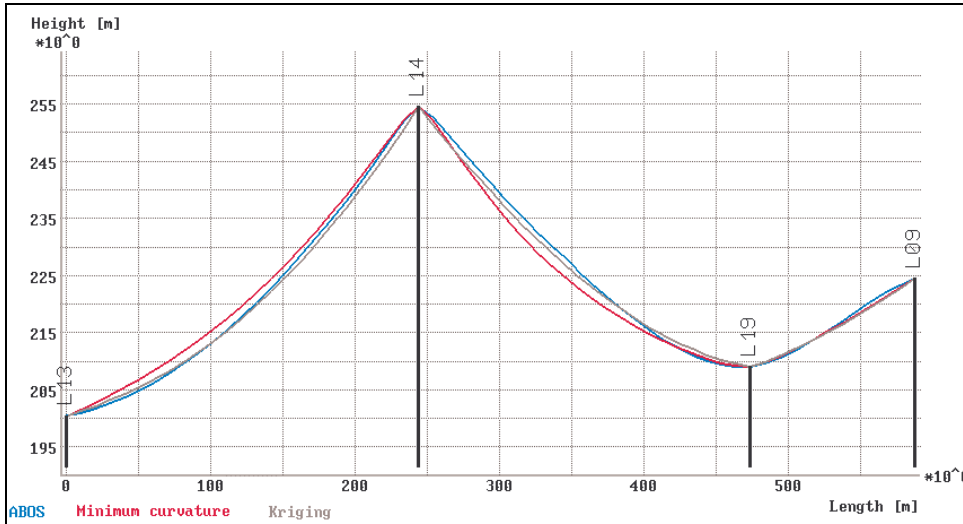Cross-sections provide another interesting comparison:

Fig. 4.4.2e: Cross-section A-A' through points L13, L14, L19 and L09.
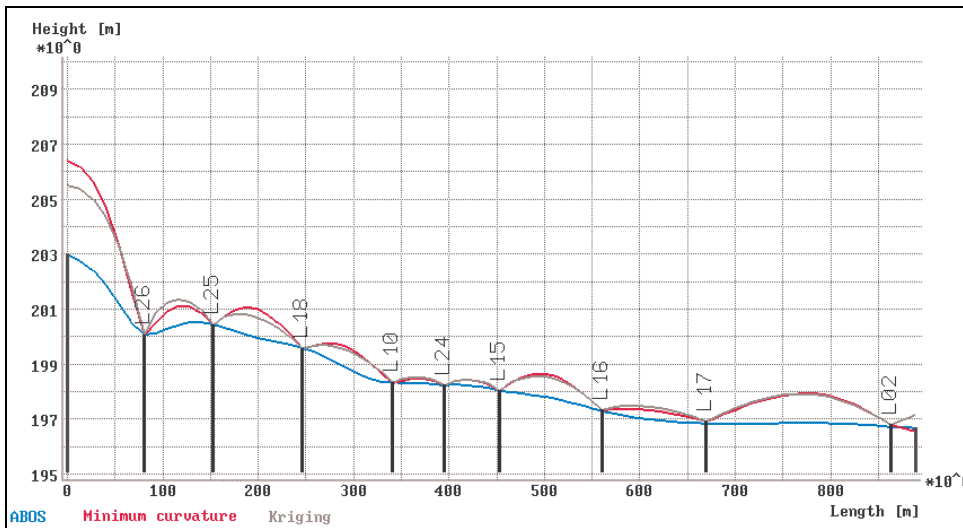
Fig. 4.4.2f: Cross-section B-B' through points L26, L25, L18, L10, L24, L15, L16, L17 and L02.

The cross-section A-A' (see figure 4.4.2e) shows that there is not significant difference between compared methods especially between the ABOS and Kriging method. Another situation is apparent in the case of cross-section B-B' leading through the "valley" (see figure 4.4.2f). The Minimum curvature method and Kriging produce "hills" between points although there is no reason for them.

27

4.4.3 Conservation of an extrapolation trend

The goal of the third example with the SIBIR data set (see the next figure) is to demonstrate the ability of tested methods to conserve trend in the regions without points.

