Appl. Math. and Comp. Sci., 1998, Vol.8, No.2, 435–444

NON-SINGLETON FUZZY MODELS

DANUTA RUTKOWSKA*

This paper presents an approach to design non-singleton fuzzy-logic systems. The relation between non-singleton and singleton fuzzy-logic systems is derived. Simulation results illustrate the effect of uncertainty of the input variables on the truck trajectories in the truck backer-upper control problem.

1. Introduction

Over the past few years fuzzy sets and fuzzy logic (see e.g. Driankov *et al.*, 1993; Mendel, 1995; Pedrycz, 1993) have been used in a wide range of problem domains including process control, pattern recognition and classification, management and decision making. Specific applications include TV colour tuning, camcorder focusing, washing-machine automation, automobile transmissions and subway operations. We are also witnessing a rapid development in the area of neural networks (see e.g. Tadeusiewicz, 1993; Korbicz *et al.*, 1994; Żurada, 1994). Fuzzy inference systems are frequently converted into fuzzy neural networks (Rutkowska, 1996, 1997a; Rutkowska *et al.*, 1997) which exhibit advantages of neural networks and fuzzy systems. In particular, fuzzy neural networks combine learning abilities of neural networks and natural language description of fuzzy systems. An excellent survey of such combinations is presented by Linkens and Nyongesa (1996). It is well-known that fuzzy inference systems process crisp data which are mapped into fuzzy sets in the fuzzifier. The most popular is the singleton fuzzifier (Wang, 1994). The non-singleton fuzzifier is applicable when the input signals are corrupted by noise and there is a need to account for uncertainty in the data.

Some heuristic methods to deal with non-singleton fuzzy systems have been cited by Mouzouris and Mendel (1997). The same authors in their paper presented a formal derivation of general non-singleton fuzzy-logic systems by making use of the modified height defuzzifier. The method developed by Mouzouris and Mendel is a sort of a direct approach to handling uncertainty in input data. Some non-direct approaches are discussed by Pedrycz (1995). In this paper, contrary to Mouzouris and Mendel, we investigate non-singleton systems by making use of the centre-of-sums defuzzifier. Moreover, we shift fuzziness from the input into rule antecedents. This operation allows us to observe (see Section 4) a deteriorating effect of fuzzy inputs on the control.

* Technical University of Częstochowa, Department of Computer Engineering, ul. Armii Krajowej 36, 42–200 Częstochowa, Poland, e-mail: drutko@ matinf.pcz.czest.pl.

In Section 2 we present singleton fuzzy-logic systems based on the centre-of-sums defuzzification. In Section 3 we derive an explicit formula describing non-singleton fuzzy systems and their relations with singleton ones. Simulation results shown in Section 4 illustrate the effect of uncertainty of the inputs (after they have been fuzzified using the non-singleton fuzzifier) on the truck trajectories in the truck backer-upper control problem.

2. Singleton Fuzzy-Logic Systems

In this paper, we consider multi-input, single-output fuzzy systems mapping $X \to Y$, where $X \subset \mathbb{R}^n$ and $Y \subset \mathbb{R}$, see Fig. 1.

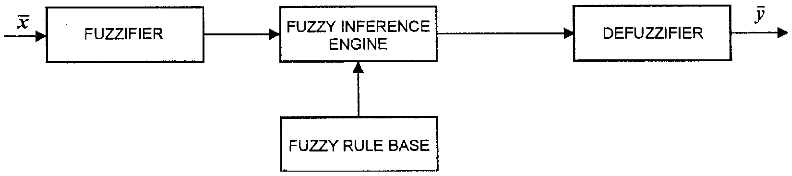

Fig. 1. Fuzzy-logic system.

