# A New Algorithm for Online Management of Fuzzy Rules Base for Nonlinear Modeling

**Krystian Łapa**

**Abstract** In this paper a new algorithm for online management of fuzzy rules base for nonlinear modeling is proposed. The online management problem is complex due to limitations of memory and time needed for calculations. The proposed algorithm allows an online creation and management of fuzzy rules base. It is distinguished, among the others, by mechanisms of: managing of number of fuzzy rules, managing of fuzzy rules weights and possibilities of background learning. The proposed algorithm was tested on typical nonlinear modeling problems.

**Keywords** Neuro-fuzzy system · Evolving fuzzy system · Background learning

## 1 Introduction

With the development of technology the amount of information (data) that can be stored and processed increases significantly. These data exceed the reach of commonly used hardware and software tools to capture, process and analyze it in an acceptable time. The typical systems have limitations due to limit of memory for storing data and time needed for iterative learning (processing of data to obtain best possible results of the system [3]). One of the solutions for this problem is online processing of the data [7]. In this solution the systems usually process incoming data, adapt to them (by modifying of its parameters) and lose information about processed data. Systems like that are classified as evolving systems (ES). The key element of processing data online is to find compromise between complexity and stability (possibilities of life-long learning - adapting of the system to new data at any point in time [15]).

The evolving systems include, among the others, the evolving fuzzy systems (EFS) [6]. The evolving fuzzy systems are based on fuzzy systems and fuzzy

K. Łapa (✉)
Institute of Computational Intelligence, Częstochowa University of Technology, Częstochowa, Poland
e-mail: krystian.lapa@iisi.pcz.pl



L. Borzemski et al. (eds.), *Information Systems Architecture and Technology: Proceedings of 37th International Conference on Information Systems Architecture and Technology—ISAT 2016—Part I*, Advances in Intelligent Systems and Computing 521, DOI 10.1007/978-3-319-46583-8_2

'IF… THEN…' rules [10]. The fuzzy rules allow not only to obtain good accuracy in the field of modelling, classification and control but also are interpretable [2]. Designing of EFS involves step by step into adding and modification of fuzzy rules and fuzzy sets to the system in certain situations (for example when new incoming data sample does not activate any existing fuzzy rule). These methods have their origin in algorithms used for creation of fuzzy rule base on the basis of data samples but they needed adaptation for larger data sets while maintaining a simple structure (see e.g. [8, 9, 13]).

On the other hand, the algorithms for designing systems offline allow to obtain better accuracy. Moreover they can work on populations of the systems to avoid local optima (see e.g. [5, 10]). These algorithms are very efficient on smaller data sets and have problems with iterative processing of larger data.

In this paper a new approach for online modelling with background learning and using fuzzy rules weights is introduced. The online designing of the system is based on analyzing of firing level of the rules and using the values of the fuzzy sets membership functions. The proposed background learning method uses limited number of prepared samples obtained from processing of data samples. Moreover, the fuzzy rules weights are calculated on the basis of counters of using fuzzy rules in a designing process. It is worth to mention that, the purpose of this paper is not only to achieve best accuracy of the system, but also to show influence of background learning and weights.

This paper is divided as follows: in Sect. 2 the proposed system is described, in Sect. 3 an idea of the design of the proposed system is presented, in Sect. 4 simulation results are shown, whereas in Sect. 5 the conclusions are drawn.

## 2 Neuro-Fuzzy System for Online Nonlinear Modelling

The proposed approach utilizes a neuro-fuzzy system of Mamdani type [10]. This system was designed to allow online building and modifying of the fuzzy rules with automatic weights selection. The fuzzy rules are based on dynamical fuzzy sets base $\mathbf{C}$ defined as follows:

$$\mathbf{C} = \left\{ \begin{array}{l} A_{1,1}, \ldots, A_{1,L_1^A}, \ldots, A_{n,1}, \ldots, A_{n,L_n^A}, \\ B_{1,1}, \ldots, B_{1,L_1^B}, \ldots, B_{m,1}, \ldots, B_{m,L_m^B} \end{array} \right\} = \{C_1, \ldots, C_{L^c}\}, \tag{1}$$

where $A_{i,l}$ stands for input fuzzy sets, $i = 1, \ldots, n$ stands for input index, $n$ stands for number of inputs, $l = 1, \ldots, L_i^A$ stands for index of fuzzy set, $L_i^A$ stands for number of input fuzzy sets from base (1) related to input $i$. $B_{j,l}$ stands for output fuzzy sets, $j = 1, \ldots, m$ stands for output index, $m$ stands for number of outputs, $l = 1, \ldots, L_j^B$ stands for index of fuzzy set, $L_j^B$ stands for number of output fuzzy sets from base (1) related to output $j$, $L^C = \sum_{i=1}^{n} L_i^A + \sum_{j=1}^{m} L_j^B$ stands for, changing in the learning process, total number of fuzzy sets. This approach allows to work with elastic number of fuzzy sets. Each fuzzy set $A_{i,l}$ is represented by membership function $\mu_{A_{i,l}}(x)$, while each fuzzy set $B_{j,l}$ is represented by membership function $\mu_{B_{j,k}}(y)$. In the proposed approach a Gaussian-type membership functions were used. Therefore, for the fuzzy sets the following parameters were assigned:

