Sborník vědeckých prací Vysoké školy báňské - Technické univerzity Ostrava
číslo 2, rok 2007, ročník LIII, řada strojní
článek č. 1570

Jolana ŠKUTOVÁ[*]

AN ORTHOGONAL NEURAL NETWORK FOR NONLINEAR FUNCTION MODELLING

MODELOVÁNÍ NELINEÁRNÍ FUNKCE ORTOGONÁLNÍ NEURONOVOU SÍTÍ

Abstract

Neural networks provide one of the means for identification or control of a nonlinear system. Multilayer feedforward network is often used in those areas. Using this kind of neural network may result in premature termination of learning before the network weights reach a global minimum, a small training convergence, an unsuitable weight parameters initialize and the stability for on-line implementation. In this paper, an orthogonal neural networks (ONN) is presented. The ONN's hidden layer consists of neurons with orthogonal activation function. The ONN have a much higher learning speed than the multilayered feedforward networks.

Abstrakt

V oblasti modelování nebo řízení neuronových sítí se často využívá neuronová síť se zpětným šířením, tzv. perceptronová neuronová síť, která se potýká s problémy lokálního minima, nízkou rychlostí konvergence, volbou počátečních parametrů vah a počtem neuronů. V tomto příspěvku je prezentována ortogonální neuronová síť s jednou skrytou vrstvou, která využívá ortogonální aktivační funkce. Tato neuronová síť umožňuje odstranit výše uvedené problémy. Ortogonální neuronová síť má mnohem vyšší rychlost učení než dopředné sítě.

1 INTRODUCTION

In area of a function approximation, a prediction as the identification of unknown system, a system control and optimization, robotics are now common used the neural networks. There are multilayer feedforward neural networks are often applied, concretely perceptron and radial neural network by type of the activation function. With their application, feedforward neural network have very problematic point of learning convergence speed, an acquirement of global minimum, a settings of initial weight parameters.

There are many publications deal with various methodologies which remove this problems and one of the radical approach is the selection novel structure of neural network to removing this sticking point. This contribution describes novel model of neural network with unconventional internal ordering of neuron's connection and an orthogonal type of activation function. Finally, it's proposed conclusion of simulation modelling of orthogonal neural network for chosen nonlinear function.

2 ORTHOGONAL NEURAL NETWORK STRUCTURE

ONN (orthogonal neural network) is feedforward neural network with multi inputs and single output (MISO – multi-input-single-output) and a hidden layer with orthogonal activation function in hidden neurons (Fig. 1). The inputs of neural networks are distributed into orthogonal neurons blocks

[*] Ing., Department of Control Systems and Instrumentation, Faculty of Mechanical Engineering, VSB – Technical University of Ostrava, 17. listopadu 15/2172, Ostrava-Poruba, tel. (+420) 59 732 4192, e-mail jolana.skutova@vsb.cz

for each input. The number of neurons for each input signals is arbitrary and *i*th neuron correspond to *i*th orthogonal function order. The number of orthogonal neurons is given by

$$N_{ortg} = \sum_{i=1}^{m} N_i, \tag{1}$$

where:

m – the number of inputs $[\text{-}]$,

N_i – the number of neurons for each input $[\text{-}]$.

The other layer of ONN is arranged to nodes which consist of products combinations of the particular outputs from orthogonal neurons and it's defined as

$$\phi_{n_1 \ldots n_m}(x) = \prod_{i=1}^{m} \phi_{n_i}(x_i), \qquad x = [x_1 \quad x_2 \quad \ldots \quad x_m]^T \tag{2}$$

where:

m – the dimension of input vector $[\text{-}]$,

ϕ_i – the output of orthogonal function implemented by each hidden layer neuron $[\text{-}]$.

