2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

The Learning of Neuro-Fuzzy Approximator with Fuzzy Rough Sets in Case of Missing Features

Robert K. Nowicki *Member, IEEE*, Bartosz A. Nowak,
Janusz T. Starczewski *Member, IEEE*, and Krzysztof Cpałka *Member, IEEE*
Institute of Computational Intelligence, Czestochowa University of Technology,
Al. Armii Krajowej 36, 42-200 Czestochowa, Poland
Email: {robert.nowicki; bartosz.nowak; janusz.starczewski; krzysztof.cpalka}@iisi.pcz.pl

***Abstract*—The architecture of neuro-fuzzy systems with fuzzy rough sets originally has been developed to process with imprecise data. In this paper, the adaptation of those systems to the missing features case is presented. However, the main considerations concern with methods of learning which could be applied to such systems for approximation tasks. Various methods for determining values of system parameters have been considered, in particular the gradient learning method. The effectiveness of proposed methods has been confirmed by many simulation experiments, which results have been supplied to this paper.**

I. INTRODUCTION

The missing data problem is a permanent element of any application of real decision systems. It concerns all application fields — industrial control and diagnosis, medical diagnosis, recognition, modelling, prediction [36], [35]. Regardless of applied methodology (neural networks, fuzzy systems, *k*-nn classifiers, svm systems etc.) there are two general methods to process data with missing values:

- marginalisation,
- imputation.

Obviously, the modified methods and hybrid solutions are also available.

Methods that belong to the first group boil down to temporary reduce the dimensionality of consideration space to the features of known values. Therefore, some elements of the system are just turned off. Therefore, sometimes the elimination of all incomplete samples includes also the marginalisation. However, it is eventually accepted only in developing time.

When we would like to use imputation, the unknown values are replaced by estimated ones. The palette of available methods is generally unlimited. The most primitive ones are confined to insertion of random, average or most common values. More sophisticated ones apply EM (Expectation Maximization) or *k* nearest neighbour algorithms, neural networks, fuzzy systems. The promising results are obtained by multiple imputation and interval imputation. If we know the probability density distribution, we can use the Bayesian solution [6], [8], [29], [30]. Then if we know the possibility distribution, we can use fuzzy imputation.

A specific approach to the problem comes from the rough set theory [23]. An object can be classified to a positive region of a class (i.e. the object certainly belongs to the class), to a negative region of the class (i.e. the object certainly not belongs to the class) or to a boundary region of the class (i.e. it is not possible to determine if the object belongs to the class or not). Membership to theese regions depends on the quality of object description. If this description is good enough, the object belongs either to the positive or negative regions. If the description is too weak, then the object belongs to the boundary region. In the rough set theory [23] as well as in the theory of evidence [37], we do not use the individual elements of the consideration but some granules [24]. The granules contain elements which are indistinguishable basing on knowledge that we dispose. Thus, the size and the shape of granules depend on the used (known) knowledge about elements. Hence, the many hybrid approaches apply rough sets together with other methods, e.g. mentioned above in the imputation context.

In this paper, we focus on a hybrid system merging fuzzy sets, rough sets and neural networks. The idea to connect the fuzzy and rough sets comes from Dubois and Prade [4], [5]. They proposed two new types of sets, i.e. rough fuzzy sets and, more general fuzzy rough sets. Specific definitions of fuzzy rough sets were proposed also by Nakamura [15] and Thiele [42]. A general approach to them was presented by Radzikowska and Kerre [27].

The first generalization of the rough set, i.e. the rough fuzzy set, allows to approximate not only a classical set, but also a fuzzy set. It is realized by the same granules as in the case of the rough set. The second generalization of the rough set, i.e. the fuzzy rough set, uses fuzzy granules to approximate either a classical or a fuzzy set.[1]

The rough fuzzy sets are used mainly in classification systems. Sarkar [32] applied rough fuzzy sets as well as "rough fuzzy membership functions" and "ownership functions" studied in [34] to support classification tasks. Nowicki proposed the rough neuro–fuzzy classifiers [16], [18], [20]. The fuzzy rough sets were employed both for classification [33] and approximation [40], [41].

In the paper, we consider the problem of approximation in case of missing data. We assume that exist function f defined in multidimensional input space, i.e $\mathbf{X} = \Re^n$. Therefore, we have $\overline{y} = f(\overline{\mathbf{x}})$, where $\overline{\mathbf{x}} = [\overline{x}_1, \overline{x}_2, \ldots, \overline{x}_n]$ is an input vector and $\overline{x}_i$, $i = 1 \ldots n$ is a value of single feature x_i.

[1] Some authors are using terms "rough fuzzy" and "fuzzy rough" in solutions which do not apply either rough fuzzy set or fuzzy rough set, but just separately fuzzy set and rough set. An example could be a fuzzy classifier containing rules generated using the rough set theory.

978-1-4799-1484-5/14/$31.00 ©2014 IEEE

Unfortunately, values of some input features are unknown. Therefore, we define the two separate sets of features — D is a set of features with known values and G is a set of features with unknown values, moreover $Q = D \cup G$. In consequence, we define two vectors covered known value — $\overline{\mathbf{v}}_D = \left[\overline{v}_{d_1}, \overline{v}_{d_2}, \ldots, \overline{v}_{d_{n_D}}\right]$ — and unknown values — $\overline{\mathbf{v}}_G = \left[\overline{v}_{g_1}, \overline{v}_{g_2}, \ldots, \overline{v}_{g_{n_G}}\right]$ — where $\{d_i\}$, $i = 1 \ldots n_D$ is the series of known feature indexes and $\{g_i\}$,$i = 1 \ldots n_G$ denotes series of unknown feature indexes. In such conditions, the goal is to determine the acceptable value $\overline{y}_D$ based on vector $\overline{\mathbf{x}}_D$. In the estimation theory, the value $\overline{y}_D$ is called $\hat{y}$. In the proposed method, the goal is somehow changed. We expect to obtain not a single, estimated value $\overline{y}_D$ but a possible narrow interval $[\overline{y}_{D*}, \overline{y}_D^*]$ which contains the correct value $\overline{y}$. The width of the interval should depend on the number and importance of missing features. Thus, we expect that in case when $G = \emptyset$ we obtain $\overline{y}_{D*} = \overline{y}_D^*$, and in case when $D = \emptyset$ we obtain $[\overline{y}_{D*}, \overline{y}_D^*] = [\min\{y\}, \max\{y\}]$ or even $[\overline{y}_{D*}, \overline{y}_D^*] = [-\infty, \infty]$. In the further parts of the paper, the output interval will described as $[\overline{y}_*, \overline{y}^*]$ regardless of D set for simplicity.

