# THE ESTIMATION OF ERRORS OF AREA MODELS DESCRIBED BY THE SHAPE FUNCTIONS BY THE MEANS OF NEURAL NETWORKS

*NEIRONU TĪKLA FUNKCIJAS KĻŪDU NOTEIKŠANAS APRAKSTS MASĪVA MODELĪ*

**Maria Mrówczyńska**
University of Zielona Góra, Institute of Building Engineering
65-516 Zielona Góra, ul. Z. Szafrana 2, Poland
Phone: +(48)683282636, e-mail: m.mrowczynska@ib.uz.zgora.pl

**Abstract**. *The article deals with the issue of estimation of the area models errors determined on the basis of a discrete points set with the given values of space coordinates (x, y, z). The object was assumed to be described by shape functions in the form of the elliptic paraboloid and the hyperbolic paraboloid. The digital task accomplishment consisted in the statistic verification of errors of the models determined by neural networks and by the accomplishment of adjustment tasks. Modeling by the means of neural networks was carried out by the unidirectional multilayer networks with the application of gradient methods of optimalization and by Resilientback Propagation algorithm (RPROP). The obtained results were compared with the following results of approximation of the second and the third degree of polynomial, the b-spline function and the kriging's method.*

**Key words**: *neural networks, gradient methods of optimalization, approximation method*

## Introduction

The problem of specifying the form of a surface over a particular set of points can be discussed from the geodesic point view in terms of statistical verification in two cases, namely [3,4]:

- points $P_i \ (i=1,2,\ldots,m)$, which hale been assigned theoretical coordinates $x_i, y_i, z_i \ (i=1,2,\ldots m)$,
- points $P_i \ (i=1,2,\ldots,m)$ with empirical coordinated $x_i, y_i, z_i \ (i=1,2,\ldots m)$.

If we put forward the hypothesis that the form function with $n$ parameters assumes the form

$$F(x,y,z)=0,$$

then for points with theoretical coordinates estimation of errors of the model consists in solving the problem of minimization

$$\min \sum_{i=1}^{m} vf_i^2 = \sum_{i=1}^{m} F^2(x_i, y_i, z_i),$$

where $vf_i \ (i=1,2,\ldots,m)$ represents the differences between the value of the function specified on the basis of theoretical coordinates and the value of the function obtained from the process of minimization.

When we consider empirical coordinates of points we will deal with an equalization task in the general form

$$\min \sum_{i=1}^{m} vx_i^2 + vy_i^2 + vz_i^2$$

at that

$$F\left(x_i^{obs} + vx_i, y_i^{obs} + vy_i, z_i^{obs} + vz_i\right) = 0 \ \text{ for } P_1,\ldots,P_m.$$

In this case errors of empirical coordinates are particularly important in terms of modeling objects.

In order to analyses estimation of errors of the model we have been trying to find the minimum function spread over the assigned set of points with theoretical coordinates. Two

cases of estimation of errors model obtained from the solution of the equalization task with regard to basis form functions have been analysed:

- equations of elliptical paraboloid $z = \frac{x^2}{a^2} + \frac{y^2}{b^2}$ (fig .1a),
- equations of hyperbolical paraboloid $z = \frac{x^2}{a^2} - \frac{y^2}{b^2}$ (fig. 1b).

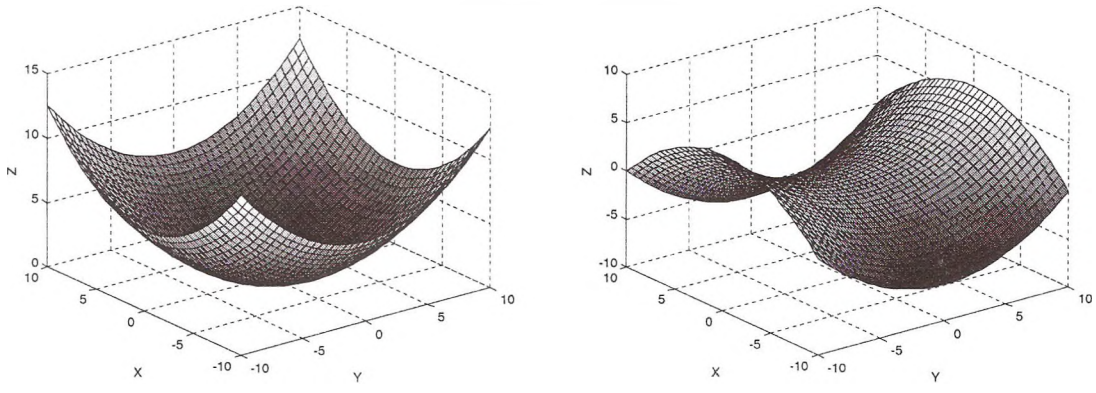


*Fig. 1a* *Fig. 1b*

*Fig.1.* **The elliptic paraboloid and the hyperbolic paraboloid for a=4 and b=4**