The fuzzifier performs a mapping from the observed crisp input space $X \subset \mathbb{R}^n$ to the fuzzy sets defined in X. The most commonly used fuzzifier is the singleton fuzzifier which maps $\bar{x} = [\bar{x}_1, \ldots, \bar{x}_n] \in X$ into a fuzzy set $A \subset X$ characterized by the membership function

$$\mu_A(x) = \begin{cases} 1 & \text{if } x = \bar{x} \\ 0 & \text{if } x \neq \bar{x} \end{cases} \tag{1}$$

The fuzzy rule base consists of a set of N rules in the following form:

$$R^{(k)}: \text{ IF } x_1 \text{ is } A_1^k \text{ AND } \ldots \text{ AND } x_n \text{ is } A_n^k \text{ THEN } y \text{ is } B^k \tag{2}$$

where A_i^k and B^k, $i = 1, \ldots, n$ and $k = 1, \ldots, N$, are linguistic terms characterized by fuzzy membership functions $\mu_{A_i^k}(x_i)$ and $\mu_{B^k}(y)$, respectively.

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 G be an arbitrary fuzzy set in X. Then each of N rules (2) determines a fuzzy set $\bar{B}^k \subset Y$ given by the *sup-star* composition

$$\mu_{\bar{B}^k}(y) = \sup_{x \in X} \left\{ \mu_G(x) * \mu_{A_1^k \times \cdots \times A_n^k \to B^k}(x, y) \right\} \tag{3}$$

where $*$ could be any operator in the class of t-norms. It is easily seen that for a crisp input $\bar{x} \in X$, i.e. a singleton fuzzifier (1), formula (3) becomes

$$\mu_{\bar{B}^k}(y) = \mu_{A_1^k \times \cdots \times A_n^k \to B^k}(\bar{x}, y) \tag{4}$$

The defuzzifier performs a mapping from fuzzy sets in Y to a crisp point $\bar{y} \in Y$. The most popular defuzzification methods are:

(i) *centre average* defuzzification

$$\bar{y} = \frac{\sum_{k=1}^{N} \bar{y}^k \mu_{\bar{B}^k}(\bar{y}^k)}{\sum_{k=1}^{N} \mu_{\bar{B}^k}(\bar{y}^k)} \tag{5}$$

where $\bar{y}^k$ is the centre of the fuzzy set B^k such that

$$\mu_{B^k}(\bar{y}^k) = 1 \tag{6}$$

(ii) *centre of area* defuzzification

$$\bar{y} = \frac{\int_Y y \mu_B(y)\, \mathrm{d}y}{\int_Y \mu_B(y)\, \mathrm{d}y} \tag{7}$$

where

$$\mu_B(y) = \max_k \mu_{\bar{B}^k}(y) \tag{8}$$

In this paper, instead of (5) and (7), we apply the *centre-of-sums* defuzzification method given by the formula

$$\bar{y} = \frac{\int_Y y \sum_{k=1}^{N} \mu_{\bar{B}^k}(y)\, \mathrm{d}y}{\int_Y \sum_{k=1}^{N} \mu_{\bar{B}^k}(y)\, \mathrm{d}y} \tag{9}$$

If we assume the product inference rule, then eqn. (4) takes the form

$$\mu_{\bar{B}^k}(y) = \mu_{B^k}(y) \prod_{i=1}^{n} \mu_{A_i^k}(\bar{x}_i) \tag{10}$$

We choose the Gaussian membership functions of fuzzy sets A_i^k and B^k given by

$$\mu_{A_i^k}(x_i) = \exp\left[-\left(\frac{x_i - \bar{x}_i^k}{\sigma_i^k}\right)^2\right] \tag{11}$$

and

$$\mu_{B^k}(y) = \exp\left[-\left(\frac{y - \bar{y}^k}{\sigma^k}\right)^2\right] \tag{12}$$

For the Gaussian functions we have

$$\int_{-\infty}^{\infty} \mu_{B^k}(y)\mathrm{d}y = \sigma^k \sqrt{\pi} \tag{13}$$

$$\int_{-\infty}^{\infty} y \mu_{B^k}(y)\mathrm{d}y = \bar{y}^k \sigma^k \sqrt{\pi} \tag{14}$$