The paper is organised as follows. The current section introduces the reader into the subject and the goal of presented researches. Section II presents fundamental information and definitions about rough sets, fuzzy sets, fuzzy rough sets, and neuro-fuzzy systems with fuzzy rough sets. Section III contains the adaptation of the neuro-fuzzy approximator to work in cases of missing features instead of imprecise data. Section IV considers learning methods concerned with systems mentioned above. Section V contains results for learning and testing of discussed systems with the use of presented learning methods. Ending Conclusions contain also plans for the future work.

II. Fundamentals

1) Fuzzy Sets: The fuzzy sets have been defined by Zadeh [43], who proposed to generalise the definition of common sets by extending the term of the characteristic function and allowing it to obtain values "between zero and one". Such function is called also a membership function. Using contemporary terminology [31], the fuzzy set is a set of pairs which includes element and its membership to the set, i.e.

$$A = \{x, \mu_A(x)\}, x \in X, \tag{1}$$

where $\mu_A \colon x \longmapsto [0,1]$ is the membership of element x to set A. Therefore, the fuzzy sets expresses the uncertainty in membership of a particular element.

2) Rough Fuzzy Sets: The rough fuzzy set has been defined by Pawlak [22]. It is a pair $\left(\underline{R}A, \overline{R}A\right)$ of fuzzy sets. $\underline{R}A$ is an R-lower approximation and $\overline{R}A$ is an R-upper approximation of fuzzy set $A \subseteq X$. The membership functions of $\underline{R}A$ and $\overline{R}A$ are defined as follows

$$\mu_{\underline{R}A}(\hat{x}) = \inf_{x \in [\hat{x}]_R} \mu_A(x) \ , \tag{2}$$

$$\mu_{\overline{R}A}(\hat{x}) = \sup_{x \in [\hat{x}]_R} \mu_A(x) \ . \tag{3}$$

where $[\hat{x}]_R$ is an equivalence class [25] dependent on the $\widetilde{D}$-indiscernibility relation which is defined by

$$x \widetilde{D} \hat{x} \Leftrightarrow \forall v \in D; f_x(v) = f_{\hat{x}}(v) \ , \tag{4}$$

where $x, \hat{x} \in \mathbf{X}$ and f_x is an information function expressing a value of feature v_i of object x.

3) Fuzzy Rough Sets: As was mentioned above, the first attempt to combine fuzzy and rough sets comes from Dubois and Prade [4], [5]. Formally, if Φ is a fuzzy partitioning of a universe U, fuzzy sets F_i are its partitions, and A is a fuzzy subset of U, i.e., $A \subseteq U$. The fuzzy rough set is defined as a pair $(\Phi_* A, \Phi^* A)$, where set $\Phi_* A$ is a Φ–lower approximation of the fuzzy set A, and set $\Phi^* A$ is its Φ–upper approximation. Accordingly, membership functions of fuzzy sets $\Phi_* A$ and $\Phi^* A$ are defined as follows:

$$\mu_{\Phi^* A}(F_i) = \sup_{x \in U} \min\left(\mu_{F_i}(x), \mu_A(x)\right), \tag{5}$$

$$\mu_{\Phi_* A}(F_i) = \inf_{x \in U} \max\left(1 - \mu_{F_i}(x), \mu_A(x)\right). \tag{6}$$

III. Neuro-Fuzzy Approximator

Continuing the development of hybrid decision systems [28], [17], [19], [20], [21], we have proposed the fuzzy rough network for approximation tasks which have been developed for imprecise input data. The system is an extension of previous rough neuro-fuzzy classifiers, however, based on fuzzy rough sets (Eqs. (5) and (6)) instead of rough fuzzy sets. It, as previous ones, consists of two specific neuro-fuzzy systems and the common knowledge base. The base of knowledge can be represented in the classical form, i.e.

$$\begin{array}{c} R^k \colon \textbf{IF } x_1 \text{ is } A_1^k \text{ AND } x_2 \text{ is } A_2^k \text{ AND} \ldots \\ \ldots \text{AND } x_n \text{ is } A_n^k \textbf{ THEN } y \text{ is } B^k \end{array} , \tag{7}$$

where $A_1^k, \ldots, A_n^k$ is an antecedent fuzzy set and B^k is a consequent fuzzy set used the k-th rule, or as Takagi-Sugeno-Kang type I rules:

$$\begin{array}{c} R^k \colon \textbf{IF } x_1 \text{ is } A_1^k \text{ AND } x_2 \text{ is } A_2^k \text{ AND} \ldots \\ \ldots \text{AND } x_n \text{ is } A_n^k \textbf{ THEN } y = \overline{y}^k \end{array} . \tag{8}$$

In such case, $\overline{y}^k = \sup_{y \in Y} \mu_{B^k}(y)$ and Y is domain of y.

The neuro-fuzzy systems can be developed using either conjunction type or logical type of reasoning [3] [28] and various methods of defuzzification. Here, we consider only centre average defuzzification and the conjunction-type reasoning. However, fuzzification is defined in the context of fuzzy rough sets. We assume that a premise is defined as follows

$$x_1 \text{ is } A_1' \text{ AND } x_2 \text{ is } A_2' \text{ AND} \ldots\ldots \text{AND } x_n \text{ is } A_n', \tag{9}$$

and the sets A_i', $i = 1, \ldots, n$ are not singletons. We treat A_i' as the F_i function in fuzzy rough definitions (Eqs. (5) and (6)). Thus, we obtain the interval $\{A_*, A^*\}$ defined as follow

$$\mu_{A_i^{*k}}(A_i') = \sup_{x_i \in \mathbf{X}_i} T\left(\mu_{A_i'}(x_i), \mu_{A_i^k}(x_i)\right), \tag{10}$$

$$\mu_{A_{i*}^k}(A_i') = \inf_{x_i \in \mathbf{X}_i} S\left(N\left(\mu_{A_i'}(x_i)\right), \mu_{A_i^k}(x_i)\right). \tag{11}$$

Now we can observe that the fuzzy rough set $\left\{A_{i*}^k, A_i^{*k}\right\}$ is formally an interval type-2 fuzzy set, although the notions of these two sets differ in interpretation. As a consequence, the well known Karnik-Mendel type-reduction iterative method

