

A New Interpretability Criteria for Neuro-Fuzzy Systems for Nonlinear Classification

Krystian Łapa^{1(✉)}, Krzysztof Cpalka¹, and Alexander I. Galushkin²

¹ Institute of Computational Intelligence, Częstochowa University of Technology,
Częstochowa Poland

{krystian.lapa,krzysztof.cpalka}@iisi.pcz.pl

² Moscow Institute of Physics and Technology, Moscow Oblast, Russia
neurocomputer@yandex.ru

Abstract. In this paper a new approach for construction of neuro-fuzzy systems for nonlinear classification is introduced. In particular, we concentrate on the flexible neuro-fuzzy systems which allow us to extend notation of rules with weights of fuzzy sets. The proposed approach uses possibilities of hybrid evolutionary algorithm and interpretability criteria of expert knowledge. These criteria include not only complexity of the system, but also semantics of the rules. The approach presented in our paper was tested on typical nonlinear classification simulation problems.

1 Introduction

The process of construction methods for the nonlinear classification is focused mostly on reaching high accuracy. The other important aim is to achieve a good clarity and interpretability of classification rules. It allows to better understand considered problem. These both aims are contradictory, so the balance between accuracy and interpretability of classifier is often investigated in the literature (see e.g. [28,36,37,71]).

A number of methods are used for nonlinear classification, i.e. neuro-fuzzy systems (see e.g. [35,47,48,55,67,90,95,102]). In those systems the knowledge is gathered in a form of rules *if then*. These rules contain linguistic variables and variables corresponding to fuzzy sets and their parameters. Methods for increasing interpretability of neuro-fuzzy system rules take an important place in the literature. This interpretability arises not only from complexity of the system, but also from semantics of the rules (see e.g. [2,30]). In the area of those research is worth to list methods focused on: **(a)** Definition and implementation of new criteria of interpretability of fuzzy rules (see e.g. [1,30,86,87]). **(b)** Appropriate aggregation of those criteria (see e.g. [3,18,31,99]) and using multi-objective methods (see e.g. [57,85,86,100]). **(c)** Use of population-based algorithms to obtain interpretable systems (see e.g. [49,60,65,74]). **(d)** Use of possibilities of non-supervising learning in the field of initialization of fuzzy rules for increasing interpretability (see e.g. [4,41,72,88]). **(e)** Use of possibilities of gradient and evolutionary methods for reduction and scaling of fuzzy rules and fuzzy sets (see e.g. [39,45,46,63]). **(f)** Use of extended structures of neuro-fuzzy systems

in purpose to increase both the accuracy and interpretability of neuro-fuzzy rules (see e.g. [20,21,79], [80]). **(g)** Extending notation of neuro-fuzzy rules for additional possibilities in the field of more precise description of considering problems (see e.g. [42,59,72,73,89]). It is worth to mention that there are many papers that focus on reviewing methods that concentrate on interpretability and interpretability-accuracy trade-off (see e.g. [1,30,86,87]). For example, in [30] authors are focused on splitting types of methods in the context of interpretability into following quadrants of interpretability: **(a)** complexity at the rule base level (it refers to i.e. number of rules, number of inputs used), **(b)** complexity at the fuzzy partition level (it refers to i.e. number of antecedences and consequents), **(c)** semantics at the rule base level (it refers to i.e. level of active rules for individual input sets), **(d)** semantics at the fuzzy partition level (it refer to i.e. equal covering the search space by fuzzy sets, shape and place of the fuzzy sets).

In this paper we propose a new method for increasing the interpretability of neuro-fuzzy systems for nonlinear classification. Neuro-fuzzy systems (see e.g. [29,52]) combine the natural language description of fuzzy systems (see e.g. [6,61]) and the learning properties of neural networks (see e.g. [7,8,9,10,11,12,13,14,15,51,94,98]). In particular we propose: **(a)** a new approach for construction and tuning of neuro-fuzzy systems (including flexible neuro-fuzzy systems, see e.g. [19,22,23,103,108,109]) **(b)** a new interpretability criteria for increasing interpretability of expert knowledge taking into account not only the complexity of the system but also semantics of the rules.

It is worth to note that many computational intelligence methods (see e.g. [5,24,25,26,62,81,82,91,92,93]) are succesfully used in pattern recognition (see e.g. [38,75,76,77,84]), modelling (see e.g. [16,69,83,97,104,105,106,107]) and many other (see e.g. [32,33,34,50]) issues.

This paper is presented as follows: in Section 2 a description of proposed system and its tuning process for nonlinear classification is presented. In Section 3 a new interpretability criteria to increase interpretability for neuro-fuzzy systems are shown. The results of simulations are presented in Section 4, finally the conclusions are described in Section 5.

2 Description of Neuro-Fuzzy System for Nonlinear Classification and Algorithm for Its Tuning

2.1 System Description

We consider multi-input, multi-output neuro-fuzzy system mapping $\mathbf{X} \rightarrow \mathbf{Y}$, where $\mathbf{X} \subset \mathbf{R}^n$ and $\mathbf{Y} \subset \mathbf{R}^m$. The flexible fuzzy rule base consists of a collection of N fuzzy if-then rules in the form

$$R^k : \left[\left(\begin{array}{l} \text{IF } (\bar{x}_1 \text{ is } A_1^k) \mid w_{k,1}^A \text{ AND } \dots \text{ AND } (\bar{x}_n \text{ is } A_n^k) \mid w_{k,n}^A \\ \text{THEN } (y_1 \text{ is } B_1^k) \mid w_{1,k}^B, \dots, (y_m \text{ is } B_m^k) \mid w_{m,k}^B \end{array} \right) \mid w_k^{\text{rule}} \right], \quad (1)$$

where n is a number of inputs, m is a number of outputs, $\bar{\mathbf{x}} = [\bar{x}_1, \dots, \bar{x}_n] \in \mathbf{X}$, $\mathbf{y} = [y_1, \dots, y_m] \in \mathbf{Y}$, A_1^k, \dots, A_n^k are fuzzy sets characterized by membership functions $\mu_{A_i^k}(x_i)$, $i = 1, \dots, n$, $k = 1, \dots, N$, B_1^k, \dots, B_m^k are fuzzy sets characterized by membership functions $\mu_{B_j^k}(y_j)$, $j = 1, \dots, m$, $k = 1, \dots, N$, $w_{k,i}^A \in [0, 1]$, $i = 1, \dots, n$, $k = 1, \dots, N$, are weights of antecedents, $w_{j,k}^B \in [0, 1]$, $k = 1, \dots, N$, $j = 1, \dots, m$, are weights of consequences, $w_k^{\text{rule}} \in [0, 1]$, $k = 1, \dots, N$, are weights of rules. In logical approach output signal \bar{y}_j , $j = 1, \dots, m$, of the neuro-fuzzy system can be described by the formula

$$\bar{y}_j = \frac{\sum_{r=1}^R \bar{y}_{j,r}^{\text{def}} \cdot T^* \left\{ S^* \left\{ 1 - \prod_{i=1}^n \left\{ \mu_{A_i^k}(\bar{x}_i); w_{k,i}^A \right\}, \mu_{B_j^k}(\bar{y}_{j,r}^{\text{def}}); 1, w_{j,k}^B \right\}; w_k^{\text{rule}} \right\}}{\sum_{r=1}^R T^* \left\{ S^* \left\{ 1 - \prod_{i=1}^n \left\{ \mu_{A_i^k}(\bar{x}_i); w_{k,i}^A \right\}, \mu_{B_j^k}(\bar{y}_{j,r}^{\text{def}}); 1, w_{j,k}^B \right\}; w_k^{\text{rule}} \right\}}, \quad (2)$$

where $T^* \{\cdot\}$ and $S^* \{\cdot\}$ are weighted triangular norms (see e.g. [20,21,78], [79,80]), $\bar{y}_{j,r}^{\text{def}}$, $j = 1, \dots, m$, $r = 1, \dots, R$, are discretization points, R is a number of discretization points. For more details see our previous papers, e.g. [20,21,78].

2.2 Description of tuning algorithm

For selection of structure and parameters of system (2) we propose an evolutionary algorithm (see e.g. [27,54,53,66,68,96,101]). In the process of evolution (evolution of parameters) all parameters of the neuro-fuzzy system (2) will be found. Moreover, in the process of evolution (evolution of the structure) we will find the number of inputs n , the number of rules N , the number of antecedents and consequents (number of fuzzy sets) and the number of discretization points R . The algorithm is based on the Pittsburgh approach ([58,78]), on the evolutionary strategy (μ, λ) for selecting parameters of system (2), on the classical genetic algorithm for choosing structure of system (2) and on the bees algorithm for fixing parameters of reduced systems (2).

The evolutionary strategy (μ, λ) starts with a random generation of the initial parents population \mathbf{P} containing μ individuals. Next, a temporary population \mathbf{T} is created by means of reproduction, whose population contains λ individuals, while $\lambda \geq \mu$. Reproduction consists in a multiple random selection of λ individuals out of the population \mathbf{P} (multiple sampling) and placing the selected ones in temporary population \mathbf{T} . Individuals of the population \mathbf{T} undergo crossover and mutation operations as a result of which an offspring population \mathbf{O} is created, which also has size λ . The purpose of the repair procedure of the population \mathbf{O} is to correct the parameters if they reach inadmissible values. The new population \mathbf{P} containing μ individuals is selected only out of the best λ individuals of the population \mathbf{O} . The bees algorithm mimics the food foraging behaviour of honey bee colonies and it is used to tuning parameters of system (2). The aim of using this algorithm is to tune the parameters of the system with recently reduced structure and to repair damaged accuracy. The behaviour of the bees

can be described as follows: **(a)** For every μ population chromosomes of \mathbf{P} and μ chromosomes are generated (scout bees), **(b)** For every chromosome of \mathbf{B} a search territory area is calculated (as an area of solution explorations coded in the population \mathbf{P}). For every iteration of the algorithm the area of exploration is decreased, **(c)** After this modification, chromosomes from \mathbf{B} are repaired and evaluated (analogically to evolutionary strategy (μ, λ)), **(d)** In the last step one solution (with best fitness function value) is picked from each group of scout bees and moved into the population \mathbf{P} . More details about bees algorithm can be found in [64].