The solution of this problem facilitates a critical analysis of the effectiveness and precision of the representation of the terrain model by means of neural networks, which are a universal approximation system reflecting multidimensional data sets without the necessity to formulate a form function. However, the use of neural networks requires a suitable network structure, a particular number of learning standards for a general number of points and choice of a suitable activation function (it can be different in the hidden layer and output later).

## Materials and methods

Artificial neural networks are systems in the form of configurations of neurons, whose computing power makes it possible to achieve a representation from input space to output space [2,5].

The basis for the algorithms applied for teaching the network is an objective function (an energy function), defined by means of the Euclides algorithm as the sum of the squares of differences between the current values of input signals of the network and the allocated values in the form:

$$E = \frac{1}{2}\sum_{i=1}^{p}\|y_i - d_i\|^2 = \frac{1}{2}\sum_{i=1}^{p}\sum_{j=1}^{M}\left(y_i^j - d_i^j\right)^2 \qquad (1)$$

where:

$p$ - number of input vectors,

$M$ - number of coordinates of the input vector,

$y_i^j$ - coordinate of the input vector $(i = 1,2,...,\mathrm{p})$, $(j = 1,2,...,\mathrm{M})$,

$d_i^j$ - coordinate of the allocated vector $(i = 1,2,...,\mathrm{p})$, $(j = 1,2,...,\mathrm{M})$.

On the assumption that the objective function is continuous and gradient optimisation methods are used, the adaptation of the vector of weights is carried out according to the rule:

$$w_{ij}^{(k+1)} = w_{ij}^{(k)} + \Delta w_{ij}^{(k)} \qquad (2)$$

where $k$ is the number of a subsequent iteration.

In the process of the minimization of the objective function and the sigmoidal activation function adopted, the increase of coordinates of the vector of weights

$$\Delta w = -\eta p(w) \tag{3}$$

where $p(w) = -\frac{\partial E}{\partial w}$, is a direction in the multidimensional space $w$, and $\eta$ is the learning coefficient.

In order to obtain similarities with the optimum solution the following quasi Newtonian methods have been used: the method of a quasi-Newton algorithm, conjugate gradients, and the Levenberg – Marquardt method.

The algorithm carried out via the quasi-Newton method uses information on the curve of the objective function being minimized. In quasi-Newton methods the hessian matrix is approximated by means of the difference of first rank derivatives. This method does not require the difficult-to-satisfy (in general) condition of the positivity of the hessian in each iteration, which facilitates the practical implementation of the algorithm. We will be looking for a stationary point $w^*$ of the minimum of the objective function $E(w)$ in the direction

$$p^{(k)} = -\left[G(w)^{(k)}\right]^{-1} g(w)^{(k)}, \tag{4}$$

and the reverse matrix of the approximated hessian $V^{(k)} = \left[G(w)^{(k)}\right]^{-1}$ as a matrix modified from the previous iteration (the starting value $V^0 = I$) has been described with the Davidon – Fletcher – Powell recurrent dependence [5]

$$\mathbf{V}^{(k)} = \mathbf{V}^{(k-1)} + \frac{s^{(k)}\left(s^{(k)}\right)^T}{\left(s^{(k)}\right)^T r^{(k)}} - \frac{\mathbf{V}^{(k-1)} r^{(k)}\left(r^{(k)}\right)^T \mathbf{V}^{(k-1)}}{\left(r^{(k)}\right)^T \mathbf{V}^{(k-1)} r^{(k)}} \tag{5}$$

where $s^{(k)}$ and $r^{(k)}$ denote respectively the increase of the vector of weights $w$ and the gradient $g(w)$ in two subsequent iterations, $s^{(k)} = w^{(k)} - w^{(k-1)}$, $r^{(k)} = g(w)^{(k)} - g(w)^{(k-1)}$.