[9], [10] can be employed to defuzzify interval fuzzy conclusions. According to this method, in each t-th iteration, the lower and upper approximations are calculated by:

$$\bar{y}_*(t) = \frac{\sum_{k=1}^{N} \bar{y}^k \cdot \mu_{A_{\mathrm{L}}^k}(\bar{\mathbf{x}}, t)}{\sum_{k=1}^{N} \mu_{A_{\mathrm{L}}^k}(\bar{\mathbf{x}}, t)} \tag{12}$$

and

$$\bar{y}^*(t) = \frac{\sum_{k=1}^{N} \bar{y}^k \cdot \mu_{A_{\mathrm{U}}^k}(\bar{\mathbf{x}}, t)}{\sum_{k=1}^{N} \mu_{A_{\mathrm{U}}^k}(\bar{\mathbf{x}}, t)}\ , \tag{13}$$

where A_{L}^k and A_{U}^k are determined as

$$A_{\mathrm{L}}^k(t) = \begin{cases} A_*^k & \text{if } \bar{y}^k > \bar{y}_*(t-1) \\ A^{*k} & \text{if } \bar{y}^k \leq \bar{y}_*(t-1) \end{cases}, \tag{14}$$

and

$$A_{\mathrm{U}}^k(t) = \begin{cases} A^{*k} & \text{if } \bar{y}^k > \bar{y}^*(t-1) \\ A_*^k & \text{if } \bar{y}^k \leq \bar{y}^*(t-1) \end{cases}, \tag{15}$$

with the initial values

$$\bar{y}^*(0) = \bar{y}_*(0) = \frac{1}{N} \sum_{k=1}^{N} \bar{y}^k. \tag{16}$$

The sets A_*^k and A^{*k} are defined by Cartesian product i.e. $A_*^k = A_{1*}^k \times A_{2*}^k \times \ldots \times A_{n*}^k$, $A^{*k} = A_1^{*k} \times A_2^{*k} \times \ldots \times A_n^{*k}$.

The content of the general defuzzification module according to the KM type-reduction algorithm in fuzzy rough network is depicted in Fig. 1. A similar system but without the Karnik-Mendel type-reduction iterative method was proposed by Simiński [39].

Another formulation of the system could be adopted for the case of missing features if we know a variation range of the features with unknown values. In such a case, fuzzy sets A_i' are substituted by intervals $\{x_{i\min}, x_{i\max}\}$ and sets A_{i*}^k and A_i^{*k} are calculated as follows

$$\mu_{A_i^{*k}}(X_i) = \sup_{x_{i\min} < x_i < x_{i\max}} \mu_{A_i^k}(x_i), \tag{17}$$

$$\mu_{A_{i*}^k}(X_i) = \inf_{x_{i\min} < x_i < x_{i\max}} \mu_{A_i^k}(x_i). \tag{18}$$

When a value $\overline{x}_i$ is known, we get

$$\mu_{A_{i*}^k}\left(\overline{x}_i\right) = \mu_{A_i^{*k}}\left(\overline{x}_i\right) = \mu_{A_i^k}\left(\overline{x}_i\right). \tag{19}$$

The structure of adapted fuzzy rough network is presented in Fig. 2.

IV. Learning

The learning of a neuro-fuzzy system is the subject of many papers. Parameters of the fuzzy sets in rules can be determined by various versions of genetic, evolutionary and gradient algorithms using the commonly known criterion:

$$Q = \frac{1}{2} \sum_{j=1}^{m} \left(\overline{y}_j - d_j\right)^2, \tag{20}$$

where $\overline{y}_j$ is a value obtained on j-th output and d_j is desired value on j-th output, m is the number of outputs. It could be applied also to above proposed system but only when values of all features are known. In this case, both subsystems work with the same data and we obtain $\overline{y}_* = \overline{y}^* = \overline{y}$, $m = 1$, and consequently j can be omitted.

When of at least one feature value is unknown, the fuzzy rough network gives mentioned above couple values $\overline{y}_*$, $\overline{y}^*$ ($m = 2$) which constitute an output interval $[\overline{y}_*, \overline{y}^*]$. It must be noted that the desired value d is common for both outputs. Below we consider a few possible criteria, some of them are new.

1) Criterion "standard": By applying directly the criterion given by (20), we obtain the following definition

$$Q = \frac{1}{2}\left(\overline{y}_* - d\right)^2 + \frac{1}{2}\left(\overline{y}^* - d\right)^2. \tag{21}$$

It can be also decomposed into two individual criteria for particular outputs,

$$Q_* = \frac{1}{2}\left(\overline{y}_* - d\right)^2 \tag{22}$$

$$Q^* = \frac{1}{2}\left(\overline{y}^* - d\right)^2 \tag{23}$$

The criterion attracts the both output values to desired value d; however, the attraction is stronger when the distance is higher. Thus, the criterion always reduce the width of the output interval. Fig 3 presents the drift of output values in three cases. The pressure of this criterion tries to involuntary narrow the output interval. The theoretically "desired" state, i.e. $Q = 0$ is reached when $\overline{y}_* = \overline{y}^* = d$ even in the case of missing input values. However, as it was mentioned above, we expect that $\overline{y}_* = \overline{y}^* = d$ occurs only when values of all input features are known. Moreover, in consequence of narrowing the output interval this criterion leads to widening of the sets in antecedents of the rules. It reduces the sensitivity of the system. Thus we can expect that the "standard" criterion is not proper for the learning with incomplete samples.

