RBF NEURO-FUZZY SYSTEM WITH NON-SINGLETON FUZZIFIER AND HYBRID LEARNING PROCEDURE

DANUTA RUTKOWSKA
Technical University of Czestochowa, Department of Computer Engineering
Armii Krajowej 36, 42-200 Czestochowa, Poland
Phone: (48-34) 3610043, Fax: (48-34) 3250546
email: drutko@kik.pcz.czest.pl

ABSTRACT: This paper presents RBF neuro-fuzzy systems with singleton and non-singleton fuzzifier. A connectionist network representing fuzzy system with non-singleton fuzzifier, product inference rule, and center-average defuzzification method, has been derived. A hybrid learning procedures based on the gradient (back-propagation) algorithm combining with genetic algorithm or clustering algorithm has been proposed. Conclusions and remarks concerning the RBF network and non-singleton fuzzifier are included.

KEYWORDS: Neuro-fuzzy system, singleton and non-singleton fuzzifier, RBF network, learning algorithm.

I. INTRODUCTION

There has been a great deal of research and many practical applications in the field of combining fuzzy systems and neural networks. This kind of synergistic combinations of these two different methods in one system is concerned as neuro-fuzzy system. There are various approaches to creating such systems. In this paper we discuss a connectionist multi-layer representation of fuzzy system, which is equivalent to RBF neural network. There has been a lot of interest in the fact that neural networks as well as RBF networks and fuzzy systems are universal approximators, i.e. they are capable of approximating any continuous function to any desired accuracy. RBF neuro-fuzzy systems possess features of RBF network local approximation and merit of learning ability.

In this paper we present RBF neuro-fuzzy systems with non-singleton fuzzifier. However the singleton fuzzifier is the most popular and commonly studied but the non-singleton one is applicable when the input signals are corrupted by noise and there is a need to account for uncertainty in the data. In case of lack of uncertainty in the input data, the RBF neuro-fuzzy systems with non-singleton fuzzifier, derived in this paper, resolve themselves to the adequate systems which are based on the singleton fuzzifier. It is worth mentioning that there are not many researchers who apply the non-singleton fuzzifier. The attempt to deal with non-singleton fuzzy systems have been made by [Mouzouris, Mendel 1997].

Hybrid learning procedures, which combine gradient algorithm (derived from the back-propagation method) with genetic algorithm or clustering method, are proposed for training systems under consideration. Remarks and conclusions concerning the application of RBF neuro-fuzzy network with non-singleton fuzzifier are included in this paper.

II. NEURO-FUZZY SYSTEM WITH SINGLETON FUZZIFIER

The multi-input, single-output neuro-fuzzy system based on the singleton fuzzification, product inference rule and center-average defuzzification is given by the following expression [Wang 1994; Rutkowska, Pilinski, Rutkowski 1997]:

$$\overline{y} = \frac{\sum_{k=1}^{N} \overline{y}^k t_k}{\sum_{k=1}^{N} t_k} \tag{1}$$

where $\overline{y}^k$ are centers of membership function m_{B^k}, and t_k are given as follows:

$$t_k = \prod_{i=1}^{n} m_{A_i^k}(\overline{x}_i) \tag{2}$$

where $\bar{x}_i$ are input values, $m_{A_i^k}$ and m_{B^k} are membership functions of fuzzy sets A_i^k and B^k, i=1,..., n, k=1,...,N, in the fuzzy rule base, which consists of a collection of N rules in the following form:

$$R^{(k)}: \ \textbf{IF} \ x_1 \ \text{is} \ A_1^k \ \text{AND} \ \dots \ \text{AND} \ x_n \ \text{is} \ A_n^k \ \textbf{THEN} \ y \ \text{is} \ B^k \tag{3}$$

where x_i are linguistic variables.

The neuro-fuzzy system described by expression (1) is shown in Fig. 1. Elements of layer L1 realize the membership functions $m_{A_i^k}$ for i=1,...,n and k=1,...,N, denoted for short as m_i^k, respectively. Outputs of layer L1 are equal to the values of these membership functions for given input values $\bar{x}_i$. The antecedent matching degrees t_k are obtained at the outputs of layer L2. The last two layers, L3 and L4, perform center-average defuzzification operation. Parameters which denote the centers $\bar{y}^k$ of the membership functions m_{B^k} are visible in layer L3. 'Crisp' output of the system, given by formula (1), is obtained at the output of layer L4.