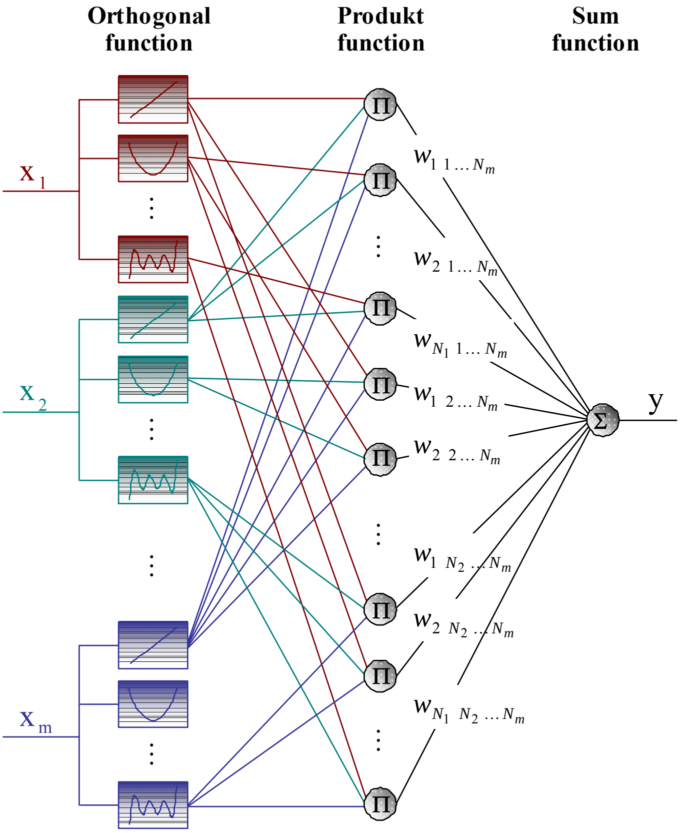

Fig. 1 The structure of orthogonal neural network

The ONN output is given by sum of the all output nodes from previous layer (2) and one is can be mathematically expressed as

$$\hat{y}(x,v)=\sum_{n_1=1}^{N_1}\cdots\sum_{n_m=1}^{N_m} w_{n_1\cdots n_m}\phi_{n_1\cdots n_m}(x)=\Phi^T(x)\hat{w}, \tag{3}$$

where:

Φ^T – the transformed input vector $[\text{-}]$,

$\hat{w}$ – the transformed weight vector $[\text{-}]$.

Orthogonal Activation Function

The rate of convergence due to using of orthogonal functions as activation function in neural network is larger then the rate of convergence for perceptron or radial neural networks when it's sticking point. The set of one-dimensional orthogonal functions ϕ_{n_i} are defined as

$$\Phi^T\int_{x_{\min}}^{x_{\max}}\phi_i^{(N)}(x)\,\phi_j^{(N)}(x)\,dx=\delta_{ij} \tag{4}$$

where:

δ_{ij} – Kronecker delta function $[\text{-}]$,

which is given by

$$\delta_{ij}=\begin{cases}1 & pokud \quad i=j\\ 0 & pokud \quad i\neq j\end{cases}. \tag{5}$$

There has been number of the orthogonal polynomials, e.g. Hermite, Legendre, Laguerre, Chebyshev and Fourier polynomials [5]. Thus far for modelling given nonlinear function applied Legendre and Fourier polynomial only. The orthogonal neural network with Fourier activation function had a sufficient modelled outputs, hence next experiments was realized by the ONN with Legendre polynomial.

3 TRAINING METHOD OF ORTHOGONAL NEURAL NETWORK

Generally, the learning process is performed by adapting the network weight such that the expected value of the mean squared error between network output and training output is minimized. The gradient descent-based learning algorithms are popular training algorithm for neural networks. To train proposed network, learning rules are determined from Lyapunov-like stability analysis. The cost function ONN is given by

$$E=\frac{1}{2}e^2=\frac{1}{2}(y-\hat{y})^2, \tag{6}$$

where:

e – the learning error $[\text{-}]$,

$\hat{y}$ – the output of ONN $[\text{-}]$,

y – the actual output $[\text{-}]$.