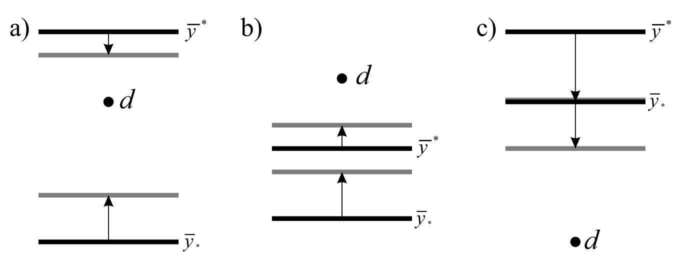

Fig. 3. The result of "standard" criterion a) d inside output interval, b) d above output interval, c) d below output interval

2) Criterion "shift of medium": This criterion attempts to set the output interval in such position that the desired value d is in its centre. It is defined as follows

$$Q = \frac{1}{2}\left(\frac{\overline{y}_* + \overline{y}^*}{2} - d\right)^2 \tag{24}$$

Note that the criterion does not refer to the width of the interval. The "desired" state $Q = 0$ is obtained when $\overline{y}_* + \overline{y}^* = 2d$

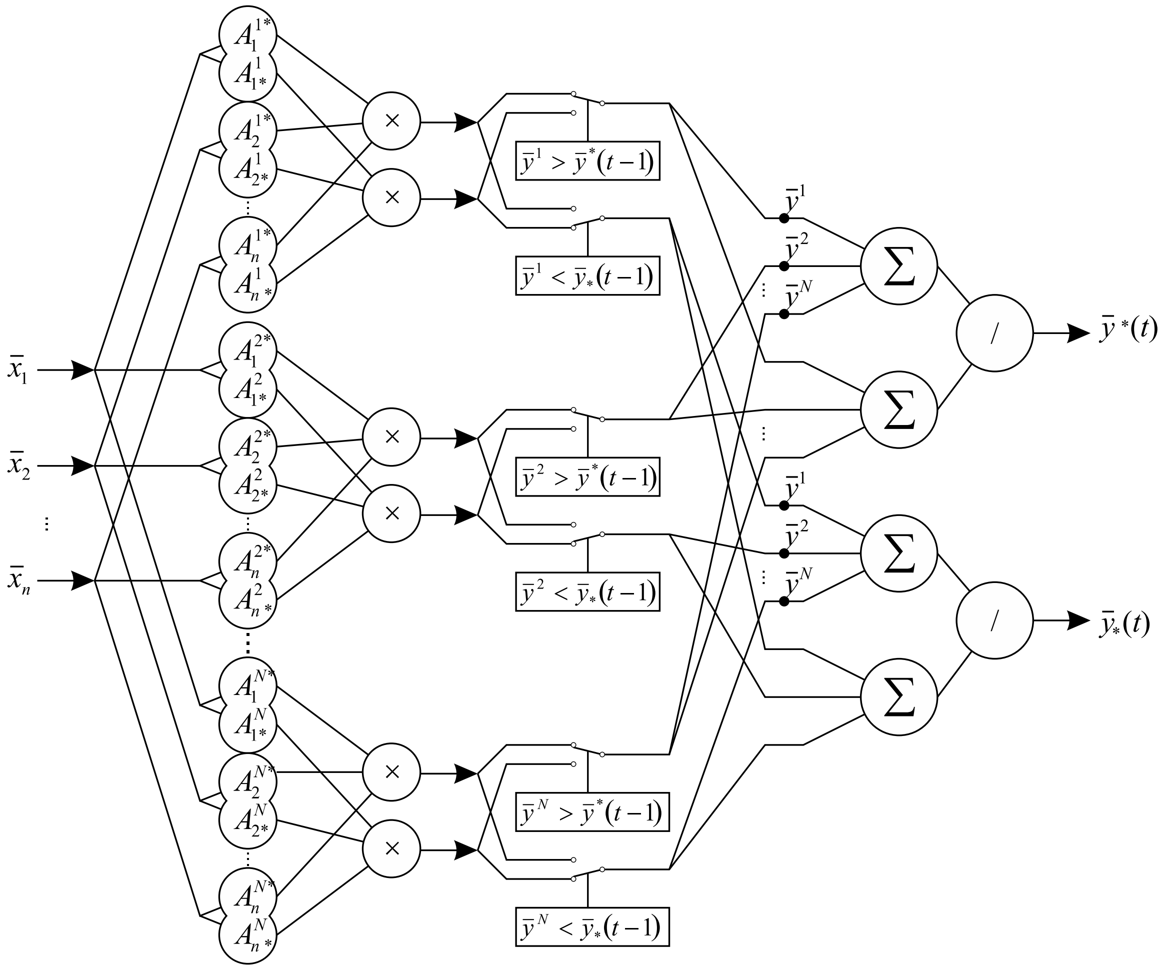

Fig. 1. The architecture of fuzzy rough network

and the width of output interval, i.e. $\overline{y}^* - \overline{y}_*$, is meaningless. In consequence, changes of the output interval width are unpredictable. A resulting impact of such changes is illustrated in Fig. 4. This criterion could be useful together with an additional criterion respecting the width of the output interval, e.g. in multicriteria genetic algorithms. The desired value of the width could depend on a number of missing values according to discussion in the Introduction of this paper.

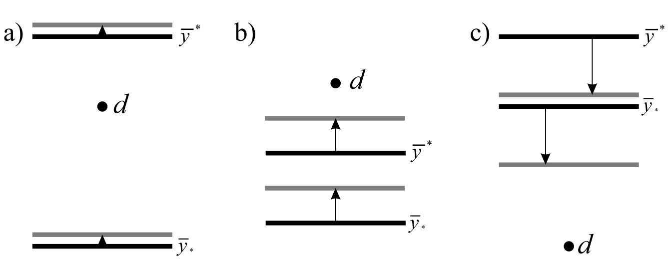

Fig. 4. The result of ”shift of medium” criterion a) d inside output interval, b) d above output interval, c) d below output interval

3) Criterion ”shift and narrow”: Another criterion is similar to the previous one but it is extended by an additional part respecting the width of the output interval

$$Q = \frac{1}{2}\left(\frac{\overline{y}_* + \overline{y}^*}{2} - d\right)^2 + \frac{1}{2}\left(\overline{y}^* - \overline{y}_*\right)^2 \qquad (25)$$

This will ensure some reduction of the width of the output interval. Therefore, during a long process of learning, it can strive near to zero even in the case of missing features. Such situation is illustrated in Fig. 5.

