Indian Journal of Geo Marine Sciences
Vol. 46 (09), September 2017, pp. 1743-1750

Forecasting geodetic measurements using finite impulse response artificial neural networks

Miloš Miljanović* [1], Toša Ninkov[2] ,Zoran Sušić[2] & Sanja Tucikesic[3]

[1]Vienna University of Technology, Vienna, Austria; [2]Faculty of Technical Sciences, Novi Sad, Serbia
[3]Faculty of Civil Engineering, Architecture and Geodesy, University Banja Luka, Bosnia and Hercegovina

*[E.Mail: m.miljanovic@gmail.com]

Received 15 October 2014 ; revised 22 November 2016

In this paper, a method for evaluating and forecasting deformation movements present in buildings during tunneling works is described. The data used for processing is gathered from the project 'Prokop' that has involved tunneling works under residential buildings, all mapped using a geodetic control network, or elevation network. Measurement results from this project are being used by the Finite Impulse Response (FIR) Neural Network as time series to predict future movements/deformations.

[Keywords: geodetic control network, artificial neural network, finite impulse response]

Introduction

Geodetic works are permanently present in all phases of space changes which include idea, realization and evaluation. These works are applied for measuring the area for groundwork, transportation of objects to the field, tracking of their construction, measurement of objects for the purpose of managing records, tracking of the behaviour of the object during exploitation, etc.

Designers in the field are securing a permanent stability of the object, however during the construction of complex buildings, some unplanned changes can occur, which can lead to catastrophic damage to the object and its surrounding. For the purpose of preventing negative consequences, it is necessary in certain time intervals to monitor civil engineering works using one of geodetic methods, with a network which can meet the demands of designer documentation[1].

Tracking of movements and deformation on every complex object, including object 'Prokop' because of its specificity, is a procedure which demands the realization of series of very complex processes and rules, as well as unavoidable cooperation from various areas of science. The quality of procedures depends not only on the quality of geodetic measurements, data processing and equalization of data, but also from the universal approach in solving engineering demands. Universality entails the knowledge of outer and inner influences on the deformation processes, compatibility of methodologies and instruments for registering shifts, geo-mechanical, construction, and other characteristics of the observed object, procedure of projecting and establishing a special purpose geodetic network, expected accuracy upon realizing control measurements, periodic control measurements, the type of model and methods used in deformation monitoring, methods for analysis and interpretation of measurement results and various other data[2].

Based on the knowledge of the procedure, characteristics of the object and all previous geodetic works (from projecting and establishing a geodetic network to the realization of control measurements that took several years), it is necessary to in a universal way show the procedure of projecting the control network, and measurement plan, acquisition and processing of data with conventional and modern instruments.

Analysis of the results of unknown parameters from these measurements, basic and other control measurements will contribute to the findings as well recommendation for all future auscultations on similar objects, and to the improvement in the approach to deformation monitoring in all civil engineering objects[3].

Recommendations, based on the analysis and investigations will be used in establishing geodetic network of specific uses, methodology of acquiring data and the choice of deformation model. The results of the previous assessment value, zero measurement on object 'Prokop' and realized control measurements will be used for the purpose of predicting movements using artificial neural networks, in particular the Finite Impulse Response (FIR) Neural Networks.

Materials and Methods

During the process of projecting geodetic control networks, for the purpose of retrieving numerical values and assessing previous accuracy of the network, it is necessary to determine the design of the network and to plan measurements in it. Temporary values of unknown coordinates or the height of points are determined mostly from existing maps or from known methods[4]. When planning the measurements, different choices are taken into consideration together with their measurement accuracy methods.

When the temporary values of coordinates and/or the height of the points are determined together with the defined plan of measured lengths, as well as their accurate measurement in the network, the matrix of design $\mathbf{A}$ and a covariance matrix of measured lengths $\mathbf{K_l} = \sigma_o \mathbf{Q_l}$ is given[5]. This way, a functional and stochastic model of indirect equalization of free or closed networks is provided.

Covariance matrix of unknown parameters for closed networks is represented as:

$$\mathbf{K}_{\hat{\mathbf{x}}} = \sigma_o^2 \cdot \mathbf{Q}_{\hat{\mathbf{x}}} = \sigma_o^2 \cdot \mathbf{N}^{-1} = \sigma_o^2 \cdot (\mathbf{A}^{\mathbf{T}}\mathbf{Q}_{\mathbf{l}}^{-1}\mathbf{A})^{-1}$$

Where $\bar{\mathbf{x}}$ is the vector of differential increments, $\boldsymbol{\sigma_o}$ is the a priori standard settlement or a priori accuracy given during the process of levelling, $\mathbf{N}$ is the matrix of normalized equations, Q1 is the cofactor matrix of measurements.

