Model of Indirect Temperature Measurement by Neural Network

Andrea Mojžišová [1] and ***Karol Kostúr*** [2]

Modelovanie nepriameho merania teploty neurónovou sieťou

Determination of temperature inside steel rolls during annealing is key to achieving the best quality of the process. Today it is not typical to measure this temperature and the time of annealing is determined empirically. In this article we present a model of neural networks for indirect measurement of surface temperature array of steel roll.

***Key words**: neural networks, indirect measurement, error in measurement.*

Introduction

The goal of our research is to develop an intelligent system for indirect measurement of inner temperature (ISNMT) of materials processed by metallurgy and machine industry. There is a wide range of sensors for surface temperature measurement; however there is no detector which could be placed inside of a material and be able to measure the inner temperature during the heat process. Existence of such a sensor would allow:

a) more effective quality control of temperature behaviour in real time,
b) development of new methods of indirect control for systems with high latency.

Intelligence of the system is based on three parallel evolving branches:

a) theory of non-stationary heat conduction, heat transfer by radiation and convection,
b) theory of neural networks,
c) theory of the Kalman filter predictor.

We developed a neural network model of indirect temperature measurement in three stages:

- design and verification of network model using measurements in lab furnace,
- design and verification of network model using measurements in lab conditions,
- design and verification of network model using measurements from U.S. Steel.

Neural Networks

A neural network consists of formal neurons which are interconnected like neurons in the brain – output of one neuron is input for several other neurons like terminals of axons are connected through synaptic connections with other dendrites.

The neural network is a universal tool capable to compute virtually anything a computer can. The main advantage of neural network is its ability to learn. An important application of neural networks is prediction and decision making.

The basic element of mathematical mode of neural network is a formal neuron. It has n real-valued inputs $x_1, ..., x_n$, which are priced by n real-valued synaptic weights $w_1, ..., w_n$. (these weights might be negative). Weighted sum of inputs is called the inner potential of the neuron.

$$\xi = \sum_{i=1}^{n} \mathrm{w_i}\, x_i \qquad (1)$$

After reaching the threshold value h, the potential ξ affects the output y of the neuron, which

[1] *RNDr. Andrea Mojžišová,* Technická univerzita v Košiciach, F. BERG, Ústav riadenia a informatizácie výrobných procesov, Boženy Němcovej 3, 043 54 Košice, andrea.mojzisova@tuke.sk
[2] *prof. Ing. Karol Kostúr, CSc.*, Technická univerzita v Košiciach, F. BERG, Ústav riadenia a informatizácie výrobných procesov, Boženy Němcovej 3, 043 54 Košice.
(Recenzovaná a revidovaná verzia dodaná 28. 11. 2008)

is an analogy to electric impulse of axon. The output value $y = \sigma(\xi)$ is defined by activation function σ:

$$y = \sigma(\xi) = \begin{cases} 1, & if\ \xi \geq 0 \\ 0, & if\ \xi < 0 \end{cases}, \text{ where } \xi = \sum_{i=1}^{n} \mathrm{w}_i\, x_i$$

(2)

The best known and most widely used model of neural network is a multilayer network with backpropagation learning algorithm. We used this type of neural network in our study.

Back-propagation Neural Networks

Let X be a set of n input neurons, Y set of m output neurons, ξ_i real input potential and y_i real output of neuron i. The neuron's output for this type of network is defined by equation:

$$y_i = \sigma_i(\xi_i), \text{ where } \sigma_i(\xi_i) = \frac{1}{1 + e^{-\lambda_i \xi}}$$

(3)

Network error $E(w)$ related to a training set is defined as the sum of partial errors $E_k(w)$ of network concerning each training example and depends on the configuration of network w.

$$E(w) = \sum_{k=1}^{p} E_k(w), \text{ where } E_k(w) = \frac{1}{2}\sum_{j \in Y}\left(y_j(w, x_k) - d_{kj}\right)^2$$

(4)

Partial network error $E_k(w)$ related to the k-th training example is directly proportional to the sum of square of difference between the real network value and the desired output where $y_j(w,x_k) - d_{kj}$ is the j-th output error for the k-th training example. If this error is zero, then the corresponding weights are non-adaptable; otherwise it is -1 or 1.

