Miran Kuhar, Bojan Stopar, Goran Turk, Tomaž Ambrožič

The use of artificial neural network in geoid surface approximation

Die lokale Geoidfläche kann durch Näherungsfunktionen mit der Methode der kleinsten Quadrate modelliert werden oder – alternativ – Künstliche Neuronale Netze (KNN) können angewandt werden. Die KNN-Approximation für ausgewählte Testgebiete wird in diesem Beitrag vorgestellt.

Introduction

Nowadays the gravimetric method is most commonly used technique for the precise determination of the geoid. The precondition for its use is the presence of high resolution gravity data set. With the lack of gravity data the geoid could be determined by means of various geometric methods: astrogeodetic method or geoid heights from GPS in conjunction with levelling.

One such geometric technique uses simple relation between GPS *ellipsoid heights h*, levelled *orthometric heights H* (or *normal heights H^N*), and the so-called *geoid heights N* (or *height anomalies ζ*), (see Fig. 1):

$$N = h - H, \tag{1a}$$

$$\zeta = h - H^N. \tag{1b}$$

As seen in Fig. 1, the geoid height is the vertical distance from the ellipsoid to the geoid level surface. If a

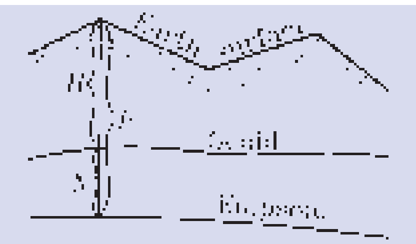

Fig. 1: Earth surface, geoid and ellipsoid

fairly large number of well distributed points with known ellipsoid and orthometric (or normal) heights are available for an area, Equation (1) can be used to determine N or ζ.

In order to derive local geoid we investigated two approaches of the geometric method: the first one is local geoid surface fitting model and the second is the use of artificial neural networks (ANN).

Local geoid surface fitting model

This method involves the development of a local geoid surface model using a surface fitting procedure. If a number of heights control stations exist in a GPS network which have both ellipsoidal and orthometric heights, the geoid height can be found directly from Equation (1).

Local geoid surface fitting model is accomplished by taking N as the function of the position of each height control stations in the network. The surface model obtains the following form:

$$\mathrm{N} = N(y,x) = A + By + Cx + Dy^2 + Eyx + Fx^2 + Gy^3 + Hy^2x + Ix^2y + Jx^3 + \ldots \tag{2}$$

where y and x represent horizontal coordinates of the height control points. The surface model can also be written in the form where geoid height differences are used instead of geoid heights itself.

If the coordinates in Equation (2) refer to the centre of gravity of the area covered by GPS network, coefficients have their geometrical interpretation [7]. A represents the parallel shift between ellipsoid and local geoid surface. The linear terms with coefficients B and C represent the difference in the inclination of the tangent plane to the ellipsoid and local geoid surface. Second degree coefficients represent the difference in the curvature of these two surfaces. The degree of the polynomial used for interpolating the surface depends on the number of available height control points. The more points there are the higher degree polynomial we could use. If there exist more points than coefficients, the interpolating surface is overdetermined and the coefficients has to evaluated by the least squares method. The size of the residuals at each control point and the standard deviation give an indication of how well the points fit the interpolating surface.

Depending on the number of control points, it is possible to generate more complex surfaces such as trigonometric functions or bicubic spline functions. However, these methods become unstable when the network points are irregularly spaced [5].

The number of points as well as their location and distribution over the area covered by the network are very important. It is recommended that height control points are well distributed geometrically throughout the project area. The number of height control points dictates the choice of the model.

As an alternative to the existing geometric models the *artificial neural network* (ANN) may be used. In the following section the basics of ANN are described.

Artificial neural network

The basic ideas and the motivation for the early developments of ANN was the study of the structure and processes in human brain. There exist several similarities between human brain and ANN. They both have units called *neurons* which are interconnected. Similarly to human brain ANN has to be taught or trained. There are two types of ANN learning procedures: supervised in which questions and answers are known and ANN has to learn the correct answers, and the unsupervised learning where the answers are not known.