And for free networks:

$$\mathbf{K}_{\hat{\mathbf{x}}} = \sigma_o^2 \cdot \mathbf{Q}_{\hat{\mathbf{x}}} = \sigma_o^2 \cdot \mathbf{N}^{+} = \sigma_o^2 \cdot (\mathbf{A}^{\mathbf{T}}\mathbf{Q}_{\mathbf{l}}^{-1}\mathbf{A})^{+}$$

where N+ denotes a pseudo inversion.

In the case of object 'Prokop', the previous value of accuracy was processed with minimal traces of covariance matrix on the stability points of geodetic control network, which have been placed outside the zone of expected deformations. Standard of horizontal and vertical directions, lengths and height differentials in the geodetic networks, as well as measurement plan have secured the targeted projected value of unknown parameters, shown in Table 1. Average values of standard coordinates and heights as well as elements of standard confidence ellipse (with probability p=0.95) is shown in table 1.

Table 1. Previous accuracy assessment of geodetic control network points and points for observation on object 'Prokop'

Accuracy assessment of the geodetic control network	[mm]	Accuracy assessment of observation points	[mm]
Coordinate standards for control network points	$\sigma_y = 0.69$	Coordinate standards for B = 1.97 observation points	$\sigma_y = 1.23$
	$\sigma_x = 0.67$		$\sigma_x = 1.30$
	$\sigma_H = 0.70$		$\sigma_H = 0.95$
Absolute confidence ellipses for control network points	A = 1.77	Absolute confidence ellipses for observation points	A = 3.97
	B = 1.55		

Figure 1 depicts previous analysis of free geodetic control network on object 'Prokop' with planned measurement directions $\sigma_\alpha = 1''$ and lengths $\sigma_D = 1\ \text{mm} + 1\ \text{ppm}$ (making the total accuracy 1mm+1mm per square kilometre), where after the analysis a homogeneous accuracy of points is preserved.