The gradient descent algorithm adjusts network weights in such a way that the square of the neural network learning error changes in a negative gradient direction and then weight adaptation is given by

$$\Delta\widehat{w} = -\delta\frac{\partial E}{\partial\widehat{w}}, \tag{7}$$

where:

$\Delta\widehat{w}$ – the variation of network weights $[-]$,

δ – the network learning rate $[-]$,

$\widehat{w}$ – the network learning rate $[-]$,

E – the mean square error $[-]$.

Thus the ONN's weight update law for the instantaneous gradient descent algorithms is given as

$$\widehat{w}(t) = \widehat{w}(t-1) + \delta e\overline{\Phi}(t), \tag{8}$$

where:

$\Delta\widehat{w}$ – the transformed input vector consisting of orthogonal functions $[-]$,

e – the learning error $[-]$.

To guarantee of ONN's training stability, the learning rate δ must be bounded by

$$0 < \delta < \frac{2}{\overline{\Phi}^T\overline{\Phi}}. \tag{9}$$

This learning rate is one of the parameters that are the part of experiment to the acquisition of neural network optimal model depending on chosen modelled system. [1]

4 MODELLING OF NONLINEAR FUNCTION

This paper presents the nonlinear function modelling results that are step to development in modelling of real-time system as well as the orthogonal neural network apply to control system subsequently. The algorithm for orthogonal neural network implementation in Matlab/Simulink program have already been designed and compiled. Further, the necessary source code optimization has been performed regarding to decrease in the computational time. The verification and experiments with particular structure design has been implemented on an approximation of nonlinear function that is expressed as

$$y(x) = \frac{2}{1+e^{-2x}} - 1 \tag{10}$$

The neural network structure is given by one input and one output of nonlinear function and hidden layer includes 20 neurons with orthogonal function of Legendre activation function kind, concretely from 1st. to 20th degree. The working range of input signal correspond to the values on the interval [-2.5, 2.5]. The ONN's training carry out in epochs (one epoch presents one training algorithm loop for whole training data). To quality results of the neural network weight update, a number of epochs have been chosen maximally 100 epochs. The learning rate value has been the question of experiments and one was settings on fourth part of maximum parameter value δ in Eq. (9).

The initial weight was selected in interval [-1; 1] randomly. The global minimum acquisition was demonstrated by multiple running of training process with different initial weight values. The final weight values have been steadied (Fig. 4) compared with weight values in first training epoch.

The orthogonal neural network was implemented in Simulink program with minimal number of the blocks. Further the individual degrees of Legendre polynomial were realized by particular M-files. The very sticking point was the composition of ONN's product nodes in Simulink program.

