COORDINATE TRANSFORMATION WITH NEURAL NETWORKS AND WITH POLYNOMIALS IN HUNGARY

Ms. Piroska Zaletnyik
Department of Geodesy and Surveying, Budapest University of Technology and Economics
H-1111 Budapest, Muegyetem rkp 3. Kmf. 16., Hungary
tel.:+36-1-4634343, fax: +36-1-4633192, e-mail: zaletnyikp@hotmail.com

1. Introduction

Nowadays in Hungary more than one coordinate systems are in use. Frequently it is necessary to transform from one coordinate system to another. The most widely used methods are Helmert and affine transformations. With these traditional methods it is impossible to make a uniform transformation for the whole area of the country, because the attainable accuracy is not sufficient. With polynomial transformation better results can be obtained.

The artificial neural network provides a new technology for coordinate transformation. The popularity of this methodology is rapidly growing.

In this research coordinate transformations with neural networks were examined between WGS-84 (coordinate system of GPS) and EOV (the Hungarian national reference system), the two most frequently used coordinate systems in Hungary. For calculating the transformation formula one needs to have points with known coordinates in both systems. Two different databases of points were examined. At the beginning of the research a database of 43 points was available. The research was begun with these data, but the number of points was not sufficient for the coordinate transformation with neural networks. Later it a bigger database of 1153 points became available and the examinations were repeated. In both cases (with the two different databases) the transformation with neural network was compared with polynomial transformation.

2. EOV-WGS84 coordinate transformation

In geodesy coordinate transformation is a very important task and up to the present several methods were worked out.

Exact projection conversion (with closed mathematical formulas) between two projection systems is possible only if the reference surfaces of the two projections are the same and points of the same triangulation network are represented in the two systems. Otherwise the conversion or coordinate transformation formula can be calculated with points the coordinates of which are known in both reference systems. These conversions have several widely used methods like the one of Helmert, affine or polynomial transformations.

Nowadays in Hungary the two most frequently used coordinate systems are the WGS-84 (coordinate system of the GPS) and the EOV (Egységes Országos Vetület = Uniform National Projection). The reference surface of the GPS is the the WGS84 ellipsoid and the one of the EOV is the new Hungarian Gaussian sphere fitting the IUGG/1967 ellipsoid.

In the WGS84 ellipsoidic reference system the point positioning data are the geodetic latitude (φ), the geodetic longitude (λ), and ellipsoidic height (h). In this research only the horizontal positioning data were used (φ, λ).

The EOV was introduced in Hungary in 1975. The EOV is an oblique-axis, reduced cylindrical projection of the new Hungarian Gaussian sphere. The point positioning data in this system are the y, x rectangular plane coordinates.

Different types of data were usedin the two systems: in the EOV plane coordinates (y, x) and in the WGS84 ellipsoidical coordinates (φ, λ). The task was to determine a coordinate transformation formula between EOV and WGS84 /F and F^{-1} in (1), (2) equations/. The coordinates of the points were given in both systems.

$$\begin{pmatrix}\varphi\\ \lambda\end{pmatrix} = F(y,x) = \begin{pmatrix}F1(y,x)\\ F2(y,x)\end{pmatrix} \tag{1}$$

$$\begin{pmatrix}y\\ x\end{pmatrix} = F^{-1}(\varphi,\lambda) = \begin{pmatrix}F1I(\varphi,\lambda)\\ F1I(\varphi,\lambda)\end{pmatrix} \tag{2}$$

There are several methods to determine F and F^{-1}. In this work polynomial transformation and neural networks were used to solve the problem.

3. Polynomial transformation

With polynomial transformation better results can be obtained in a big area (like Hungary) than with the traditional methods (affine, Helmert). Affine and Helmert transformation can be used in local transformations in smaller areas. If we try to use them to solve the conversion problem in the whole area of Hungary we receive too big errors in the known points. Therefore polynomial transformation was used to compare the results of neural networks with a traditional method.

According to the Hungarian Projection Regulation the maximum order of these polynomial series is 5^{th} order.