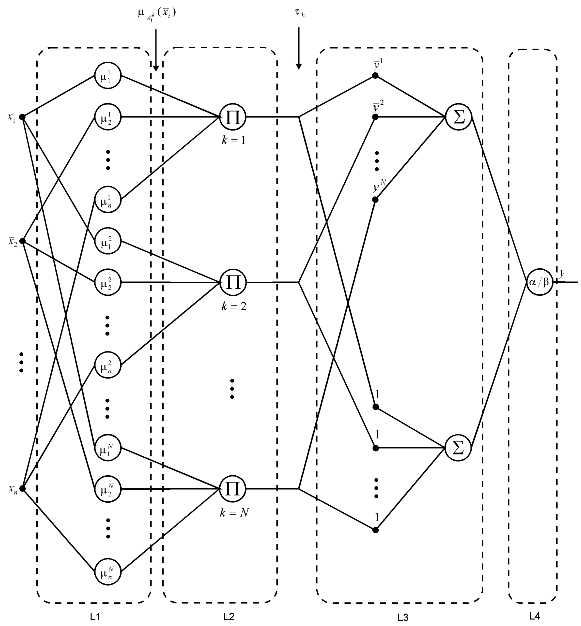

Figure 1: Neuro-fuzzy system with singleton fuzzifier, product inference rule and center-average defuzzifier

Similar connectionist networks of fuzzy systems as shown in Fig. 1 based on different defuzzification methods have been presented in [Rutkowska 1998a; Rutkowska 1999].

The membership functions, realized by the elements of layer L1, can be bell-shaped (Gaussian), triangular or for example trapezoidal functions. The most often used in neuro-fuzzy systems are Gaussian and triangular membership functions. Elements of layer L2 perform multiplication function, as we see in Fig. 1, when the inference process is based on the product operation (Larsen's rule). For fuzzy inference process based on min-operation (Mamdani's rule) the elements of layer L2 realize minimum instead of multiplication. Learning procedures for neuro-fuzzy systems based on Larsen's and Mamdani's rule with Gaussian and triangular membership functions are presented in [Rutkowska 1997].

III. RBF NEURO-FUZZY SYSTEM

Let us consider the neuro-fuzzy system presented in section II based on the product inference rule (Fig. 1) with Gaussian membership functions. We choose the Gaussian membership functions $m_{A_i^k}(x_i)$ with centers $\bar{x}_i^k$ and widths s_i^k, given by the following formula

$$m_{A_i^k}(x_i) = \exp\left[-\left(\frac{x_i - \bar{x}_i^k}{s_i^k}\right)^2\right] \tag{4}$$

for i=1,...,n and k=1,...,N.

It is easy to notice that the neuro-fuzzy system shown in Fig. 1 with the elements of layer L1 realizing the Gaussian membership functions given by expression (4) is a normalized version of RBF neural network proposed by [Moody, Darken 1989]. If $s_i^k = s^k$ for all i=1,...,n , then layers L1 and L2 in Fig. 1 represent one hidden layer with N hidden units which realize Gaussian RBF as a product

$$G_k(\bar{\mathbf{x}}) = \exp\left[-\left(\frac{\left\|\bar{\mathbf{x}} - \bar{\mathbf{x}}^{\mathbf{k}}\right\|}{s^k}\right)^2\right] = \prod_{i=1}^{n} \exp\left[-\left(\frac{\bar{x}_i - \bar{x}_i^k}{s^k}\right)^2\right] \tag{5}$$

where $\bar{\mathbf{x}} = \left[\bar{x}_1 \ldots \bar{x}_n\right]^T$ and $\bar{\mathbf{x}}^{\mathbf{k}} = \left[\bar{x}_1^k \ldots \bar{x}_n^k\right]^T$ for k=1,...,N. The output $\bar{y}$ of the normalized RBF neural network illustrated in Fig. 1 is given by the following expression

$$\bar{y} = \frac{\sum_{k=1}^{N} \bar{y}^k \, G_k(\bar{\mathbf{x}})}{\sum_{k=1}^{N} G_k(\bar{\mathbf{x}})} \tag{6}$$

Formula (6) describes the normalized RBF neural network with N radial basis functions G_k given by equation (5) in the hidden layer and the weights $\bar{y}^k$ in the output layer. The weights of this RBF network are equal to the centers of the membership functions m_{B^k} . The value of the radial basis functions G_k for given input vector $\bar{\mathbf{x}}$ equals to t_k which are defined by formula (2) and called the degree of activation of the rule $R^{(k)}$ in the form (3). Expression (6) is the same as formula (1). Thus neuro-fuzzy system described in section II is equivalent to the normalized RBF neural network. The equivalence of fuzzy system and RBF network has been shown in [Jang, Sun 1993]. Both of these systems are universal approximators, what means that they are capable of approximating any continuous function to any desired accuracy [Kosko 1992; Poggio, Girosi 1990]. It is well known that multi-layer neural networks possess similar feature [Cybenko 1989] but they perform global approximation, while RBF networks construct local approximation to nonlinear input-output mapping.