Design and verification of network model using measurements from U.S. Steel

In this stage of research we measured ambient temperature of a steel roll during annealing and created a model for determination of the surface temperature of the roll. These experiments took place at U.S. Steel.

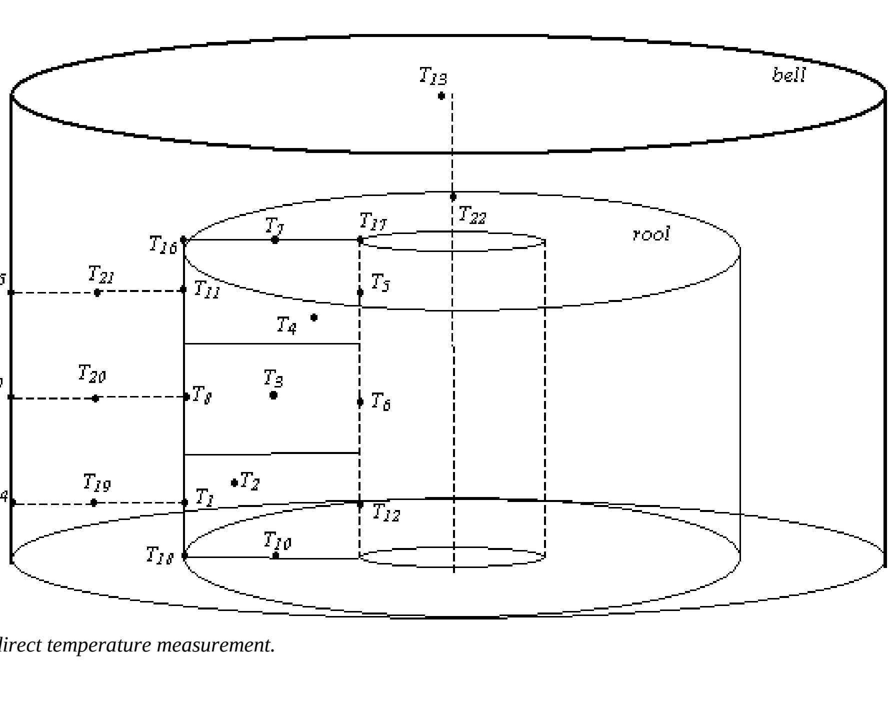

Fig. 1. The scheme of direct temperature measurement.

Learning data

Learning data is a set of inputs and desired outputs. By assuming these patterns and by modifying weights we train the network how to answer input values. Afterwards the network answers unknown patterns by interpolation of known patterns.

Input data for temperatures T1, T2, T3, T4, T45, T5, T9, T910, T10, T11, T12 and T13 is temperature A4 (also time and Etalon temperatures).

Input data were captured during a single survey at U.S. Steel. Having only one batch of data we had to split it optimally between learning and testing datasets. Initially, we used first 50000 inputs as a learning set. However, the learning process failed for this setup. Finally we used first 30000 inputs as a learning set and all (200000) inputs as a testing set for networks 1-3-1 and 1-5-1. For further networks 2-5-1 and 3-5-1 we used full new 7 measurements (number 35-41).

Design Structure of Neural Networks

The topology of the network is the number of input neurons, the number of neurons in the hidden layer, and the number of output neurons.

For our research we prepared networks with topology 1-3-1,1-5-1, 2-5-1, and 3-5-1. We measured temperatures for three steel rolls. However, we present results for the first roll only. We prepared neural networks for each temperature and for each topology.