The following two variables 5^{th} order polynomial was used:

$$\begin{aligned} x' = {} & A_0 + A_1 \cdot x + A_2 \cdot y + A_3 \cdot x^2 + A_4 \cdot x \cdot y + A_5 \cdot y^2 + A_6 \cdot x^3 + A_7 \cdot x^2 \cdot y + A_8 \cdot x \cdot y^2 + A_9 \cdot y^3 \\ & + A_{10} \cdot x^4 + A_{11} \cdot x^3 \cdot y + A_{12} \cdot x^2 \cdot y^2 + A_{13} \cdot x \cdot y^3 + A_{14} \cdot y^4 + A_{15} \cdot x^5 + A_{16} \cdot x^4 \cdot y + A_{17} \cdot x^3 \cdot y^2 \\ & + A_{18} \cdot x^2 \cdot y^3 + A_{19} \cdot x \cdot y^4 + A_{20} \cdot y^5 \end{aligned} \tag{3}$$

$$\begin{aligned} y' = {} & B_0 + B_1 \cdot x + B_2 \cdot y + B_3 \cdot x^2 + B_4 \cdot x \cdot y + B_5 \cdot y^2 + B_6 \cdot x^3 + B_7 \cdot x^2 \cdot y + B_8 \cdot x \cdot y^2 + B_9 \cdot y^3 \\ & + B_{10} \cdot x^4 + B_{11} \cdot x^3 \cdot y + B_{12} \cdot x^2 \cdot y^2 + B_{13} \cdot x \cdot y^3 + B_{14} \cdot y^4 + B_{15} \cdot x^5 + B_{16} \cdot x^4 \cdot y + B_{17} \cdot x^3 \cdot y^2 \\ & + B_{18} \cdot x^2 \cdot y^3 + B_{19} \cdot x \cdot y^4 + B_{20} \cdot y^5 \end{aligned} \tag{4}$$

where x and y are the coordinates in the first system and x' and y' are the coordinates in the second system.

In the Department of Geodesy and Surveying of the Budapest University of Technology and Economics a software (called VETULET) was worked out to solve coordinate transformation problems. In this software 5^{th} order polynomial transformation is used for EOV-WGS84 conversion using also an auxiliary plane projection system. The calculation is as follows: first the ellipsoidical coordinates of the WGS84 are transformed to the new Hungarian Gaussian sphere, then transformed to the auxiliary plane coordinate system and then transformed to the EOV with 5^{th} order polynomial. The conversion from EOV to WGS84 is happening in reverse order but can be calculated only with iteration steps. In a former research it was proved that there is no need for the auxiliary plane coordinate system. Using only 5^{th} order polynomials between the EOV and the WGS84, the same results can be obtained with a simpler calculation.

The necessary number of points to calculate a two variables polynomial transformation formula is the next:

$$p = \frac{(r+2)\cdot(r+1)}{2} \tag{5}$$

where r is the number of order of the polynomial and p is the number of the necessary points.

In a 5th order polynomial case 21 points are needed. In the case of more points there is a possibility for adjustment calculation, for polynomial fitting. These calculations were carried out with Mathematica software.

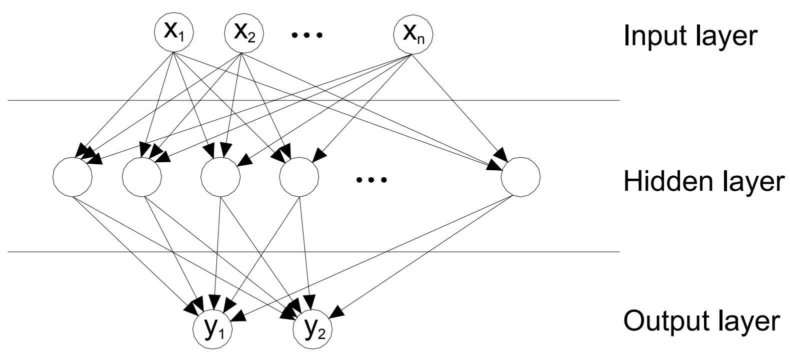
4. Transformation with neural networks

The artificial neural network is a very effective tool of artificial intelligence. The expansion of the biological knowledge, particularly the recognition of the functioning of the human neurons lead to the creation of artificial neural networks.

A very important characteristic of these networks is the approximation feature. These networks can solve problems which are too difficult to solve by applying algorithms. Using closely related input and output data (teaching set) unknown continuous functions can be approached with least square method. This approximation feature was used for coordinate transformation also. The conversion formula was carried out by Mathematica software and it's Neural Network module. The advantage of this tool is the ability of symbolic calculations. Other neural network softwares solve problems like a black box, where we do not get the function determined during the learning procedure. We can give the input data and receive the output data. Using the Neural Network module of Mathematica we can give variables as input data and we receive the unknown function as output data. This is very important if we want to use the function in another programme language.

4.1. Structure of neural networks

The neural networks can be imagined like a system containing several layers with nodes called neurons in the layers. In a network there are one input layer, one or more hidden layers and one output layer.