$$A_{i,l} = \left\{x^A_{i,l}, \sigma^A_{i,l}, c^A_{i,l}\right\}; \quad B_{j,l} = \left\{y^B_{j,l}, \sigma^B_{j,l}, c^B_{j,l}\right\}, \tag{2}$$

where $x^A_{i,l}$ and $y^B_{j,l}$ stands for centers of fuzzy sets, $\sigma^A_{i,l}$ and $\sigma^B_{j,l}$ stands for widths of fuzzy sets, $c^A_{i,l}$ and $c^B_{j,l}$ stands for counters (treat as heaviness) of using fuzzy sets. Fuzzy set is "used" when fuzzy set membership function achieve high value for specified data sample.

The fuzzy rules base contains fuzzy rules $\mathbf{R}_k$, where $k = 1, \ldots, N$ stands for fuzzy rule index, $N$ stands for actual number of fuzzy sets. The number of fuzzy rules can change in a learning process (see Sect. 3). In the proposed approach fuzzy rules are defined as follows:

$$\mathbf{R}_k = \left\{I^A_{1,k}, \ldots, I^A_{n,k}, I^B_{1,k}, \ldots, I^B_{m,k}, c^R_k\right\}, \tag{3}$$

where each index $I^A_{i,k}$ refers to one input fuzzy set from base $\mathbf{C}$ and each index $I^B_{j,k}$ refers to one output fuzzy set from base $\mathbf{C}$, $c^R_k$ is fuzzy rule counter (treat as heaviness) of using fuzzy rule. Fuzzy rule is "used" when fuzzy rule firing level achieves high value for specified data sample. This approach allows sharing of single fuzzy sets by many fuzzy rules, which notation is defined as:

$$R_k | w_k \colon \text{IF} \begin{pmatrix} x_1 \text{ is } A_{1,I^A_{1,k}} \text{ AND} \\ x_2 \text{ is } A_{2,I^A_{2,k}} \text{ AND} \\ \cdots \\ x_n \text{ is } A_{n,I^A_{n,k}} \end{pmatrix} \text{ THEN } \begin{pmatrix} y_1 \text{ is } B_{1,I^B_{1,k}} \text{ AND} \\ y_2 \text{ is } B_{2,I^B_{2,k}} \text{ AND} \\ \cdots \\ y_m \text{ is } B_{m,I^B_{m,k}} \end{pmatrix}, \tag{4}$$

where $w_k$ is a weight of $k$-th fuzzy rule calculated on the basis of fuzzy rules counters in a following way:

$$w_k = \alpha + (1 - \alpha) \cdot \left( \frac{c^R_k - \min\limits_{l=1,\ldots,N}\left\{c^R_l\right\}}{\max\limits_{l=1,\ldots,N}\left\{c^R_l\right\} - \min\limits_{l=1,\ldots,N}\left\{c^R_l\right\} + \xi} \right), \tag{5}$$

where $\alpha$ stands for parameter specifying minimum value of fuzzy rule weight ($w_k \in [\alpha, 1]$), $\xi > 0$ stands for very small value which prevents division by zero. Equation (5) assigns automatically higher values of weights for rules with higher value of fuzzy rule counters. It not only allows better interpretation of the rules (giving information which rule is more important) but also increases the possibilities of obtaining higher accuracy of the system.

The firing level (activation level) of fuzzy rule $\mathbf{R}_k$ is calculated as:

$$\tau_k(\bar{\mathbf{x}}) = \mathop{T}_{i=1}^{n} \left\{ \mu_{A_{i,I^A_{i,k}}}(\bar{x}_i) \right\} = T \left\{ \mu_{A_{1,I^A_{1,k}}}(\bar{x}_1), \ldots, \mu_{A_{n,I^A_{n,k}}}(\bar{x}_n) \right\}, \tag{6}$$

where $T(\cdot)$ is any triangular t-norm [10]. In case of singleton fuzzification the interferences from $k$-th rule are calculated independently for each $j$-th output using triangular t-norm (which is an interference operator in the Mamdani type of fuzzy system):

$$\mu_{\bar{B}_{j,k}}(\bar{\mathbf{x}}, y) = \mu_{\mathbf{A}_k \to B_{j,k}}(\bar{\mathbf{x}}, y) = T \left\{ \tau_k(\bar{\mathbf{x}}), \mu_{B_{I^B_{j,k}}}(y) \right\}. \tag{7}$$