IV. RBF NEURO-FUZZY SYSTEM WITH NON-SINGLETON FUZZIFIER

In this paper we consider multi-input, single-output fuzzy systems mapping X→Y, where X ⊂ $\mathbf{R^n}$ and Y ⊂ $\mathbf{R}$, see Fig. 1. A fuzzifier performs a mapping from observed crisp input space X ⊂ $\mathbf{R^n}$ to the fuzzy sets defined in X. In section II and section III the neuro-fuzzy systems with singleton fuzzifier has been studied. It is the most commonly used fuzzifier, which maps input vector $\bar{\mathbf{x}} = \left[\bar{x}_1 \ldots \bar{x}_n\right]^T \in \mathrm{X}$ into fuzzy set $A' \subset \mathrm{X}$ characterized by a membership function $m_{A'}(\mathbf{x})$ defined by

$$m_{A'}(\mathbf{x}) = \begin{cases} 1 & \text{if} \quad \mathbf{x} = \bar{\mathbf{x}} \\ 0 & \text{if} \quad \mathbf{x} \neq \bar{\mathbf{x}} \end{cases} \tag{7}$$

The non-singleton fuzzifier is characterized by the

$$m_{A'}(\mathbf{x}) = \exp\left[-\left(\frac{\left\|\mathbf{x} - \bar{\mathbf{x}}\right\|}{s}\right)^2\right] = \prod_{i=1}^{n} \exp\left[-\left(\frac{x_i - \bar{x}_i}{s}\right)^2\right] \tag{8}$$

where $\bar{\mathbf{x}} = \left[\bar{x}_1 \ldots \bar{x}_n\right]^T$ denotes center and s width of this non-singleton membership function.

The fuzzy inference engine determines a mapping from the fuzzy sets in the input space X to the fuzzy sets in the output space Y. Let $\bar{B}^k \subset \mathrm{Y}$ be fuzzy set given by the *sup-star* composition

$$m_{\overline{B}^k}(y) = \sup_{\mathbf{x}\in X} \left\{ m_{A'}(\mathbf{x}) * m_{A_1^k \times \cdots \times A_n^k \to B^k}(\mathbf{x}, y) \right\} \tag{9}$$

where $*$ could be any operator in the class of *t*-norms. For the product inference rule, using expressions (4) and (8), formula (9) becomes

$$m_{\overline{B}^k}(y) = m_{B^k} \prod_{i=1}^{n} g_i^k \tag{10}$$

where

$$g_i^k = \sup_{x_i} \left\{ \exp\left[-\left(\frac{x_i - \overline{x}_i}{s} \right)^2 \right] \exp\left[-\left(\frac{x_i - \overline{x}_i^k}{s_i^k} \right)^2 \right] \right\} \tag{11}$$

We find the value of g_i^k by maximizing the expression in curly brackets in formula (11). It attains its maximum at

$$\tilde{x}_i^k = \frac{s^2\, \overline{x}_i^k + (s_i^k)^2\, \overline{x}_i}{s^2 + (s_i^k)^2} \tag{12}$$

Consequently,

$$g_i^k = \exp\left[-\left(\frac{\tilde{x}_i^k - \overline{x}_i^k}{s_i^k} \right)^2 \right] \exp\left[-\left(\frac{\tilde{x}_i^k - \overline{x}_i}{s_i} \right)^2 \right] \tag{13}$$