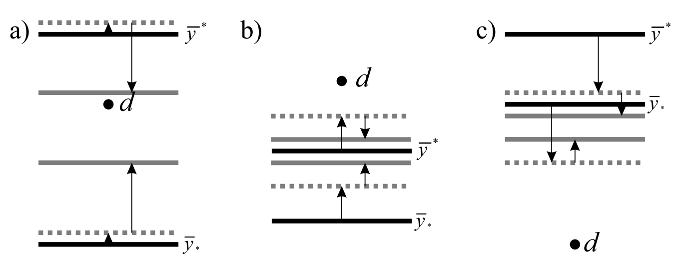

Fig. 5. The result of ”shift and narrow” criterion a) d inside output interval, b) d above output interval, c) d below output interval

In the case of problems which require a very long learning process, the criterion should be modified by weights, e.g.

$$Q = \alpha\frac{1}{2}\left(\frac{\overline{y}_* + \overline{y}^*}{2} - d\right)^2 + \beta\frac{1}{2}\left(\overline{y}^* - \overline{y}_*\right)^2 \qquad (26)$$

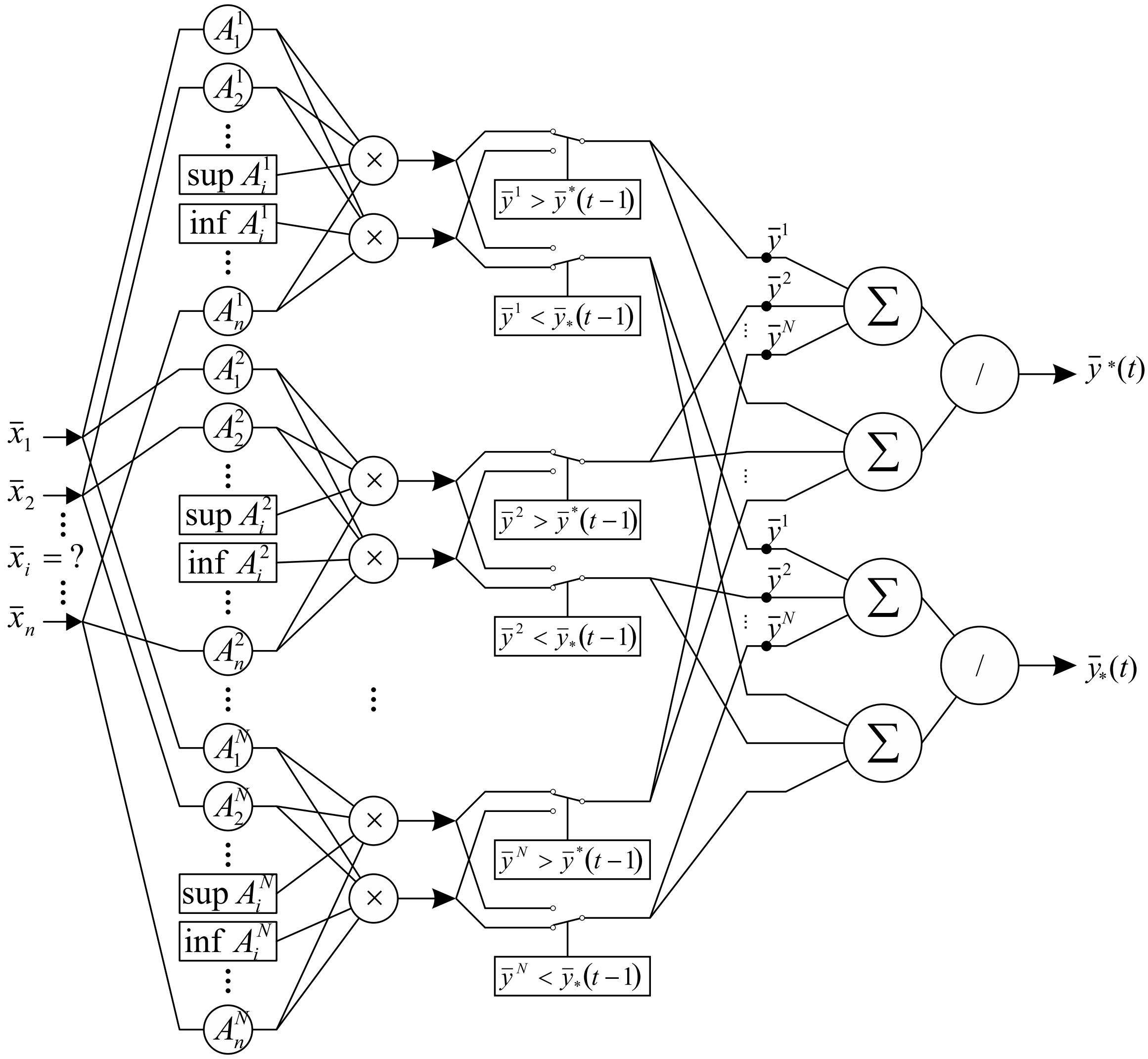

Fig. 2. The architecture of fuzzy rough network adapted for missing values

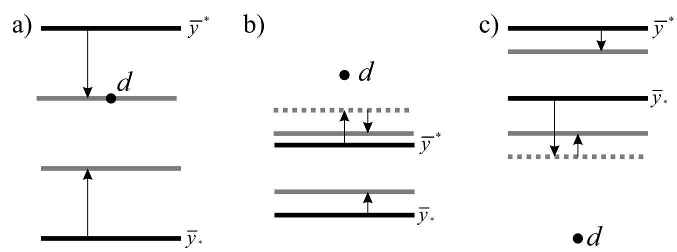

Fig. 6. The result of "bound and narrow" criterion a) d inside output interval, b) d above output interval, c) d below output interval

and the weight β should be decreasing with respect to the expected length of the learning process.

4) Criterion "bound and narrow": This criterion attracts the nearest bound of the output interval to the desired value as long as the desired value is outside of the interval. If d is within this interval, the criterion performs no change. Besides, it tries to reduce the width of the output interval.

$$Q = \frac{1}{2}\left(\max\left\{0, d - \overline{y}^{*}, \overline{y}_{*} - d\right\}\right)^{2} + \frac{1}{2}\left(\overline{y}^{*} - \overline{y}_{*}\right)^{2} \qquad (27)$$

The performance of the criterion is shown in Fig. 6.

In the case of problems which required very long learning process the criterion should be modified as previous one.