One of the common definitions of ANN is: ANN is a network of simple units which operate locally. The units are connected by connections which may reduce or amplify the signal from one unit to another. Each unit receives signals from other units, processes these signals and transmits the signals to other units.

There are several types of ANN geometry. A review of different ANN is given in several papers, books and internet sites e.g. [9], [10], [12]. The *multi-layer feed-forward network* is usually chosen in functional approximation. Since it is our aim to approximate the geoid heights, the multi-layer feed-forward network and the supervised learning, were chosen in our research.

Multi-layer feed-forward network

The geometry of a multi-layer feed-forward neural network is shown in Fig. 2. Input units are connected to the first layer of hidden units which are further connected to the units of the second hidden layer. The units of the last hidden layer are connected to the output units. The multi-layer feed-forward networks are usually employed as the approximators of the unknown functional relation. In fact, it was shown that any continuous functions may be accurately approximated by the multi-layer feed-forward neural network [3], [6].

The input units represent the input data, whereas the output units represent the output data. The hidden layers may be considered as a black box which perform the necessary transformations of input data so that the expected output data are obtained.

Each connection between the units is represented by its weight w_{ij}^k, where index i corresponds to the unit number of $(k-1)^{\text{th}}$ layer, while index j corresponds to the unit number of k^{th} layer. The input layer is denoted by 0, whereas the output layer is denoted by n_l. The signals travel in only one direction, i.e. from the input layer toward the output layer. The value of a unit is multiplied by the corresponding weight and added to the value of the signal in the unit of the next layer. The sum of all signals from neurons in the previous layer is transformed by the activation function f(.)

$$y_i^k = \mathrm{f}(y_i'^k) = \mathrm{f}\left(\sum_{j=1}^{n_k-1} w_{ij}^k \, y_j^{k-1}\right) \tag{3}$$

Activation function f(.) is needed in order to enable modelling of an arbitrary non-linear relation between input and output units. Although many different functions could be successful activation function, usually a differentiable and bounded function is used. One of the usual choices of activation function is a sigmoid function $1/(1+e^{-y})$, $\tanh(y)$, or Gaussian. The results of the neural network depend on the values of the weights w_{ij}^k which have to be determined by the learning (training) procedure.

As mentioned above the supervised learning was employed in our research. Therefore, the values of input units and the corresponding output units are known. The set of known input and output values is termed as *input-output pair.*

All input-output pairs are usually divided into three sets. The first is termed as *learning* or *training set* which is used to determine the connection weights w_{ij}^k. The second, named *validation set,* is used to choose the optimal parameters of neural network, i.e. the number of hidden layers and the number of units in each hidden layer. Finally a chosen and taught neural network is tested, using the *testing set* of data.

When the learning procedure is completed, meaning that the neural network performs adequately for all input-output pairs in the learning set, the neural network is assessed on the validation set of input-output pairs and the optimal neural network is chosen.

For numerical reasons the values of input and output units have to be normalized. The normalization of the values of output units depends on the range of activation function. Usually, the linear transformation works well, although sometimes a non-linear transformation may is preferred if the data are clustered.

The supervised learning is in fact a general optimization problem in which the minimum of error E_p

$$E_p = \frac{1}{2}\sum_{i=1}^{n_o} (t_{pi} - y_{pi}^{n_l})^2 \tag{4}$$

is sought. We have to find weights w_{ij}^k which give minimum error E_p. The problem is not easy to solve since function E_p of many variables w_{ij}^k is nonlinear and may have a large number of local minima. In Equation (4) t_{pi} is the target output where as $y_{pi}^{n_l}$ is the output obtained by the neural network, n_o is the number of output units.