Coding of Parameters and Structure. The parameters of system (2) are coded in the following chromosome (Pittsburgh approach)

$$\mathbf{X}_{ch}^{\text{par}} = \left\{ \begin{array}{l} \bar{x}_{1,1}^A, \sigma_{1,1}^A, \dots, \bar{x}_{n,1}^A, \sigma_{n,1}^A, \dots \\ \bar{x}_{1,Nmax}^A, \sigma_{1,Nmax}^A, \dots, \bar{x}_{n,Nmax}^A, \sigma_{n,Nmax}^A, \\ \bar{y}_{1,1}^B, \sigma_{1,1}^B, \dots, \bar{y}_{m,1}^B, \sigma_{m,1}^B, \dots \\ \bar{y}_{1,Nmax}^B, \sigma_{1,Nmax}^B, \dots, \bar{y}_{m,Nmax}^B, \sigma_{m,Nmax}^B, \\ w_{1,1}^A, \dots, w_{1,n}^A, \dots, w_{Nmax,1}^A, \dots, w_{Nmax,n}^A, \\ w_{1,1}^B, \dots, w_{m,1}^B, \dots, w_{1,Nmax}^B, \dots, w_{m,Nmax}^B, \\ w_1^{\text{rule}}, \dots, w_{Nmax}^{\text{rule}}, \\ \bar{y}_{1,1}^{\text{def}}, \dots, \bar{y}_{1,Rmax}^{\text{def}}, \dots, \bar{y}_{m,1}^{\text{def}}, \dots, \bar{y}_{m,Rmax}^{\text{def}} \end{array} \right\} = \left\{ X_{ch,1}^{\text{par}}, \dots, X_{ch,L}^{\text{par}} \right\}, \quad (3)$$

where $L = Nmax \cdot (3 \cdot n + 3 \cdot m + 1) + Rmax \cdot m$ is the length of the chromosome $\mathbf{X}_{ch}^{\text{par}}$, $ch = 1, \dots, \mu$ for the parent population or $ch = 1, \dots, \lambda$ for the temporary population, $\{\bar{x}_{i,k}^A, \sigma_{i,k}^A\}$, $i = 1, \dots, n$, $k = 1, \dots, N$, are parameters of Gaussian membership functions $\mu_{A_i^k}(x_i)$ of the input fuzzy sets A_1^k, \dots, A_n^k (were used in our simulations), $\{\bar{y}_{j,k}^B, \sigma_{j,k}^B\}$, $k = 1, \dots, N$, $j = 1, \dots, m$, are parameters of Gaussian membership functions $\mu_{B_j^k}(y_j)$ of the output fuzzy sets B_1^k, \dots, B_m^k , $Nmax$ is the maximum number of rules, $Rmax$ is the maximum number of discretization points. The maximum number of rules $Nmax$ should be selected individually to the problem from the range $[1, Nmax]$. Analogously, the maximum number of discretization points $Rmax$ should also be selected individually to the problem from the range $[1, Rmax]$ ([20,21,78]). The purpose of the algorithm is also to select the number of antecedents (from the range $[1, n]$) and consequents (from the range $[1, m]$) within each rule from rule base. The reduction of the system is done using additional chromosome $\mathbf{X}_{ch}^{\text{red}}$. Its genes take binary values and indicate which rules, antecedents, consequents, inputs, and discretization points are selected. The chromosome $\mathbf{X}_{ch}^{\text{red}}$ is given by

$$\mathbf{X}_{ch}^{\text{red}} = \left\{ \begin{array}{l} x_1, \dots, x_n, \\ A_1^1, \dots, A_n^1, \dots, A_1^{Nmax}, \dots, A_n^{Nmax}, \\ B_1^1, \dots, B_m^1, \dots, B_1^{Nmax}, \dots, B_m^{Nmax}, \\ \text{rule}_1, \dots, \text{rule}_{Nmax}, \\ \bar{y}_{1,1}^{\text{def}}, \dots, \bar{y}_{1,Rmax}^{\text{def}}, \dots, \bar{y}_{m,1}^{\text{def}}, \dots, \bar{y}_{m,Rmax}^{\text{def}} \end{array} \right\} = \left\{ X_{ch,1}^{\text{red}}, \dots, X_{ch,L^{\text{red}}}^{\text{red}} \right\}, \quad (4)$$

where $L^{\text{red}} = Nmax \cdot (n + m + 1) + n + Rmax \cdot m$ is the length of the chromosome $\mathbf{X}_{ch}^{\text{red}}$. Its genes indicate which rules (rule $_k$, $k = 1, \dots, Nmax$), antecedents (A_i^k , $i = 1, \dots, n$, $k = 1, \dots, Nmax$), consequents (B_j^k , $j = 1, \dots, m$, $k = 1, \dots, Nmax$), inputs (\bar{x}_i , $i = 1, \dots, n$), and discretization points (\bar{y}^r , $r = 1, \dots, Rmax$) are taken to the system. We can easily notice that the number of inputs used in the system and encoded in the chromosome ch can be determined as follows

$$n_{ch} = \sum_{i=1}^n \mathbf{X}_{ch}^{\text{red}} \{x_i\}, \quad (5)$$

where $\mathbf{X}_{ch}^{\text{red}} \{x_i\}$ means gene of the chromosome $\mathbf{X}_{ch}^{\text{red}}$ associated with the input x_i (as previously mentioned, if the value of the gene is 1, the associated input is taken into account during work of the system). The number of rules (N_{ch}) used in the system and encoded in the chromosome ch may be determined analogously. Implementation of the strategy (μ, λ) uses an additional chromosome

$$\sigma_{ch}^{\text{par}} = \left(\sigma_{ch,1}^{\text{par}}, \dots, \sigma_{ch,L}^{\text{par}} \right). \quad (6)$$

This allows the implementation of the mechanism of self-adaptive range of mutation (see e.g. [78]). At the beginning of the operation of evolutionary strategy the range is large, while during the convergence its gradual reduction is observed. This results in a smooth transition from exploration (occurring at the beginning of the algorithm) to exploitation of the promising areas.

Evolution of Parameters and Structure. This hybrid population-based method allows for tuning both structure and parameters of system (2) with various interpretability criteria. It is worth to mention that: **(a)** An evolutionary strategy (μ, λ) is used for tuning the parameters of system (2). It processes chromosomes $\mathbf{X}_{ch}^{\text{par}}$ and σ_{ch}^{par} from the population **P**, **T** and **O**. The details about crossover and mutation operators from this strategy can be found in [78]. **(b)** For the structure evolution of system (2), a classic genetic algorithm is chosen. It processes chromosomes $\mathbf{X}_{ch}^{\text{red}}$ from the population **P**, **T** and **O**. The details about crossover and mutation operators from this strategy can be found in [58]. It is important to mention that genetic algorithm works together with evolutionary strategy (μ, λ) and it allows to reduce any element of the system structure, such as antecedences, consequents, inputs, rules and discretization points. **(c)** A bees algorithm is additionally used for tuning parameters of the system (2). It processes chromosomes $\mathbf{X}_{ch}^{\text{par}}$ from the population **B**. The purpose of the use of the bees algorithm is to search neighbourhood around chromosomes from population **B** (chromosomes with reduced structure of the system) and replace them with fitter solutions. The details about bees algorithm can be found in [64]. **(d)** The important mechanism of our method is an evaluation process of the chromosomes from populations **P**, **T**, **O** and **B** described in next subsection. It takes into account an accuracy-interpretability trade-off and allows to obtain a balanced dependent from weights of the fitness function components solutions (see e.g. [30]).