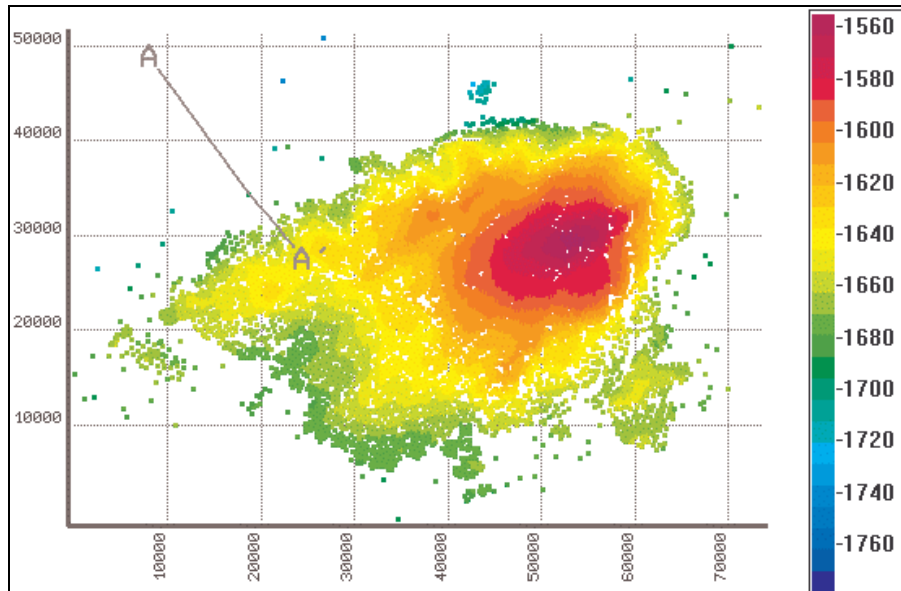Fig. 4.4.3a: Distribution of points in the example SIBIR.

Again, the next three figures enable to compare the surface generated by the Minimum curvature, Kriging and ABOS methods. In the case of the ABOS method the interpolation with trend function was applied, which is the procedure implemented in the graphical user interface SurGe.
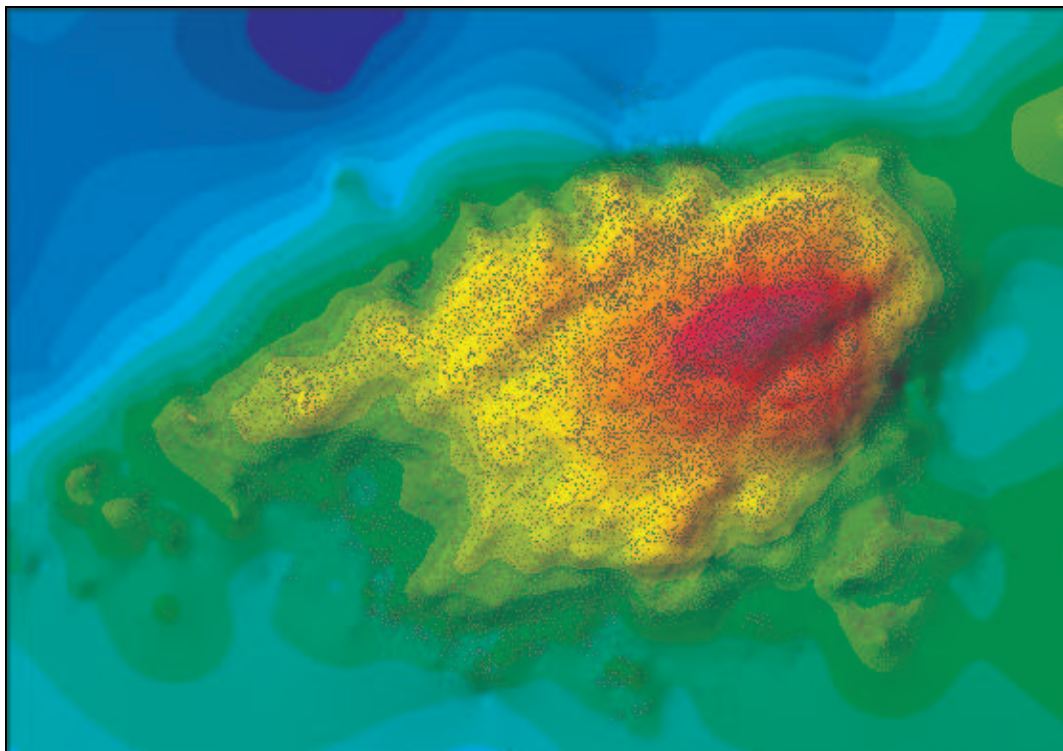