The aggregation of interference of fuzzy rules is calculated as follows:

$$\mu_{\bar{B}_{j,k}}(\bar{\mathbf{x}}, y) = \mathop{S^*}_{k=1}^{N} \left\{ \mu_{\bar{B}_{j,k}}(\bar{\mathbf{x}}, y), w_k \right\} \tag{8}$$

where $S^*(\cdot)$ is a triangular t-conorm with weights of arguments. The defuzzificated values of fuzzy system of its $j$-th output can be calculated with (for example) the center of area method:

$$y_j(\bar{\mathbf{x}}) = \frac{\sum_{l=1}^{L^B_j} y^B_{j,l} \cdot \mu_{B'_j}\left(\bar{\mathbf{x}}, y^B_{j,l}\right)}{\sum_{l=1}^{L^B_j} \mu_{B'_j}\left(\bar{\mathbf{x}}, y^B_{j,l}\right)}, \tag{9}$$

where $y^B_{j,l}$ are values equal to maximum (isolated) points of the functions $\mu_{B_{j,k}}(y)$ (which are centers of used in simulations Gaussian-type fuzzy sets).

## 3 Description of Proposed Method

The proposed method is based on three mechanisms: preliminary analysis of data samples, building of fuzzy system and background learning. The main idea of proposed method is presented on Fig. 1. The proposed mechanisms are described in detail in the further part of this section.

### 3.1 Preliminary Analysis of Data Samples

The purpose of preliminary analysis of data samples is to estimate initialization values of widths of fuzzy sets. Each sample $\bar{\mathbf{x}}$ consist of $n$ input signals and $m$ output signals: $\bar{\mathbf{x}} = \{\bar{x}_1, \ldots, \bar{x}_n, \bar{x}_{n+1}, \ldots, \bar{x}_{n+m}\} = \{\bar{x}_1, \ldots, \bar{x}_n\}$, for which a

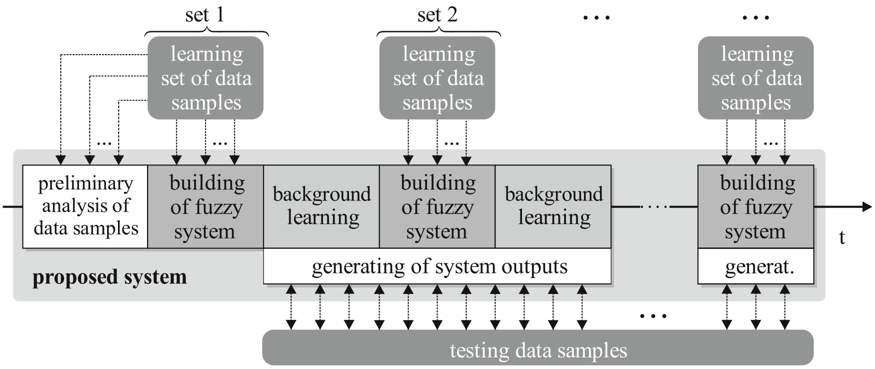


**Fig. 1** The idea of proposed evolving fuzzy system

minims $\mathbf{x}^{\min}$ and maxims $\mathbf{x}^{\max}$ are determined. The minims and maxims allows to determine widths of fuzzy sets in the following way:

$$\sigma_i = \sigma^{\text{par}} \cdot \left(x_i^{\max} - x_i^{\min}\right), \tag{10}$$

where $\sigma^{\text{par}} \in [0, 1]$ stands for parameter specifying the initial width of fuzzy sets. The minimums and maximums are initially calculated only for first set of data samples, and they are updated with every further data sample.

## 3.2 *Building of Fuzzy System*

In the proposed method a new fuzzy rules are added into the system when actual data sample $\bar{\mathbf{x}}$ does not activate with (specified by parameter $\tau^{\text{akt}} \in [0, 1]$) level any of existed fuzzy rules in form of (4). This condition can be write as:

$$\tau^{\text{akt}} < \max_{k=1,\ldots,N}\left\{\sqrt[n]{\tau_k(\bar{\mathbf{x}})}\right\}. \tag{11}$$

The use of the square root in the Eq. (11) reduces the impact of the system (9) inputs number on results of fuzzy rules activation. If the condition (11) is met the counter of the fuzzy rule $c_k^R$ (where $k$ stands for index of fuzzy rule with highest value of firing level) is increased and the parameters of fuzzy rule are modified as follows:

$$\begin{cases} x_{i,l}^A = \frac{c_{i,l}^A \cdot x_{i,l}^A + 1 \cdot \bar{x}_i}{c_{i,l}^A + 1}; & c_{i,l}^A = c_{i,l}^A + 1 \quad \text{for input fuzzy set } (i \le n) \\ y_{i-n,l}^B = \frac{c_{i-n,l}^B \cdot y_{i-n,l}^B + 1 \cdot \bar{x}_i}{c_{i-n,l}^B + 1}; & c_{i-n,l}^B = c_{i-n,l}^B + 1 \quad \text{for output fuzzy set } (i > n) \end{cases}, \tag{12}$$

Due to using heaviness of fuzzy sets in this modification, the insensitivity of changes slowly decreases. Thanks to that the fuzzy sets retain in a clear (interpretable) positions (they do not overlap each other). This idea is based on the idea of moving clusters from the Ward method [16].