There are two essentially different approaches: *error back-propagation*

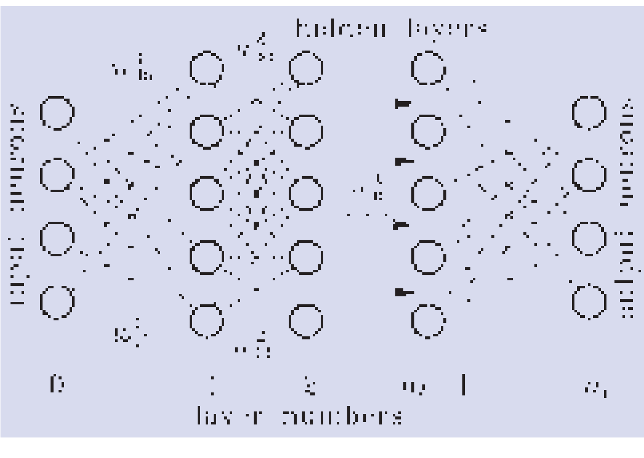

***Fig. 2:* Multi-layer feed-forward neural network**

algorithms which is a gradient method and *genetic algorithms* which is in fact a stochastic search [1], [4], [16]. There are many variations and combinations of the above mentioned method e.g. [2], [14]. If the number of weights is relatively small, the gradient method is a good choice. The error back-propagation (or "generalized delta rule" as it was termed by its authors [11]) is a gradient method in which the weights are changed for a chosen step size in the direction of the maximum descent for each input-output pair. However, there is always a possibility of finding only a local minimum which may not give satisfactory set of weights. One solution of this problem is simply to run the error back-propagation procedure for different starting points and then choose the best result. If the number of weights is larger, the genetic algorithms are better.

In the error back-propagation the weights are changed for a chosen *step size* ΔW in the direction of maximum descent of each input-output pair

$$\Delta w_{ij}^{k} = -\Delta W \frac{\partial E_p}{\partial w_{ij}}. \qquad (5)$$

The derivatives in Equation (5) are determined consecutively from the weights between the output layer and the last hidden layer towards the weight between the input layer and the first hidden layer by the chain rule. Therefore the weights are changed in this order, and the term error back-propagation makes sense, since the error $(t_{pi} - y_{pi})$ is gradually propagated from the output toward the input layer.

The procedure is repeated for each input-output pair until the error is smaller than prescribed for all input-output pairs. If the prescribed error is too small, *overfitting* may occur. Overfitting means that the neural network may reproduce input-output pairs used in the learning procedure, but fails to generalize them and may produce erroneous results for values of the input units for which the ANN were not trained.

There are two major difficulties when using error back-propagation: it is almost impossible to choose the optimal step size ΔW, and quite often the procedure converges to a local minimum. If the step size is too large, we may overshoot the global minimum. On the other hand, if the step size is too small, the convergence is very slow. Both difficulties may be overcome by different procedures with adaptive step size [8], or with the introduction of inertial term [9], [11].

The parameters, i.e. the number of hidden layers and the number of hidden neurons, of the optimal neural network are problem dependent. One of the methods to choose the right network is by using the validation set to determine which one performs best. However, some general guidelines can be given. If the number of units is very large the learning procedure may be very slow, since each forward calculation takes a substantial computational effort. Although larger networks are usually able to learn the sought relationship, this may sometimes be a drawback. A large network may easily reproduce the training set of input-output pairs but fails to generalize yielding to a poor testing performance. Networks with insufficient units may have problems to learn properly during the learning procedure.

Numerical examples

As an alternative to linear or second order approximation the artificial neural network is used to approximate the geoid heights as a function of coordinates y and x. Therefore, the input-output pair consists of horizontal coordinates y and x as input variables, and geoid height N (or height anomaly ζ) as single output variable.

ANN is trained by the error back-propagation procedure. The iterations are repeated until the relative difference between the ANN prediction and the target value of geoid height is lower than 5 % for all input-output pairs of the training set. The target relative difference may have been set lower than 5 %. However, in this case the overfitting may occur which means that ANN would not be able to generalize the approximation and the error in the testing data set would increase considerably.