1. Figure Structure of a general neural network

Every neuron in the hidden layer has one or more input data and one or more output data. These neurons are the basic elements of the networks. A general neuron first calculates the weighted sum of the inputs and then carries out a transfer or activation function on this sum, this is the output of the neuron. The most important difference among the networks is the type of the activation function. The most frequently used functions are RBF (Radial Basis Function) and sigmoid functions.

The parameters of the activation functions and the weights are determined during the learning procedure. This is the first part of the functioning. This section is a slow procedure with a lot of iteration steps using the least square method to determine the optimum weights and parameters. The most popular learning algorithm is the back-propagation algorithm. For this section one part of the known points is used, the teaching set. The remainder points are used in the second section of functioning to test the results of the neural network. This part is very fast This is the reason why neural networks can be applied very successfully.

The number of parameters (p) to determine depends on the number of used layers and the number of used neurons (n). In case of the RBF network (with one hidden layer) it is as follows:

$$p=4n+3 \tag{6}$$

In case of back-propagation network with one hidden layer and with sigmoid activation functions it is as follows:

$$p=4n+1 \tag{7}$$

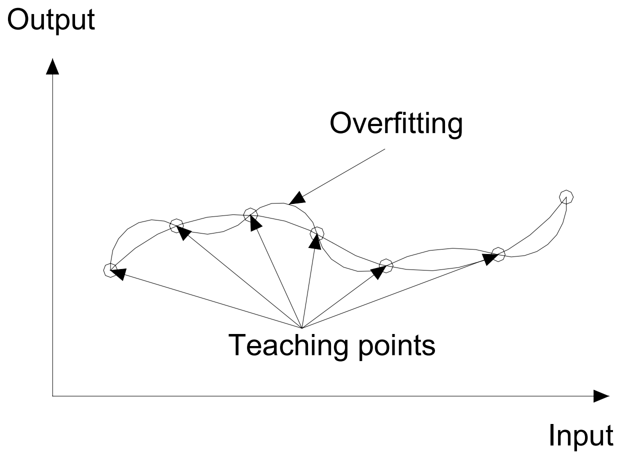
For coordinate transformation these types of networks were used.

To design the structure of the networks one has to choose the number of the layers, number of the neurons and type of activation functions, and make an effort to approach the biggest accuracy with the simplest structure of network.

4.2. The overfitting problem

The neural network can be used only if the determined function works not only in the teaching points but between these points, too. Therefore in addition to the teaching set a testing set is also needed for the qualification of the results. The testing set or test points are points which were not used during the learning algorithm but the related input and output data of these points are known. If we don't have a testing set a so called overfitting problem can occur.

Overfitting means that the error of the teaching set is reducing while the error of the testing set is growing, the network excessively fits to the teaching points.



2. Figure Overfitting problem

5. Transformation using 43 points

In the first part of our investigation 43 points were used to determine coordinate transformation formulas with polynomials and with neural networks.

43 points for the territory of Hungary are not enough for choosing teaching and testing points for neural networks. It would be difficult to decide which points to leave out from the teaching (calculating) procedure. But without a testing set we can't control if the conversion formula works outside the teaching points also. In some way we have to control the results to avoid the overfitting problem.

The coordinate transformation has two directions, from the EOV to the WGS84 and from the WGS84 to the EOV. We have to calculate the formulas for both directions. If both formulas are good the "there and back" transformation have to work also (transforming the data from the EOV to the WGS84 and then, in the reverse direction, transforming from the WGS84 to the EOV with the calculated formulas). We can use this method for controlling the results of the formulas not only in the teaching points but between them also. We can choose arbitrary points from the whole area and then do the “there and back” transformation. If there is a big difference between the original and the calculated data the formulas are not good between the teaching points.

The results of neural networks were compared with 5^{th} order polynomial transformation described in chapter 3. To choose the most suitable neural network for this problem various types were examined. The RBF and the back-propagation networks with sigmoid activation functions were tried with 10 neurons. This is the maximum applicable number of neurons according to (6) and (7) equations. Using more neurons we have to determine more parameters than the number of points we have and this leads to the overfitting problem. Using ten neurons the number of parameters to determine in the case of the RBF network is 43 and in the case of the sigmoid activation function is 41.

The back-propagation network with sigmoid activation function was better for this problem. Therefore I will describe only the results of this network and compare it with the polynomial method.

To choose which transformation method is better, two things have to be examined. It can be examined how the calculated formula for the teaching points works and how the “there and back” transformation works for the arbitrary chosen points (for this examination about 1000 points were chosen arbitrarily for the whole area of Hungary). Let’s see the results!