Chromosome Population Evaluation. Each individual \mathbf{X}_{ch} of the parental and temporary populations is represented by a sequence of chromosomes $\langle \mathbf{X}_{ch}^{par}, \sigma_{ch}^{par}, \mathbf{X}_{ch}^{red} \rangle$, given by formulas (3), (4) and (6). Genes of two first chromosomes take real values, whereas the genes of the last chromosome take integer values from the set $\{0, 1\}$. The system aims to minimize the following fitness function

$$ff(\mathbf{X}_{ch}) = T^* \left\{ \begin{array}{c} ffaccuracy(\mathbf{X}_{ch}), ffinterpretability(\mathbf{X}_{ch}); \\ w_{ffaccuracy}, w_{ffinterpretability} \end{array} \right\}, \quad (7)$$

where $T^* \{ \cdot \}$ is the algebraic weighted t-norm (see e.g. [20,21,78]), $w_{ffaccuracy} \in (0, 1]$ is a weight of the component $ffaccuracy(\mathbf{X}_{ch})$ and $w_{ffinterpretability}$ is a weight of the component $ffinterpretability(\mathbf{X}_{ch})$. The component $ffaccuracy(\mathbf{X}_{ch})$ determines the accuracy of the system (2) i.e. average normalized percentage classification error for all data from learning sequence. A purpose of normalization of the component $ffaccuracy(\mathbf{X}_{ch})$ is to ensure an influence on every component of the function (7). The component $ffinterpretability(\mathbf{X}_{ch})$ determines complexity-based (component $ffint_A(\mathbf{X}_{ch})$) and semantic-based (components $ffint_B(\mathbf{X}_{ch})$ - $ffint_G(\mathbf{X}_{ch})$) interpretability of the system (2) encoded in the tested chromosome

$$T^* \left\{ \begin{array}{c} ffinterpretability(\mathbf{X}_{ch}) = \\ ffint_A(\mathbf{X}_{ch}), ffint_B(\mathbf{X}_{ch}), ffint_C(\mathbf{X}_{ch}), ffint_D(\mathbf{X}_{ch}), \\ ffint_E(\mathbf{X}_{ch}), ffint_F(\mathbf{X}_{ch}), ffint_G(\mathbf{X}_{ch}); \\ w_{ffintA}, w_{ffintB}, w_{ffintC}, w_{ffintD}, w_{ffintE}, w_{ffintF}, w_{ffintG} \end{array} \right\}, \quad (8)$$

where $w_{ffintA} \in (0, 1]$ denotes weight of the component $ffint_A(\mathbf{X}_{ch})$, etc. The individual components of the formula (8) are defined in the next section.

3 A New Interpretability Criteria for Neuro-Fuzzy System for Nonlinear Classification

In this section a new interpretability criteria for neuro-fuzzy system for nonlinear classification are described, each criterion is a component of fitness function responsible for interpretability (8) of the system. The criteria are defined as follows:

- (a) The component $ffint_A(\mathbf{X}_{ch})$ determines complexity of the system (2) i.e. a number of reduced elements of the system (rules, input fuzzy sets, output fuzzy sets, inputs, and discretization points) in relation to length of the chromosome \mathbf{X}_{ch}^{red} (it allows to increase complexity-based interpretability)

$$\text{ffint}_A(\mathbf{X}_{ch}) = \frac{\left(\sum_{i=1}^n \mathbf{X}_{ch}^{\text{red}} \{x_i\} \cdot \sum_{k=1}^{Nmax} \mathbf{X}_{ch}^{\text{red}} \{\text{rule}_k\} \cdot \mathbf{X}_{ch}^{\text{red}} \{A_i^k\} + \sum_{j=1}^m \sum_{k=1}^{Nmax} \mathbf{X}_{ch}^{\text{red}} \{\text{rule}_k\} \cdot \mathbf{X}_{ch}^{\text{red}} \{B_j^k\} + \sum_{j=1}^m \sum_{r=1}^{Rmax} \mathbf{X}_{ch}^{\text{red}} \{\bar{y}_{m,r}^{\text{def}}\} \right)}{N_{ch} \cdot (n_{ch} + m) + m \cdot Rmax}, \quad (9)$$

where $\mathbf{X}_{ch}^{\text{red}} \{x_i\}$ means a gene of the chromosome $\mathbf{X}_{ch}^{\text{red}}$ associated with the input x_i , etc.

(b) The component $\text{ffint}_B(\mathbf{X}_{ch})$ minimizes number of rules fired at the same time in the system (2) for fuzzy sets

$$\text{ffint}_B(\mathbf{X}_{ch}) = 1 - \frac{1}{Z} \sum_{z=1}^Z \frac{\left(\max_{k=1, \dots, Nmax} \{ \mathbf{X}_{ch}^{\text{red}} \{\text{rule}_k\} \cdot \tau_k(\bar{\mathbf{x}}_z) \} \right)^2}{\sum_{k=1}^{Nmax} \mathbf{X}_{ch}^{\text{red}} \{\text{rule}_k\} \cdot \tau_k(\bar{\mathbf{x}}_z)}, \quad (10)$$

where $\bar{\mathbf{x}}_z$ is a vector of input signals learning sequence ($z = 1, \dots, Z$), Z is the number of samples of learning sequence and $\tau_k(\bar{\mathbf{x}}_z)$ is the flexible firing strength of the k -th rule described by the formula

$$\tau_k(\bar{\mathbf{x}}) = \mu_{\mathbf{A}^k}(\bar{\mathbf{x}}) = T_{i=1}^n \left\{ \mu_{A_i^k}(\bar{x}_i); w_{k,i}^A \right\}. \quad (11)$$

(c) The component $\text{ffint}_C(\mathbf{X}_{ch})$ maximizes a fitness of training data to input fuzzy sets of the system (2) encoded in the tested chromosome

$$\text{ffint}_C(\mathbf{X}_{ch}) = \frac{\sum_{z=1}^Z \sum_{i=1}^n \mathbf{X}_{ch}^{\text{red}} \{x_i\} \cdot \left(1 - \max_{k=1, \dots, Nmax} \left\{ \mathbf{X}_{ch}^{\text{red}} \{\text{rule}_k\} \cdot \mu_{A_i^k}(\bar{x}_{z,i}) \right\} \right)}{Z \cdot n_{ch}}. \quad (12)$$

(d) The component $\text{ffint}_D(\mathbf{X}_{ch})$ reduces overlapping of input and output fuzzy sets of the system (2) encoded in the tested chromosome

$$\frac{1}{4} \cdot \left(\begin{array}{c} \text{ffint}_D(\mathbf{X}_{ch}) = \\ \left(\sum_{i=1}^{n_{ch}} \text{noifs}(i) \left(\left| \text{Cffintc} - \exp \left(- \left(\frac{\mathbf{X}_{ch}^{\text{supp}} \{ \bar{x}_{i,k}^A \} - \mathbf{X}_{ch}^{\text{supp}} \{ \bar{x}_{i,k+1}^A \}}{\mathbf{X}_{ch}^{\text{supp}} \{ \sigma_{i,k}^A \} + \mathbf{X}_{ch}^{\text{supp}} \{ \sigma_{i,k+1}^A \}} \right)^2 \right) \right| + \right. \\ \left. + \left| - \exp \left(- \left(\frac{\mathbf{X}_{ch}^{\text{supp}} \{ \bar{x}_{i,k}^A \} - \mathbf{X}_{ch}^{\text{supp}} \{ \bar{x}_{i,k+1}^A \}}{\mathbf{X}_{ch}^{\text{supp}} \{ \sigma_{i,k}^A \} - \mathbf{X}_{ch}^{\text{supp}} \{ \sigma_{i,k+1}^A \}} \right)^2 \right) \right| \right) \right) + \\ \left(\sum_{j=1}^m \text{noofs}(j) \left(\left| \text{Cffintc} - \exp \left(- \left(\frac{\mathbf{X}_{ch}^{\text{supp}} \{ \bar{y}_{j,k}^B \} - \mathbf{X}_{ch}^{\text{supp}} \{ \bar{y}_{j,k+1}^B \}}{\mathbf{X}_{ch}^{\text{supp}} \{ \sigma_{j,k}^B \} + \mathbf{X}_{ch}^{\text{supp}} \{ \sigma_{j,k+1}^B \}} \right)^2 \right) \right| + \right) \\ \left. + \left| - \exp \left(- \left(\frac{\mathbf{X}_{ch}^{\text{supp}} \{ \bar{y}_{j,k}^B \} - \mathbf{X}_{ch}^{\text{supp}} \{ \bar{y}_{j,k+1}^B \}}{\mathbf{X}_{ch}^{\text{supp}} \{ \sigma_{j,k}^B \} - \mathbf{X}_{ch}^{\text{supp}} \{ \sigma_{j,k+1}^B \}} \right)^2 \right) \right| \right) \right) \right) \end{array} \right) \quad (13)$$

where $\mathbf{X}_{ch}^{\text{supp}}$ stands for additional chromosome with list of non-reduced fuzzy sets

$$\mathbf{X}_{ch}^{\text{supp}} = \left\{ \begin{array}{c} \bar{x}_{1,1}^A, \sigma_{1,1}^A, \bar{x}_{1,2}^A, \sigma_{1,2}^A, \dots, \\ \bar{x}_{n_{ch},1}^A, \sigma_{n_{ch},1}^A, \bar{x}_{n_{ch},2}^A, \sigma_{n_{ch},2}^A, \dots, \\ \bar{y}_{1,1}^B, \sigma_{1,1}^B, \bar{y}_{2,N_{ch}}^B, \sigma_{2,N_{ch}}^B, \dots, \\ \bar{y}_{m,N_{ch}}^B, \sigma_{m,N_{ch}}^B, \bar{y}_{2,N_{ch}}^B, \sigma_{2,N_{ch}}^B, \dots \end{array} \right\} = \{ X_{ch,1}^{\text{supp}}, \dots, X_{ch,L^{\text{supp}}}^{\text{supp}} \}, \quad (14)$$

where $L^{\text{supp}} = 2 \cdot \left(\sum_{i=1}^{n_{ch}} \text{noifs}(i) + \sum_{j=1}^m \text{noofs}(j) \right)$ stands for length of the chromosome $\mathbf{X}_{ch}^{\text{supp}}$. Moreover, a number of fuzzy sets for input i from equation (13) can be reached using function $\text{noifs}(i)$ defined as follows

$$\text{noifs}(i) = \sum_{k=1}^{N_{ch}} \mathbf{X}_{ch}^{\text{red}} \{ \text{rule}_k \} \cdot \mathbf{X}_{ch}^{\text{red}} \{ A_i^k \}. \quad (15)$$

A number of fuzzy sets for output j can be calculated analogously. The lists of parameters encoded in the chromosome $\mathbf{X}_{ch}^{\text{supp}}$ does not have specified final elements - their amount depends on a structure of the chromosome $\mathbf{X}_{ch}^{\text{red}}$. It is worth to mention that lists of parameters are sorted by centres of fuzzy sets. Single rows from the $\mathbf{X}_{ch}^{\text{supp}}$ contain parameters connected with specified input and output fuzzy sets. Due to that this approach is different from the one that uses chromosome $\mathbf{X}_{ch}^{\text{red}}$.