If the condition (11) is not met a new fuzzy rule is added to the system (4). Newly created fuzzy rules can use both the existing in the base (1) and newly created fuzzy sets. To check if the fuzzy rule can use already existing fuzzy set for each input signal $\bar{x}_i$ from data sample $\bar{\mathbf{x}}$ the following condition is checked:

$$\begin{cases} \mu^{\text{akt}} > \max\limits_{l=1,\ldots,L_i^A} \left\{ \mu_{A_{h,l}}(\bar{x}_i) \right\} & \text{for input fuzzy set } (i \le n) \\ \mu^{\text{akt}} > \max\limits_{l=1,\ldots,L_{i-n}^B} \left\{ \mu_{B_{h-n,l}}(\bar{x}_i) \right\} & \text{for output fuzzy set } (i > n) \end{cases}, \tag{13}$$

where $\mu^{\text{akt}} \in [0, 1]$ stands for threshold value specifying when existing fuzzy set might be used (it acts similar to a function parameter $\tau^{\text{akt}}$ of fuzzy rules). If the condition (13) is met, a fuzzy rules can use existing fuzzy set (with highest value of membership function), the parameters of fuzzy set are modified according to (12), and the index $I_i^A$ or $I_{i-n}^B$ of the fuzzy rule is set to index of this fuzzy set. In the other case, a new fuzzy set is inserted into fuzzy sets base (1) with parameters initialized as follows:

$$\begin{cases} x_{i,l}^A = \bar{x}_i;\, \sigma_{i,l}^A = \sigma_i;\, c_{i,l}^A = 1 & \text{for input fuzzy set } (i \le n) \\ y_{j,l}^B = \bar{x}_{n+j};\, \sigma_{j,l}^B = \sigma_{n+j};\, c_{j,l}^B = 1 & \text{for output fuzzy set } (i > n) \end{cases}. \tag{14}$$

The building of fuzzy system is based on standard fuzzy system mechanisms: the analysis of firing (activation) of fuzzy rules and analysis of the values of fuzzy sets membership functions. The weights of fuzzy rules are calculated automatically on the basis of fuzz rules counters (heaviness). This approach is new in the literature.

## *3.3 Background Learning*

EFS are, by default, systems which parameters cannot be tuned. It results from theoretically infinite number of incoming online data samples. However a method of background learning (tuning) of parameters of fuzzy sets is presented (as an additional option—see Fig. 1). It was achieved by creating and storing maximum of $R^{\max}$ auxiliary samples for each fuzzy rule. Each auxiliary sample of $k$-th rule is stored in the form of cluster as in Ward's method [16]. Each cluster is represented by centers $x_{h,k,d}^R$ and heaviness $c_{k,d}^R$ where $d = 1, \ldots, R^{\max}$ is an index of auxiliary sample and $h = 1, \ldots, n + m$.

The process of creating auxiliary samples is connected to the process of creating and modifying fuzzy rules. Each of incoming data samples becomes an auxiliary sample for those rules for with highest value of firing (activation) level:

$$x^R_{h,k,d} = \bar{x}_h; \quad c^R_{k,d} = 1. \tag{15}$$

When the number of auxiliary samples for specified fuzzy rule is higher than maximum number of data samples $R^{\max}$ then, two closest auxiliary samples are merged. The distance between auxiliary samples (with taking into account heaviness) is calculated as follows:

$$dist_{d1,d2} = \frac{c^R_{k,d1} \cdot c^R_{k,d2}}{c^R_{k,d1} + c^R_{k,d2}} \cdot \sum_{h=1}^{n+m} \left| \frac{x^R_{h,k,d1} - x^R_{h,k,d2}}{x^{\max}_h - x^{\min}_h} \right|, \tag{16}$$

where $d1$, $d2$ are indexes of two comparing auxiliary samples. Therefore, the number $R^{\max}$ cannot be very high. The merging of two closest auxiliary samples is performed as follows:

$$x^R_{h,k,d3} = \frac{c^R_{k,d1} \cdot x^R_{h,k,d1} + c^R_{k,d2} \cdot x^R_{h,k,d2}}{c^R_{k,d1} + c^R_{k,d2}}; \quad c^R_{k,d3} = c^R_{k,d1} + c^R_{k,d2}, \tag{17}$$

where $d3$ is an index of newly created auxiliary sample. For the background learning, a genetic algorithm (GA) [10] was used (however any other learning methods can be also used—see e.g. [11, 12]), which aims to minimize error obtained for all auxiliary samples in all rules (in the learning process the auxiliary samples are treated as normal learning data samples).