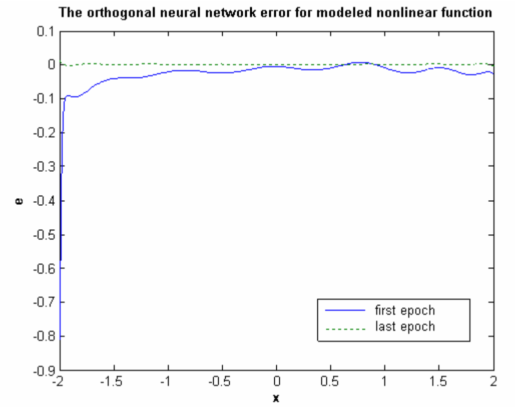

Fig. 2 The ONN's error for given nonlinear function

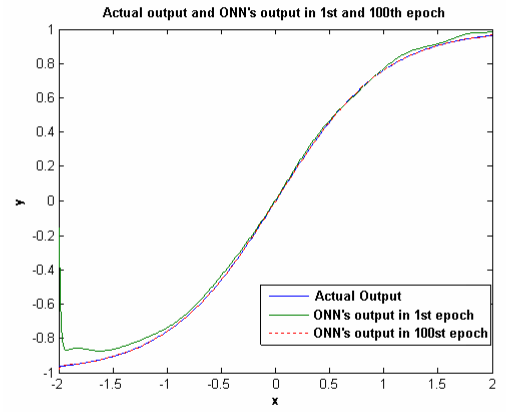

Fig. 3 The desired actual output and ONN's output for given nonlinear function

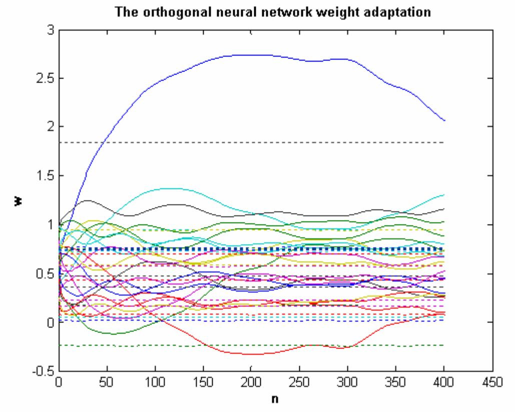

Fig. 4 The ONN's weight values for first epoch (solid line) and last epoch (dash line) of training process

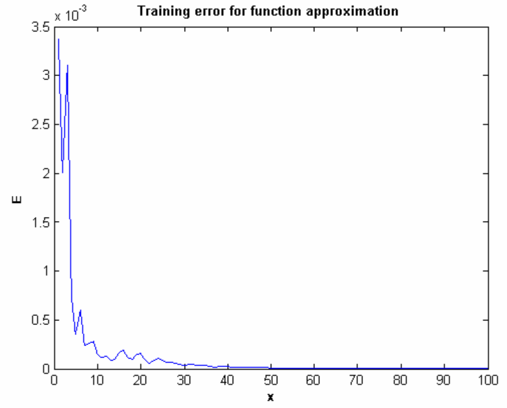

Fig. 5 The ONN' training convergence

5 CONCLUSIONS

The orthogonal neural networks pruning away some fundamental sticking point of perceptron neural networks applications, as a rate of convergence speed of the ONN learning process, an achieving of global minimum while neural network training. In contrast with perceptron networks, it requires different training data to acquirement of a robust neural network.

It now proceed tests and experiments with the selection of inputs number and inputs pattern, the selection of training data sets and the number of orthogonal neurons (Legendre polynomials order) for real pressure-air system. For a system control using neural network exists many control strategies using neural networks that design of a suitable neural networks structure is examined and testing in order to a quality control.

This work was developed with the support of the research project GACR 101/06/0491.

REFERENCES

[1] LEONDES, C. T. *Control and Dynamic Systems.* Academic Press, 1998. 438 pp. ISBN 0124438679.

[2] YANG, S. S. & TSENG, C. S. An Orthogonal Neural Network for Function Approximation. In *IEEE Transaction on Systems, Man, Cybernetics, Part B.* 1996, Vol. 26, Issue. 5, pp. 779-785. ISSN 1083-4419.

[3] SOLOWAY, D. & HALEY, P. J. Neural Generalized Predictive Control. A Newton-Raphson Implementation. In *Proceedings of the IEEE International Symposium on Intelligent Control*. Dearborn: ISICs, 1996, pp. 277-282. ISBN 978-0780329782.

[4] GUO, B. & YU, J. Model Adaptive Control Based on a Compound Orthogonal Neural Network. *International Journal of Information Technology.* 2006, Vol. 12, Issue. 5, pp. 99-107. ISSN 0218-7957.

[5] WIKIPEDIA. The Free Encyclopedia. [online] available from www: <URL: http://en.wikipedia.org/

Reviewer: doc. Ing. Zora Jančíková, CSc., VŠB - Technical University of Ostrava