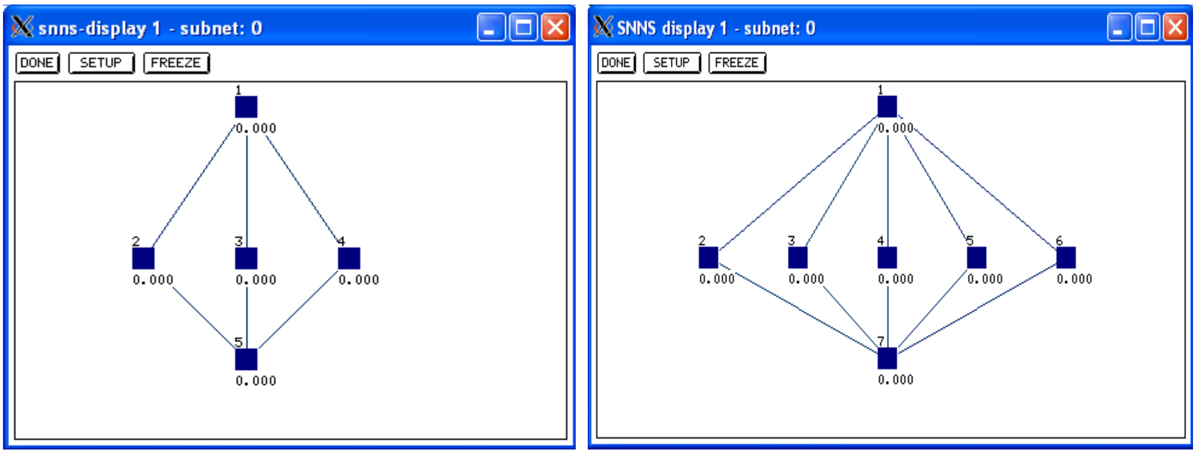

Fig. 2. Neural Network 1-3-1 and 1-5-1.

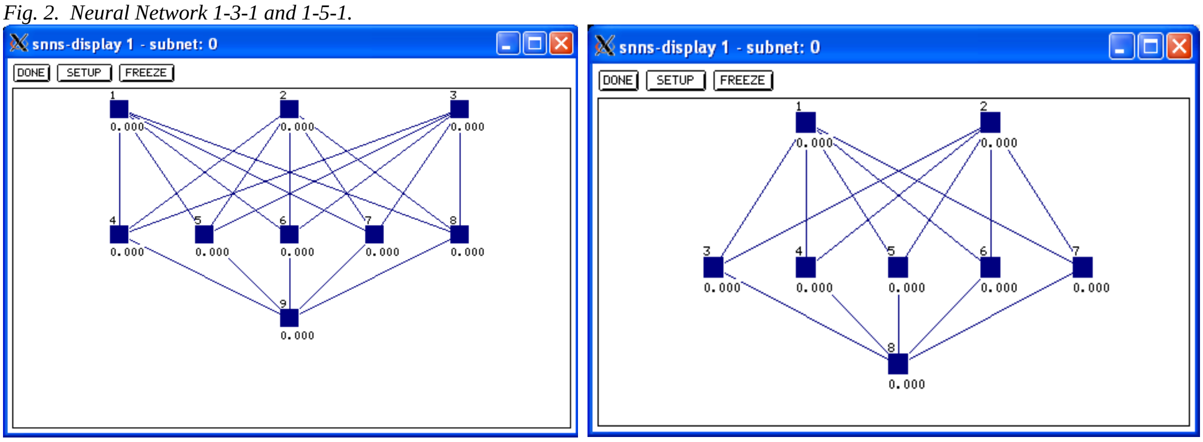

Fig. 3. Neural Network 2-5-1 and 3-5-1.

<u>Topology of network 1-3-1 (1-5-1):</u>
Input neurons: neuron 1 = A1
Hidden layer: neurons 2—4, (2—6)
Output neurons: neuron 5 (7) = T1 (T2, T3, T4, T45, T5, T9, T910, T10, T11, T12 , T13)

<u>Topology of network 2-5-1 (3-5-1):</u>
Input neurons: neuron 1 = A1, 2=time (3=etalon temperature)
Hidden layer: neurons 3—7, (4—8)
Output neurons: neuron 8 (9) = T1 (T2, T3, T4, T5, T9, T10, T11, T12 , T13)

Test Phase Results

We designed neural networks in the SNNS environment (Stuttgart Neural Network Simulator). For each input set and for topology 1-3-1 and 1-5-1 we trained 8 networks with different number of training cycles (5000, 10000, ..., 40000). We used the Std_Backpropagation learning function. The initial setup of weights was random. An example of learning process for network 1-3-1 and temperature T1_1 is depicted in figure 4.