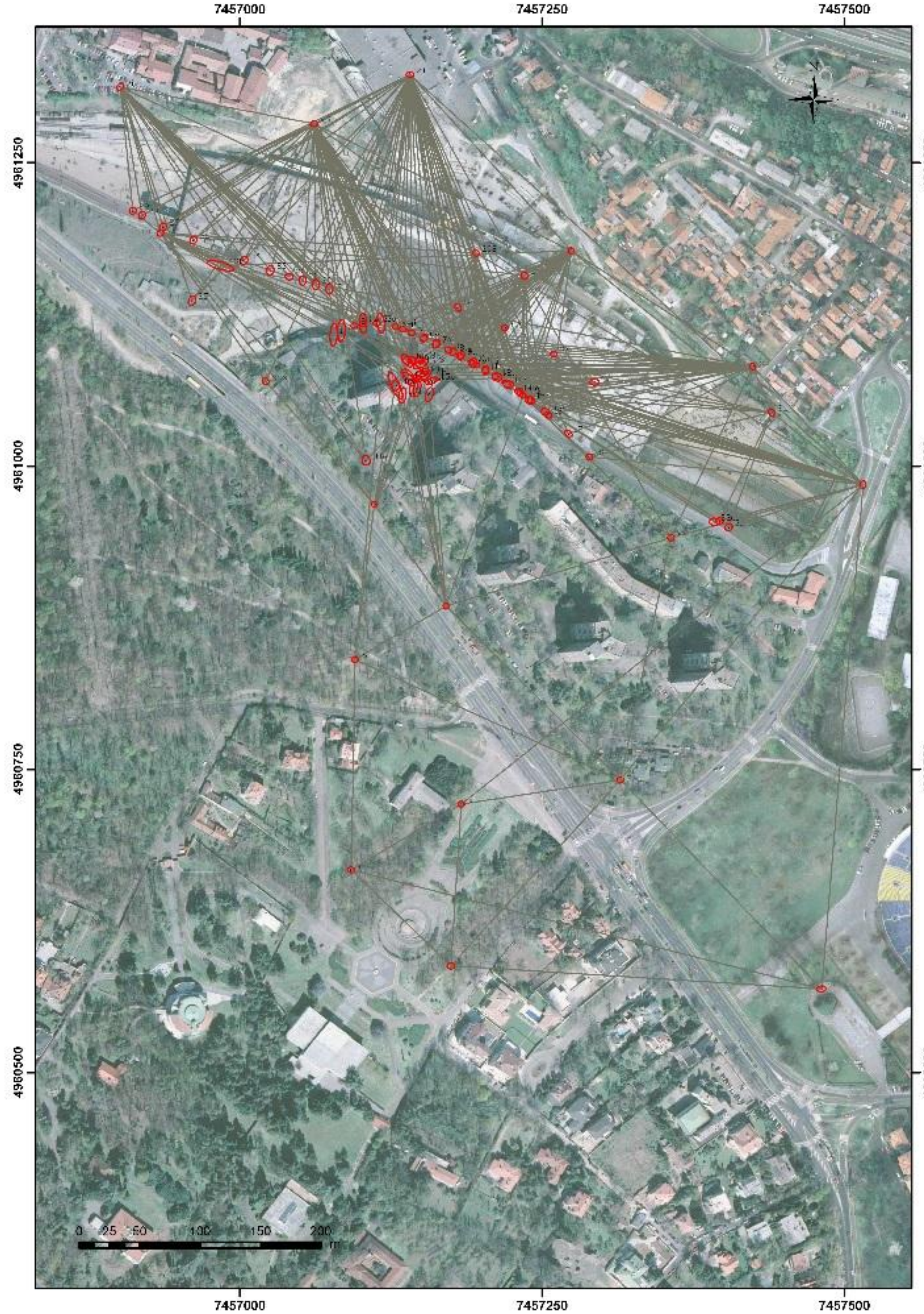

Figure 1. Sketch of the points in the micro network and observation points, with absolute confidence ellipses (p=0.95), object Prokop, Belgrade (coordinates are shown in state plane coordination system, Gauss-Krueger projection, seventh zone)

After field investigation control measurements, in vertical and horizontal plane, calculation of definite directions and lengths from all observed gyres has been performed. When processing conventional length measurements, appropriate adjustments based on atmospheric influence, and altitude have been applied (project was done at the altitude of 97m).For GPS measurements, simultaneous registered data in receivers at the final points of the measured vector were grouped in three files. The data was transferred and processed using software package PROCESS for each individual vector. For measured vectors the same adjustments like in length measurements need to be applied.

As a result of GPS measurements, based on the least squares method, length of the vector was calculated, including average squared error, spatial components of the vector together with the least squared errors[6].

When the errors were not in the allowed threshold, the vectors were re-processed interactively by the system. In this variant the software enables the user to omit the results from certain satellites and time intervals that are not aligned with the others, as well as the possibility of changing the referred satellite. The built-in software comes with the modules necessary modules to deal with the result variance.

When the results have been processed, the same ones are being forwarded to the software to perform the leveling of the elevation network, which is done by the method of group leveling with the condition that the least squared errors are minimal.

The necessary steps needed to perform the leveling are: error equations, inversion and control of inverted matrices for accuracy measurements, calculation of definite coordinates, mean error unit, and the mearn error for the (x,y) coordinates.

$m_o = \pm 0.72$mm

$mx_{max} = \pm 1.20$mm

$my_{max} = \pm 1.00$mm

With the processed coordinates, the stability of the elevation network has been determined using the Helmert transformation. For stable points, elevation network has been leveled together with the points for observation on abutment diaphragms and residential buildings[7]. Leveling has been performed based on the groups the points belong to. The exact information that is being fed to the software is:

- id and name of stability points
- id and name of points that determine movements
- measured values, directions, and lengths
- spatial coordinates of the points in elevation network together with the approximate coordinates of the observed points

Elevation network with the observed points has been leveled in the local coordination system which uses radial axis R(x) that matches the axis of the metro line, with positive direction towards the metro platform, and tangent axis T(Y) perpendicular to R, with a positive direction towards the tunnel 'Dedinje'.

The leveling of the elevation network and the calculations of height differences has been done using a benchmark (GN 236) placed on the portal of the 'Dedinje' tunnel.

Processing of this data using the software has been done using:

- id and name of stability benchmark with value
- id and name of the stability points that determine movements with their approximate values
- measured height differences
- weight $p = \frac{1}{n}$, where n is the number of stations in one direction

With the averaging method (which is in fact Helmert transformation by vertical axis) stable benchmarks are being identified. The movements for these benchmark points have been marked as 0mm, as they were identified as stable points[8].