Combining (9)–(12) and using (13) and (14), one gets

$$\bar{y} = \frac{\sum_{k=1}^{N} \bar{y}^k \sigma^k \left(\prod_{i=1}^{n} \exp\left[-\left(\frac{\bar{x}_i - \bar{x}_i^k}{\sigma_i^k} \right)^2 \right] \right)}{\sum_{k=1}^{N} \sigma^k \left(\prod_{i=1}^{n} \exp\left[-\left(\frac{\bar{x}_i - \bar{x}_i^k}{\sigma_i^k} \right)^2 \right] \right)} \tag{15}$$

Formula (15) describes the fuzzy logic system with the centre-of-sums defuzzification, product inference rule and singleton fuzzifier.

3. Non-Singleton Fuzzy-Logic Systems

The *non-singleton* fuzzifier is characterized by the membership function

$$\mu_{G_i}(x_i) = \exp\left[-\left(\frac{x_i - \hat{x}_i}{\sigma_i} \right)^2 \right] \tag{16}$$

For the product inference rule, formula (3) becomes

$$\mu_{\bar{B}^k}(y) = \sup_{x_1,\dots,x_n} \left\{ \mu_{B^k}(y) \prod_{i=1}^{n} \left(\mu_{G_i}(x_i) \mu_{A_i^k}(x_i) \right) \right\} \tag{17}$$

Let

$$\mu_{\bar{B}^k}(y) = \mu_{B^k} \prod_{i=1}^{n} \gamma_i^k \tag{18}$$

where

$$\gamma_i^k = \sup_{x_i} \left\{ \left(\mu_{G_i}(x_i) \mu_{A_i^k}(x_i) \right) \right\} \tag{19}$$

$i = 1, 2, \dots, n,\ k = 1, 2, \dots, N$. We find the value of γ_i^k by maximizing the expression

$$\exp\left[-\left(\frac{x_i - \bar{x}_i^k}{\sigma_i^k} \right)^2 \right] \exp\left[-\left(\frac{x_i - \hat{x}_i}{\sigma_i} \right)^2 \right] \tag{20}$$

which attains its maximum at

$$\tilde{x}_i^k = \frac{\sigma_i^2 \bar{x}_i^k + \left(\sigma_i^k\right)^2 \hat{x}_i}{\sigma_i^2 + \left(\sigma_i^k\right)^2} \tag{21}$$

Consequently,

$$\gamma_i^k = \exp\left[-\left(\frac{\tilde{x}_i^k - \bar{x}_i^k}{\sigma_i^k} \right)^2 \right] \exp\left[-\left(\frac{\tilde{x}_i^k - \hat{x}_i}{\sigma_i} \right)^2 \right] \tag{22}$$

A simple algebra leads to the final result

$$\gamma_i^k = \exp\left[-\left(\frac{\hat{x}_i - \bar{x}_i^k}{\bar{\sigma}_i^k}\right)^2\right] \tag{23}$$

where

$$\bar{\sigma}_i^k = \sqrt{\sigma_i^2 + \left(\sigma_i^k\right)^2} \tag{24}$$

Now we apply the centre-of-sums defuzzification method. Let

$$\lambda_k = \prod_{i=1}^{n} \gamma_i^k \tag{25}$$

Substituting (25) into (18) and using (9), one gets

$$\bar{y} = \frac{\sum_{k=1}^{N} \lambda_k \int_Y y \mu_{B^k}(y) \mathrm{d}y}{\sum_{k=1}^{N} \lambda_k \int_Y \mu_{B^k}(y) \mathrm{d}y} \tag{26}$$

From (13), (14), (25) and (23) it follows that formula (26) takes the form

$$\bar{y} = \frac{\sum_{k=1}^{N} \bar{y}^k \sigma^k \prod_{i=1}^{n} \exp\left[-\left(\frac{\hat{x}_i - \bar{x}_i^k}{\bar{\sigma}_i^k}\right)^2\right]}{\sum_{k=1}^{N} \sigma^k \prod_{i=1}^{n} \exp\left[-\left(\frac{\hat{x}_i - \bar{x}_i^k}{\bar{\sigma}_i^k}\right)^2\right]} \tag{27}$$