Fig. 4.4.3b: Interpolation of the SIBIR data set using the Minimum curvature method.
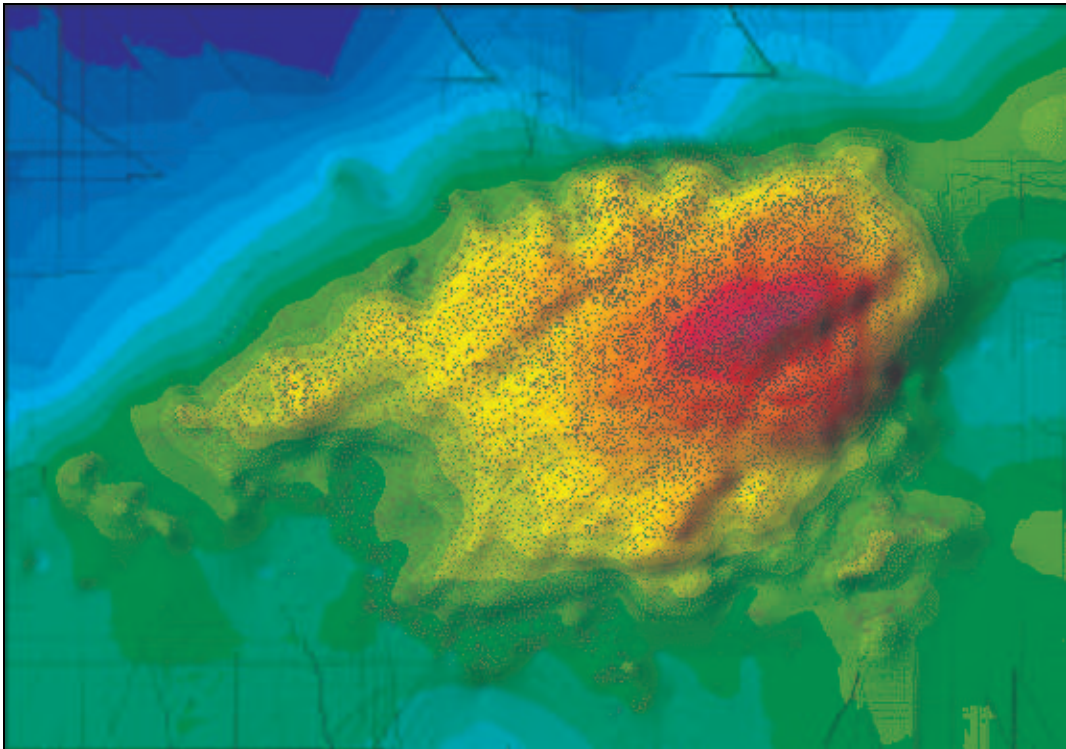
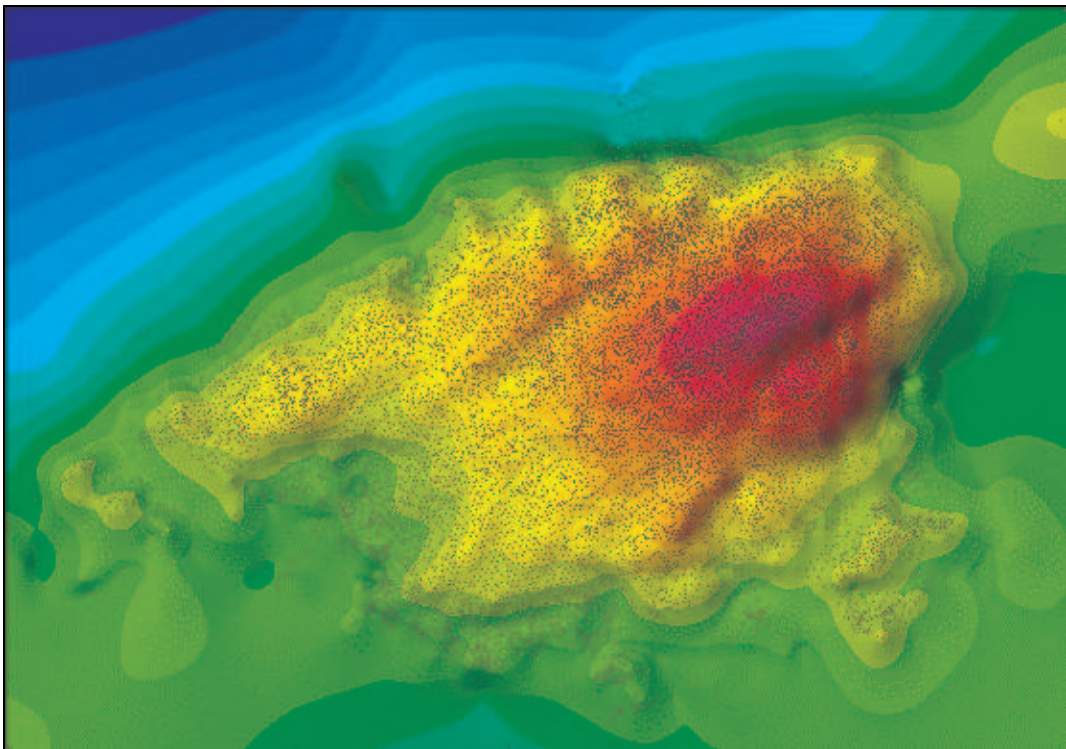Fig. 4.4.3c: Interpolation of the SIBIR data set using the Kriging method.