5) Criterion "expand then narrow": The last proposition acts in two overlapping stages. When the desired output value is outside of the interval, it tends to be extended in the direction of d, otherwise, when the desired value is inside, the width of the interval is asymmetrically decreasing. It is defined as follows

$$Q = \begin{cases} \frac{1}{2}\left(\overline{y}^{*} - d\right)^{2} & \text{if } d > \overline{y}^{*} \\ \frac{1}{2}\left(\overline{y}_{*} - d\right)^{2} + \frac{1}{2}\left(\overline{y}^{*} - d\right)^{2} & \text{if } \overline{y}_{*} < d < \overline{y}^{*} \\ \frac{1}{2}\left(\overline{y}_{*} - d\right)^{2} & \text{if } d < \overline{y}_{*}, \end{cases} \qquad (28)$$

or, alternatively, as two separate criteria for the particular outputs

$$Q_* = \begin{cases} \frac{1}{2}\left(\overline{y}_* - d\right)^2 & \text{if } d \leq \overline{y}^* \\ 0 & \text{if } d > \overline{y}^* \end{cases}, \quad (29)$$

$$Q^* = \begin{cases} \frac{1}{2}\left(\overline{y}^* - d\right)^2 & \text{if } d \geq \overline{y}_* \\ 0 & \text{if } d < \overline{y}_* \end{cases}. \quad (30)$$

This solution applied to any gradient algorithm will result with the width of the output interval being extended in the direction of values d being outside and the width of the interval will be reduced for values d being within the interval. This behaviour is illustrated in Fig. 7. We expect that in the first stage of learning, in most cases, the desired value will be outside of the output interval, so it will be extended. If the size of the system (the number of rules) is enough, in the second stage the interval will be sufficiently narrow to contain all the desired values.

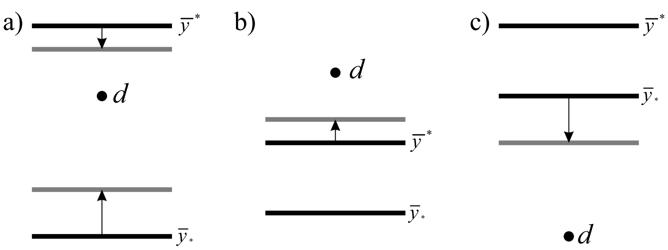

Fig. 7. The result of "expand then narrow" criterion a) d inside output interval, b) d above output interval, c) d below output interval

The consideration presented above have been verified in tests.

V. Experimental results

In order to analyse the fuzzy rough network, we have chosen the *pumadyn* dataset. As we can read in the documentation attached to the data [7], "the pumadyn datasets are a family of datasets synthetically generated from a realistic simulation of the dynamics of a Puma 560 robot arm". We have decided to choose four sets, all with 8 inputs but with various characteristics. The set *pumadyn8fh* is characterised by fairy linearity and high noise, the set *pumadyn8fm* is characterised by fairy linearity and medium noise, the set *pumadyn8nh* is characterised by nonlinearity and high noise, and the set *pumadyn8nm* is nonlinear with medium noise. The eight input features consist three angles between robot elements with range $\left[-\frac{1}{2}\beta\pi, \frac{1}{2}\beta\pi\right]$, its velocities with the same range and two torques at robot joints with range $\left[-\frac{1}{2}\beta, \frac{1}{2}\beta\right]$. The index β is fixed as 0.6 for sets *pumadyn8fm* and *pumadyn8fh* and as 1.2 for sets *pumadyn8nm* and *pumadyn8nh*.

The *pumadyne* dataset had been modelled by Shmilovici and Ben-Shimon [38] for a complete input interval and after a feature selection. Their results (RMSE) in the first case for selected above datasets are quoted in Table I. We will treat these results as a reference.

The fuzzy rough network gives on its output an answer in the form of an interval. If values of all features are known, the width of the interval is zero and should increase as a number and importance of unknown values increase. However, in our intention, the correct (desired) output value should be inside the interval. Otherwise, the distance to the interval constitutes the error measure. Therefore, we have got two measures that characterise the performance of the approximator. The first one is the root mean square of the distance between a desired value and the nearest boundary of an output interval,

$$\mathrm{RMSE_{Bnd}} = \sqrt{\frac{\sum_{s=1}^{M}\left(\max\{0, d_s - \overline{y}_s^*, \overline{y}_{*s} - d_s\}\right)^2}{M}}, \quad (31)$$

where s is an index of a sample, M is a number of samples, d_s is a desired value of s-th sample, $\overline{y}_s^*, \overline{y}_{*s}$ are upper and lower approximations of s-th sample. The second measure is given by the mean width of output intervals. It can be defined as follows

$$\mathrm{IWidth} = \frac{\sum_{s=1}^{M}\left(\overline{y}_s^* - \overline{y}_{*s}\right)}{M}. \quad (32)$$

TABLE I. RMSE obtained in selected *pumadyn* datasets by [38]

Database	RMSE
pumadyn8fh	3.16
pumadyn8fm	1.05
pumadyn8nh	3.28
pumadyn8nm	1.26

Tables II-V contain the average results ($\mathrm{RMSE_{Bnd}}$ and IWidth measures) obtained for fuzzy rough networks with 7 rules. The number of rules has been selected during multiple experiments. The number of missing features (1, 2 or 4) has been changing for each optimisation criteria (1 – 5). For evaluation purpose, the systems have been tested using 10-cross validation. The distribution of the missing values has been chosen randomly. In order to obtain comparable results, parameters of the learning process, as the learning coefficient, the momentum coefficient and the number of iterations, have been constant in all experiments. Therefore, it is quite possible to improve the presented $\mathrm{RMSE_{Bnd}}$ results by continuing the learning process or changing some of its parameters.

TABLE II. Impact of criteria on error and average interval width for *pumadyn8fh* database

Criterion	Measure	#missing features 1	2	4
"standard"	$\mathrm{RMSE_{Bnd}}$	2.94	3.13	3.87
	IWidth	1.76	2.35	2.04
"shift of medium"	$\mathrm{RMSE_{Bnd}}$	2.26	1.83	1.23
	IWidth	3.71	6.59	10.83
"shift and narrow"	$\mathrm{RMSE_{Bnd}}$	3.55	4.14	4.71
	IWidth	1.10	0.99	0.59
"bound and narrow"	$\mathrm{RMSE_{Bnd}}$	3.14	3.59	3.99
	IWidth	1.82	1.89	1.89
"expand then narrow"	$\mathrm{RMSE_{Bnd}}$	2.03	2.23	2.47
	IWidth	4.04	4.56	5.20

The results contained in Tables II-V show that the choice of criterion is extremely important. The presumptions presented in Section IV, that refer to the criteria, have been partly confirmed. The criterion "shift of medium" results with a huge width of intervals and therefore is of low importance. The criteria "standard", "shift and narrow" and "bound and narrow" have not reduced the width of intervals to zero and consequently both the width and the level of error have remained on a restrained level. The positively dominant criterion