Assuming that the centre $\hat{x}_i$ of the fuzzy input set G_i is approximated by the crisp input x_i, we get the description of the non-singleton fuzzy inference system in the form

$$\bar{y} = \frac{\sum_{k=1}^{N} \bar{y}^k \sigma^k \prod_{i=1}^{n} \exp\left[-\left(\frac{x_i - \bar{x}_i^k}{\bar{\sigma}_i^k}\right)^2\right]}{\sum_{k=1}^{N} \sigma^k \prod_{i=1}^{n} \exp\left[-\left(\frac{x_i - \bar{x}_i^k}{\bar{\sigma}_i^k}\right)^2\right]} \tag{28}$$

where $\bar{y}^k$ and σ^k are the parameters of the membership function (12), $\bar{x}_i^k$ is the centre of the membership function (11), $\bar{\sigma}_i^k$ is given by (24) and depends on the parameters σ_i and σ_i^k of the membership functions (16) and (11), respectively. Comparing formulae (15) and (28), we see that fuzziness has been shifted from the input into rule antecedents.

It is easily seen that when the uncertainty of the input is zero, i.e. $\sigma_i^2 = 0$, then the non-singleton fuzzy inference system (28) reduces to the singleton one described by formula (15). It should also be noted that the singleton system (15) can be transformed to the non-singleton one with fuzzifier (16) if the parameters σ_i^k are replaced by $\sqrt{(\sigma_i^k)^2 - \sigma_i^2}$, where $\sigma_i^k > \sigma_i$, $i = 1, \ldots, n$ and $k = 1, \ldots, N$.

4. Simulation Study

The purpose of the simulation is to illustrate the effect of uncertainty of the fuzzy system inputs, after they are fuzzified using the non-singleton fuzzifier, on the truck trajectories in the truck backer-upper control problem. Subsections 4.1 and 4.2 are based on the concept developed by Wang and Mendel (1991) and Wang (1994).

4.1. Problem Formulation

Figure 2 shows the truck and its loading zone. The truck position is exactly determined by three state variables $x \in [-150, 150]$, $y \in [0, 300]$, $\phi \in [-180^o, 180^o]$, where ϕ specifies the angle of the truck with the vertical. Control to the track is the steering angle $\theta \in [-45^o, 45^o]$. The truck moves backward by a fixed unit distance every stage. For simplicity, we assume that there exists enough clearance between the truck and the loading dock so we can ignore the y-position coordinate. The goal is to design a fuzzy inference system making the truck arrive at the loading dock at the angle $\phi(t_f) = 0$ and the final position $x(t_f) = 0$.

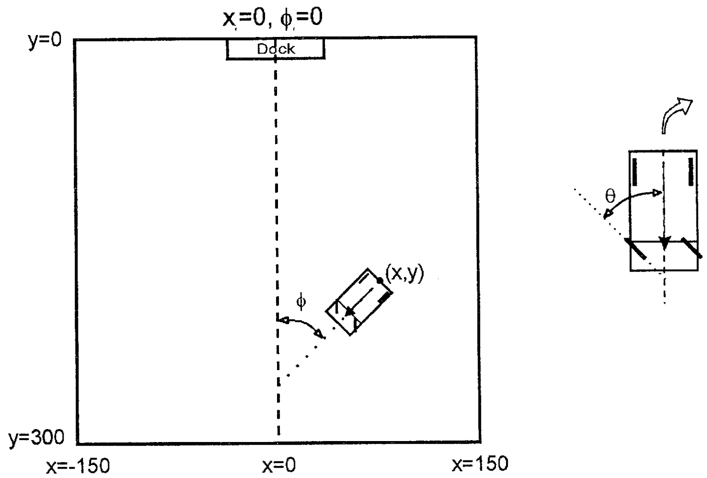

Fig. 2. Truck and loading dock illustration.