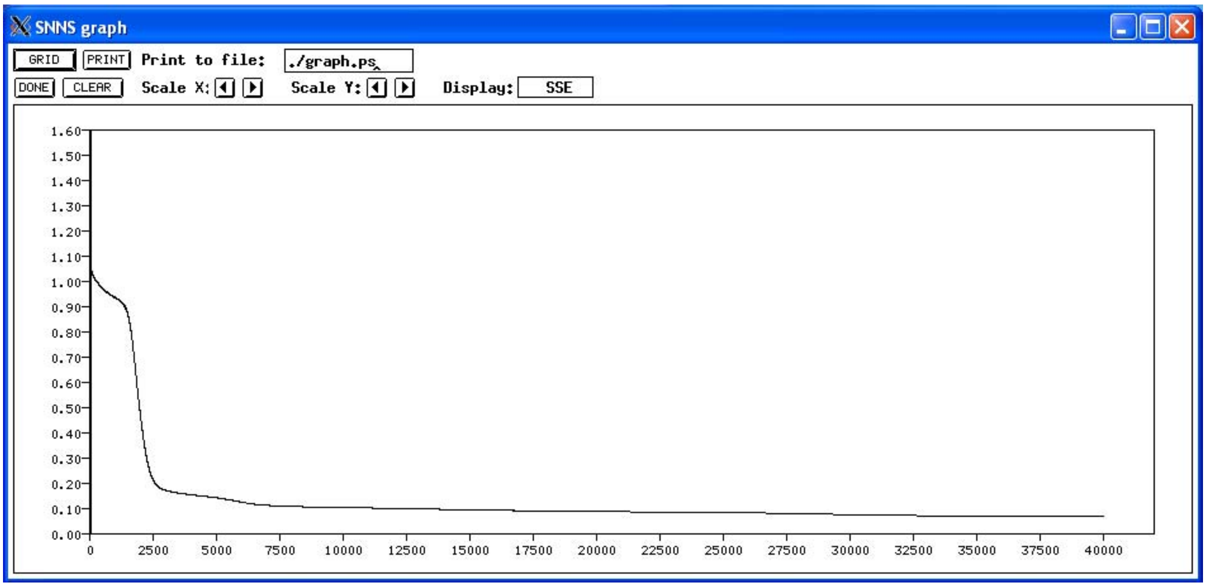

Fig. 4. Learning process for network 1-3-1.

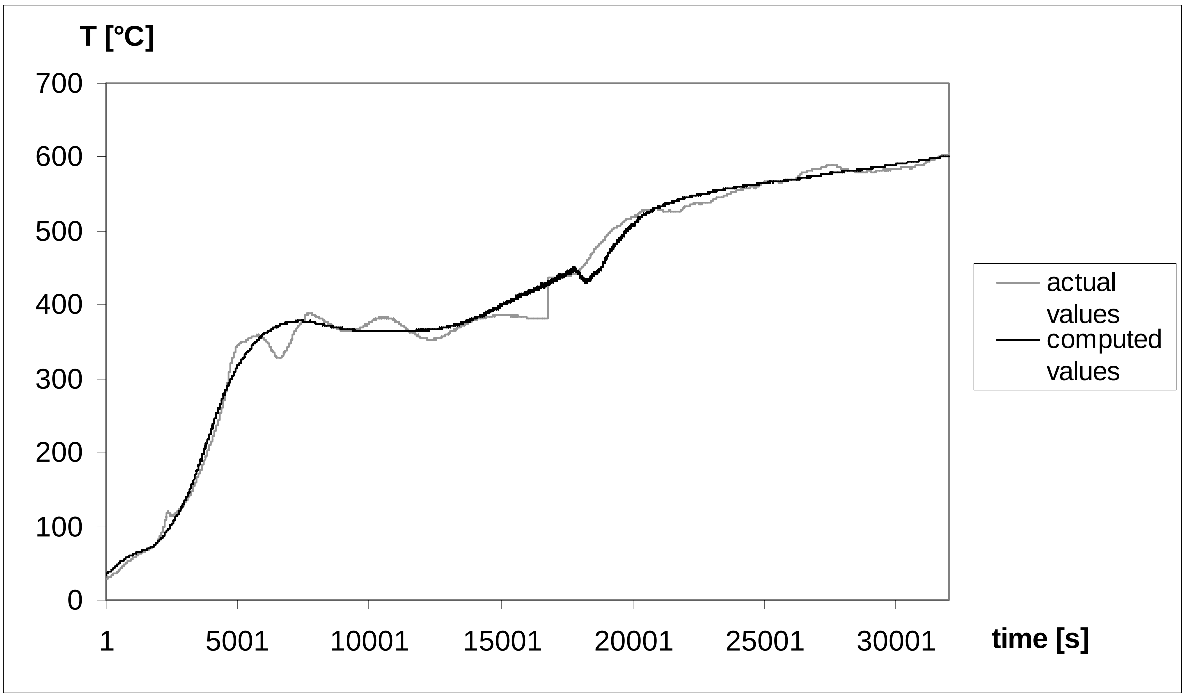

Fig. 5. Comparisons of actual and computed values for temperature T1_5.