Eastern Florida test area

The region of eastern Florida, covering the area of approximately $d\varphi = 0.6°$ and $d\lambda = 1.1°$ is shown in Fig. 3. The geoid heights in the area of interest vary from –29.7 to –28.7 m. The actual geoid heights are known in 71 points [13]. Among 71 points 23 (in *Fig. 3* marked with squares) were selected to form a validation set. The remaining 48 (in Fig. 3 marked with filled circles) were employed in the training procedure. The same 48 points were used for the 2nd order polynomial fitting of the geoid at the region. The training procedure of ANN was performed on different configurations of ANN. The best configuration was

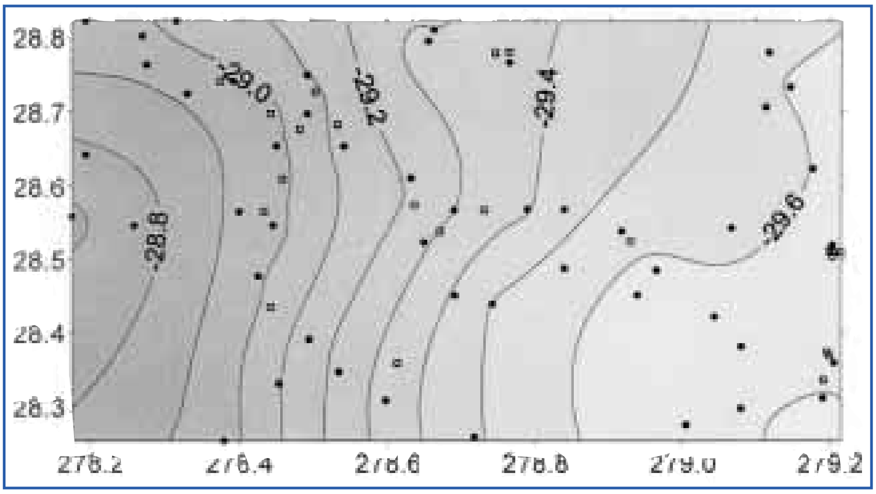

Fig. 3: Actual geoid shape in eastern Florida

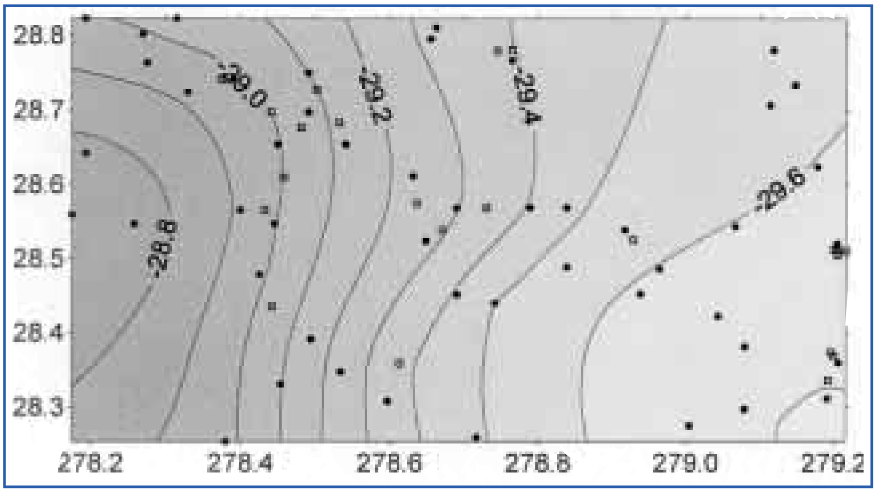

Fig. 4: ANN approximation of geoid in eastern Florida

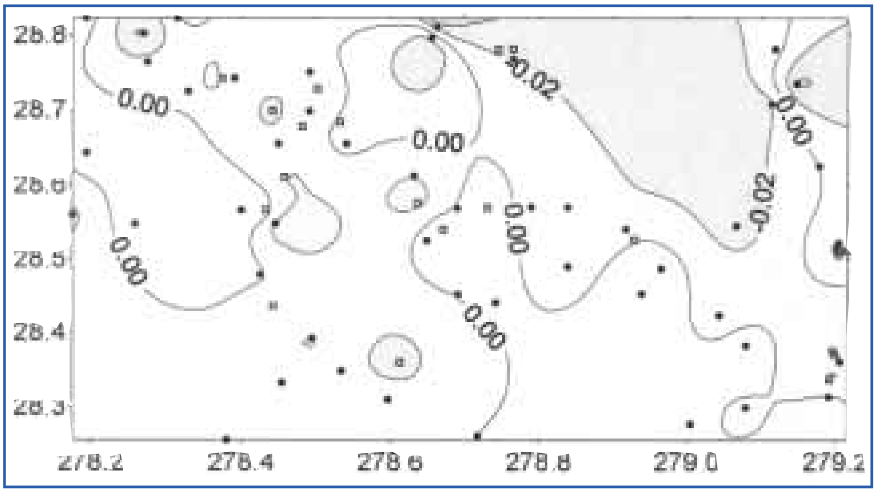

Fig. 5: Differences between geoid and ANN approximation of geoid in eastern Florida