## 3.4 System Evaluation

The evaluation function for GA includes both the complexity and accuracy of the system (9). The complexity of system (9) is defined as follows:

$$CMPL = w^{\text{rule}} \cdot N + w^{\text{fset}} \cdot \left( \sum_{i=1}^{n} L^{\text{A}}_i + \sum_{j=1}^{m} L^{\text{B}}_j \right), \tag{18}$$

where $w^{\text{rule}} \in [0, 1]$ stands for weight of fuzzy rules (set experimentally to 1.0), $w^{\text{fset}} \in [0, 1]$ stands for weight of fuzzy sets (set experimentally to 0.5). The accuracy of the system (9) is determined by *RMSE*:

$$RMSE = \frac{1}{Z \cdot m} \sum_{i=1}^{Z} \sum_{j=1}^{m} \sqrt{\left( \bar{y}_j(\bar{\mathbf{x}}_z) - x_{z,n+j} \right)^2}. \tag{19}$$

## 4 Simulations

In this paper a four simulation cases were tested (see Table 1). In the case I the proposed mechanisms of background learning and using fuzzy rule weights were turned-off (for a comparison). In the case II the fuzzy rule weights were turned-off to compare influence of weights on the results (case III). The case IV purpose was to achieve systems with simpler structures (by changing values of system parameters $\sigma^{par}$, $\tau^{akt}$, $\mu^{akt}$—see Table 1).

In this paper a set of modelling problems was used (presented in Table 2). The used data sets were modified in a way to allow to test the background learning in an online creation of fuzzy system: (a) from data a 10 % randomly chosen samples are selected and delivered into fuzzy system, (b) those samples are used to create auxiliary samples, (c) the background learning mechanism is applied (in case II–IV), (d) the 100 % of data is used for testing the system (9). This test procedure was repeated 20 times. Typical EFS are tested with each simulation for each problem, and it was repeated 50 times and then results were averaged.

The simulation parameters $\sigma^{par}$, $\tau^{akt}$, and $\mu^{akt}$ have a deciding impact on accuracy and complexity of the system and they were set experimentally according to Table 1 (for example an increase of $\tau^{akt}$ causes creating more fuzzy rules). The weight parameter $\alpha$ was set to 0.2. The background learning parameters were chosen experimentally: mutation probability $p_m = 0.15$, crossover probability $p_c = 0.75$, number of maximum auxiliary samples $R^{max} = 3$. In equations that define firing level of rules (6), interference of rules (7) and aggregation of interference (8) and the product triangular norms were used.

**Table 1** Considered simulation cases

| Case | BL | Weights | $\sigma^{par}$ | $\tau^{akt}$ | $\mu^{akt}$ | Case description |
|---|---|---|---|---|---|---|
| I | No | No | 0.15 | 0.25 | 0.25 | Typical EFS |
| II | Yes | No | 0.15 | 0.25 | 0.25 | Proposed EFS + BL |
| III | Yes | Yes | 0.15 | 0.25 | 0.25 | Proposed EFS + BL + rule weights |
| IV | Yes | Yes | 0.18 | 0.20 | 0.20 | Proposed EFS + lower complexity |

BL stands for background learning

**Table 2** List of used simulation problems

| Problem id (#) | Problem name | Reference | Inputs | Outputs | Samples |
|---|---|---|---|---|---|
| 1 | Hang function | [14] | 2 | 1 | 50 |
| 2 | Chemical plant | [1] | 3 | 1 | 70 |
| 3 | Abalone | [17] | 8 | 1 | 4178 |
| 4 | Machine CPU | [4] | 7 | 1 | 210 |

**Table 3** Linguistic labels used for fuzzy rule weights

| Weight value | Linguistic label | Definition |
|---|---|---|
| <0.4 | *l* | Low important |
| $\in \langle 0.4, 0.7 \rangle$ | *i* | Important |
| >0.7 | *v* | Very important |

**Table 4** Summary results of *RMSE* and *CMPL* (best *CMPL* concern best *RMSE* result)

| # | Case | Avg. *RMSE* | Avg. *CMPL* | Best *RMSE* | Best *CMPL* | Avg. *N* | Avg. fuzzy sets |
|---|---|---|---|---|---|---|---|
| 1 | I | 0.3798 | 12.1868 | 0.1371 | 13.5000 | 7.33 | 9.70 |
| | II | 0.3215 | 12.7000 | 0.1221 | 16.0000 | 7.66 | 10.08 |
| | III | **0.3040** | 12.9052 | 0.1153 | 14.5000 | 7.81 | 10.19 |
| | IV | 0.3711 | **11.1184** | **0.0946** | **13.0000** | **6.66** | **8.92** |
| 2 | I | 0.0483 | 13.8392 | 0.0223 | **13.5000** | 6.98 | **13.46** |
| | II | 0.0383 | 14.0454 | **0.0117** | 15.0000 | 7.24 | 15.22 |
| | III | **0.0379** | 14.0846 | 0.0144 | 16.0000 | 7.25 | 16.53 |
| | IV | 0.0399 | **12.5100** | 0.0165 | 14.0000 | **6.35** | 14.89 |
| 3 | I | 3.0375 | 24.2604 | 1.7169 | 25.5000 | 10.30 | 27.91 |
| | II | 2.5806 | 25.7413 | 1.5007 | 27.0000 | 11.19 | 29.10 |
| | III | **2.4931** | 25.4470 | 1.5251 | 26.5000 | 10.97 | 28.95 |
| | IV | 2.6760 | **21.4996** | **1.1781** | **22.5000** | **8.90** | **25.21** |
| 4 | I | 95.3925 | 22.8983 | 36.8803 | 22.0000 | 9.46 | 26.88 |
| | II | 58.7915 | 23.3000 | 18.1972 | 24.0000 | 9.80 | 27.00 |
| | III | **53.1230** | 23.3774 | **15.1279** | 24.0000 | 9.84 | 27.08 |
| | IV | 64.9428 | **20.7622** | 17.3769 | **21.0000** | **8.68** | **24.17** |

