APPROXIMATION ABILITIES OF NEURO-FUZZY SYSTEMS

Maria Mrówczyńska, Józef Gil
University of Zielona Góra, Institute of Building Engineering

ABSTRACT

The paper presents the operation of neuro-fuzzy systems of an adaptive type as specific structures of mathematical models, intended for the approximation of multi-variable functions in the domain of real numbers. Neuro-fuzzy systems operate on the basis of a set of inferences "if-then", generated with the use of algorithms for the self-organisation of data grouping and the estimation of relations between fuzzy experiment results.
The article includes a description of models of neuro-fuzzy systems by Takaga, Sugeno, Kang (TSK), and Wang and Mendel (WM), which, when defined as continuous functions, enable the approximation of any multi-variable non-linear functions. Moreover, the module structure of the models enables the implementation of a parallel multi-layer structure, analogous to the structure of classic neural networks.

1. INTRODUCTION

The approximation of non-linear functions, specified in multidimensional data sets, can be performed with the use of neural networks, which serve as universal approximators. They enable the representation of non-linear functions $y = f(x)$, where x is the input vector, and y the output vector. In spite of obvious advantages, neural networks are often criticised for the lack of the possibility of explaining the decisions made. For this reason, knowledge contained in neural networks represented by weights does not have a clear form and is not helpful to the user (Łęcki, 2008; Osowski, 2006).
Fuzzy systems in contrast to neural networks, which process data in a numerical form, can also process data by means of linguistic values (in a symbolical form) written in the form of fuzzy formulas (Duch, 2001). Fuzzy systems have solid theoretical bases and describe imprecise values in a precise manner on the basis of a membership function, which leads to coherent and logical conclusions. Both types of networks have a common feature, which is parallel data processing. This aspect enables the use of fuzzy systems and neural networks for solving the same tasks, and explaining the purposefulness of the approach connecting both methods.
As a result of connecting the operation procedure of neural networks and fuzzy systems, a system with the ability to process information has been created, which is known as fuzzy systems or neuro-fuzzy networks (Jang, 1996; Łęcki, 2008). The article presents an application of a neuro-fuzzy system for approximating a multivariable function on the assumption that the input vectors, and consequently values of the membership function as well as output values oh the system are void of disturbances. The paper describes the operation and architecture of networks by Takaga – Sugeno – Kang, and Wang – Mendel, including algorithms for the specification of an optimum number of reasoning formulas.

2. FUZZY INFERENCE SYSTEM

Fuzzy sets, as a generalisation of ordinary sets, are characterised by fuzzy membership of a particular set, i.e. a particular member can belong to a number of sets in part. A fuzzy set is described as a set of ordered pairs $(x, \mu_A(x))$, where $\mu_A(x)$ denotes a membership function. The value of the membership function is within the range $[0,1]$. Therefore,

$$0 \leq \mu_A(x) \leq 1. \tag{1}$$

The membership extent $\mu_A(x) = 1$ denotes a full membership of the member x of the set A, $\mu_A(x) = 0$ denotes the lack of membership, and intermediate values $\mu_A(x)$ denote partial membership.

In practice, membership functions of the Gaussian type, triangular functions, and trapezoidal functions are the most frequently used. For example, the form of the Gaussian membership function is defined as follows (Osowski, 2006)

$$\mu_A(x_i) = \exp\left[-\left(\frac{x_i - c_i}{\sigma_i}\right)^2\right] \tag{2}$$

where x is a variable with the centre in c and the width c .

An essential feature of fuzzy sets is inference, which determines their practical importance. The basic inference rule "if – then" also called a fuzzy implication has the form (Markowska – Kaczmar, 2006)

if x is A then y is B,

where x and y denote variables, and A and B are values defined by membership functions of the variables x and y. For the variables x_i, $(i = 1,2,\ldots,N)$, the conclusion has the form:

if x_1 is A_1 and x_2 is A_2 and … and x_N is A_N then y is B.