TABLE III. IMPACT OF CRITERIA ON ERROR AND AVERAGE INTERVAL WIDTH FOR *pumadyn8fm* DATABASE

Criterion	Measure	#missing features 1	2	4
”standard”	$RMSE_{Bnd}$	1.36	1.97	3.14
	IWidth	1.97	2.59	2.36
”shift of medium”	$RMSE_{Bnd}$	0.52	0.47	0.42
	IWidth	3.95	6.25	10.15
”shift and narrow”	$RMSE_{Bnd}$	2.38	3.34	4.13
	IWidth	1.29	1.21	0.72
”bound and narrow”	$RMSE_{Bnd}$	2.21	2.87	3.47
	IWidth	1.60	1.81	1.79
”expand then narrow”	$RMSE_{Bnd}$	1.13	1.47	2.02
	IWidth	2.83	3.92	4.57

TABLE IV. IMPACT OF CRITERIA ON ERROR AND AVERAGE INTERVAL WIDTH FOR *pumadyn8nh* DATABASE

Criterion	Measure	#missing features 1	2	4
”standard”	$RMSE_{Bnd}$	3.12	3.66	4.59
	IWidth	2.25	2.55	1.86
”shift of medium”	$RMSE_{Bnd}$	2.08	1.63	1.10
	IWidth	5.26	8.48	13.42
”shift and narrow”	$RMSE_{Bnd}$	4.15	4.78	5.31
	IWidth	1.17	1.04	0.52
”bound and narrow”	$RMSE_{Bnd}$	3.64	4.10	4.56
	IWidth	2.00	2.11	1.98
”expand then narrow”	$RMSE_{Bnd}$	2.24	2.44	2.78
	IWidth	4.47	5.31	5.88

has turned out to be the criterion marked as ”bound and narrow”. Unfortunately, the results obtained using the criterion ”expand then narrow” have proved to be a surprise; the width of intervals is unacceptably high and the level of error not satisfactory. Note, that we had been expected the process of interval narrowing during the second stage of the learning process. This large divergence with respect to our expectations requires further study.

VI. CONCLUSIONS

In the paper, we have shown the structure of fuzzy rough network and its adaptation to handle with missing features in approximation tasks. We have studied the learning process in terms of various criteria. The criteria relate to the case of missing input features and an interval output. The appropriate criterion should allow to learn the system using even samples with missing features without imputation or marginalisation. Details on gradient learning of neuro-like networks was omitted as they were extensively studied in the literature. A general methodology presented in this paper can be adopted to other, more sophisticated methods of learning as well as to genetic or evolutionary algorithms. The presented research confirm that the results obtained by a single fuzzy-rough or a rough-fuzzy system are not completely satisfactory, hence such systems seem to be predisposed to work in ensembles [14]. The future work will be focused on improving the results e.g. by using other methods of learning or by constructing ensembles of fuzzy rough network as in the cases of rough fuzzy networks [13], [12], [11]. Moreover, we would like also extend the proposed model with an additional weight [44], which could result with reducing the number of rules [2]. Next, the logical methods of reasoning as well as the flexible reasoning schemes [1] will hopefully allow to obtain new desired properties of the system as interpretability of fuzzy rules [26].

TABLE V. IMPACT OF CRITERIA ON ERROR AND AVERAGE INTERVAL WIDTH FOR *pumadyn8nm* DATABASE

Criterion	Measure	#missing features 1	2	4
”standard”	$RMSE_{Bnd}$	1.86	2.83	4.14
	IWidth	2.58	3.08	2.27
”shift of medium”	$RMSE_{Bnd}$	0.81	0.82	0.70
	IWidth	5.31	7.76	12.21
”shift and narrow”	$RMSE_{Bnd}$	3.64	4.33	5.02
	IWidth	1.15	1.31	0.70
”bound and narrow”	$RMSE_{Bnd}$	2.91	3.68	4.26
	IWidth	1.91	2.16	2.00
”expand then narrow”	$RMSE_{Bnd}$	1.45	1.80	2.46
	IWidth	3.66	5.06	5.55

ACKNOWLEDGMENT

The project was funded by the National Science Centre under decision number DEC-2012/05/B/ST6/03620.

REFERENCES

[1] K. Cpałka. On evolutionary designing and learning of flexible neuro-fuzzy structures for nonlinear classification. *Nonlinear Analysis: Theory, Methods & Applications*, 71(12):1659–1672, 2009.

[2] K. Cpałka and L. Rutkowski. A new method for designing and reduction of neuro–fuzzy systems. In *Proceedings 2006 IEEE International Conference on Fuzzy Systems, IEEE World Congress on Computational Intelligence*, pages 8510–8516, Vancouver, BC, Canada, 2006.

[3] E. Czogała and J. Łęski. *Fuzzy and Neuro–Fuzzy Intelligent Systems*. Physica–Verlag, A Springer–Verlag Company, Heidelberg, New York, 2000.

[4] D. Dubois and H. Prade. Rough fuzzy sets and fuzzy rough sets. *International Journal of General Systems*, 17(2–3):191–209, 1990.

[5] D. Dubois and H. Prade. Putting rough sets and fuzzy sets together. In R. Sowiski, editor, *Intelligent Decision Support: Handbook of Applications and Advances of the Rough Sets Theory*, pages 203–232. Kluwer, Dordrecht, 1992.

[6] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. A Wiley–Interscience Publication, John Wiley & Sons, INC., New York/Chichester/Weinheim/Brisbane/Singapore/Toronto, 2001.

[7] Z. Ghahramani. The pumadyn datasets, 1996.

[8] W. Greblicki and L. Rutkowski. Density-free Bayes risk consistency of nonparametric pattern recognition procedures. *Proceedings of the IEEE*, 69(4):482–483, April 1981.

[9] N. N. Karnik and J. M. Mendel. Centroid of a type-2 fuzzy set. *Information Sciences*, 132:195–220, 2001.

[10] N. N. Karnik, J. M. Mendel, and Q. Liang. Type-2 fuzzy logic systems. *IEEE Transactions on Fuzzy Systems*, 7(6):643–658, 1999.

[11] M. Korytkowski, R. Nowicki, L. Rutkowski, and R. Scherer. Adaboost ensemble of dcog roughneurofuzzy systems. *Lectures Notes in Computer Systems*, 6922:62–71, 2011.

[12] M. Korytkowski, R. K. Nowicki, L. Rutkowski, and R. Scherer. Micog defuzzification roughneurofuzzy system ensemble. In *2010 IEEE International Conference on Fuzzy Systems, IEEE World Congress on Computational Intelligence*, pages 2268–2273, Barcelona, Spain, 2010.