(e) The component $\text{ffint}_E(\mathbf{X}_{ch})$ increases the integrity of the shape of the input and output fuzzy sets associated with the inputs and outputs of the system (2) encoded in the tested chromosome

$$\text{ffint}_D(\mathbf{X}_{ch}) = \left(\begin{aligned} & \sum_{i=1}^n \mathbf{X}_{ch}^{\text{red}} \{x_i\} \cdot \sum_{k=1}^{Nmax} \mathbf{X}_{ch}^{\text{red}} \{\text{rule}_{k1}\} \cdot \\ & \cdot \left(1 - \frac{\min(\mathbf{X}_{ch}^{\text{par}} \{\sigma_{i,k1}^A\}, \frac{1}{N_{ch}} \sum_{k=2=1}^{Nmax} (\mathbf{X}_{ch}^{\text{red}} \{\text{rule}_{k2}\} \cdot \mathbf{X}_{ch}^{\text{par}} \{\sigma_{i,k2}^A\}))}{\max(\mathbf{X}_{ch}^{\text{par}} \{\sigma_{i,k1}^A\}, \frac{1}{N_{ch}} \sum_{k=2=1}^{Nmax} (\mathbf{X}_{ch}^{\text{red}} \{\text{rule}_{k2}\} \cdot \mathbf{X}_{ch}^{\text{par}} \{\sigma_{i,k2}^A\}))} \right) + \\ & + \sum_{j=1}^m \left(1 - \frac{\min(\mathbf{X}_{ch}^{\text{par}} \{\sigma_{j,k1}^B\}, \frac{1}{N_{ch}} \sum_{k=2=1}^{Nmax} (\mathbf{X}_{ch}^{\text{red}} \{\text{rule}_{k2}\} \cdot \mathbf{X}_{ch}^{\text{par}} \{\sigma_{j,k2}^B\}))}{\max(\mathbf{X}_{ch}^{\text{par}} \{\sigma_{j,k1}^B\}, \frac{1}{N_{ch}} \sum_{k=2=1}^{Nmax} (\mathbf{X}_{ch}^{\text{red}} \{\text{rule}_{k2}\} \cdot \mathbf{X}_{ch}^{\text{par}} \{\sigma_{j,k2}^B\}))} \right) \end{aligned} \right) \cdot \frac{\sum_{k=1}^{Nmax} \mathbf{X}_{ch}^{\text{red}} \{\text{rule}_{k1}\}}{n_{ch} + m}, \quad (16)$$

where $\mathbf{X}_{ch}^{\text{par}} \{\sigma_{i,k}^A\}$ stands for a gene of the chromosome $\mathbf{X}_{ch}^{\text{par}}$ associated with the parameter $\sigma_{i,k}^A$ (the width of the Gaussian function used in the simulations), $\mathbf{X}_{ch}^{\text{par}} \{\sigma_{j,k}^B\}$ means gene of the chromosome $\mathbf{X}_{ch}^{\text{par}}$ associated with the parameter $\sigma_{j,k}^B$.

(f) The component $\text{ffint}_F(\mathbf{X}_{ch})$ increases complementarity of the input fuzzy sets of system (2) encoded in the tested chromosome

$$\text{ffint}_F(\mathbf{X}_{ch}) = \frac{\sum_{z=1}^Z \sum_{i=1}^n \left(\cdot \max \left(1, \left| 1 - \sum_{k=1}^{Nmax} \mathbf{X}_{ch}^{\text{red}} \{\text{rule}_k\} \cdot \mu_{A_i^*}(\bar{x}_{z,i}) \right| \right) \right)}{Z \cdot n_{ch}}. \quad (17)$$

(g) The component $\text{ffint}_G(\mathbf{X}_{ch})$ increases readability of the antecedents and weights of rules of system (2) encoded in the tested chromosome

$$\text{ffint}_G(\mathbf{X}_{ch}) = 1 - \frac{\left(\sum_{k=1}^{Nmax} \mathbf{X}_{ch}^{\text{red}} \{\text{rule}_k\} \cdot \left(\frac{\sum_{i=1}^n \mathbf{X}_{ch}^{\text{red}} \{x_i\} \cdot \mu_w(w_{i,k}^A)}{n_{ch}} \right) + \sum_{k=1}^{Nmax} \mathbf{X}_{ch}^{\text{red}} \{\text{rule}_k\} \cdot \mu_w(w_k^{\text{rule}}) \right)}{2 \cdot N_{ch}}, \quad (18)$$

where $\mu_w(w_{i,k}^A)$ is a function defining congeries around values 0, 0.5 and 1 (in simulations we assumed that $a = 0.25$, $b = 0.50$ i $c = 0.75$). This function is described as follows

$$\mu_w(x) = \begin{cases} \frac{a-x}{a} & \text{for } x \geq 0 \text{ and } x \leq a \\ \frac{x-a}{b-a} & \text{for } x \geq a \text{ and } x \leq b \\ \frac{c-x}{c-b} & \text{for } x \geq b \text{ and } x \leq c \\ \frac{x-c}{1-c} & \text{for } x \geq c \text{ and } x \leq 1 \end{cases}. \quad (19)$$

Table 1. Values of the weights of the components $f_{accuracy}(\mathbf{X}_{ch})$ and $f_{interpretability}(\mathbf{X}_{ch})$ for variants considered in simulation: Case I-Case V. Weights of remaining criteria was set as follows: $w_{ffintA} = 0.50$, $w_{ffintB} = 0.10$, $w_{ffintC} = 0.10$, $w_{ffintD} = 1.00$, $w_{ffintE} = 1.00$, $w_{ffintF} = 0.10$, $w_{ffintG} = 0.20$.

Name of the weight	Case I	Case II	Case III	Case IV	Case V
$f_{accuracy}(\mathbf{X}_{ch})$	1.00	1.00	1.00	0.75	0.25
$f_{interpretability}(\mathbf{X}_{ch})$	0.25	0.75	1.00	1.00	1.00

Table 2. The accuracy (%) of the neuro-fuzzy classifier (2) for learning phase, testing phase and average value of them both for simulation variants Case I-Case V

Problem	Sequence	Case					Other authors testing results
		I	II	III	IV	V	
wine recognition problem	testing	92.21	91.53	90.59	88.48	87.88	85.00-98.61 [43,70]
	learning	98.95	98.70	98.73	98.31	97.50	
	average	95.58	95.11	94.66	93.39	92.69	
glass identification problem	testing	73.81	71.98	64.29	59.43	50.08	49.99-74.00 [17,43,70]
	learning	76.77	75.19	73.10	71.09	68.86	
	average	75.29	73.58	68.69	65.26	59.47	
Pima Indians diabetes problem	testing	75.39	74.92	73.34	70.86	69.08	45.90-80.00 [17,44]
	learning	79.40	78.61	77.10	74.56	71.63	
	average	77.40	76.77	75.22	72.71	70.35	
iris classification problem	testing	93.33	93.00	91.33	90.83	88.50	81.80-97.84 [17,40]
	learning	98.42	98.59	98.15	98.24	98.24	
	average	95.88	95.80	94.74	94.54	93.37	
Wisconsin breast cancer problem	testing	96.27	96.12	95.90	95.72	95.56	90.00-97.24 [40,70]
	learning	97.97	97.83	97.94	97.42	96.88	
	average	97.12	96.97	96.92	96.57	96.22	

4 Simulation Results

In our simulations we considered five typical problems from field of nonlinear classification ([56]): **(a)** wine recognition problem, **(b)** glass identification problem, **(c)** Pima Indians diabetes problem, **(d)** iris classification problem, **(e)** Wisconsin breast cancer problem. For each problem a 10-fold cross validation was used, and the process was repeated 5 times. Moreover, for each simulation problem a five variants of learning were applied. Each variant had different set of weights of fitness function (7) - see Table 1. These variants can be described as follows: **(a)** Case "high accuracy", where an accuracy component (2) is the most important in the chromosome evaluation (marked further as "Case I"). **(b)** Case "high interpretability", where the interpretability component (2) is the most important in the chromosome evaluation ("Case V"). **(c)** Case "balanced accuracy and interpretability", where both components of the fitness function have the same meaning (equal values of the weights) ("Case III"). **(d)** Two cases between