Fig. 4.4.3d: Interpolation of the SIBIR data set using the ABOS method.

All three methods generate apparently almost the same surface in the area densely covered by points *XYZ*, but at peripheral parts there are differences.

The surface created by the Minimum curvature method is smooth but it tends to be flat at the peripheral parts without points and even increases (see the upper right corner of the map in figure 4.4.3b) while it should decrease.

The Kriging method produces apparent "jumps" (discontinuities) at peripheral areas as expected due to the selection of points (max. 64) falling into the specified circle surrounding with a radius of 44600 units. The radius and maximal number of selected points was set by the used software – if one or both of these parameters were smaller, the jumps would be even higher. Without respect to the discontinuities, the ability of the Kriging method to conserve the extrapolation trend is better than in the case of the Minimum curvature method.

The surface created by the ABOS method does not have the deficiencies observed in the previous two cases and the cross-section A-A' confirms that the conservation of extrapolation trend is the best (see the next figure).
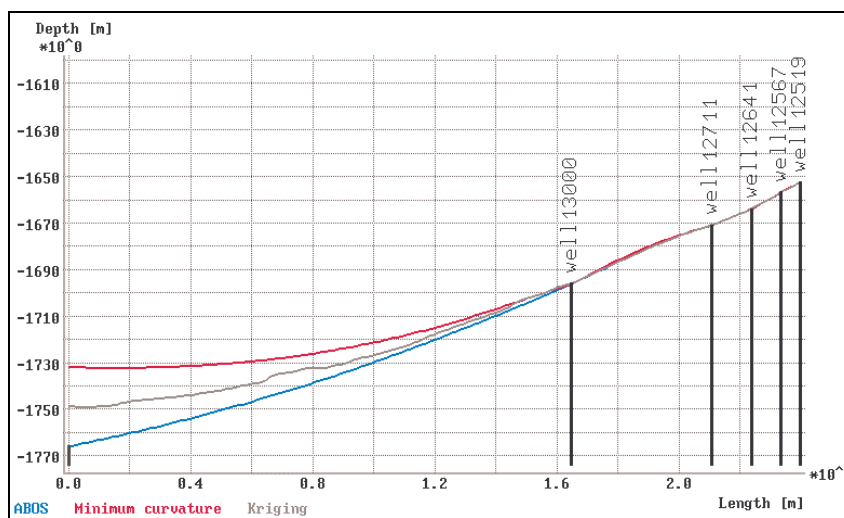
Fig. 4.4.2e: Cross-section A-A'.