## 4.1 Simulation Results

The results obtained for considered problems for all cases are shown in Table 4. The *RMSE* from all test phases is shown in Fig. 3. The examples of obtained fuzzy sets for case IV are shown in Fig. 2 and corresponding fuzzy rules are presented in Table 5 (fuzzy sets were replaced by their linguistic labels '*v.low*', '*low*', '*medium*', '*high*', '*v.high*'—see Fig. 2 and the weights values were replaced by linguistic labels '*l*', '*i*', '*v*'—see Table 3).

## 4.2 Simulations Conclusions

The conclusions from simulations can be summed up as follows: (a) the results obtained for the proposed approach are close (in a field of accuracy) to the results

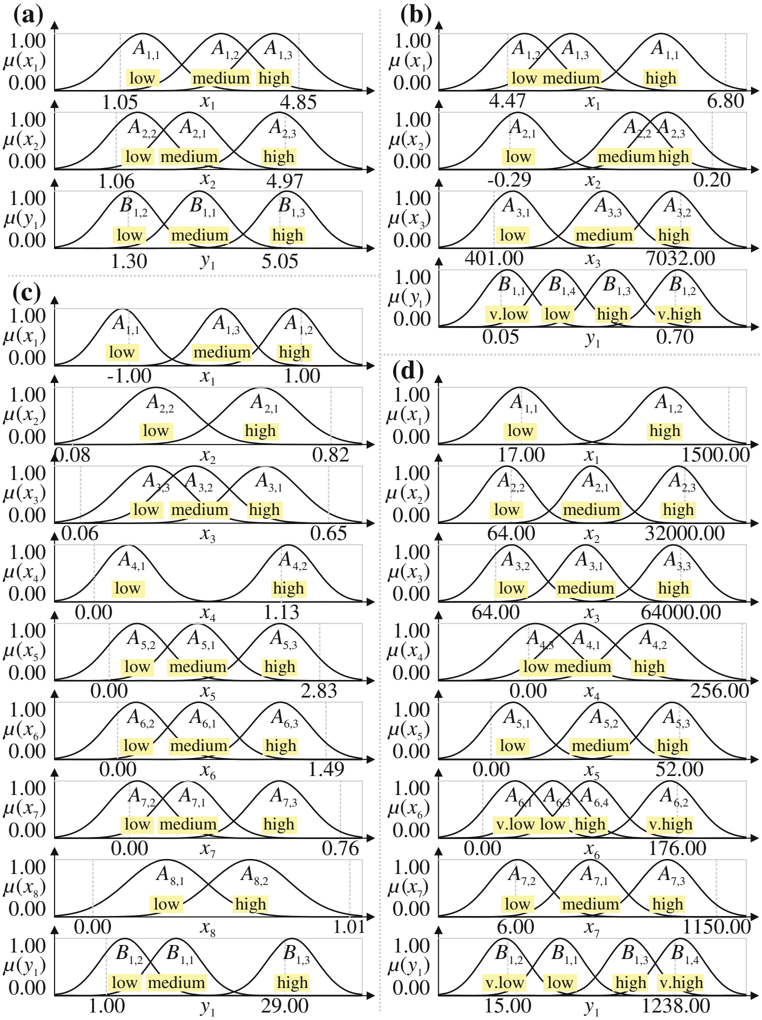


**Fig. 2** The examples of fuzzy sets for considered problems: **a** #1, **b** #2, **c** #3, **d** #4. *Dotted lines* stands for minimums (*left line*) and maximums (*right line*) of learning data

obtained by other authors for offline learning and (for most cases) with use of not interpretable systems, (b) the proposed approach allowed to achieve interpretable fuzzy rules (see Fig. 2 and Table 5), (c) the background learning improved EFS by approximately 15–20 % (see Table 4), (d) the use of automatically calculated fuzzy rule weights improved EFS by additional approximately 5 %, (e) the case IV allows to obtain good accuracy with lower number of rules and lower number of fuzzy sets, (f) the method without background learning stops improving after processing $\sim$50 % of data sets (see test phase number 5 on Fig. 3).