A simple calculus leads to the following result

$$g_i^k = \exp\left[-\left(\frac{\overline{x}_i - \overline{x}_i^k}{\overline{s}_i^k} \right)^2 \right] \tag{14}$$

where

$$\overline{s}_i^k = \sqrt{s^2 + (s_i^k)^2} \tag{15}$$

Applying center-average defuzzification method, given by the following formula

$$\overline{y} = \frac{\sum_{k=1}^{N} m_{\overline{B}^k}\left(\overline{y}^k\right) \overline{y}^k}{\sum_{k=1}^{N} m_{\overline{B}^k}\left(\overline{y}^k\right)} \tag{16}$$

and using expressions (14) and (10), one can easily obtain the description of neuro-fuzzy system with non-singleton fuzzifier, product inference rule, and center-average defuzzifier, as follows

$$\overline{y} = \frac{\sum_{k=1}^{N} \overline{y}^k \prod_{i=1}^{n} \exp\left[-\left(\frac{\overline{x}_i - \overline{x}_i^k}{\overline{s}_i^k} \right)^2 \right]}{\sum_{k=1}^{N} \prod_{i=1}^{n} \exp\left[-\left(\frac{\overline{x}_i - \overline{x}_i^k}{\overline{s}_i^k} \right)^2 \right]} \tag{17}$$

Comparing expression (17) with the description of neuro-fuzzy system with singleton fuzzifier, given by formula (1) and equations (2) and (4), we notice the difference, which is the value of width of the Gaussian functions. For neuro-fuzzy systems with singleton fuzzifier formula (1) contains the membership functions of antecedents of the rules in the form (3), given by equation (4) with the values of widths equals to s_i^k and centers $\bar{x}_i^k$. For neuro-fuzzy systems with non-singleton fuzzifier formula (17) contains the Gaussian functions with the same values of centers, denoted as $\bar{x}_i^k$ but the widths of these Gaussian functions are given by expression (15), so their values depend on the values of the widths of both membership functions $m_{A'}$ and $m_{A_i^k}$.

Similarly as in section II, we can present the neuro-fuzzy systems with non-singleton fuzzifier, given by formula (17), in the form of connectionist network, which is illustrated in Fig. 2. This network differs from the neuro-fuzzy system shown in Fig. 1 in the layer L1, which contains elements realizing Gaussian functions with different width patrameters.

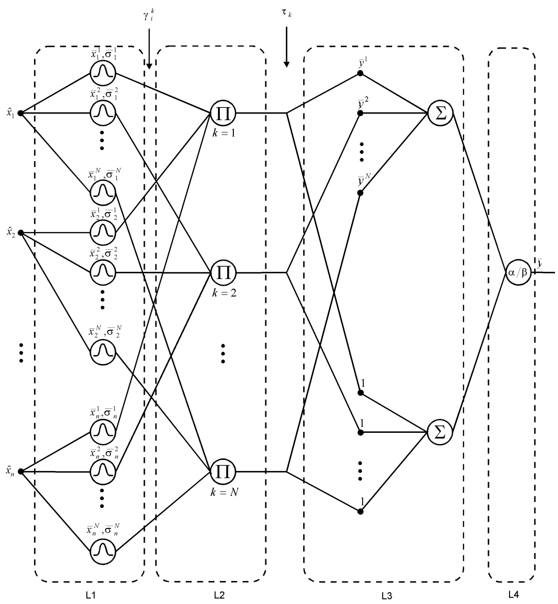

Figure 2: Neuro-fuzzy system with non-singleton fuzzifier, product inference rule and center-average defuzzifier

Analogous networks for different defuzzification methods have been studied in [Rutkowska 1997; Rutkowska 1998b; Rutkowska, Nowicki, Rutkowski 1999].

Since the network shown in Fig. 2 is very similar to that which is illustrated in Fig. 2, it is obvious that it can be regarded as the normalized RBF neural network, similarly as the RBF neuro-fuzzy system discussed in section III. However, it is necessary to assume that $\bar{s}_i^k = \bar{s}^k$ for all i=1,...,n and we have to take into account this assumption in formula (17). In this case equation (15) takes simpler form, such that

$$\bar{s}^k = \sqrt{s^2 + (s^k)^2} \tag{18}$$

where s^k is the width parameter of Gaussian radial basis function G_k given by expression (5), for k=1,...,N, and s is the width parameter of Gaussian non-singleton membership function.

V. HYBRID LEARNING ALGORITHM

The four-layer feed-forward networks shown in Fig. 1 and Fig. 2 can be trained by applying the back-propagation idea using for training neural networks (see e.g. [Zurada 1994]). Based on the learning sequence $(\bar{x}_1(1),\ \ldots,\ \bar{x}_n(1);\ d(1)),$ $(\bar{x}_1(2),\ \ldots,\ \bar{x}_n(2);\ d(2)),\ \ldots$, where $d(t),\ t=1,\ 2,\ \ldots,$ are desired outputs of the system, we wish to obtain the optimal values of the parameters (centers and widths) of membership functions $\mu_{A_i^k}$ and μ_{B^k}, which minimize the error defined as follows:

$$e(t) = \frac{1}{2}\left(\bar{y}(t) - d(t)\right)^2 \tag{19}$$

In this paper we employ the center-average defuzzification method, so only one parameter (center) of the membership function μ_{B^k} is taken into account. Both parameters have to be trained when we apply center-of-sums defuzzification formula [Rutkowska 1997; Rutkowska 1998a, 1998b].