The Levenberg – Marquardt method is very similar to the quasi-Newton method, which also uses the square calculation of the objective function $E(w)$ and an approximated value of the Hessian $G(w)$ including a regularization factor. When the objective function is defined as (1) the approximated matrix of the Hessian has the form

$$G(w) = J(w)^T J(w) + R(w) \tag{6}$$

where:

$J(w)$ - Jacobian of the function (1)

$R(w)$ - summands of the expansion of the exact value of the hessian $H(w)$ by means of the regularization factor $r\mathbf{1}$ (or $r\mathbf{I}$).

After introducing the regularization factor into the formula (6) we obtain a form of the matrix of the hessian equivalent to the expression (6)

$$G(w)^{(k)} = J(w)^{(k)^T} J(w)^{(k)} + r^{(k)}\mathbf{1} \tag{7}$$

The efficiency of the algorithm depends on the choice of the scalar coefficient $r^{(k)}\mathbf{1}$. At the beginning of the learning process, when the value of the function $E(w)$ is great, the regularization factor $r^{(k)}\mathbf{1}$ assumes great values. As error is reduced and the solution gets closer the parameter $r^{(k)}\mathbf{1}$ is reduced down to zero.

The method of connected gradients uses the square model of the objective function without the necessity to use a number of matrix calculations in each iteration. The directions $p_1, p_2, \ldots, p_n$ are called directions connected to the symmetrical, strictly positive matrix $G$, if

$$\boldsymbol{p}_i^T \boldsymbol{G} \boldsymbol{p}_j = 0 \qquad i \neq j \qquad (i, j = 1,2,\ldots,n) \tag{8}$$

Directions of the connections are generated on the basis of information about new derivatives and former directions. Using concise notation including the assumption (8), the direction vector in the $k^{\text{th}}$ iteration

$$\boldsymbol{p}^{(k)} = -\boldsymbol{g}(\boldsymbol{w})^{(k)} + \beta^{(k-1)} \boldsymbol{p}^{(k-1)} \tag{9}$$

where:

$\boldsymbol{g}(\boldsymbol{w})^{(k)}$ - gradient vector,

$\beta^{(k-1)}$ - connection coefficient.

The coefficient $\beta^{(k-1)}$ should be chosen in such a way that the direction $\boldsymbol{p}^{(k)}$ is connected to the direction $\boldsymbol{p}^{(k-1)}$. The best known theorem for specifying the coefficient $\beta^{(k-1)}$ (a method by the Polish scientist Riebiere) is expressed with the formula [5]:

$$\beta^{(k-1)} = \frac{\boldsymbol{g}(\boldsymbol{w})^{(k)T} \left( \boldsymbol{g}(\boldsymbol{w})^{(k)} - \boldsymbol{g}(\boldsymbol{w})^{(k-1)} \right)}{\left( \boldsymbol{g}(\boldsymbol{w})^{(k-1)} \right)^T \boldsymbol{g}(\boldsymbol{w})^{(k-1)}} \tag{10}$$

The essence of the RPROP algorithm (Resilient Backpropagation algorithm) consists in updating weights according to the sign of only the summands of the gradient, regardless of its value [1,5].

Correction of weights is effected according to the dependence

$$\boldsymbol{w}^{(k+1)} = \boldsymbol{w}^{(k)} - \eta^{(k)} \operatorname{sgn}\left( \nabla E(\boldsymbol{w})^{(k)} \right), \tag{11}$$

where the learning coefficient $\eta$ depends on the sign of the gradient. The coefficient $\eta$ is chosen in each cycle for each weight $w$ individually. The value of this coefficient increases when the sign of the gradient in two subsequent iterations is the same, if not it decreases. Thus

$$\eta^{(k)} = \begin{cases} \min\left( a\eta^{(k-1)}, \eta_{\min} \right) & \text{dla } \nabla E(\boldsymbol{w})^{(k)} \nabla E(\boldsymbol{w})^{(k-1)} > 0 \\ \max\left( b\eta^{(k-1)}, \eta_{\max} \right) & \text{dla } \nabla E(\boldsymbol{w})^{(k)} \nabla E(\boldsymbol{w})^{(k-1)} < 0 \\ \eta^{(k-1)} & \text{w innym przypadku} \end{cases} \tag{12}$$