4.4.4 Speed of interpolation

It has no sense to compare the speed of interpolation methods for small problems i.e. if the number of points is up to a few thousand, because all tested methods finish within a few seconds. However, if the number of points is greater, the computational speed becomes an important issue.

In the following list there are computational times consumed during solving interpolation with the SIBIR data set mentioned in the previous paragraph.

The Triangulation with linear interpolation …   5 seconds
The Natural neighbour method   …   16 seconds
The Inverse distance method   … 350 seconds
The Radial basis functions method   … 515 seconds
The Minimum curvature method   …   11 seconds
The Kriging method   … 530 seconds
The ABOS method   …   18 seconds

The time was, of course, measured on the same computer with the CPU Intel Pentium M 1.6GHz and memory 512 MB RAM.

The time consumed by the Kriging method and the Radial basis functions method may seem to be curious, but with respect to the selected grid consisting of 737x513 nodes and the maximal number of selected points 64, the system of up to 64 linear equations had to be solved 378081 times.

4.5. Summary

The presented examples show that the ABOS method provides results comparable with highly sophisticated interpolation / approximation methods such as Minimum curvature, Kriging or Radial basis functions even if its mathematical basis is very poor. Moreover, ABOS is able to solve large problems without necessity to search points in specified surrounding of grid nodes and to solve systems of equations. As for computational speed, ABOS is about 20 times faster than Kriging or Radial basis function method.

5. Conclusion

The most important result of this work is the design and especially the computer implementation of the new interpolation / approximation method ABOS for digital generation of surface passing through Z coordinates of points irregularly distributed in 3D Euclidean space. In contrast to sophisticated interpolation methods such as Radial basis functions, Kriging or Minimum curvature, the ABOS method does not fulfil any mathematical formulation such as linear combination of basis functions, statistical formulation of the best linear unbiased estimate or requirement of minimal curvature of the resulting surface. Instead, it provides tools for modelling the resulting surface based on numerical tensioning and smoothing enabling to create a broad range of surface shapes and to accommodate the resulting surface according to the user's conception. Moreover, the ABOS method makes it possible to solve large problems without the necessity to search points in a specified surrounding of the grid nodes and without the necessity to solve a system of equations. As for computational speed, ABOS is up to twenty times faster than methods based on solving equation systems, for example the Kriging method.

After more than three-years of publication through the medium of Internet WWW pages (see [13]) and through the medium of more than 300 software distributors (see *List of selected SurGe distributors* in *References*), the ABOS method and SurGe software has proven wide applicability. Now thousands of people worldwide use the software not only in traditional fields as reservoir engineering, geology and geophysics, but also in hydrology, geodesy, archeology, bathymetry, gravimetry, genetics, neurology, electronics, aerodynamics or nuclear physics; some people use SURGEF.EXE as an interpolation engine in their own mapping software (see [XXVI]).

The first part of *References* contains references to some of the more than 1800 people who contacted me during the last three years to discuss or solve interpolation problems, to suggest improvements to SurGe or to explain inexplicit descriptions in the SurGe documentation.

References

Selected contacts to SurGe users

[I] P. E. Pingrey: ***Forest mapping Geographic Information Systems.***
Wisconsin, USA
Paul.Pingrey@dnr.state.wi.us, http://www.forestpal.com/,
http://www.forestpal.com/Toolbox_mappers.html#surge

[II] J. Parkkinen: ***Spatial statistical and structural modeling of the Keivitsa Ni-Cu-PGE deposit, northern Finland.***
Parkkinen GeoConsulting, Alavus, Finland
jyrki_parkkinen@yahoo.co.uk, http://koti.mbnet.fi/parkkine/mikropc/

[III] W. Wieselquist: ***Neutron transport on unstructured grids.***
Department of Nuclear Engineering, North Carolina State University, USA
william.wieselquist@gmail.com

[IV] J. McLean: ***Electrical resistivity surveys of karst features near Fort Stanton, Lincoln County, New Mexico.***
Franktown, Colorado, USA
johnmclean@interfold.com

[V] M. Ratkiewicz: ***Construction of synthetic genetic maps.***
Instute of Biology, University of Bialystok, Poland
ermi@uwb.edu.pl