In figure 5 there are comparisons of actual and computed values for temperature T1_5 in the best network. We evaluated the error for each topology and each learning set. The partial error of each output at time k was calculated using the following formula:

$$Ei_k = \frac{|STi_k - MTi_k|}{STi_k} \times 100 \quad , \qquad (5)$$

where STi_k is the actual temperature of i-th output at time k and MTi_k is computed temperature of the corresponding output at time k.

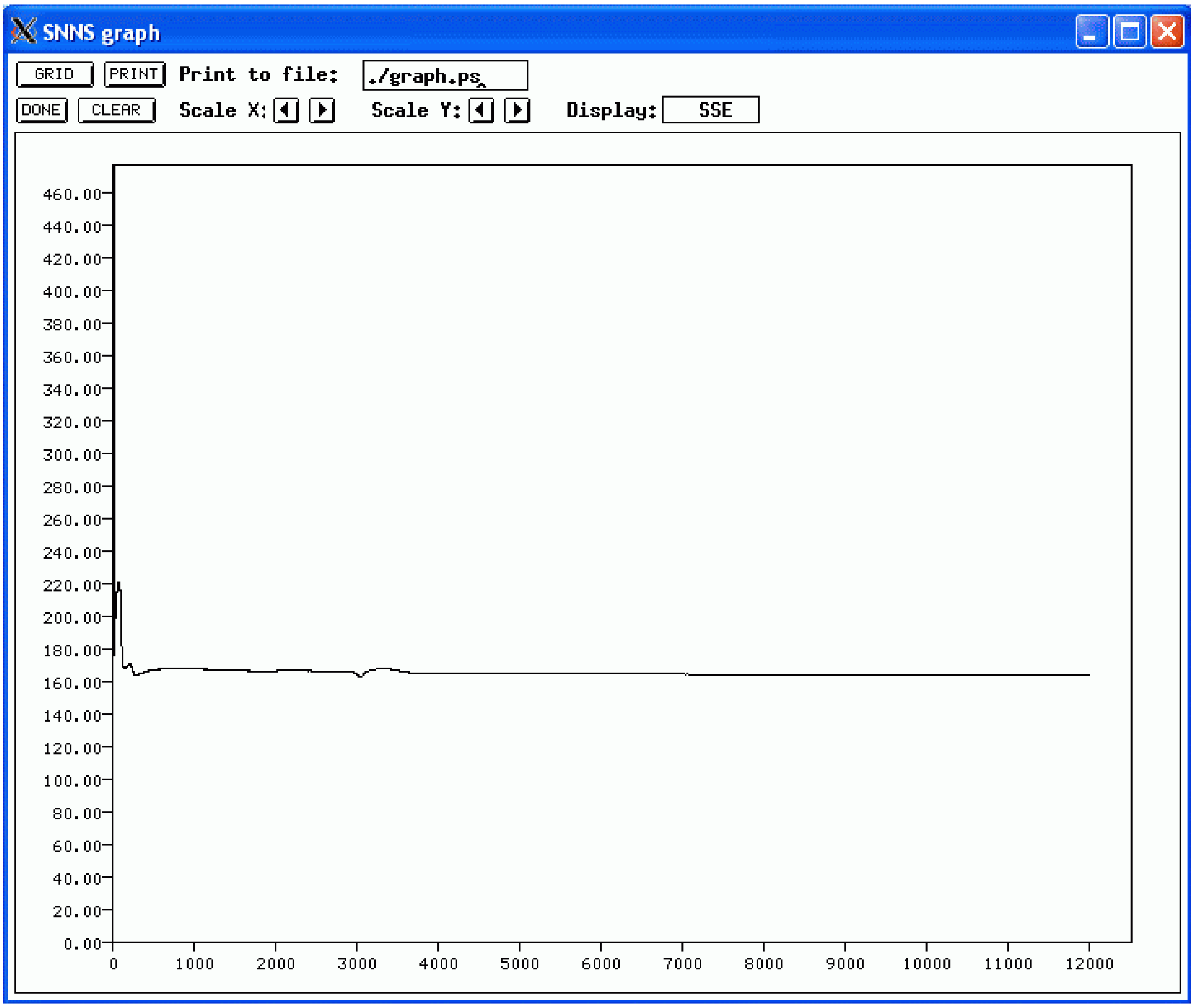

Fig. 6. Learning process for network 2-5-1.