Results

The data to be used are the (x,y) coordinates of the points at residential buildings. These coordinates are in meters, and the data chosen to be forecasted is $(x - \lfloor x \rfloor)$, and $(y - \lfloor y \rfloor)$, because the deformation are monitored in millimeters. Forecasting results are shown in Figure 5, with x coordinates on the left side, and y on the left; predictions are marked in blue, actuals in red.

FIR training is an iterative process where each cycle consists of one or more forward propagations through the network, and one back propagation to obtain derivatives of the cost function with respect to the network weights. The number of taps in first layer is 5, and 3 in second. The learning rate was set to 0.001, and the number of hidden layers was set to 2.

By increasing the number of neurons in hidden layers, or tapped delays, more features are extracted from the data (and hence the predictions are made more accurate). However, this may lead to over-fitting, and the predictions might go out of range. Over fitting occurs when a forecasting model has too few degrees of freedom. In other words it has relatively few observations in relation to its parameters and therefore is able to memorize individual points rather than learn general patterns. Because the data set is small (a time series of 30 entries), the number of hidden layers was set to 2, and not more.

Discussions

Finite Impulse Response (FIR) Neural Network

In this section, the neural network architecture that will be used to predict future values of (x,y) coordinates (in the local coordination system) is described. Neural network must contain memory in order to process temporal information. There are two basic ways to build memory into neural networks. The first is to introduce time delays in the network and to adjust their parameters during the learning phase. The second way is to introduce positive feedback, thus making the network recurrent. This paper will concentrate on two architectures: finite impulse response (FIR) and recurrent neural networks.

FIR neural network uses the unfold-in-time static approach, and is a functional equivalent of the time delay neural network (TDNN). They do not have feedback connections between units. TDNN provide simple forms of dynamics by buffering lagged input variables at the input layer and/or lagged hidden unit outputs at the hidden layer. FIR network is a feed forward network whose static connection weights between units are replaced by an FIR linear filter that can be modeled with tapped delay lines. After applying the unfold-in-time technique to a FIR, all delays will be removed by expanding the network into a large equivalent static structure. Standard back propagation algorithm is then applied for training. Formally, time delays are identical to time windows and can thus be viewed as and can thus be viewed as autoregressive models.

Linear Systems

It is possible that P, the process whose output is trying to get predicted is governed by linear dynamics. The study of linear systems is the domain of Digital Signal Processing (DSP).

DSP is concerned with linear, translation-invariant (LTI) operations on data systems. Those operations are implemented by filters. The analysis and design of filters effectively forms the core of this field.

Filters operate on an input sequence u[t], producing an output sequence x[t]. They are typically described in terms of their frequency response, i.e. low pass, high-pass, band-stop.

There are two basic filter architectures, known as the Finite Impulse Response (FIR) filter and the Infinite Impulse Response (IIR) filter. FIR filters are characterized by q+1 coefficients:

$$x[t] = \sum_{i=0}^{q} \beta_i u[t-i]$$

These filters implement the convolution of the input signal with a given coefficient vector β_i. The input in these filters $u[i]$ is the impulse function, and the output x is long as $q+1$ which must be finite.

IIR filters are characterized by p coefficients.

$$x[t] = \sum_{i}^{p} \alpha_i x[t-i] + u[t]$$

The input u_i contributes directly to x_i at time t, but, crucially, x_t is otherwise a weighted sum of its own past samples. Because both the input signal and vector α_i are finite in duration, the response asymptotically decays to zero. Once x_i is non-zero, it will make non-zero contributions to future values of x_t infinite number of times.

FIR Filters in ANNs

In Finite impulse response (FIR) Neural networks, each neuron is extended to be able to process temporal features by replacing synapse weights by finite impulse response filters[9]. A general structure of this filter is shown in figure 2.

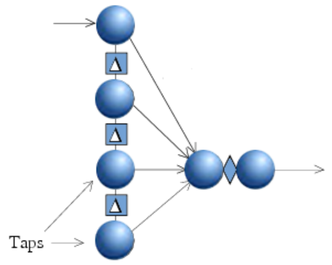

Figure 2. Finite Impulse Response Filter

A multilayer feed forward network is then built using these neurons as shown in figure 3. Network input layer consists of FIR filters feeding the data into neurons in hidden layer. Output of a layer may only connect to the first tap of a node in next layer. Network may have one or several hidden layers. Output layer consists of neurons which receive their inputs from previous hidden layer.