[13] M. Korytkowski, L. Rutkowski, and R. Scherer. From ensemble of fuzzy classifiers to single fuzzy rule base classifier. In L. Rutkowski, R. Tadeusiewicz, L. Zadeh, and J. Zurada, editors, *Artificial Intelligence and Soft Computing ICAISC 2008*, volume 5097 of *Lecture Notes in Computer Science*, pages 265–272. Springer Berlin Heidelberg, 2008.

[14] M. Korytkowski, R. Scherer, and L. Rutkowski. On combining back-propagation with boosting. In *2006 International Joint Conference on Neural Networks, IEEE World Congress on Computational Intelligence, Vancouver, BC, Canada*, pages 1274–1277, 2006.

[15] A. Nakamura. Fuzzy rough sets. *Note Multiple-Valued Logic in Japan*, 9(8):1–8, 1988.

[16] R. Nowicki. On combining neuro–fuzzy architectures with the rough set theory to solve classification problems with incomplete data. *IEEE Trans. on Knowledge and Data Engineering*, 20(9):1239–1253, Sept. 2008.

[17] R. Nowicki. On combining neuro-fuzzy architectures with the rough set theory to solve classification problems with incomplete data. 20(9):1239–1253, 2008.

[18] R. Nowicki. Rough–neuro–fuzzy structures for classification with missing data. *IEEE Trans. on Systems, Man, and Cybernetics—Part B: Cybernetics*, 39(6):1334–1347, 2009.

[19] R. Nowicki. Rough neuro-fuzzy structures for classification with missing data. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics archive*, 39(6):1334–1347, 2009.

[20] R. Nowicki. On classification with missing data using rough-neuro-fuzzy systems. *International Journal of Applied Mathematics and Computer Science*, 20(1):55–67, 2010.

[21] R. K. Nowicki and J. T. Starczewski. On non-singleton fuzzifcation with dcog defuzzifcation. *Lecture Notes in Computer Science*, 6113:168–174, 2010.

[22] Z. Pawlak. Rough sets. *International Journal of Computer and Information Sciences*, 11(5):341–356, 1982.

[23] Z. Pawlak. *Rough Sets: Theoretical Aspects of Reasoning About Data*. Kluwer, Dordrecht, 1991.

[24] W. Pedrycz and A. Bargiela. Granular clustering: A granular signature of data. *IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics*, 32(2):212–224, Apr. 2002.

[25] L. Polkowski. *Rough Sets. Mathematical Foundation*. Physica–Verlag, A Springer-Verlag Company, Heidelberg, New York, 2002.

[26] A. Przybył and K. Cpałka. A new method to construct of interpretable models of dynamic systems. In L. Rutkowski, M. Korytkowski, R. Scherer, R. Tadeusiewicz, L. A. Zadeh, and J. Zurada, editors, *Lecture Notes in Computer Science*, volume 7268, pages 697–705. Springer Berlin Heidelberg, 2012.

[27] A. M. Radzikowska and E. E. Kerre. A comparative study of fuzzy rough sets. *Fuzzy sets and systems*, 126:137–155, 2002.

[28] D. Rutkowska and R. Nowicki. Implication-based neuro–fuzzy architectures. *International Journal of Applied Mathematics and Computer Science*, 10(4):675–701, 2000.

[29] L. Rutkowski. On Bayes risk consistent pattern recognition procedures in a quasi-stationary environment. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-4(1):84–87, Jan 1982.

[30] L. Rutkowski. Adaptive probabilistic neural networks for pattern classification in time-varying environment. *Neural Networks, IEEE Transactions on*, 15(4):811–827, July 2004.

[31] L. Rutkowski. *Flexible Neuro–Fuzzy Systems: Structures, Learning and Performance Evaluation*. Kluwer, 2004.

[32] M. Sarkar. Rough fuzzy functions in classification. *Fuzzy Sets and Systems*, 132(2):353–369, 2002.

[33] M. Sarkar and B. Yegnanarayana. Application of fuzzy–rough sets in modular neural networks. In *Proceedings IEEE World Congress on Computational Intelligence*, pages 741–746, Alaska, USA, May 1998. may 4–9.

[34] M. Sarkar and B. Yegnanarayana. Rough fuzzy membership functions. In *Proceedings IEEE World Congress on Computational Intelligence*, pages 796–801, Alaska, USA, May 1998. may 4–9.

[35] R. Scherer. Neuro-fuzzy relational systems for nonlinear approximation and prediction. *Nonlinear Analysis*, 71:e1420–e1425, 2009.

[36] R. Scherer and L. Rutkowski. Connectionist fuzzy relational systems. In S. Hagamuge and L. Wang, editors, *Studies in Computational Intelligence, Computational Intelligence for Modelling and Control*, pages 35–47. Springer-Verlag, 2005.

[37] G. Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, Princeton, New Jork, 1976.

[38] A. Shmilovici and D. Ben-Shimon. A feature selection strategy for the relevance vector machine. In Z. Chen and E. Lopez-Neri, editors, *Recent Advances in Knowledge Engineering and Systems Science*, pages 73–78. WSEAS Press, 2013.

[39] K. Simiński. Neuroroughfuzzy approach for regression modelling from missing data. *Int. J. Appl. Math. Comput. Sci.,*, 22(22):461–476, 2012.

[40] J. T. Starczewski. General type-2 FLS with uncertainty generated by fuzzy rough sets. In *Proc. IEEE-FUZZ 2010*, pages 1790–1795, Barcelona, 2010.

[41] J. T. Starczewski. Generalized uncertain fuzzy logic systems. In *Advanced Concepts in Fuzzy Logic and Systems with Membership Uncertainty*, volume 284 of *Studies in Fuzziness and Soft Computing*, pages 137–179. Springer Berlin Heidelberg, 2013.

[42] H. Thiele. Fuzzy rough sets versus rough fuzzy sets — an interpretation and a comparative study using concepts of modal logics. In *Proceedings of EUFIT97*, pages 159–167, 1997.

[43] L. A. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.

[44] M. Zalasiński and K. Cpałka. Novel algorithm for the on-line signature verification. In L. Rutkowski, M. Korytkowski, R. Scherer, R. Tadeusiewicz, L. A. Zadeh, and J. M. Zurada, editors, *Lecture Notes in Computer Science*, volume 7268, pages 362–367. Springer Berlin Heidelberg, 2012.