4.2. Generation of the Learning Sequences

We describe the movement of the track by making use of the following approximate kinematics derived by Wang and Mendel (1991):

$$x(t+1) = x(t) + \sin\left[\theta(t) + \phi(t)\right] - \sin\left[\theta(t)\right]\cos\left[\phi(t)\right] \tag{29}$$

$$\phi(t+1) = \phi(t) - \arcsin\left[\frac{2\sin[\theta(t)]}{b}\right] \tag{30}$$

where b is the length of the truck. In the simulation $b = 20$.

Based on the above equations we generate 14 learning sequences

$$\big(x(t_0), \phi(t_0), \theta(t_0)\big), \big(x(t_1), \phi(t_1), \theta(t_1)\big), \ldots, \big(x(t_f), \phi(t_f), \theta(t_f)\big)$$

starting from different initial states $(x(t_0), \phi(t_0))$. The steering angle θ at every stage is chosen by a trial-and-error method such that the kinematic equations finally give

$$x(t_f) \approx 0, \quad \phi(t_f) \approx 0$$

The fourteen learning sequences of different lengths form one epoch of length 282.

4.3. Learning Procedures

Based on the learning sequences we train the parameters $\bar{y}^k$, σ^k, $\bar{x}_i^k$ and σ_i^k of the singleton fuzzy logic system (15) such that the error

$$e = \frac{1}{2}\left[\bar{y}(t) - \theta(t)\right]^2 \tag{31}$$

for $t = 0, 1, 2, \ldots$, is minimized. The linguistic variables x_1 and x_2 in the fuzzy rule base (2) correspond to the truck position x and the angle ϕ, respectively. In the simulation $N = 36$, the variables x_1 and x_2 are represented by six different linguistic terms, i.e. some fuzzy sets A_i^k in (2) overlap.

We apply the following training procedure (for details, see Rutkowska, 1997b):

$$\begin{aligned} \bar{y}^k(t+1) &= \bar{y}^k(t) - \eta(\bar{y} - d)\sigma^k \left.\frac{\prod_{i=1}^n \mu_{A_i^k}(\bar{x}_i)}{\sum_{j=1}^N \sigma^j \prod_{i=1}^n \mu_{A_i^j}(\bar{x}_i)}\right|_{(t)} \\ &\quad + \alpha\big(\bar{y}^k(t) - \bar{y}^k(t-1)\big) \end{aligned} \tag{32}$$

$$\begin{aligned} \sigma^k(t+1) &= \sigma^k(t) - \eta(\bar{y} - d)(\bar{y}^k - \bar{y}) \left.\frac{\prod_{i=1}^n \mu_{A_i^k}(\bar{x}_i)}{\sum_{j=1}^N \sigma^j \prod_{i=1}^n \mu_{A_i^j}(\bar{x}_i)}\right|_{(t)} \\ &\quad + \alpha\big(\sigma^k(t) - \sigma^k(t-1)\big) \end{aligned} \tag{33}$$

$$\begin{aligned} \bar{x}_i^k(t+1) &= \bar{x}_i^k(t) - 2\eta(\bar{y} - d)(\bar{y}^k - \bar{y})(\bar{x}_i - \bar{x}_i^k) \\ &\quad \times \frac{\sigma^k}{\left(\sigma_i^k\right)^2} \left.\frac{\prod_{p=1}^n \exp\left[-\left(\dfrac{\bar{x}_p - \bar{x}_p^k}{\sigma_p^k}\right)^2\right]}{\sum_{j=1}^N \sigma^j \prod_{p=1}^n \exp\left[-\left(\dfrac{\bar{x}_p - \bar{x}_p^j}{\sigma_p^j}\right)^2\right]}\right|_{(t)} \\ &\quad + \alpha\big(\bar{x}_i^k(t) - \bar{x}_i^k(t-1)\big) \end{aligned} \tag{34}$$