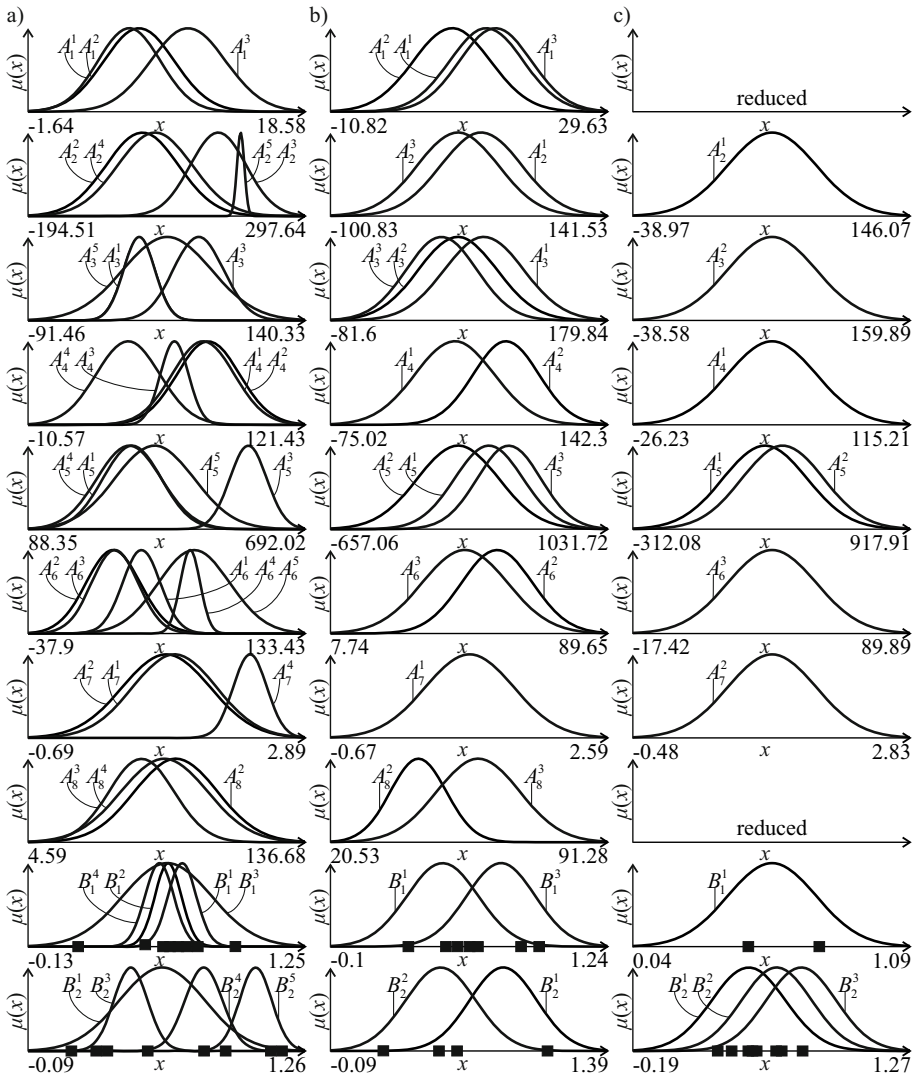


Fig. 1. Example input and output fuzzy sets of the neuro-fuzzy system (2) for the Pima Indians diabetes problem for three various settings of the function (7): a) Case I, b) Case III, c) Case V. The position of the discretization points was marked as black rectangles.

described above cases: Case II (case between Case I and Case III), Case IV (case between Case III and Case V). Obtained results are presented on Fig. 3 and in Table 2. Other relevant simulation properties can be summarized as follows: (a) For the system (2) algebraic triangular norms were used. (b) The following properties of evolutionary algorithm were assumed: the number of chromosomes in the population was set to 100, the algorithm performs 10 000 steps (gener-

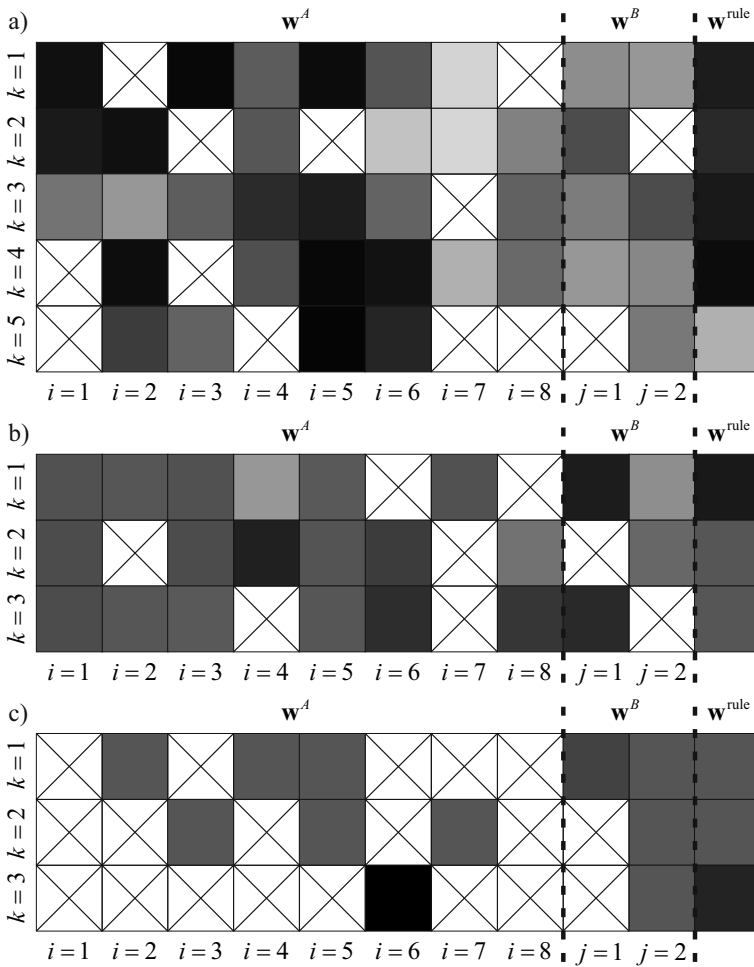


Fig. 2. Example weights representation in the neuro-fuzzy system (2) (dark areas correspond to low values of weights and vice versa) for Pima Indians diabetes problem for different weights configuration of the function (7): a) Case I, b) Case III, c) Case V

ations), the crossover probability was set as $p_c = 0.8$, the mutation probability was set as $p_m = 0.2$, the mutation intensity was set as $\sigma = 0.3$.

The conclusions from simulations can be summarized as follows: (a) The effect of the weights of (7) is significant, which can be seen on Fig. 3. (b) Case III from the simulations allowed to reach good interpretability of fuzzy sets (see Fig. 1 and Fig. 2) and high accuracy (see Table 2). (c) Case I allowed (as expected) to achieve significantly higher accuracy of the system (see Table 2) in comparison with other cases and satisfactory results in comparison with other authors' results.

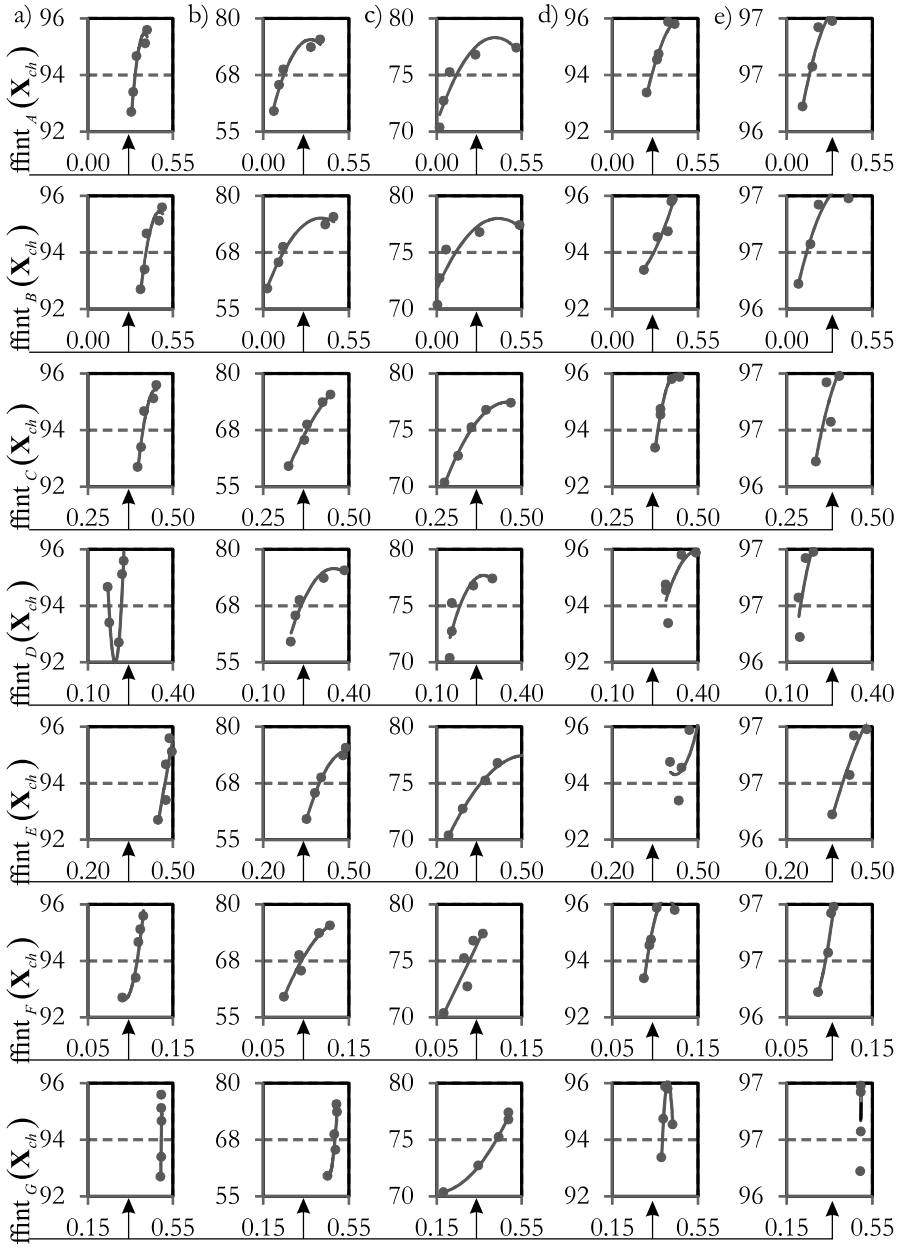


Fig. 3. Dependence between accuracy (%) of neuro-fuzzy classifier (2) (average for learning and testing phase) and values of interpretability components $\text{ffint}_A(\mathbf{X}_{ch})$ - $\text{ffint}_G(\mathbf{X}_{ch})$ for considered simulation variants Case I-Case V for following simulation problems: a) wine recognition problem, b) glass identification problem, c) Pima Indians diabetes problem, d) iris classification problem, e) Wisconsin breast cancer problem

5 Conclusions

In this paper, we proposed a new approach to the construction of interpretable neuro-fuzzy systems for nonlinear classification. It allows to select a structure and parameters of the neuro-fuzzy classifier. Moreover, it allows to take into account different interpretability criteria. The method for selection and configuration of those criteria is very flexible. Due to that fact it allows to obtain the balance between accuracy and interpretability of neuro-fuzzy system classifier in a convenient way. Simulations confirmed the effectiveness of the proposed approach.