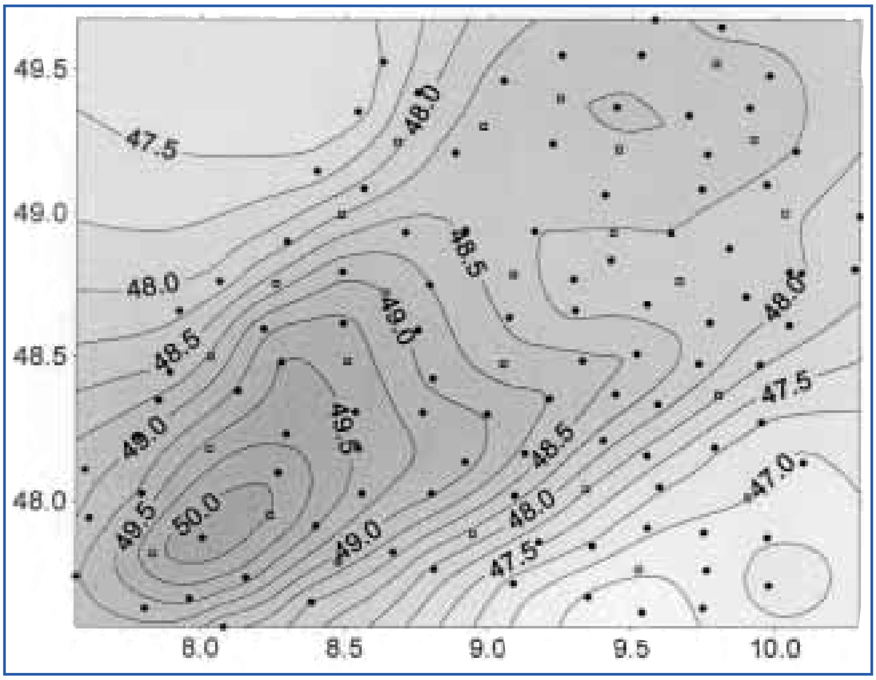

Fig. 6: Actual quasigeoid shape in Baden-Württemberg

found to be two hidden layers neural network with 50 and 40 hidden neurons, respectively.

The efficiency of the ANN approximation was compared to the results obtained by the second order model. The efficiency of the approximation was tested at all 71 points and at 23 new points. The differences between actual geoid heights (or height anomaly) and approximated values are summarized in Table 1. It can be seen that the ANN approximation results are considerably better than the second order model.

Another comparison between the actual geoid heights and the approximation results was done with the computed correlation coefficient, where actual geoidal heights were compared with those obtained by the approximation procedures. Although the coefficient correlation is quite high for both types of the approximation, the comparison of the mean square error of the approximation shows that the ANN approximation is several times better.

Baden-Württemberg test area

In the second case the quality of the ANN approximation was tested over a larger area. From the Prof. Wenzel's web page – GPS/Levelling derived height anomalies or geoid heights [15], 125 points were chosen in the state of Baden-Württemberg, covering the area of approximately $\Delta\varphi = 2°$ and $\Delta\lambda = 3°$. Quite irregular quasigeoid in this area varies from 46.6 to 50.2 m (*Fig. 6*).

Since the Baden-Württemberg test area is quite large and fitting the quasigeoid with the lower degree polynomial surface would not produce practically useful results, we only performed ANN surface fitting procedure for this area. From the set of 125 points with known height anomalies, 99 selected points were employed in the training procedure, the remaining 26 points were chosen for validation. In this case the optimal neural network configuration was found to be two hidden layers with 20 and 15 hidden neurons, respectively. The quasigeoid shape of the test area obtained by the ANN fitting procedure is shown in *Fig. 7*, and the differences between the actual quasigeoid shape and the quasigeoid ANN approximation are given in *Fig. 8*.