[VI] T. Church: ***Contour maps of phosphate levels on archaeological sites in the South Dakota.***
Lone Mountain Archaeological Services, Inc., El Paso, Texas, USA
tchurch@lone-mtn.com, http://www.archaeological.org/

[V] M. Chesters: ***Recording the eroding archeology of Scotland's coasts.***
Great Britain
marian@ronaldsay.plus.com, http://www.shorewatch.co.uk/

[VI] S. J. Morris: ***Calculation of new snow volumes within the avalanche event.***
New Zealand
simon.morris@xtra.co.nz

[VII] F. Jallon: ***Mapping of mud layer using laser triangulation process and computation of mud volume.***
Trappes, France
frederic.jallon@esic-sn.fr

[VIII] M. Sieczka: ***Interpolation of DEM files including faults.***
Institute of Plant Biology, Wroclaw University, Poland
sieczka@biol.uni.wroc.pl

[IX] E. Rangelova: *Implementation of Geographic Information Systems.*
Department of Geomatics Engineering, University of Calgary, Canada
evrangel@ucalgary.ca, http://www.geomatics.ucalgary.ca

[X] M. Isaac: *Interpretation of aerogeophysical anomalies of southwestern Tanzania.*
University of Dar Es Salaam, Tanzania
marobhe@udsm.ac.tz

[XI] D. Brozik: *Lectures in mapping and gridding methods.*
Marshal University, Huntington, USA
brozik@marshall.edu, http://www.marshall.edu/

[XII] J. Gottfried: *Lectures in mapping and gridding methods.*
Department of Earth Resources Engineering, Faculty of Engineering, Kyushu University, Japan
jgottfried@iol.cz

[XIII] A. Mariko: *Lectures on 2D interpolation methods and mapping techniques, Geophysical and hydrological research in Mali and Nigeria.*
DER Géologie, École Nationale d'Ingénieurs (ENI), Bamako, Mali
amariko@ird.fr, http://www.mali.ird.fr/

[XIV] M. Marzano: *Development of interpolation and approximation methods.*
Department of Computer Sciences, Universidad de Buenos Aires, Argentina
mmarzano@fibertel.com.ar

[XV] Daju Dali: *Construction of iso-resistivity maps.*
Ahmadu Bello University, Zaire, Nigeria
ddaju@yahoo.com,

[XVI] A. Urseguia: *Sonar bathymetric measurements of sea-floor.*
Barcelona, Spain
ANDRESURSUEGUIA@terra.es

[XVII] D. Nikolic: *Tests of metal layers thickness on silicon chips.*
Lexington, Massachusetts, USA
dgn@cypress.com, www.cypress.com

[XVIII] R. Coben: *Temperature distribution in the frontal part of brain.*
New York, USA
robcoben@optonline.net, http://providers.eegspectrum.com/US/RobertCobenPhD/

[XIX] M. Dmitrievsky: *Mapping of geological structures in Siberia, development of original interpolation / approximation method.*
Russia
mishadm@mail.ru, http://surfit.sourceforge.net/

[XX] A. Marosszeky: *Gridding / contouring of airborne Gravity and Magnetic data.*
Australia
amoscs@tpg.com.au

[XXI] K. Niklass: *Models and extrapolation of aerodynamic resistance maps for racing cars.*
white-smoke@tirol.com

[XXII] A. Palmisano: *Map of the Colo River area in New South Wales.*
Australia
a_palmo2003@hotmail.com

[XXIII] D. Bruehl: *Resources for geological and hydrogeological mapping.*
dhb@geologicresources.com, http://www.geologicresources.com/

[XXIV] Prof. K. Loga: *Research of Bain Blood Flow by Transcranial Doppler (TCD).*
Neurosurgeon, Chennai, Tamil Nadu, India
logamuthu@gmail.com

[XXV] G. Ebert: *Grids for flood hazard maps.*
ERNST+CO, Freiburg, NSR
gernot.ebert@ernst-co.de, http://www.ernst-co.de/

[XXVI] D. Wilbourn: *ArcheoSurveyor – software for geophysical surveys.*
DW Consulting, Barneveld, The Netherlands
dwilbourn@tiscali.nl, http://www.dwconsulting.nl/