The random value of the membership function is the most frequently interpolated in the form an algebraic product

$$\mu_A(x) = \prod_{i=1}^{N} \mu_A(x_i). \tag{3}$$

In order to connect variables and fuzzy rules, a fuzzy system has been introduced in the form of a fuzzificator on the input and a defuzzificator on the output (fig. 1). The fuzzificator transforms input data into a fuzzy set, and the defuzzificator transforms the fuzzy set into a unique solution point. The transformation of the N dimensional vector $\boldsymbol{x} = [x_1, x_2, \ldots, x_N]^T$ into the fuzzy set A is characterised by the membership function $\mu_A(x)$. A diagram of a fuzzy system with a fuzzificator and a defuzzificator has been presented in fig. 1 (Osowski, 2006).

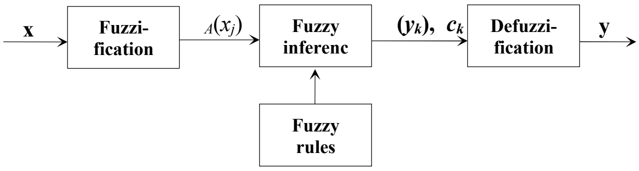

Fig. 1. A diagram of a fuzzy system.

Defuzzification can be carried out, for example, on the basis of average values of centres (centre of gravity)

$$y = \frac{\sum_{k=1}^{M} c_k \mu_A^{(k)}\left(\boldsymbol{x}^{(k)}\right)}{\sum_{k=1}^{M} \mu_A^{(k)}\left(x^{(k)}\right)} \tag{4}$$

The symbols $\mu_A^{(k)}$ and c_k included in the formula (4) denote respectively: the membership function of fuzzy sets A corresponding to the input vector, and the parameter c_k is the centre of the membership function, i.e. the point where the membership function reaches its highest value.

3. MODELS OF NEURO-FUZZY NETWORKS BY TAKAGA – SUGENO – KANG (TSK)

In a model of a TKS type neuro-fuzzy network, the basic inference pattern for M inference rules and N variables x_i is as follows:

if x_1 **is** $A_1^{(1)}$ **and** x_2 **is** $A_2^{(1)}$ **and** … **and** x_N **is** $A_N^{(1)}$ **then** $y_1 = p_{10} + \sum_{j=1}^{N} p_{1j} x_j$

… … …

if x_1 **is** $A_1^{(M)}$ **and** x_2 **is** $A_2^{(M)}$ **and** … **and** x_N **is** $A_N^{(M)}$ **then** $y_M = p_{M0} + \sum_{j=1}^{N} p_{Mj} x_j$ **.**

The fuzzification function has the form of a generalised Gaussian function for each variable x_i separately (Duch, 2001; Osowski, 2006)

$$\mu_A(x_i) = \frac{1}{1 + \left(\frac{x_i - c_i^{(k)}}{\sigma_i^{(k)}}\right)^{2b_i^{(k)}}}, \tag{5}$$

and for M inference rules the output signal from the data network is the formula

$$y(\boldsymbol{x}) = \frac{1}{\sum_{k=1}^{M} w_k} \sum_{k=1}^{M} w_k y_k(\boldsymbol{x}), \tag{6}$$

where $y_k(\boldsymbol{x}) = p_{k0} + \sum_{j=1}^{N} p_{kj} x_j$, and the weights w_k are interpreted as the value $\mu_A^{(k)}(\boldsymbol{x})$ of a datum aggregated according to the formula (5).

The TSK neuro-fuzzy network, whose detailed architecture has been presented in fig.2, has two parametric layers (the first and the third), whose parameters are adapted in the process of learning. In the first layer the variables x_i are fuzzified, and the parameters $c_i^{(k)}, \sigma_i^{(k)}, b_i^{(k)}$ are specified. In the second layer a product aggregation of particular variables x_i is carried out, which is expressed in the form of a random value of the membership ratio $w_k = \mu_A^{(k)}(\boldsymbol{x})$. In the third layer values of the function $y_k(\boldsymbol{x}) = p_{k0} + \sum_{j=1}^{N} p_{kj} x_j$ (a TSK function) are generated on the basis of the adaptation of the parameters p_{kj} as linear weights. The fourth layer consists of two summation neurons, where one calculates the sum of signals weighed $y_k(\boldsymbol{x})$, and the other the sum of weights $\sum_{k=1}^{M} w_k$. The fifth layer contains one neuron which calculates the network output signal $y(\boldsymbol{x})$ according to the dependence (4).