Some measures of the achieved ANN approximation quality at the area were again computed for the whole set of 125 points, and separately for only 26 points where quasigeoidal heights were obtained with the ANN approximation. Some quality measures are given in *Table 2*.

From Tables 1 and 2 it is easy to conclude that the quality measures obtained only for the new points with the geoid/quasigeoid heights show better results than in the case where all points are checked. This situation is dictated by the learning procedure, which stops when the relative difference between the ANN predicted and the actual value of geoid height is lower than 5 % for all input-output training set pairs, and by the distribution of the new points, which are mainly situated inside the region of investigation.

The correlation coefficient value computed for the actual quasigeoidal heights and those obtained with the ANN approximation is again very high for both cases, when all and only new points are considered. This comparison with all 125 points at the Baden-Württemberg area is sketched in *Fig. 9*.

Conclusion

The actual geoid shape may be approximated by different functions using the least squares method. An alternative method which employs ANN is presented here. The method is satisfactory if the accuracy requirements are not too severe.

Presently, no perfect solution for the accurate determination of geoid heights exists. The number of arguments may be offered to support or reject these approaches of orthometric (normal) heights determination in GPS-levelling procedure. This paper presents one such tool in an attempt to benefit from rapid technological progress and suggests a need for further investigations. One of possible approaches in further im-

Table 1: Comparison between ANN and the second order approximations (Florida)

Approximation type	Min. [m]	Max. [m]	Mean [m]	St. dev. [m]	Correlation coefficient
71 points ANN	−0.044	0.059	0.005	0.018	0.99805
2nd order	−0.161	0.104	0.011	0.055	0.98154
23 points ANN	−0.022	0.029	0.006	0.017	0.99815
2nd order	−0.084	0.088	−0.026	0.040	0.99076

Table 2: Accuracy of ANN approximation at the Baden-Württemberg

Approximation type	Min. [m]	Max. [m]	Mean [m]	St. dev. [m]	Correlation coefficient
125 points ANN	−0.189	0.185	0.002	0.076	0.99517
26 points ANN	−0.126	0.113	0.001	0.067	0.99633