The appropriate gradient descent recursions, with momentum terms, take the form

$$\bar{y}^k(t+1) = \bar{y}^k(t) - \eta \frac{\partial e(t)}{\partial \bar{y}^k(t)} + \alpha\left(\bar{y}^k(t) - \bar{y}^k(t-1)\right) \tag{20}$$

for parameter $\bar{y}^k$ and similar forms for parameters $\bar{x}_i^k$ and σ_i^k, where $\eta \in (0, 1)$ is the learning coefficient and $\alpha \in (0, 1)$ is the momentum coefficient. Calculating the partial derivatives in these recursions, we can get the learning procedures for neuro-fuzzy system with singleton fuzzifier, similar as in [Wang 1994; Rutkowska, Pilinski, Rutkowski 1997], and for neuro-fuzzy system with non-singleton fuzzifier, analogous as in [Rutkowska 1997].

Now we present the learning procedures with momentum terms for neuro-fuzzy system with non-singleton defuzzifier, depicted in Fig. 2. The parameters $\bar{y}^k$, $\bar{x}_i^k$, and $\bar{\sigma}_i^k$ are modified according to these procedures:

$$\bar{y}^k(t+1) = \bar{y}^k(t) - \eta(\bar{y} - d)\left.\frac{\prod_{p=1}^{n}\exp\left[-\left(\frac{\bar{x}_p - \bar{x}_p^k}{\bar{\sigma}_p^k}\right)^2\right]}{\sum_{j=1}^{N}\prod_{p=1}^{n}\exp\left[-\left(\frac{\bar{x}_p - \bar{x}_p^k}{\bar{\sigma}_p^k}\right)^2\right]}\right|_{(t)} + \alpha\left(\bar{y}^k(t) - \bar{y}^k(t-1)\right) \tag{21}$$

$$\bar{x}_i^k(t+1) = \bar{x}_i^k(t) - 2\eta(\bar{y} - d)(\bar{y}^k - \bar{y})(\bar{x}_i - \bar{x}_i^k)\frac{1}{\left(\bar{\sigma}_i^k\right)^2}\left.\frac{\prod_{p=1}^{n}\exp\left[-\left(\frac{\bar{x}_p - \bar{x}_p^k}{\bar{\sigma}_p^k}\right)^2\right]}{\sum_{j=1}^{N}\prod_{p=1}^{n}\exp\left[-\left(\frac{\bar{x}_p - \bar{x}_p^k}{\bar{\sigma}_p^k}\right)^2\right]}\right|_{(t)} + \alpha\left(\bar{x}_i^k(t) - \bar{x}_i^k(t-1)\right) \tag{22}$$

$$\bar{\sigma}_i^k(t+1) = \bar{\sigma}_i^k(t) - 2\eta(\bar{y} - d)(\bar{y}^k - \bar{y})(\bar{x}_i - \bar{x}_i^k)^2\frac{1}{\left(\bar{\sigma}_i^k\right)^3}\left.\frac{\prod_{p=1}^{n}\exp\left[-\left(\frac{\bar{x}_p - \bar{x}_p^k}{\bar{\sigma}_p^k}\right)^2\right]}{\sum_{j=1}^{N}\prod_{p=1}^{n}\exp\left[-\left(\frac{\bar{x}_p - \bar{x}_p^k}{\bar{\sigma}_p^k}\right)^2\right]}\right|_{(t)} + \alpha\left(\bar{\sigma}_i^k(t) - \bar{\sigma}_i^k(t-1)\right) \tag{23}$$

For the RBF neuro-fuzzy system with non-singleton fuzzifier recursions (21) - (23) take simpler form since $\bar{\sigma}_i^k = \bar{\sigma}^k$.