**Table 5** The example notations of fuzzy rules obtained for considered problems

| # | Fuzzy rules notation |
|---|---|
| 1 | $R_1 \vert v$ : IF ( $x_1$ is low AND $x_2$ is medium ) THEN ( $y_1$ is medium ) |
| | $R_2 \vert v$ : IF ( $x_1$ is medium AND $x_2$ is medium ) THEN ( $y_1$ is low ) |
| | $R_3 \vert i$ : IF ( $x_1$ is high AND $x_2$ is low ) THEN ( $y_1$ is medium ) |
| | $R_4 \vert i$ : IF ( $x_1$ is low AND $x_2$ is low ) THEN ( $y_1$ is high ) |
| | $R_5 \vert i$ : IF ( $x_1$ is high AND $x_2$ is high ) THEN ( $y_1$ is low ) |
| 2 | $R_1 \vert i$ : IF ( $x_1$ is high AND $x_2$ is low AND $x_3$ is low ) THEN ( $y_1$ is v.low ) |
| | $R_2 \vert v$ : IF ( $x_1$ is low AND $x_2$ is medium AND $x_3$ is high ) THEN ( $y_1$ is v.high ) |
| | $R_3 \vert i$ : IF ( $x_1$ is medium AND $x_2$ is low AND $x_3$ is medium ) THEN ( $y_1$ is high ) |
| | $R_4 \vert i$ : IF ( $x_1$ is high AND $x_2$ is medium AND $x_3$ is low ) THEN ( $y_1$ is v.low ) |
| | $R_5 \vert i$ : IF ( $x_1$ is medium AND $x_2$ is medium AND $x_3$ is low ) THEN ( $y_1$ is low ) |
| | $R_6 \vert i$ : IF ( $x_1$ is medium AND $x_2$ is high AND $x_3$ is medium ) THEN ( $y_1$ is high ) |
| 3 | $R_1 \vert v$ : IF ( $x_1$ is low AND $x_2$ is high AND $x_3$ is high AND $x_4$ is low AND $x_5$ is medium AND $x_6$ is medium AND $x_7$ is medium AND $x_8$ is low ) THEN ( $y_1$ is medium ) |
| | $R_2 \vert v$ : IF ( $x_1$ is high AND $x_2$ is high AND $x_3$ is medium AND $x_4$ is low AND $x_5$ is low AND $x_6$ is low AND $x_7$ is medium AND $x_8$ is low ) THEN ( $y_1$ is medium ) |
| | $R_3 \vert v$ : IF ( $x_1$ is medium AND $x_2$ is low AND $x_3$ is low AND $x_4$ is low AND $x_5$ is low AND $x_6$ is low AND $x_7$ is low AND $x_8$ is low ) THEN ( $y_1$ is low ) |
| | $R_4 \vert i$ : IF ( $x_1$ is low AND $x_2$ is high AND $x_3$ is high AND $x_4$ is low AND $x_5$ is high AND $x_6$ is high AND $x_7$ is high AND $x_8$ is high ) THEN ( $y_1$ is medium ) |
| | $R_5 \vert i$ : IF ( $x_1$ is high AND $x_2$ is high AND $x_3$ is high AND $x_4$ is low AND $x_5$ is high AND $x_6$ is high AND $x_7$ is high AND $x_8$ is high ) THEN ( $y_1$ is medium ) |
| | $R_6 \vert i$ : IF ( $x_1$ is high AND $x_2$ is high AND $x_3$ is high AND $x_4$ is low AND $x_5$ is high AND $x_6$ is mediumAND $x_7$ is medium AND $x_8$ ishigh ) THEN ( $y_1$ is high ) |
| | $R_7 \vert i$ : IF ( $x_1$ is high AND $x_2$ is high AND $x_3$ is medium AND $x_4$ is high AND $x_5$ is low AND $x_6$ is low AND $x_7$ is low AND $x_8$ is low ) THEN ( $y_1$ is medium ) |

(continued)

**Table 5** (continued)

| # | Fuzzy rules notation |
|---|---|
| 4 | $R_1\|i$ : IF ($x_1$ is low AND $x_2$ is medium AND $x_3$ is medium AND $x_4$ is medium AND $x_5$ is low AND $x_6$ is v.low AND $x_7$ is medium) THEN ($y_1$ is low) |
| | $R_2\|v$ : IF ($x_1$ is low AND $x_2$ is low AND $x_3$ is low AND $x_4$ is medium AND $x_5$ is low AND $x_6$ is v.low AND $x_7$ is low) THEN ($y_1$ is v.low) |
| | $R_3\|i$ : IF ($x_1$ is low AND $x_2$ is low AND $x_3$ is high AND $x_4$ is medium AND $x_5$ is low AND $x_6$ is v.high AND $x_7$ is high) THEN ($y_1$ is high) |
| | $R_4\|i$ : IF ($x_1$ is low AND $x_2$ is high AND $x_3$ is high AND $x_4$ is high AND $x_5$ is medium AND $x_6$ is low AND $x_7$ is high) THEN ($y_1$ is v.high) |
| | $R_5\|i$ : IF ($x_1$ is high AND $x_2$ is low AND $x_3$ is low AND $x_4$ is low AND $x_5$ is low AND $x_6$ is v.low AND $x_7$ is low) THEN ($y_1$ is v.low) |
| | $R_6\|i$ : IF ($x_1$ is low AND $x_2$ is low AND $x_3$ is medium AND $x_4$ is high AND $x_5$ is high AND $x_6$ is high AND $x_7$ is medium) THEN ($y_1$ is low) |