Acknowledgment. The project was financed by the National Science Centre (Poland) on the basis of the decision number DEC-2012/05/B/ST7/02138.

References

1. Alonso, J.M.: Embedding HILK in a three-objective evolutionary algorithm with the aim of modeling highly interpretable fuzzy rule-based classifiers, pp. 15–20. European Centre for Soft Computing (2010)
2. Alonso, J.M., Cordon, O., Quirin, A., Magdalena, L.: Analyzing interpretability of fuzzy rule-based systems by means of fuzzy inference-grams. In: 1st World Conference on Soft Computing, pp. 181.1–181.8 (2011)
3. Alonso, J.M., Magdalena, L., Guillaume, S.: HILK: A new methodology for designing highly interpretable linguistic knowledge bases using the fuzzy logic formalism. *International Journal of Intelligent Systems* 23(7), 761–794 (2008)
4. Aziz, D., Ali, M.A.M., Gan, K.B., Saiboon, I.: Initialization of Adaptive Neuro-Fuzzy Inference System Using Fuzzy Clustering in Predicting Primary Triage Category. In: 2012 4th International Conference on Intelligent and Advanced Systems (ICIAS), pp. 170–174. Dept. of Electr., Electron. & Syst. Eng., Univ. Kebangsaan (2012)
5. Bartczuk, Ł., Dziwiński, P., Starczewski, J.T.: New method for generation type-2 fuzzy partition for FDT. In: Rutkowski, L., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2010, Part I. LNCS (LNAI)*, vol. 6113, pp. 275–280. Springer, Heidelberg (2010)
6. Bartczuk, Ł., Dziwiński, P., Starczewski, J.T.: A new method for dealing with unbalanced linguistic term set. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2012, Part I. LNCS*, vol. 7267, pp. 207–212. Springer, Heidelberg (2012)
7. Bilski, J.: Momentum modification of the RLS algorithms. In: Rutkowski, L., Siekmann, J.H., Tadeusiewicz, R., Zadeh, L.A. (eds.) *ICAISC 2004. LNCS (LNAI)*, vol. 3070, pp. 151–157. Springer, Heidelberg (2004)
8. Bilski, J., Rutkowski, L.: Numerically robust learning algorithms for feed forward neural networks. *Advances in Soft Computing*, pp. 149–154 (2003)
9. Bilski, J., Smoląg, J.: Parallel realisation of the recurrent RTRN neural network learning. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2008. LNCS (LNAI)*, vol. 5097, pp. 11–16. Springer, Heidelberg (2008)

10. Bilski, J., Smola, J.: Parallel Realisation of the Recurrent Elman Neural Network Learning. In: Rutkowski, L., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2010, Part II. LNCS (LNAI), vol. 6114, pp. 19–25. Springer, Heidelberg (2010)
11. Bilski, J., Smola, J.: Parallel Realisation of the Recurrent Multi Layer Perceptron Learning. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2012, Part I. LNCS (LNAI), vol. 7267, pp. 12–20. Springer, Heidelberg (2012)
12. Bilski, J., Smola, J.: Parallel approach to learning of the recurrent jordan neural network. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2013, Part I. LNCS (LNAI), vol. 7894, pp. 32–40. Springer, Heidelberg (2013)
13. Bilski, J., Smola, J.: Parallel architectures for learning the RTRN and Elman dynamic neural networks. *IEEE Trans. Parallel and Distributed Systems* PP(99) (2014)
14. Bilski, J., Smola, J., Galushkin, A.I.: The Parallel Approach to the Conjugate Gradient Learning Algorithm for the Feedforward Neural Networks. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2014, Part I. LNCS (LNAI), vol. 8467, pp. 12–21. Springer, Heidelberg (2014)
15. Bilski, J., Litwiński, S., Smola, J.: Parallel realisation of QR algorithm for neural networks learning. In: Rutkowski, L., Siekmann, J.H., Tadeusiewicz, R., Zadeh, L.A. (eds.) ICAISC 2004. LNCS (LNAI), vol. 3070, pp. 158–165. Springer, Heidelberg (2004)
16. Bartczuk, Ł., Przybył, A., Koprinkova-Hristova, P.: New method for nonlinear fuzzy correction modelling of dynamic objects. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2014, Part I. LNCS (LNAI), vol. 8467, pp. 169–180. Springer, Heidelberg (2014)
17. Bostanci, B., Bostanci, E.: An Evaluation of Classification Algorithms Using Mc Nemar's Test. In: Bansal, J.C., Singh, P.K., Deep, K., Pant, M., Nagar, A.K. (eds.) *Proceedings of Seventh International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA 2012)*. AISC, vol. 201, pp. 15–26. Springer, India (2013)
18. Chen, J.L., Hou, Y.L., Xing, Z.Y., Jia, L.M., Tong, Z.Z.: A Multi-objective Genetic-based Method for Design Fuzzy Classification Systems. *IJCSNS International Journal of Computer Science and Network Security* 6(8), 110–117 (2006)
19. Cpałka, K., Łapa, K., Przybył, A., Zalaśiński, M., Rutkowski, L.: A new method for designing neuro-fuzzy systems for nonlinear modelling with interpretability aspects. *Neurocomputing* 135, 203–217 (2014)
20. Cpałka, K.: A New Method for Design and Reduction of Neuro-Fuzzy Classification Systems. *IEEE Transactions on Neural Networks* 20, 701–714 (2009)
21. Cpałka, K.: On evolutionary designing and learning of flexible neuro-fuzzy structures for nonlinear classification. *Nonlinear Analysis Series A: Theory, Methods and Applications* 71, 1659–1672 (2009)
22. Cpałka, K., Zalaśiński, M.: On-line signature verification using vertical signature partitioning. *Expert Systems with Applications* 41, 4170–4180 (2014)
23. Cpałka, K., Zalaśiński, M., Rutkowski, L.: New method for the on-line signature verification based on horizontal partitioning. *Pattern Recognition* 47, 2652–2661 (2014)

24. Dziwiński, P., Bartczuk, Ł., Starczewski, J.T.: Fully controllable ant colony system for text data clustering. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *SIDE 2012 and EC 2012. LNCS*, vol. 7269, pp. 199–205. Springer, Heidelberg (2012)
25. Dziwiński, P., Starczewski, J.T., Bartczuk, Ł.: New linguistic hedges in construction of interval type-2 FLS. In: Rutkowski, L., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2010, Part II. LNCS (LNAI)*, vol. 6114, pp. 445–450. Springer, Heidelberg (2010)
26. Dziwiński, P., Bartczuk, Ł., Przybył, A., Avedyan, E.D.: A New Algorithm for Identification of Significant Operating Points Using Swarm Intelligence. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2014, Part II. LNCS (LNAI)*, vol. 8468, pp. 349–362. Springer, Heidelberg (2014)
27. El-Abd, M.: On the hybridization on the artificial bee colony and particle swarm optimization algorithms. *Journal of Artificial Intelligence and Soft Computing Research* 2(2), 147–155 (2012)
28. Fazzolari, M., Alcalá, R., Herrera, F.: A multi-objective evolutionary method for learning granularities based on fuzzy discretization to improve the accuracy-complexity trade-off of fuzzy rule-based classification systems: D-MOFARC algorithm. *Applied Soft Computing* 24, 470–481 (2014)
29. Gabryel, M., Cpałka, K., Rutkowski, L.: Evolutionary strategies for learning of neuro-fuzzy systems. In: *Proceedings of the I Workshop on Genetic Fuzzy Systems, Granada*, pp. 119–123 (2005)
30. Gacto, M.J., Alcalá, R., Herrera, F.: Interpretability of linguistic fuzzy rule-based systems: An overview of interpretability measures. *Information Sciences* 181, 4340–4360 (2011)
31. Gacto, M.J., Alcalá, R., Herrera, F.: A Multiobjective Evolutionary Algorithm for Tuning Fuzzy Rule Based Systems with Measures for Preserving Interpretability. In: *Proc. of the Joint International Fuzzy Systems Association World Congress and the European Society for Fuzzy Logic and Technology Conference (IFSA/EUSFLAT 2009)* (2009)
32. Gałkowski, T.: Kernel estimation of regression functions in the boundary regions. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2013, Part II. LNCS (LNAI)*, vol. 7895, pp. 158–166. Springer, Heidelberg (2013)
33. Galkowski, T., Pawlak, M.: Nonparametric function fitting in the presence of non-stationary noise. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2014, Part I. LNCS (LNAI)*, vol. 8467, pp. 531–538. Springer, Heidelberg (2014)
34. Galkowski, T., Rutkowski, L.: Nonparametric fitting of multivariate functions. *IEEE Trans. Automatic Control* AC-31(8), 785–787 (1986)
35. Gao, M., Hong, X., Harris, C.J.: Construction of Neurofuzzy Models For Imbalanced Data Classification. *IEEE Transactions on Fuzzy Systems* 22(6), 1472–1488 (2014)
36. Ghandar, A., Michalewicz, Z.: An experimental study of Multi-Objective Evolutionary Algorithms for balancing interpretability and accuracy in fuzzy rule base classifiers for financial prediction. In: *2011 IEEE Symposium on Computational Intelligence for Financial Engineering and Economics*, pp. 1–6 (2011)
37. Gorzałczany, M.B., Rudziński, F.: Accuracy vs. interpretability of fuzzy rule-based classifiers: An evolutionary approach. In: Rutkowski, L., Korytkowski, M.,

- Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) EC 2012 and SIDE 2012. LNCS, vol. 7269, pp. 222–230. Springer, Heidelberg (2012)
38. Grycuk, R., Gabryel, M., Korytkowski, M., Scherer, R., Voloshynovskiy, S.: From single image to list of objects based on edge and blob detection. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2014, Part II. LNCS (LNAI), vol. 8468, pp. 605–615. Springer, Heidelberg (2014)
 39. Guillaume, S., Charnomordic, B.: Generating an Interpretable Family of Fuzzy Partitions From Data. *IEEE Transactions on Fuzzy Systems* 12(3), 324–335 (2004)
 40. Hossen, J., Sayeed, S., Yusof, I., Kalaiarasi, S.M.A.: A Framework of Modified Adaptive Fuzzy Inference Engine (MAFIE) and Its Application. *International Journal of Computer Information Systems and Industrial Management Applications* 5, 662–670 (2013)
 41. Icke, I., Rosenberg, A.: Multi-objective Genetic Programming for Visual Analytics. In: Silva, S., Foster, J.A., Nicolau, M., Machado, P., Giacobini, M. (eds.) EuroGP 2011. LNCS, vol. 6621, pp. 322–334. Springer, Heidelberg (2011)
 42. Ishibuchi, H., Nakashima, T.: Effect of the rule weights in fuzzy rule-based classification systems. *IEEE Trans. Fuzzy Syst.* 9, 506–515 (2001)
 43. Jensen, R., Cornelis, C.: Fuzzy-Rough Nearest Neighbour Classification. In: Peters, J.F., Skowron, A., Chan, C.-C., Grzymala-Busse, J.W., Ziarko, W.P. (eds.) *Transactions on Rough Sets XIII*. LNCS, vol. 6499, pp. 56–72. Springer, Heidelberg (2011)
 44. Kalaiselvi, C., Nasira, G.M.: A Novel Approach for the Diagnosis of Diabetes and Liver Cancer using ANFIS and Improved KNN. *Research Journal of Applied Sciences, Engineering and Technology* 8(2), 243–250 (2014)
 45. Kaur, G.: Similarity measure of different types of fuzzy sets. *School of Mathematics and Computer Applications, Tharpar University* (2010)
 46. Kenesei, T., Abonyi, J.: Interpretable Support Vector Machines in Regression and Classification - Application In Process Engineering. *Hungarian Journal of Industrial Chemistry, Veszprém* 35, 101–108 (2007)
 47. Korytkowski, M., Nowicki, R., Rutkowski, L., Scherer, R.: AdaBoost Ensemble of DCOG Rough-Neuro-Fuzzy Systems. In: Jędrzejowicz, P., Nguyen, N.T., Hoang, K. (eds.) ICCCI 2011, Part I. LNCS, vol. 6922, pp. 62–71. Springer, Heidelberg (2011)
 48. Korytkowski, M., Nowicki, R., Scherer, R.: Neuro-fuzzy rough classifier ensemble. In: Alippi, C., Polycarpou, M., Panayiotou, C., Ellinas, G. (eds.) ICANN 2009, Part I. LNCS, vol. 5768, pp. 817–823. Springer, Heidelberg (2009)
 49. Kumar, G., Rani, P., Devaraj, C., Victoire, D.: Hybrid Ant Bee Algorithm for Fuzzy Expert System Based Sample Classification. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 11(2), 347–360 (2014)
 50. Laskowski, Ł., Laskowska, M.: Functionalization of SBA-15 mesoporous silica by Cu-phosphonate units: Probing of synthesis route. *Journal of Solid State Chemistry* 220, 221–226 (2014)
 51. Laskowski, Ł., Laskowska, M., Jelonkiewicz, J., Boullanger, A.: Spin-glass implementation of a Hopfield neural structure. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2014, Part I. LNCS (LNAI), vol. 8467, pp. 89–96. Springer, Heidelberg (2014)
 52. Łapa, K., Zalasinski, M., Cpałka, K.: A new method for designing and complexity reduction of neuro-fuzzy systems for nonlinear modelling. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.)

- ICAISC 2013, Part I. LNCS (LNAI), vol. 7894, pp. 329–344. Springer, Heidelberg (2013)
53. Lobato, F.S., Steffen Jr., V.: A new multi-objective optimization algorithm based on differential evolution and neighborhood exploring evolution strategy. *Journal of Artificial Intelligence and Soft Computing Research* 1(4), 259–267 (2011)
 54. Lobato, F.S., Steffen Jr., V., Silva Neto, A.J.: Solution of singular optimal control problems using the improved differential evolution algorithm. *Journal of Artificial Intelligence and Soft Computing Research* 1(3), 195–206 (2011)
 55. Luukka, P.: A New Nonlinear Fuzzy Robust PCA Algorithm and Similarity Classifier in Classification of Medical Data Sets. *International Journal of Fuzzy Systems* 13(3), 153–163 (2011)
 56. Machine Learning Repository [Online], <https://archive.ics.uci.edu/ml/datasets.html> (accessed: December 16, 2014)
 57. Marquez, A.A., Marquez, F.A., Peregrin, A.: A multi-objective evolutionary algorithm with an interpretability improvement mechanism for linguistic fuzzy systems with adaptive defuzzification. In: 2010 IEEE International Conference on Fuzzy Systems (FUZZ), pp. 1–7 (2010)
 58. Michalewicz, Z.: *Genetic Algorithms + Data Structures = Evolution Programs*. Springer (1999)
 59. Nauck, D., Kruse, R.: How the Learning of the RuleWeight Affects the Interpretability of the Fuzzy Systems. In: Proceedings of 1998 IEEE International Conference on Fuzzy Systems, vol. 2, pp. 1235–1240 (1998)
 60. Nouri, J.D., Abadeh, S.M., Mohammadi, G.F.: HYEI: A New Hybrid Evolutionary Imperialist Competitive Algorithm for Fuzzy Knowledge Discovery. *Advances in Fuzzy Systems* 2014, 1–9 (2014)
 61. Nowicki, R., Rutkowski, L., Scherer, R.: A method for learning of hierarchical fuzzy systems. In: *Intelligent Technologies - Theory and Applications*, pp. 124–129 (2002)
 62. Pławiak, P., Tadeusiewicz, R.: Approximation of phenol concentration using novel hybrid computational intelligence methods. *Applied Mathematics and Computer Science* 24(1) (2014)
 63. Paiva, R.P., Dourado, A.: Interpretability and learning in neuro-fuzzy systems. *Fuzzy Sets and Systems* 147, 17–38 (2004)
 64. Pham, D.T., Ghanbarzadeh, A., Koc, E., Otri, S., Rahim, S., Zaidi, M.: The Bees Algorithm, A Novel Tool for Complex Optimisation Problems. In: Proceedings of the 2nd International Virtual Conference on Intelligent Production Machines and Systems, pp. 454–459 (2006)
 65. Pouyan, B.M., Yousefi, R., Ostadabbas, S., Nourani, M.: A Hybrid Fuzzy-Firefly Approach for Rule-Based Classification. In: *The Twenty-Seventh International Flairs Conference* (2014)
 66. Prampero, P.S., Attux, R.: Magnetic particle swarm optimization. *Journal of Artificial Intelligence and Soft Computing Research* 2(1), 59–72 (2012)
 67. Przybył, A., Er, M.J.: The idea for the integration of neuro-fuzzy hardware emulators with real-time network. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2014, Part I. LNCS (LNAI)*, vol. 8467, pp. 279–294. Springer, Heidelberg (2014)
 68. Przybył, A., Jelonkiewicz, J.: Genetic algorithm for observer parameters tuning in sensorless induction motor drive. In: *Neural Networks and Soft Computing*, pp. 376–381 (2003)