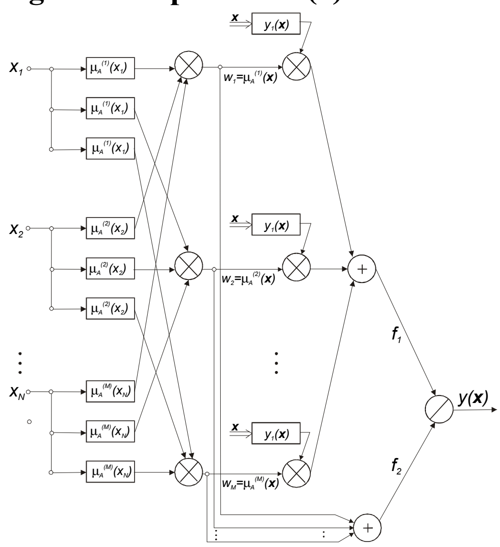

Fig. 2. The Architecture of a TSK neuro-fuzzy network (Osowski, 2006).

4. MODEL OF A NEURO-FUZZY NETWORK BY WANG-MANDEL (WM)

A simplified version of a TSK network is a neuro-fuzzy network by Wang – Mendel, which has four layers. The first layer fuzzifies input variables, the second layer aggregates the firing values of the predecessor. The third layer has the form of a product of the aggregation of inference rules and a zero rank polynomial (normalisation signal

generation). The fourth layer is the output layer, where a neuron calculates the output signal $y(\boldsymbol{x})$ according to the dependence

$$y(\boldsymbol{x}) = \frac{\sum_{k=1}^{M} c_k \, \mu_A^{(k)}(\boldsymbol{x})}{\sum_{k=1}^{M} \mu_A^{(k)}(\boldsymbol{x})}. \tag{7}$$

In a WM network the first layer and the third layer are parameter layers. In the first layer the parameters $c_i^{(k)}, \sigma_i^{(k)}, b_i^{(k)}$ are adapted, and in the third layer values of weights are specified, whose physical interpretation corresponds to the centres c_k of the membership function of the successor of the kth fuzzy inference rule.

5. SELECTED SELF-ORGANISATION ALGORITHMS IN NEURO-FUZZY SYSTEMS

Self-organisation is included in localised areas. The model of self-organisation used in the article consists in assigning the vector x to a data group which is represented by the centre c. Cluster centres dividing multidimensional space are used to initiate inference rules which can ensure convergence to a global minimum (Osowski, 2006).
The self-organisation process on the basis of the set of input vectors x can be obtained by means of one of the following algorithms: mountain clustering method, subtractive clustering, C – means clustering, Gustafson – Kessel method. The first two algorithms are used to preliminarily initiate centres, and the other two, according to the shape of the location area of the data constituting the cluster, enable the specification of optimum values of centres by means of the formula

$$c_j = \frac{\sum_{j=1}^{p} \mu_{ij}^{m} \boldsymbol{x}_j}{\sum_{j=1}^{p} \mu_{ij}^{m}} \tag{8}$$

where μ_{ij} is the extent of membership of all the vectors $\boldsymbol{x}_j$ $(j = 1,2,\dots p)$ of the centres c_j $(j = 1,2,\dots,M)$ and m is the weight ratio (usually $m = 2$). Details of the procedure have been presented in the article (Osowski, 2006).