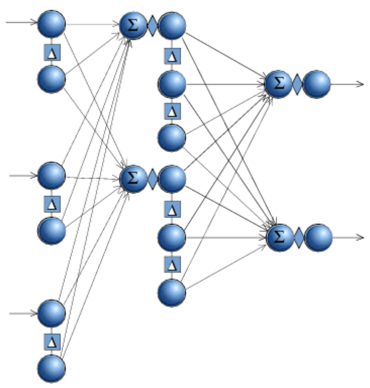

Figure 3 Multilayer Feedforward Neural Network

At each time increment, one new value is fed to input filters, and output neuron produces one scalar value. In effect this structure has the same functional properties as the Time Delayed Neural Networks (TDNN). However, the FIR neural network is interpreted as a vectoral and temporal extension of MLP. This interpretation leads to the temporal back propagation algorithm.

Temporal back propagation

The basic back propagation algorithm assumes that the neural network is a combinational circuit, providing an output for a given input. However, many applications suitable for adaptive learning have inherently temporal structures.

Every time-delay neural network can be represented as a standard static network simply by duplicating nodes and removing delays.

The resultant net is much larger, contains a large number of weight duplications (or triplications), and is not fully interconnected. The process of creating the static equivalent can be thought of as 'unfolding' the network. Once the network is unfolded, the back propagation algorithm can be applied directly to solve the static network.
The output layer of the static network contains the same number of nodes as the output layer of the temporal network. Because this layer has no delay taps, the next layer has no non-physical nodes, and the number of virtual nodes of the static equivalent is equal to the number of filters in that layer times the number of placeholders in each filter. For each layer back to the input, the number of total virtual nodes is a cumulative sum of the number of virtual nodes in that layer plus the number of virtual nodes calculated for the previous layer minus one (because the first placeholder in each filter accepts and propagates its input without delay to the next layer). Mathematically, the notation for total virtual nodes at a layer is:

$$T_l = \begin{cases} 1 & l = L \\ T_{(l+1)} + T_l - 1 & 1 \le l \le L \end{cases}$$

where T_l is the physical number of taps per filter.
The next stage is to actually unfold the network. The first step is to copy down the output nodes, then to copy down all the placeholders of the next layer back, and make each one into a node by prepending a processing element to it. The result is a partial network shown in figure 4. So far, the training algorithm is simply a standard back propagation without modifications, except that the hidden-layer nodes are referenced with three, instead of two variables, where l is used as a subscript to denote that there is only one tap s=1 associated with the output layer L. The network still has the exact physical layout of the temporal net, but without the delay units. The next step is to copy the first layer and second layer weights downward, overlapping placeholders when necessary, until the number of inputs in the first layer equals the number of accumulated inputs calculated.

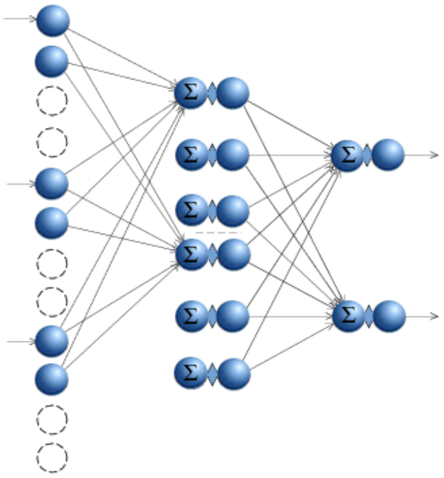

Figure 4. Partial static neural network in the unfolding process

$$\Delta w_{lijt} = -\eta \sum_{n=1}^{T_l} \delta_{lin} y_{(l-1)j(t+n-1)}$$

$$1 \leq l \leq L, \quad 1 \leq i \leq I_l$$
$$1 \leq j \leq I_{(l-1)}, \quad 1 \leq t \leq T_{(l-1)}$$