Polynomial transformation in 43 points			
	Standard deviation	*Maximum positive error*	*Maximum negative error*
y	0.035 m	0,058 m	-0.088 m
x	0.036 m	0,074 m	-0.082 m
ϕ	0.0011”	0,0026”	-0.0024”
λ	0,0018”	0,0044”	-0,0029”

Transformation with neural network in the 43 teaching points			
	Standard deviation	*Maximum positive error*	*Maximum negative error*
y	0.003 m	0,010 m	-0.010 m
x	0.003 m	0,007 m	-0.011 m
ϕ	0.0002”	0,0006”	-0.0005”
λ	0,0003”	0,0009”	-0,0008”

Comparing these results it is evident that with neural networks better results can be obtained. The results of the neural network are 5-10 times are better than the others. But is it the same situation when we examine the “there and back” transformation between the teaching points?

Polynomial "there and back" transformation in 1000 points			
	Standard deviation	*Maximum positive error*	*Maximum negative error*
y	0,003 m	0,012 m	-0.011 m
x	0,002 m	0,012 m	-0,007 m

"There and back" transformation with neural networks in 1000 points			
	Standard deviation	*Maximum positive error*	*Maximum negative error*
y	0,098 m	0,286 m	-0,382 m
x	0,456 m	5,208 m	-2,212 m

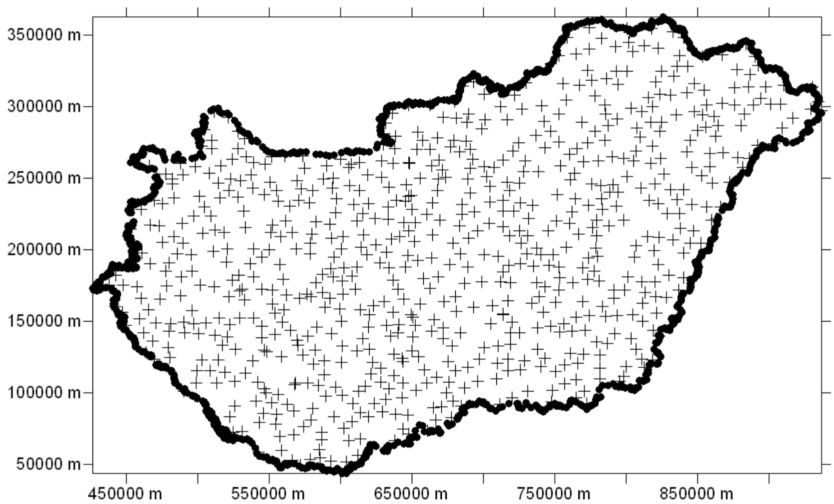
Examining these results we see that the polynomial transformation is better. It seems that this method works not only for the teaching points but between the points also. The results of the neural networks are unreliable, there are very big (a few meter big) errors also. This could be the result of the overfitting problem due to the small number of points. But what happen if we have more data available? Can neural networks work better?

6. Transformation using 1153 points

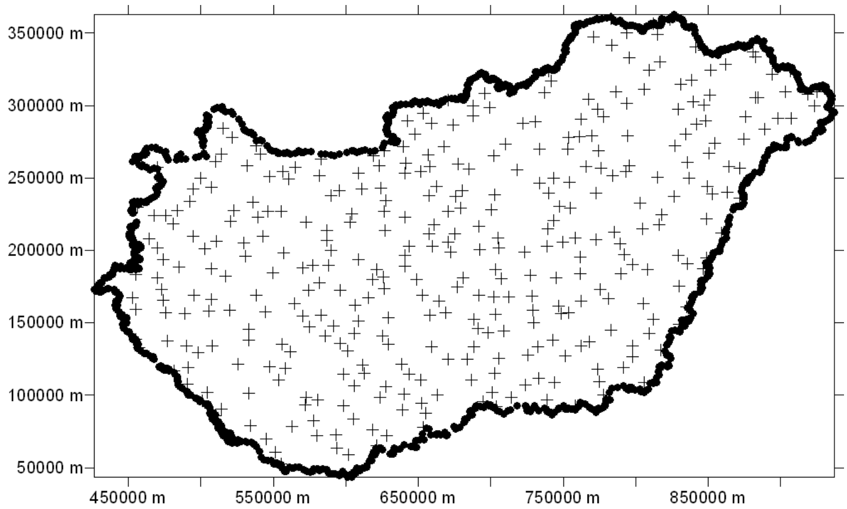
In the first part of the research we had only 43 points, but later we received a bigger database with 1153 points. 43 points were not enough to choose teaching and testing points therefore we used the "there and back" transformation to determine if the calculated formulas are working between the known points also.

With 1153 points this problem does not exist. There is no need to use the "there and back" transformation. Two-thirds of the data was chosen for teaching points and one-third of the data for testing the calculated formulas.