A drawback of gradient algorithms is that they can get ‘trapped’ in local optimum when the starting point is not chosen well. In order to avoid this drawback genetic algorithm is proposed for finding the starting point for the gradient algorithm, so the learning procedure is the hybrid one. First we use genetic algorithm and then gradient procedures (21) - (23). Chromosomes of the genetic algorithm represent the parameters of Gaussian functions: $\bar{y}^k$, $\bar{x}_i^k$ and $\bar{\sigma}_i^k$. Fitness function is defined using the learning sequence and the error (19). The best chromosom (evaluated by the fitness function) is chosen after some generations of crossovers and mutations. Then the gradient algorithm, starting from that point, quickly leads to global optimum. Thus we have neuro-fuzzy-genetic system [Rutkowska 1997]. Genetic algorithm can also be used in this system for fuzzy rules generating [Rutkowska 1998c].

Another hybrid learning approach is possible for optimizing the RBF neuro-fuzzy system with singleton and non-singleton fuzzifier. This is a two-stage procedure based on a clustering algorithm at first stage and the gradient algorithm at second stage. This approach has been applied to singleton neuro-fuzzy system in [Rutkowska, Starczewski 1999].

VI. CONCLUSIONS

In this paper RBF neuro-fuzzy system with non-singleton fuzzifier is presented. This system is applicable when the input data are corrupted by noise. It is easily seen that the neuro-fuzzy system with non-singleton fuzzifier reduces to the neuro-fuzzy system based on the singleton fuzzifier when the width parameter of fuzzy non-singleton membership function $s = 0$. In this case the width parameters of Gaussian radial basis functions reduce to the width parameters of Gaussian membership functions; see equation (15) or (18) and compare expressions (1) and (17).

Simulation studies have shown that for precise training data, with no uncertainty and without noise, the smaller is a value of the width parameter s of fuzzy non-singleton membership function, what means that the non-singleton is closer to singleton, the better is the performance of neuro-fuzzy system with non-singleton fuzzifier, [Rutkowska 1998b].

REFERENCES

Cybenko, G., 1989, ”Approximation by superpositions of a sigmoidal functions”, Mathematics of Control, Signals, and Systems, Vol. 2, pp. 303-314.

Kosko, B.,1992, ”Fuzzy systems as universal approximators”, Proc. IEEE Int. Conf. Fuzzy Systems, pp. 1153-1162.

Jang, J.-S.R.; Sun, C.-T., 1993, ”Functional equivalence between radial basis function networks and fuzzy inference systems”, IEEE Trans. on Neural Networks, Vol. 4, no.1, pp. 156-159.

Moody, J.; Darken, C., 1989, ”Fast learning in networks of locally-tuned processing units”, Neural Computation, Vol.1, no.2, pp. 281-294.

Mouzouris, G. C.; Mendel, J. M., 1997, ”Non-singleton fuzzy logic system: theory and applications”, IEEE Transactions on Fuzzy Systems, Vol. 5, no. 1, pp. 56-71.

Poggio, T.; Girosi, F., 1990, ”Networks for approximation and learning”, Proc. of the IEEE, Vol. 78, no. 9, pp. 1481-1497.

Rutkowska, D.; Pilinski, M.; Rutkowski, L., 1997, ”Neural Networks, Genetic Algorithms and Fuzzy Systems”, PWN, Warsaw (in Polish).

Rutkowska, D., 1997, ”Intelligent Computational Systems”, Academic Publishing House, Warsaw, (in Polish).

Rutkowska, D. 1998a., ”Fuzzy-neural-GA systems with different defuzzification methods”, Proc. of EUFIT’98, Aachen, pp. 713-717.

Rutkowska, D. 1998b., ”Non-singleton fuzzy models”, Journal of Applied Mathematics and Computer Science, Vol.8, no.2, pp. 435-444.

Rutkowska, D., 1998c., ”On generating fuzzy rules by an evolutionary approach”, Cybernetics and Systems, Vol.29, no. 4, pp. 391-407.

Rutkowska, D.; Nowicki R.; Rutkowski L., 1999, ”Singleton and non-singleton fuzzy systems with nonparametric defuzzification”, In: Computational Intelligence and Applications, Szczepaniak P.S. (ed.). Springer-Verlag, pp.292-301.

Rutkowska, D.; Starczewski, A., 1999, ”Fuzzy inference neural networks and their application to medical diagnosis”, In: Fuzzy Systems in Medicine, Springer-Verlag, to be published.

Rutkowska, D., 1999, ”Neural-fuzzy-genetic systems with RBFN defuzzifier”, In: Computational Intelligence for Modelling, Control and Automation, Mohammadian M. (ed.), IOS Press, pp. 184-189.

Wang, L. X., 1994, ”Adaptive Fuzzy Systems and Control”, PTR Prentice Hall.

Zurada, J., 1994, ”Introduction to Artificial Neural Systems”, West Publishing Company.