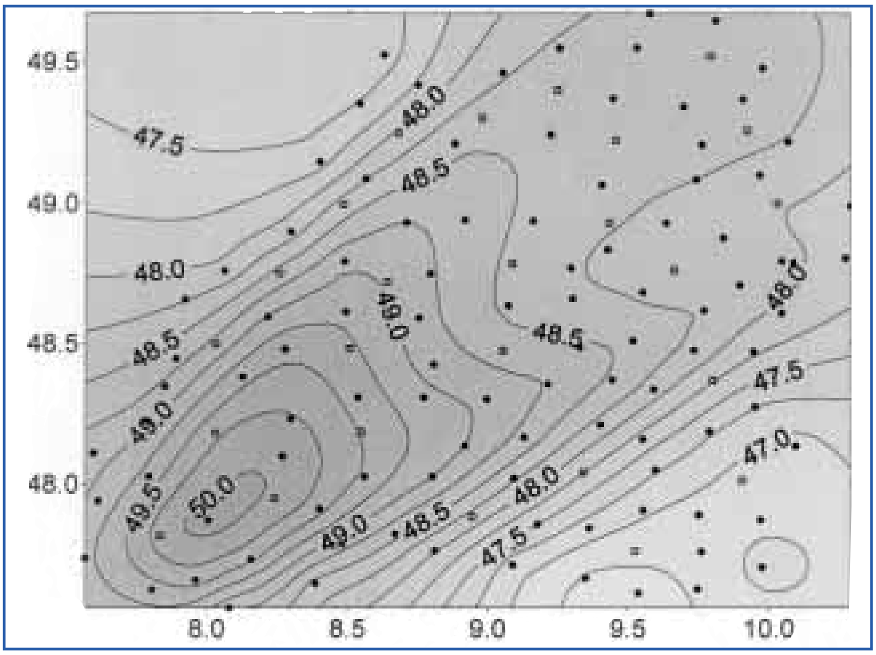

Fig. 7: ANN approximation of quasigeoid in Baden-Württemberg

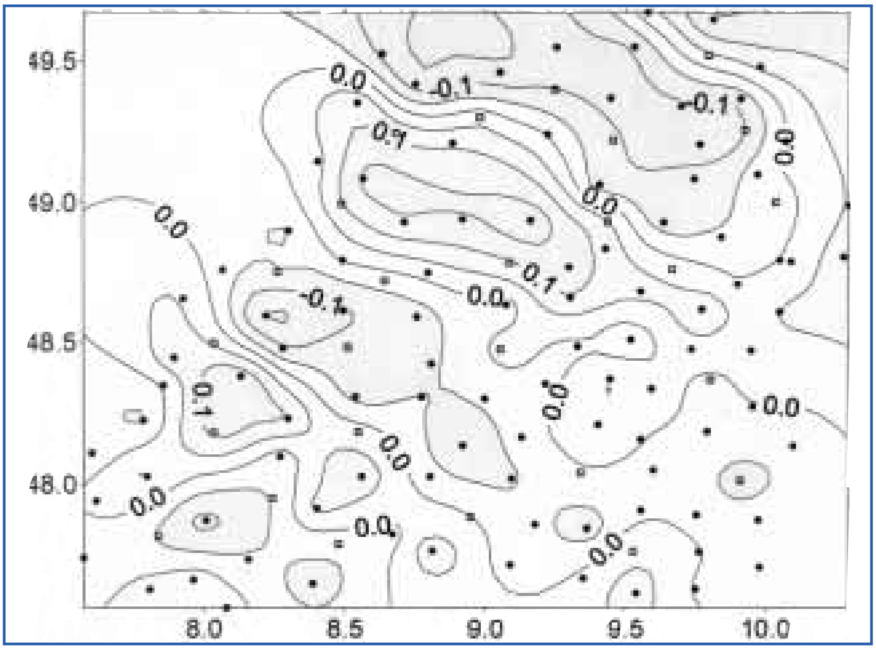

Fig. 8: Differences between quasigeoid and ANN approximation of quasigeoid in Baden-Württemberg

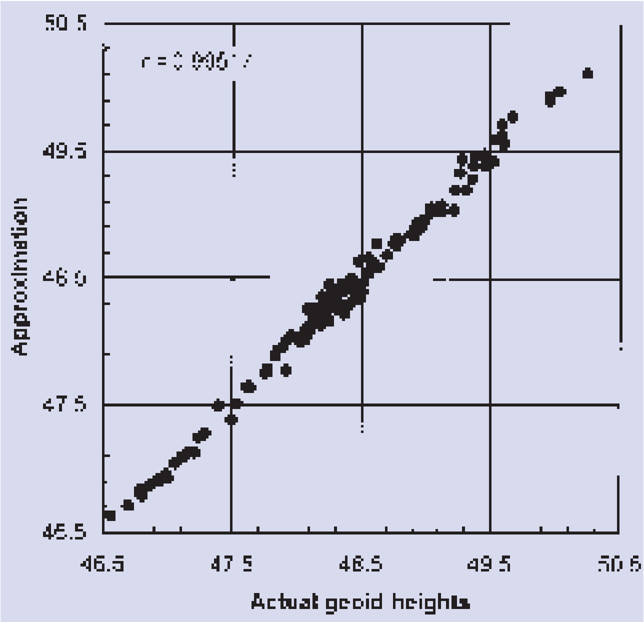

Fig. 9: Comparison of actual quasigeoid heights and ANN quasigeoid approximation in Baden-Württemberg

provement of geoid heights approximation is the combination of the least squares collocation and the ANN approximation in which the ANN approximation serves as the trend function.

Literature

[1] BABA, N.: A New Approach for Finding the Global Minimum of Error Function of Neural Networks, Neural Networks, Vol. 2, pp. 267–273, 1989.