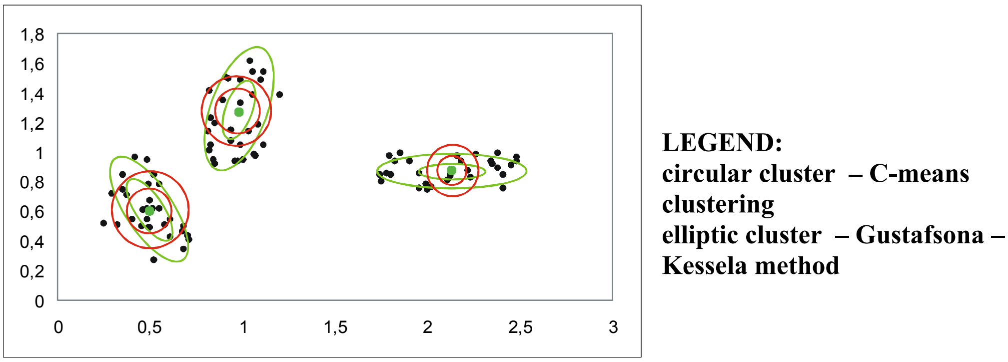

Fig. 3. Activity areas of rules for the c – means clustering and the Gustafson– Kessel method.

6. ALGORITHM FOR THE SPECIFICATION OF AN OPTIMUM NUMBER OF INFERENCE RULES

The determination of the number of inference rules whose activity corresponds to the data group contained in the cluster area is an important component of the structure of fuzzy systems. The criteria for the quality of input data grouping are as follows (Osowski, 2006):

fuzzy cluster volume V_h

$$V_h = \sum_{i=1}^{M} \sqrt{\det(\boldsymbol{S}_i)} \tag{9}$$

where M **- the number of inference rules,** $\boldsymbol{S}$ **- the covariance matrix expressed by the formula**

$$\boldsymbol{S}_i = \frac{\sum_{j=1}^{p} \mu_{ij}^m (\boldsymbol{x}_j - \boldsymbol{c}_i)(\boldsymbol{x}_j - \boldsymbol{c}_i)^T}{\sum_{j=1}^{p} \mu_{ij}^m} \tag{10}$$

average partition density D_A

$$D_A = \frac{1}{M} \sum_{i=1}^{M} \frac{U_i}{\sqrt{\det(\boldsymbol{S}_i)}} \tag{11}$$

where the value $U_i = \sum_k u_{ik}$ **is calculated for** $(\boldsymbol{x}_k - \boldsymbol{c}_i)^T \boldsymbol{S}_i^{-1} (\boldsymbol{x}_k - \boldsymbol{c}_i) < 1$.

average internal distance D_w **between cluster data and its centre**

$$D_w = \frac{1}{M} \sum_{i=1}^{M} \frac{\sum_{k=1}^{p} \mu_{ik}^m d_{ik}^2}{\sum_{k=1}^{p} \mu_{ik}^m} \tag{12}$$

average cluster fattening t_A

$$t_A = \frac{1}{M} \sum_{i=1}^{M} t_i \tag{13}$$

where t_i **is the numerical ratio between the smallest and the greatest eigenvalue of the covariance matrix** $\boldsymbol{S}_i$.

Small values of the factors V_h **and** t_A**, and great values of the factors** D_A **and** D_w **mean good quality of the division of data into clusters. Because of the fact that it is impossible to simultaneously meet these conditions, a global statistical measure is usually defined**

$$\alpha = a_1 V_h - a_2 D_A - a_3 D_w + a_4 t_A \tag{14}$$

where $a_i > 0$, for $i = 1,...,4$ are scale factors.

By changing the number of clusters *M* it is possible to specify the dependence between the global measure and the real value of *M*, and on this basis to choose an optimum number of inference rules. The minimum value of the global factor of grouping quality α corresponds to the sub-optimum number of clusters. It results from the diagram presented in fig. 4 that the optimum number of inference rules for the set of points in question is 16.

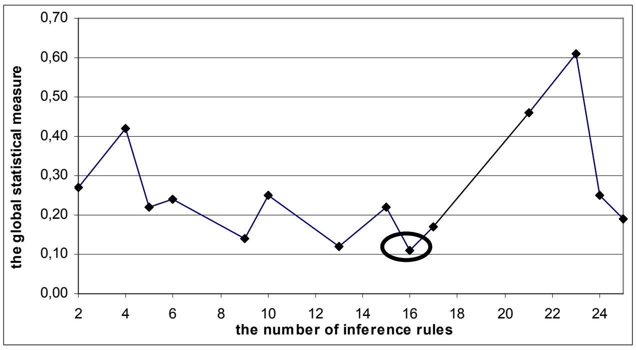

Fig. 4. Values of the global statistical measure.