The location of the teaching and testing points is as follows:



3. Figure Teaching points



4. Figure Testing points

The polynomial transformation was the same as in the previous chapter but using 1153 points. We tried to use higher order polynomials than 5^{th} also but it did not work because of the deterioration of conditions of the equations. The results of this transformation are in the next table. The horizontal error was calculated in every point with the $\sqrt{(y^2 + x^2)}$ formula.

Polynomial transformation in 1153 points			
	Standard deviation	*Maximum positive error*	*Maximum negative error*
y	0.044 m	0,205 m	-0.199 m
x	0.043 m	0,152 m	-0.219 m
horizontal	0.033 m	0,255 m	

For the transformation with the neural network with 1153 points both the RBF and the sigmoid activation functions were tried. Also in this case the back-propagation network with sigmoid activation function was better. We used 30 neurons in the hidden layer. With this structure very good results were obtained in the teaching and in the testing points also.

Teaching set for transformation with a neural network (768 points)			
	Standard deviation	*Maximum positive error*	*Maximum negative error*
y	0.028 m	0,130 m	-0.123 m
x	0.029 m	0,088 m	-0.109 m

Testing set for transformation with a neural network (385 points)			
	Standard deviation	*Maximum positive error*	*Maximum negative error*
y	0.036 m	0,113 m	-0.107 m
x	0.033 m	0,126 m	-0.131 m

Between the results of the teaching and testing set there is no big difference. It can be declared that in this case neural network can be used not only for the teaching points but also for the whole area.

For better comparison with the polynomial fitting the neural network transformation with the calculated formulas were executed to the all (1153) data.

Transformation with neural network in 1153 points			
	Standard deviation	*Maximum positive error*	*Maximum negative error*
y	0.031 m	0,130 m	-0.123 m
x	0.031 m	0,126 m	-0.131 m
horizontal	0.023 m	0,165 m	

The maximum horizontal error with neural network (16.5 cm) is 9 cm less than with the polynomial transformation (25.5 cm). The standard deviation is also better with neural network (2.3 cm, and the other is 3.3 cm).

These results proved our assumption that the neural networks can be better than using polynomials above a certain number of data points.

7. Summary

In this research the application of a new informatics tool, the neural network was examined for the coordinate transformation problem between the WGS-84 and the EOV, the Hungarian National Projection System. The results were compared with a traditional method, the polynomial transformation.

The use of the different sized databases of points highlighted one dangerous source of errors for neural networks, the overfitting problem. The neural network is a very effective tool for many problems like approximating functions, but have to be used very carefully. First we have to determine the requested accuracy and examine the available quantity of data. If there is another applicable traditional method with acceptable accuracy then we should use that simpler solution. If the quantity of data is bigger than better results can be obtained with neural networks.

The greatest advantage of neural networks is that it can be used very successfully with a huge quantity of data when polynomials cannot be applied because of the deterioration of conditions of the equations. For this reason I think that in the future this tool will be applied in many areas of geodesy and geoinformatics.

8. Acknowledgements

This investigation was supported by the Hungarian National Research Fund (OTKA), contract No. T043007 (Investigation of the geodetic datums and planary coordinate systems in Hungary).

9. References

Barsi Á. 1999: Koordináta-transzformáció megoldása neurális hálózatokkal, *Budapest, Geodézia és Kartográfia LI, No. 10. pp. 12-18.*

Horváth G. 1995: Neurális hálózatok és műszaki alkalmazásaik, *Budapest, Műegyetemi Kiadó*

Paláncz B. – Völgyesi L. 2003: High accuracy data representation via sequence of neural networks, *Acta Geod. Geoph. Hung., Vol. 38 (3) pp. 337-343*

Scarselli, Franco-Ah Chung Tsoi 1998: Universal Approximation Using Feedforward Neural Networks: A Survey of Some Existing Methods, and some New Results, *Neural Networks, Vol. 11, No. 1, pp. 15-37, Elsevier Science Ltd., Great Britain*

Vetületi szabályzat az egységes országos vetületi rendszer alkalmazására 1975 *(Hungarian Projection Regulation), Budapest, MÉM Országos Földügyi és Térképészeti Hivatal*

Zaletnyik P. 2003 (1): Neurális hálózatok alkalmazása a geodéziában, *MSc. Thesis, Budapest University of Technology and Economics*

Zaletnyik P. 2003 (2): WGS-84 –EOV koordináta transzformáció neurális hálózattal, *Scientific Student Conference at the Budapest University of Technology and Economics, Department of Geodesy and Surveying (www.agt.bme.hu/staff_e/zaletnyik_e.html)*