$$\sigma_i^k(t+1) = \sigma_i^k(t) - 2\eta(\bar{y} - d)(\bar{y}^k - \bar{y})(\bar{x}_i - \bar{x}_i^k)^2$$
$$\times \frac{\sigma^k}{\left(\sigma_i^k\right)^3} \frac{\prod_{p=1}^n \exp\left[-\left(\dfrac{\bar{x}_p - \bar{x}_p^k}{\sigma_p^k}\right)^2\right]}{\sum_{j=1}^N \sigma^j \prod_{p=1}^n \exp\left[-\left(\dfrac{\bar{x}_p - \bar{x}_p^j}{\sigma_p^j}\right)^2\right]}\Bigg|_{(t)}$$
$$+ \alpha\big(\sigma_i^k(t) - \sigma_i^k(t-1)\big) \tag{35}$$

where $\eta \in (0,1)$ and α are the learning and momentum coefficients, respectively. We assumed that $\eta = 0,25$, $\alpha = 0,1$. Algorithms (32)–(35) run for 10 000 iterations.

4.4. Simulation Results

Once the parameters $\bar{y}^k$, σ^k, $\bar{x}_i^k$ and σ_i^k are optimized, we check the performance of the system (15) starting from various initial testing states. The result is shown in Fig. 3. We observe that the singleton fuzzy logic system successfully controls the truck to the desired position. Next we assume some level of uncertainty about the inputs $x_1 = x$ and $x_2 = \phi$. The truck trajectories due to the non-singleton fuzzy logic system (28) are shown in Fig. 4 for $\sigma_1 = \sigma_2 = 20, 40, 60$. We conclude that the effect of uncertainty is minor for $\sigma_1 = \sigma_2 = 20$, becomes visible for $\sigma_1 = \sigma_2 = 40$ and is not acceptable for $\sigma_1 = \sigma_2 = 60$.

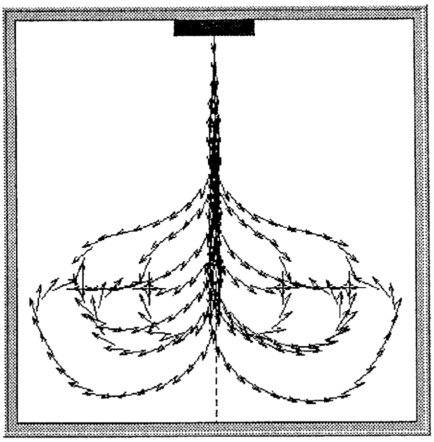

Fig. 3. Truck trajectories using the singleton fuzzy-logic system ($\sigma_1 = \sigma_2 = 0$).