7. NUMERICAL EXAMPLE

An evaluation of the approximation abilities of neural networks and fuzzy systems by Wang – Mandel, and Takaga - Sugano – Kang has been presented on the example of a non-linear function (Osowski et al., 2006).

$$z(\mathbf{x}) = z(x_1, x_2) = \sin(x_1)\sin(x_2)\exp\left(-x_1^2 - x_2^2\right) \tag{15}$$

The size of the teaching and testing set was 200 data pairs (x, *z*(x)), randomly generated within the variation range [-2, 2]. The results of the calculations have been presented in table 1, and error surfaces for the most favourable approximation method (TSK 16 inference rules) have been graphically presented in fig. 5.

Table 1. Mean square errors *RMSE* of approximation according to the number of inference rules

The approximation method	The number of inference rules		
	5 rules	10 rules	16 rules
neuro-fuzzy systems TSK	0,07992	0,01205	0,00736
neuro-fuzzy systems WM	0,14116	0,01949	0,01254
multi-layered perceptron (Levenberg – Marquardt learning rule)	0,02891	0,0135	0,00732
spline			0,01997

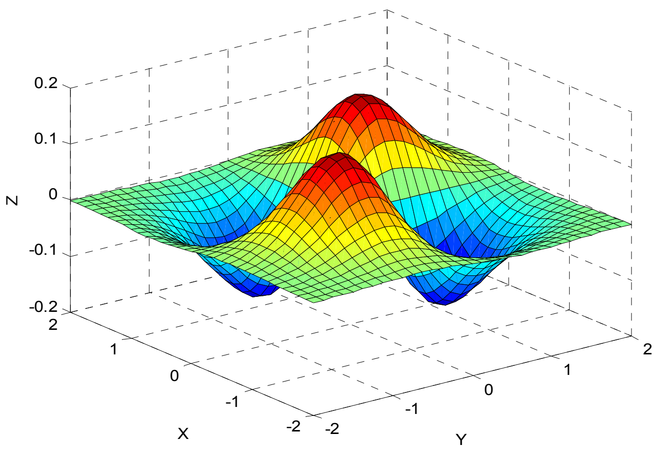

Fig. 5. The form of the function (15) being represented.

8. CONCLUSIONS

Neuro-fuzzy networks enable the approximation of a non-linear multi-variable function with unrestricted precision. At the same time, it is necessary to realise that the result will be somewhat different from the results of interpolation carried out in the test set, regardless of the number of clusters used. The numerical experiments carried out by the authors indicate that the quality of approximation evaluated on the basis of the testing error with the use of the TSK fuzzy system and neural networks can be compared on condition that the right method of teaching neural networks is applied. In the case in question the Levenberg – Marquardt algorithm exceeds other methods of teaching neural networks as far as accuracy is concerned. In order to solve the problem of approximation different structures of networks can be used (a multi-layer perceptron, a radial network etc.), but in most cases, the use of the TSK system brings the best results.

REFERENCES

Duch W., Korbicz J., Rutkowski L., Tadeusiewicz L., *Sieci neuronowe, t.6.*, Akademicka Oficyna wydawnicza EXIT, Warszawa 2000.

Jang J.S., Sun C.T., Mitzutani E., *Neuro – Fuzzy and Soft Computing*. Prentice Hall, New York 1997.

Łęski J., *Systemy neuronowo - rozmyte.* Wydawnictwo Naukowo – Techniczne, Warszawa 2008.

Markowska-Kaczmar U., *Ekstrakcja reguł z sieci neuronowych. Podejście ewolucyjne.* Oficyna Wydawnicza Politechniki Wrocławskiej. Wrocław 2006.

Osowski S., *Sieci neuronowe do przetwarzania informacji,* Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa 2006.

Osowski S., Cichocki A., Siwek K., *MATLAB w zastosowaniu do obliczeń obwodowych o przetwarzania sygnałów*. Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa 2006.