69. Przybył, A., Smolag, J., Kimla, P.: Distributed control system based on real time ethernet for computer numerical controlled machine tool (in Polish). *Przegląd Elektrotechniczny* 86(2), 342–346 (2010)
70. Qu, Y., Shang, C., Shen, Q., Parthala, M., Wei, W.N.: Kernel-based fuzzy-rough nearest neighbour classification. In: 2011 IEEE International Conference on Fuzzy Systems (FUZZ), pp. 1523–1529 (2011)
71. Rey, M.I., Galende, M., Sainz, G.I., Fuente, M.J.: Checking orthogonal transformations and genetic algorithms for selection of fuzzy rules based on interpretability-accuracy concepts. In: 2011 IEEE International Conference on Fuzzy Systems, pp. 1271–1278 (2011)
72. Riid, A., Rustern, E.: Interpretability improvement of fuzzy systems: Reducing the number of unique singletons in zeroth order Takagi-Sugeno systems. In: 2010 IEEE International Conference on Fuzzy Systems (FUZZ), pp. 1–6 (2010)
73. Riid, A., Rüstern, E.: Interpretability, Interpolation and Rule Weights in Linguistic Fuzzy Modeling. In: Fanelli, A.M., Pedrycz, W., Petrosino, A. (eds.) WILF 2011. LNCS (LNAI), vol. 6857, pp. 91–98. Springer, Heidelberg (2011)
74. Rini, D.P., Shamsuddin, S.M., Yuhaniz, S.S.: Balanced the Trade-offs Problem of ANFIS using Particle Swarm Optimization. *Telkomnika* 11(3), 611–616 (2013)
75. Rutkowski, L.: On Bayes risk consistent pattern-recognition procedures in a quasi-stationary environment. *IEEE Trans. Pattern Analysis and Machine Intelligence* 4(1), 84–87 (1982)
76. Rutkowski, L.: Online Identification of Time-Varying Systems by Nonparametric Techniques. *IEEE Trans. Automatic Control* 27(1), 228–230 (1982)
77. Rutkowski, L.: On nonparametric identification with prediction of time-varying systems. *IEEE Trans. Automatic Control* 29(1), 58–60 (1984)
78. Rutkowski, L.: *Computational Intelligence*. Springer (2008)
79. Rutkowski, L., Cpałka, K.: Flexible structures of neuro-fuzzy systems. In: Sincak, P., Vascak, J. (eds.) *Quo Vadis Computational Intelligence*. STUDEFUZZ, vol. 54, pp. 479–484. Springer, Heidelberg (2000)
80. Rutkowski, L., Cpałka, K.: Compromise approach to neuro-fuzzy systems. In: Sincak, P., Vascak, J., Kvasnicka, V., Pospichal, J. (eds.) *Intelligent Technologies - Theory and Applications*, vol. 76, pp. 85–90. IOS Press (2002)
81. Rutkowski, L., Jaworski, M., Pietruczuk, L., Duda, P.: Decision Trees for Mining Data Streams Based on the Gaussian Approximation. *IEEE Transactions on Knowledge and Data Engineering* 26, 108–119 (2014)
82. Rutkowski, L., Jaworski, M., Pietruczuk, L., Duda, P.: The CART decision tree for mining data streams. *Information Sciences* 266, 1–15 (2014)
83. Rutkowski, L., Przybył, A., Cpałka, K., Er, M.J.: Online speed profile generation for industrial machine tool based on neuro-fuzzy approach. In: Rutkowski, L., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2010, Part II*. LNCS (LNAI), vol. 6114, pp. 645–650. Springer, Heidelberg (2010)
84. Rutkowski, L., Rafajłowicz, E.: On optimal global rate of convergence of some nonparametric identification procedures. *IEEE Trans. Automatic Control* 34(10), 1089–1091 (1989)
85. Sánchez, G., Jiménez, F., Sánchez, J.F., Alcaraz, J.M.: A Multi-objective Neuro-evolutionary Algorithm to Obtain Interpretable Fuzzy Models. In: Meseguer, P., Mandow, L., Gasca, R.M. (eds.) *CAEPIA 2009*. LNCS, vol. 5988, pp. 51–60. Springer, Heidelberg (2010)
86. Shukla, P.K., Tripathi, S.P.: A Review on the Interpretability-Accuracy Trade-Off in Evolutionary Multi-Objective Fuzzy Systems (EMOFS). *Information* 3, 256–277 (2012)

87. Shukla, P.K., Tripathi, S.P.: A new approach for tuning interval type-2 fuzzy knowledge bases using genetic algorithms. *Journal of Uncertainty Analysis and Applications* 2, 4 (2014)
88. Shukla, P.K., Tripathi, S.P.: Handling High Dimensionality and Interpretability-Accuracy Trade-Off Issues in Evolutionary Multiobjective Fuzzy Classifiers. *International Journal of Scientific & Engineering Research* 5(6) (2014)
89. Siminski, K.: Rule Weights in a Neuro-Fuzzy System with a Hierarchical Domain Partition. *Int. J. Appl. Math. Comput. Sci.* 20(2), 337–347 (2010)
90. Sood, A., Aggarwal, S.: Crossroads in Classification: Comparison and Analysis of Fuzzy and Neuro-Fuzzy Techniques. *International Journal of Computer Applications* (0975-8887) 24(2), 13–17 (2011)
91. Starczewski, J., Rutkowski, L.: Connectionist structures of type 2 Fuzzy Inference Systems. In: Wyrzykowski, R., Dongarra, J., Paprzycki, M., Waśniewski, J. (eds.) *PPAM 2001*. LNCS, vol. 2328, pp. 634–642. Springer, Heidelberg (2002)
92. Starczewski, J., Rutkowski, L.: Interval type 2 neuro-fuzzy systems based on interval consequents. *Advances in Soft Computing*, pp. 570–577 (2003)
93. Starczewski, J.T., Bartczuk, Ł., Dziwiński, P., Marvuglia, A.: Learning methods for type-2 FLS based on FCM. In: Rutkowski, L., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2010, Part I*. LNCS (LNAI), vol. 6113, pp. 224–231. Springer, Heidelberg (2010)
94. Szarek, A., Korytkowski, M., Rutkowski, L., Scherer, R., Szyprowski, J.: Application of neural networks in assessing changes around implant after total hip arthroplasty. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2012, Part II*. LNCS (LNAI), vol. 7268, pp. 335–340. Springer, Heidelberg (2012)
95. Szarek, A., Korytkowski, M., Rutkowski, L., Scherer, R., Szyprowski, J.: Forecasting wear of head and acetabulum in hip joint implant. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2012, Part II*. LNCS (LNAI), vol. 7268, pp. 341–346. Springer, Heidelberg (2012)
96. Szczypta, J., Przybył, A., Cpałka, K.: Some aspects of evolutionary designing optimal controllers. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2013, Part II*. LNCS (LNAI), vol. 7895, pp. 91–100. Springer, Heidelberg (2013)
97. Szczypta, J., Przybył, A., Wang, L.: Evolutionary approach with multiple quality criteria for controller design. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2014, Part I*. LNCS (LNAI), vol. 8467, pp. 455–467. Springer, Heidelberg (2014)
98. Tadeusiewicz, R., Chaki, R., Chaki, N.: *Exploring Neural Networks with C#*. CRC Press, Taylor & Francis Group, Boca Raton (2014)
99. Troiano, L., Ranilla, J., Díaz, I.: Interpretability of Fuzzy Association Rules as means of Discovering Threats to Privacy (CMMSE 2010). *International Journal of Computer Mathematics*, 325–333 (2011)
100. Wang, H., Kwong, S., Jin, Y., Wei, W., Man, K.F.: Multi-objective hierarchical genetic algorithm for interpretable fuzzy rule-based knowledge extraction. *Fuzzy Sets and Systems* 149(1), 149–186 (2005)
101. Woźniak, M., Kempa, W.M., Gabryel, M., Nowicki, R.: A finite-buffer queue with single vacation policy-analytical study with evolutionary positioning. *Int. Journal of Applied Mathematics and Computer Science* 24, 887–900 (2014)

102. Yang, Z., Wang, Y., Ouyang, G.: Adaptive Neuro-Fuzzy Inference System for Classification of Background EEG Signals from ESES Patients and Controls. *The Scientific World Journal* 2014, 1–8 (2014)
103. Zalaśiński, M., Cpałka, K.: A new method of on-line signature verification using a flexible fuzzy one-class classifier, pp. 38–53. Academic Publishing House EXIT (2011)
104. Zalaśiński, M., Łapa, K., Cpałka, K.: New Algorithm for Evolutionary Selection of the Dynamic Signature Global Features. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2013, Part II. LNCS (LNAI), vol. 7895, pp. 113–121. Springer, Heidelberg (2013)
105. Zalaśiński, M., Cpałka, K.: Novel algorithm for the on-line signature verification. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2012, Part II. LNCS (LNAI), vol. 7268, pp. 362–367. Springer, Heidelberg (2012)
106. Zalaśiński, M., Cpałka, K.: New approach for the on-line signature verification based on method of horizontal partitioning. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2013, Part II. LNCS (LNAI), vol. 7895, pp. 342–350. Springer, Heidelberg (2013)
107. Zalaśiński, M., Cpałka, K.: Novel Algorithm for the On-Line Signature Verification Using Selected Discretization Points Groups. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2013, Part I. LNCS (LNAI), vol. 7894, pp. 493–502. Springer, Heidelberg (2013)
108. Zalaśiński, M., Cpałka, K., Er, M.J.: New Method for Dynamic Signature Verification Using Hybrid Partitioning. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2014, Part II. LNCS (LNAI), vol. 8468, pp. 216–230. Springer, Heidelberg (2014)
109. Zalaśiński, M., Cpałka, K., Hayashi, Y.: New Method for Dynamic Signature Verification Based on Global Features. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2014, Part II. LNCS (LNAI), vol. 8468, pp. 231–245. Springer, Heidelberg (2014)