Software and data source

[S1] Generic Mapping Tools, http://gmt.soest.hawaii.edu/

[S2] Golden Software, http://www.goldensoftware.com/products/surfer/surfer.shtml

[S3] K. L. Clarkson, http://cm.bell-labs.com/cm/cs/who/clarkson/2dch.c

[S4] LandSerf, http://www.landserf.org/

[S5] Global Mapper, http://www.globalmapper.com/

[S6] ESRI Shapefile, http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf

[S7] Inno Setup, http://www.jrsoftware.org/

[S8] Atlas DTM, http://www.atlasltd.cz/

[S9] USGS, http://rockyweb.cr.usgs.gov/nmpstds/demstds.html

[S10] Schlumberger, http://www.slb.com/content/services/software/reseng/index.asp

[S11] Mathworld, http://mathworld.wolfram.com/VoronoiDiagram.html

List of selected SurGe software distributors

http://www.simtel.net/product.php[id]77464[sekid]0[SiteID]simtel.net
http://www.download3k.com/search/surge.html
http://www.downloadplanet.net/info37162.htm
http://www.fileboost.net/directory/education/science/surge/029209/review.html
http://www.findmysoft.com/SurGe-download.html
http://www.icq-monitor.com/SurGe-download_9334.html
http://www.softaward.com/4965.html
http://www.softpicks.net/software/SurGe-8048.htm
http://superdownloads.uol.com.br/download/99/surge/
http://www.robdelorenzo.com/win/science-cad/surge-5-70./
http://www.softlookup.com/download.asp?id=60919
http://www.softpedia.com/get/Science-CAD/SurGe.shtml
http://www.softreviewer.com/info37162.htm
http://www.devgalaxy.com/program-j-x-r-x-i.aspx
http://www.download-up.com/download-SurGe-4532.html
http://www.programurl.com/surge.htm
http://www.sharewareconnection.com/surge.htm
http://www.softhypermarket.com/SurGe-download_9334.html
http://www.softizer.com/show_product/home_education/mathematics/surge/
http://www.wowload.com/loadnow-SurGe-259.html

http://ua.t.u-tokyo.ac.jp/okabelab/freesat/LINKS/RESULT_LINKS/surface_SurGe.htm
http://www.nr.no/pages/sand/wwwlinks
http://surfit.sourceforge.net/links.html
http://www.icomvis.una.ac.cr/telesig/sitios_interes.htm
http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1360692
http://www.geotrack.com.br/pdown.htm
http://w3.univ-tlse2.fr/geoprdc/bazarcarto/index.php?m=c&c=9
http://www.ruraltech.org/gis/map_info/gis_links/
http://www.telefonica.net/web2/petiandres/petibati.htm
http://typhoon.mines.edu/zipfiles/
http://www.geotests.net/mambo/index.php?option=com_content&task=view&id=20&Itemid=2

Bibliography

[1] W.H.F. Smith, P.Wessel, *Gridding with continuous curvature splines in tension*, Geophysics, Vol.55, No.3, p.293-305, 1990

[2] D.G. Krige, *A statistical approach to some basic mine valuation problems on the Witwatersrand*, Journal of the Chemical, Metallurgical, and Mining Society of South Africa, vol. 52, pp. 119-139, Dec. 1951

[3] C. V. Deutsch, A. G. Journel, *GSLIB. Geostatistical Software Library and User's Guide*. Oxford University Press, New York, NY, 1998

[4] T. H. Mayer, *The Discontinuous Nature of Kriging Interpolation for Digital Terrain Model*, Cartography and Geographic Information Science, Volume 31, Number 4, 2004, pp. 209-216(8)

[5] C. de Boor, *A practical guide to splines*, Springer-Verlag., 1978

[6] Ch. Yang, S. Kao, F. Lee, P. Hung, *Twelve Different Interpolation methods: A Case Study of Surfer 8.0,* Geo-Imagery Bridging Continents, XXth ISPRS Congress, 12-23 July 2004 Istanbul, 2006

[7] N. Sukumar, *A Note on Natural Neighbor Interpolation and the Natural Element Method (NEM),* Theoretical and Applied Mechanics, Northwestern University, 1997

[8] H. Mitasova, L. Mitas, *Interpolation by Regularized Spline with Tension: I. Theory and Implementation*, Mathematical Geology, 25, 641-655, 1993

[9] I.C. Briggs, *Machine Contouring using minimum curvature*, Geophysics, 39, 39-48, 1974

[10] M. Dressler, *Computation and presentation of surface in geology*, Zemní plyn a nafta, 40, 4, 317-323, 1996

[11] J. Hora, M. Dressler, V. Wasserbauer, *Model Diamo for ground-water flow and transport of chemicals*. Proceedings from Environmental Statistics and Earth Science, Brno, 1996.