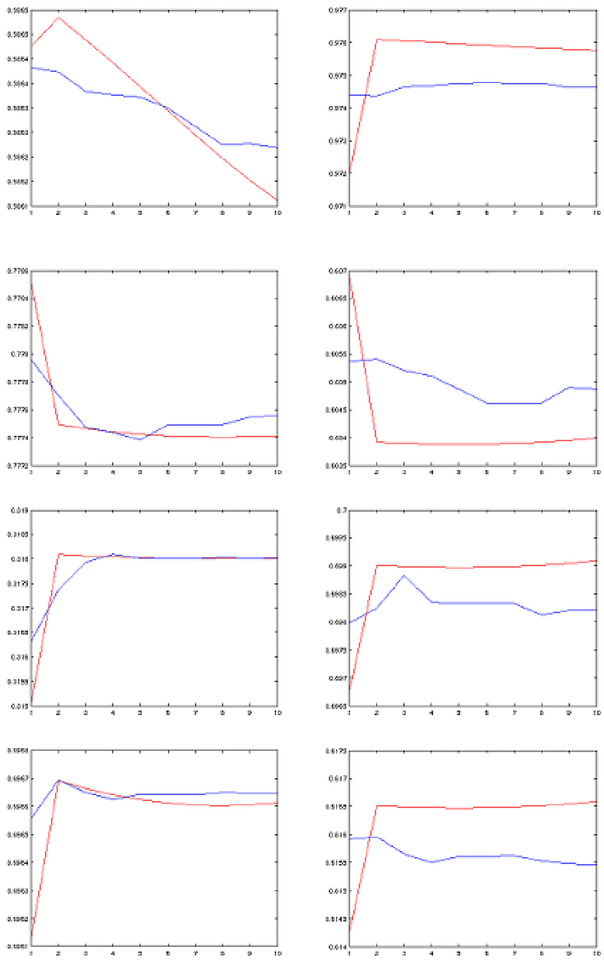

Figure 5. Forecasting results (left: X-axis, right Y-axis).

Conclusion

In this paper, a neural network for forecasting future values of horizontal and vertical movements is described. The points chosen for prediction are located in residential building. Alternative methods include Kalman filtering, and the Finite Element Method (which are widely used in geodetic measurements). Advantage of using ANNs (FIR in this case) is that the user does not need to be an expert in geodesy, or to be familiar with structural engineering in order to make the prediction. Overall the network has performed well, as in the observed cases even though the predictions were not completely accurate, the movement trend was detected in all cases. In order to increase the accuracy of the forecast, a bigger data set is required, which would greatly improve the results.

Acknowledgment

Authors are grateful to the Vienna University of Technology, Austria, University of Novi Sad, Serbia, and University of Banja Luka, Bosnia and Hercegovina and to their research associates for their assistance and encouragement.

References

1 Alexander Reiterer, Martin Lehmann, Milos Miljanovic, Haider Ali, Gerhard Paar, Uwe Egly, Thomas Eiter, Heribert Kahmen, ‘A 3D optical deformation measurement system supported by knowledge-based and learning techniques’, Journal of Applied Geodesy, 3 (2008), 1–13,

2 Bulatović V, Sušić Z & Ninkov T, ‘Estimate of the ASTER-GDEM regional systematic errors and their removal’, INTERNATIONAL JOURNAL OF REMOTE SENSING, ISSN: 0143-1161, (2012), 5915-5926.

3 Caspary W F, *Concepts of Network and Deformation Analysis*, (Monograph 11, Kensington, Australia: University of New South Wales, School of Surveying), 1987.

4 Egger K & Keller W, *New Instruments and their Applications for Geodetic Deformation Measurements on Dams*, Transactions, 12th International Congress on Large Dams, Mexico, Reprinted by Kern & Co AG (now Leica-Geosystems), Aarau, Switzerland, 1976.

5 Ninkov T, *Deformaciona analiza i njena praktična primena*. Geodetski list, 39 /62/ (7–9), (1985) 167–178.

6 Egger K, *Terrestrial Measurement of Deformation', in: Swiss National Committee on Large Dams – Working Group on Dam Monitoring, The Geodetic and Photogrammetric Deformation Measurement of Dams*, published on the occasion of the 19th ICOLD Congress, Florence, Italy, 1997

7 Egger K, ‘Mesures terrestres de deformation’, in: Swiss National Committee on Large Dams – Working Group on Dam Monitoring, 1993

8 Heck B, Das Analyseverfahren Des Geodätischen Instituts Der Universität Karlsruhe Stand 1983. V Deformationsanalysen ’83. Geometrische Analyse und

Interpretation von Deformationen Geodätischer Netze (str. 153– 182
9 Milos Miljanovic, 'Comparative analysis of Recurrent and Finite Impulse Response Neural Networks in Time Series Prediction', Indian Journal of Computer Science and Engineering, (2012), 180-191.