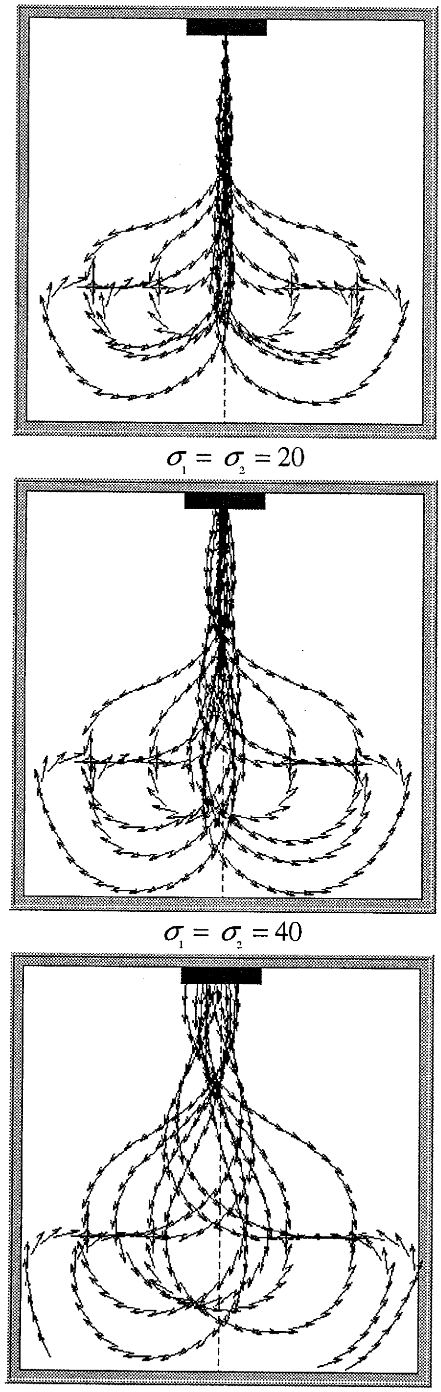

Fig. 4. Truck trajectories using the non-singleton fuzzy-logic system $(\sigma_1 = \sigma_2 = 20, 40, 60)$.

5. Conclusion

In this paper we derived an explicit formula describing non-singleton fuzzy systems with the *centre-of-sums* defuzzification method. By shifting fuzziness from the input into rule antecedents we observed a deteriorating effect of fuzzy inputs on the control of the truck. This effect can be neglected for small σ_1 and σ_2 and becomes visible (or unacceptable) for larger values of these parameters.

Acknowledgment

The author would like to thank the anonymous reviewers for their helpful comments.

References

Driankov D., Hellendoorn H. and Reinfrank M. (1993): *An Introduction to Fuzzy Control.* — Berlin: Springer-Verlag.

Korbicz J., Obuchowicz A. and Uciński D. (1994): *Artificial Neural Networks. Foundations and Applications.* — Warsaw: Akademicka Oficyna Wydawnicza PLJ, (in Polish).

Linkens D.A. and Nyongesa H.O. (1996): *Learning systems in intelligent control: An appraisal of fuzzy, neural and genetic algorithm control applications.* — IEEE Proc. Control Theory Appl., Vol.143, No.4, pp.367–386.

Mendel J.M. (1995): *Fuzzy logic systems for engineering: A tutorial.* — Proc. IEEE, Vol.83, No.3, pp.345–377.

Mouzouris G.C. and Mendel J.M. (1997): *Non-singleton fuzzy logic systems: theory and applications.* — IEEE Trans. Fuzzy Syst., Vol.5, No.1, pp. 56–71.

Pedrycz W. (1993): *Fuzzy Control and Fuzzy Systems.* — New York: J. Wiley.

Pedrycz W. (1995): *Fuzzy Sets Engineering.* — Boca Raton, FL: CRC Press.

Rutkowska D. (1996): *On application of genetic algorithm to fuzzy neural network learning.* — Proc. 2nd Conf. *Neural Networks and Their Applications*, Szczyrk, Poland, pp.420–425.

Rutkowska D. (1997a): *Neural structures of fuzzy systems.* — Proc. IEEE Int. Symp. *Circuits and Systems*, Hong Kong, pp.601–604.

Rutkowska D. (1997b): *Intelligent Computational Systems. Genetic Algorithms and Neural Networks in Fuzzy Systems.* — Warsaw: Akademicka Oficyna Wydawnicza PLJ, (in Polish).

Rutkowska D., Piliński M. and Rutkowski L. (1997): *Neural Networks, Genetic Algorithms and Fuzzy Systems.* — Warsaw: PWN, (in Polish).

Tadeusiewicz R. (1993): *Neural Networks.* — Warsaw: Akademicka Oficyna Wydawnicza RM, (in Polish).

Wang L.X. (1994): *Adaptive Fuzzy Systems and Control.* — Englewood Cliffs: PTR Prentice Hall.

Wang L.X. and Mendel J. M. (1991): *Generating Fuzzy Rules from Numerical Data, with Applications.* — University of Southern California, SIPI-Report No.169.

Żurada J. (1994): *Introduction to Artificial Neural Systems.* — St. Paul: West Publishing Company.

Received: 2 June 1997
Revised: 26 September 1997