[2] BURSHTEIN, D.: Neural Network Training using the Simulated Annealing Method, World Congress on Neural Networks, WCNN '93, Portland, OR, pp. III-397–III-400, 1993.

[3] FUNAHASHI, K.: On the Approximate Realization of Continuous Mappings by Neural Networks, Neural Networks, Vol. 2, pp. 183–192, 1989.

[4] GOLDBERG, D. E.: Genetic algorithms in search, optimization and machine learning, Addison-Wesley, Reading, MA, 1989.

[5] HOLLOWAY, R. D.: The Integration of GPS Heights into the Australian Height Datum, Unisurv reports S-33, School of Surveying Engineering, Kensington, 1988.

[6] HORNIK, K., STINCHOMBE, M., WHITE, H.: Multi-layer Feed-forward Networks are Universal Approximators, Neural Networks, Vol. 2, pp. 359–366, 1989.

[7] ILLNER, M., JÄGER, R.: Integration von GPS-Höhen ins Landesnetz – Konzept und Realisierung in Programmsystem HEIDI, AVN, 1/1995, 1995.

[8] JANAKIRAMAN, J., HONAVAR, V.: Adaptive Learning Rate for Increasing Learning Speed in Backpropagation Networks, World Congress on Neural Networks, WCNN '93, Portland, OR, pp. IV-378–IV-381. 1993.

[9] LIPPMANN, R. P.: An Introduction to Computing with Neural Nets, IEEE ASSP Magazine, Vol. 4, No. 2, pp. 4–22, 1987.

[10] MÜLLER, B., REINHARDT, J., STRICKLAND, M. T.: Neural Networks, An Introduction, Springer-Verlag, Berlin, 1995.

[11] RUMELHART, D. E., MCCLELLAND J. L.: Parallel Distributed Processing, Vol. 1: Foundations, The MIT Press, Cambridge, MA, 1986.

[12] SARLE, W. S., ed.: Neural Network FAQ, Periodic posting to the Usenet newsgroup comp.ai.neural-nets, URL: ftp://ftp.sas.com/pub/neural/FAQ.html, 1998.

[13] SMITH, D. A., ROMAN, D. D.: Documentation for the GPS data set of August 27, 1999, URL: http://www.ngs.noaa.gov/GEOID/GPSonBM99/gpsbm99.html, 1999.

[14] TREADGOLD, N. K., GEDEON, T. D.: Simulated Annealing and Weight Decay in Adaptive Learning: The SARPROP Algorithm, IEEE Transactions on Neural Networks, Vol. 9, No. 4, pp. 662–668, 1998.

[15] WENZEL, H. G.: GPS/Levelling derived height anomalies or geoid heights, URL: http://www.gik.uni-karlsruhe.de/~wenzel/gpslevha/bwref97b.dat, 1999.

[16] YIP, P. P. C., PAO, Y.-H.: A fast universal training algorithm for neural networks, World Congress on Neural Networks, WCNN '93, Portland, OR, pp. III-614-III–621, 1993.

Address of the authors:
MIRAN KUHAR, PH. D.,
BOJAN STOPAR, PH. D.,
GORAN TURK, PH. D.,
TOMAŽ AMBROŽIČ, M.Sc.,
University of Ljubljana, Faculty of Civil and Geodetic Engineering,
Jamova 2, SI-1115 Ljubljana,
Slovenia
E-Mail: mkuhar@fgg.uni-lj.si, bstopar@fgg.uni-lj.si, gturk@fgg.uni-lj.si, tambrozi@fgg.uni-lj.si

Abstract

Local geoid surface can be modelled by the approximation function by the least squares method. An alternative method of approximation is the use of artificial neural networks (ANN). The ANN approximation has been tested in various test areas. It has been concluded that ANN may efficiently approximate real geoid heights on the chosen area.

Zusammenfassung

Die lokale Geoidfläche kann durch Approximationsfunktionen mit der Methode der kleinsten Quadrate modelliert werden. Alternativ werden Künstliche Neuronale Netze (KNN) angewandt. Die KNN-Approximation wurde auf ausgewählten Gebieten getestet. Das Testergebnis sind die auf den ausgewählten Gebieten erfolgreich approximierten Geoidhöhen.