For each input set and for topology 2-5-1 and 3-5-1 we trained 5 networks with different number of training cycles (1000, 2000, 3000, 4000 and 5000). We used the Std_Backpropagation learning function. Initial setup of weights was random. An example of learning process for network 1-3-1 and temperature T1_1 is depicted in figure 6. The final distribution of weights (fig. 7) is interesting too. SNNS emphasizes neurons with specific character.

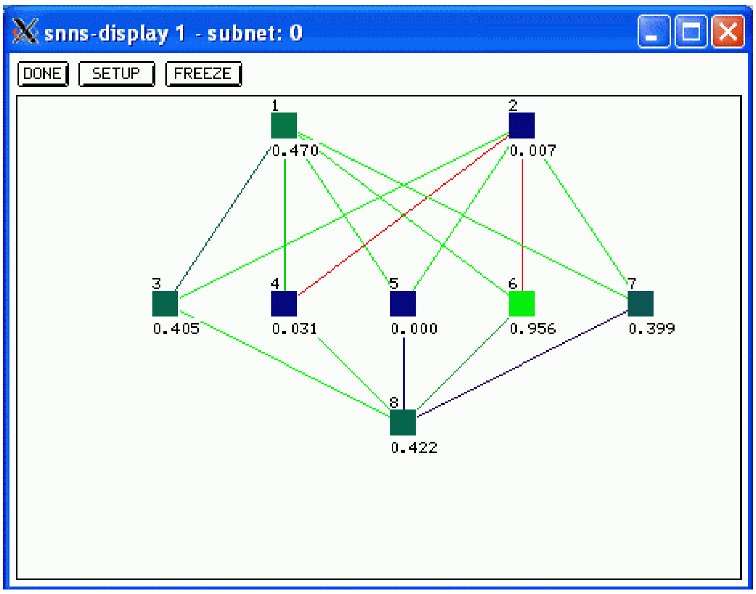

Fig. 7. Trained network 2-5-1.

Conclusion

Results presented in this article suggest that this type of neural networks is appropriate as a model of indirect measurement of surface temperature of a steel roll using ambient temperature.

Errors are mostly caused by defective input data. The learning set contains pairs with equal input but different output temperatures. In these cases the neural network is not capable of determinig the correct value and has to approximate results. If the faulty data were separated out, the network would be able to better understand dependencies between inputs and outputs.

In the future we shall evaluate neural networks with different topologies. We shall acquire more appropriate input data. Our goal will be to optimize the model to decrease the error level.

This paper was created in thanks for projects VEGA 1/3346/06, VEGA 1/2179/05, VEGA 1/2160/05 and AV 4/0016/05 from the Slovak Grant Agency for Science., soluted on faculty BERG, TU in Košice.

References

Stuttgart Neural Network Simulator User Manual, *VERSION 4.1*

Sinčák, Andrejková: Neurónové siete I.; *ELFA Press, Košice, 1996.*

Šíma J., Neruda R. 1996. Theoretical Issues of Neural Networks; MATFYZPRESS, Prague, 1996.

Bulnová, A., Kostúr, K.: Indirect measurement of temperatures by neural networks; In: ICCC'2006: International Carpathian Control Conference, Rožňov pod Radhoštěm, Czech Republic, *May 29-31, 2006.*

Mojžišová, A., Kostúr, K.: Indirect measurement of temperatures by neural networks. In: ICCC'2006 : Proceedings of 7th International Carpathian Control Conference : Rožnov pod Radhoštěm, May 29-31, 2006. *Ostrava: VŠB-TU, p. 385-388. ISBN 80-248-1066-2.*

Mojžišová, A., Kostúr, K.: Neural network model of indirect temperature measurement. In: ICCC'2007 : Proceedings of 8th International Carpathian Control Conference: Štrbské Pleso, Slovak Republic, May 24-27, 2007. *Košice: TU, s. 474-477. ISBN 978-80-8073-805-1.*