The symbols $a$ and $b$ in the formulas are constants: $a = 1.2$, $b = 0.5$ and $\eta_{\min}$ and $\eta_{\max}$ denote respectively the minimum and maximum value of the learning coefficient, equal in the ROROP algorithm respectively $10^{-6}$ and 50 [5,6]

## Results and discussion

The representation of surfaces described by means of form functions – elliptical paraboloid and hyperbolical paraboloid has been effected for a training set of 20 points, a test set of 320 points (fig. 2a and 2b) for a network with the architecture 2_5_1 and 2_10_1 with the use of the bipolar activation function in the form

$$y = f(net) = \frac{1 - \exp(-\lambda net)}{1 + \exp(-\lambda net)}, \quad \lambda > 0 \tag{13}$$

In the learning process the change of the learning terror of two consecutive iterations 1e-10 and the number of iterations on the level of 20000 have been adopted as the criterion for the termination of the minimization process.

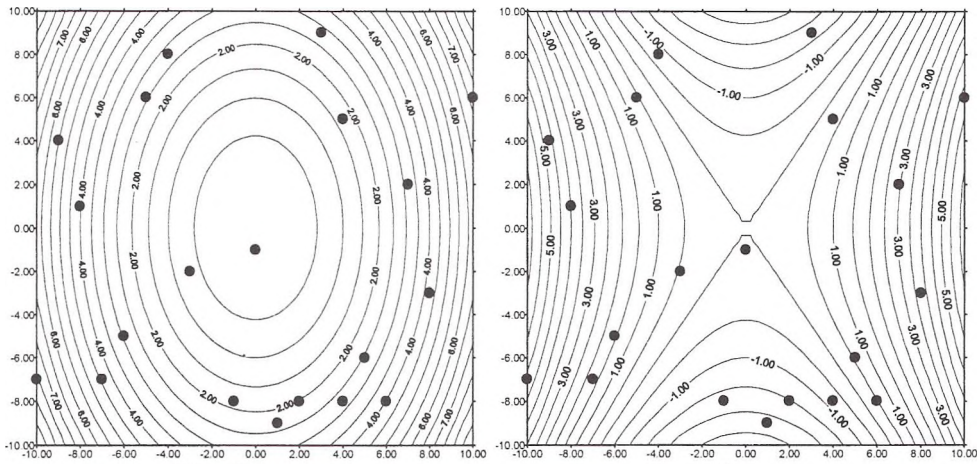

*Fig. 2a* *Fig. 2b*

*Fig.2.* **The layout of training set points – the elliptic paraboloid (2a) and the hyperbolic paraboloid (2b)**

The results of learning and testing the network in the form of the mean error

$$RMSE = \frac{1}{\sqrt{n}} \sqrt{\sum_{i=1}^{n} (d_i - y_i)^2} \tag{14}$$

and the results of obtained by means of spline and kriging approximations have been compared in table 1 and 2 an represented graphically in fig. 3.

*Table 1.*

**The results of the area models projection – the networks architecture 2 5 1**

| The approximation method | The elliptic paraboloid error *RMSE* [m] | | The hyperbolic paraboloid error *RMSE* [m] | |
|---|---|---|---|---|
| | training set | testing set | training set | testing set |
| RPROP algorithm | 0.04 | 0.07 | 0.06 | 0.07 |
| Backpropagation algorithm | 0.65 | 0.78 | 0.72 | 0.79 |
| Conjugate gradient algorithm | 0.17 | 0.19 | 0.25 | 0.28 |
| Quasi-Newton algorithm | 0.01 | 0.00 | 0.02 | 0.00 |
| Levenberga – Marquardta | 0.02 | 0.02 | 0.01 | 0.01 |
| Spline approximation | 0.02 | 0.03 | 0.01 | 0.02 |
| Kriging approximation | 0.07 | 0.10 | 0.07 | 0.08 |