[12] V. Onderka, M. Dressler, *STRAZ BLOCK GROUNDWATER POLLUTION - PHASE I*, GeoGas Brno, 1995.

[13] M.Dressler, *Approximation / interpolation of surfaces based on numerical tensioning and smoothing,* http://www.sweb.cz/M.Dressler/ABOS.htm

Anotace

Předkládaná disertační práce se zabývá metodami pro interpolaci bodů v třírozměrném euklidovském prostoru pomocí spojité funkce dvou nezávisle proměnných. V první části jsou popsány matematické základy nejčastěji používaných metod zpravidla založených na exaktních matematických formulacích jako je lineární kombinace bázových funkcí, statistická formulace nejlepšího nestranného lineárního odhadu či požadavek minimální křivosti výsledné funkce. U každé metody jsou zhodnoceny její přednosti, nedostatky a omezení a dále jsou uvedeny disciplíny, ve kterých se metoda využívá.

Cílem druhé části práce je navrhnout a implementovat novou interpolační metodu ABOS (Approximation Based On Smoothing), která by eliminovala omezení existujících metod, byla dostatečně robustní a natolik flexibilní, aby umožnila snadno realizovat představy člověka interpretujícího nějaké měření – ať už je to komplex geologických a seismických měření, rozložení teploty, výška sněhové vrstvy, koncentrace látek ve zvodni nebo digitální model terénu.

Nová metoda není založena na matematické definici vlastností výsledné interpolační funkce. Místo toho poskytuje nástroje pro modelování tvaru plochy (tři typy vážených průměrů umožňujících „napínat" a „vyhlazovat" plochu mezi interpolovanými body), jejichž vhodnou kombinací spolu s nastavením přesnosti lze dosáhnout jak hladké interpolace nebo aproximace, tak i interpolace s ostrými lokálními extrémy.

Motivací pro vývoj nové metody nebyla pouze eliminace omezení existujících metod a dostatečná flexibilita, ale i rychlost interpolace při řešení velkých interpolačních úloh, kdy počet bodů dosahuje desítek až stovek tisíců – metoda ABOS je v takových případech až dvacetkrát rychlejší než metoda Kriging, která dala vzniknout v padesátých létech minulého století novému vědnímu oboru geostatistice.

Počítačová implementace metody ABOS se skládá ze dvou základních programů. První, napsaná v jazyce FORTRAN 77, realizuje vlastní interpolační algoritmus a druhá, napsaná v jazyce C, poskytuje grafické uživatelské rozhraní pro zobrazení a snadnou manipulaci jak se vstupními daty, tak i s výslednými plochami.

Prakticky byla metoda ABOS využita k vytvoření geologických modelů zásobníků plynu v české republice (Dolní Dunajovice, Štramberk, Lobodice, 9.-11. sarmatský obzor zásobníku Tvrdonice), které pak byly aplikovány v třírozměrném dvoufázovém matematickém modelu proudění tekutin v porézním prostředí ECLIPSE. Dále byla metoda použita ke konstrukci modelů zvodnělých geologických struktur při řešení sanačních projektů v lokalitách CHZ Sokolov, Stráž pod Ralskem a Licoměřice.

Dnes, po více jak tříleté prezentaci metody ABOS a grafického uživatelského rozhraní SurGe prostřednictvím stránek WWW, je metoda využívána tisíci uživateli po celém světě v nejrůznějších oblastech – kromě tradičního využití v ložiskovém inženýrství, geologii a geofyzice je to rovněž hydrologie, geodesie, archeologie, batymetrie, gravimetrie, genetika, neurologie, aerodynamika či jaderná fyzika.