*Table 2.*

**The results of the area models projection – the networks architecture 2 10 1**

| The approximation method | The elliptic paraboloid error *RMSE* [m] | | The hyperbolic paraboloid error *RMSE* [m] | |
|---|---|---|---|---|
| | training set | testing set | training set | testing set |
| RPROP algorithm | 0.02 | 0.04 | 0.05 | 0.06 |
| Backpropagation algorithm | 0.53 | 0.64 | 0.62 | 0.65 |
| Conjugate gradient algorithm | 0.12 | 0.14 | 0.20 | 0.24 |
| Quasi-Newton algorithm | 0.00 | 0.00 | 0.00 | 0.00 |
| Levenberga – Marquardta | 0.00 | 0.00 | 0.00 | 0.00 |
| Spline approximation | 0.02 | 0.03 | 0.01 | 0.02 |
| Kriging approximation | 0.07 | 0.10 | 0.07 | 0.08 |

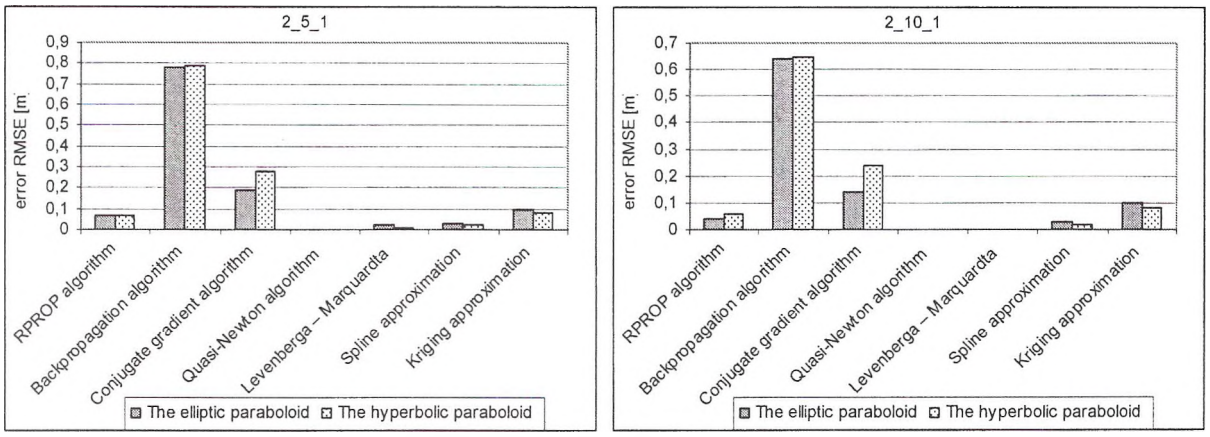


*Fig.3.* **The errors values *RMSE* [m] for the testing set depending on the approximation method and the networks architecture**

## Conclusions

The approximation of the square form of three real numbers $x, y, z$ (elliptical paraboloid and hyperbolic paraboloid) by means of second and third rate polynomials and by means of one direction multi – layer networks taught by means of the Levenberg – Marquardt changeable metric method brings the best results in terms of quality. Slightly greater errors are obtained by the use of the heuristic algorithm *RPROP* and spline and kriging approximation. At this point it is necessary to add that the algorithm *RPROP* requires approximation 10 times more iterations in order to achieve convergence than basic gradient methods. The worst results have been obtained by the use of the method of the greatest fall, because of linear convergence and slow progress of minimization around the optimum point.

On the basis of the results obtained it is possible to say that the approximation of surface models described by means of basic form functions does not differ in terms of quality form the approximation obtained by means of traditional methods. It is merely necessary to meet all the requirements concerning the obtainment of the local minimum, located near the global minimum.

### Bibliography

1. Duch W., Korbicz J., Rutkowski L., Tadeusiewicz R., Sieci neuronowe, Akademicka Oficyna Wydawnicza Exit, Warszawa 2000.
2. Faulkner J., Einführung in Neuronale Netze, Universität Tübingen 2001.
3. Nowak E., Estymacja i weryfikacja numerycznego modelu terenu. XI Konf. Naukowo – Techniczna „Systemy Informacji Przestrzennej", Warszawa 28 – 30 maj 2001.
4. Nowak E., Wyznaczanie kształtu poprzez estymację błędów pomiaru i modelu. V Konf. Naukowo – Techniczna „Problemy Automatyzacji w Geodezji Inżynieryjnej", Warszawa 29 – 30 marca 2001.
5. Osowski S., Sieci neuronowe w ujęciu algorytmicznym. WNT, Warszawa 1996.
6. Riedmiller M., Braun H., A fast adaptive learning algorithm, Technical Report, University Karslruhe, Germany 1992.