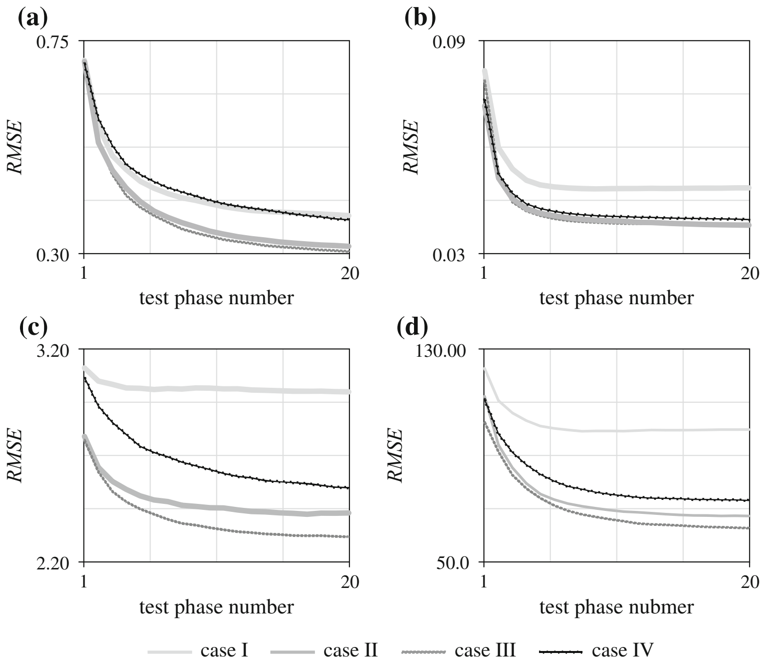


**Fig. 3** RMSE for test phases for considered problems: **a** #1, **b** #2, **c** #3, **d** #4

## 5 Conclusions

In this paper a new algorithm for online management of fuzzy rules base for nonlinear modeling was proposed. This algorithm allows, among the others, online creating of fuzzy rules and fuzzy sets, assigning weights to fuzzy rules weights and use proposed background learning mechanism. The proposed approach allows to achieve good accuracy in comparison to standard EFS. The biggest improvement was obtained by using the background learning on the basis of auxiliary data samples. The proposed method of calculating fuzzy rules weights allows for further improvement in accuracy of the system. Moreover, the proper selection of the system parameters allows to obtain clear fuzzy rules simultaneously with low complexity of the system. The obtained results can be considered as satisfactory.

**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. Cpałka, K., Łapa, K., Przybył, A., Zalasiński, M.: A new method for designing neuro-fuzzy systems for nonlinear modelling with interpretability aspects. Neurocomputing **135**, 203–217 (2014)
2. Gacto, M.J., Alcalá, R., Herrera, F.: Interpretability of linguistic fuzzy rule-based systems: an overview of interpretability measures. Inf. Sci. **181**(20), 4340–4360 (2011)
3. Kaisler, S., Armour, F., Espinosa, J.A., Money, W.: Big data: issues and challenges moving forward, system sciences (HICSS). In: Proceedings of the 46th Hawaii International Conference on, IEEE, pp. 995–1004 (2013)
4. Kibler, D., Aha, D.: Instance-based prediction of real-valued attributes. In Proceedings of the CSCSI (Canadian AI) Conference (1988)
5. Łapa, K., Cpałka, K.: On the application of a hybrid genetic-firework algorithm for controllers structure and parameters selection. Adv. Intel. Syst. Comput. **429**, 111–123 (2016)
6. Lughofer, E.: Evolving Fuzzy Systems—Methodologies. Advanced Concepts and Applications. Springer, Berlin (2011)
7. Lughofer, E.: On-line assurance of interpretability criteria in evolving fuzzy systems—achievements, new concepts and open issues. Inf. Sci. **251**, 22–46 (2013)
8. Lughofer, E., Hüllermeier, E.: On-line redundancy elimination in evolving fuzzy regression models using a fuzzy inclusion measure. In: Proceedings of the EUSFLAT 2011 Conference, Elsevier, Aix-Les-Bains, France, pp. 380–387 (2011)
9. Maciel, L., Lemos, A., Gomide, F., Ballini, R.: Evolving fuzzy systems for pricing fixed income options. Evolv. Syst. **3**(1), 5–18 (2012)
10. Rutkowski, L.: Computational Intelligence. Springer, Berlin (2008)
11. Rutkowski, L., Cpałka, K.: Flexible weighted neuro-fuzzy systems. In: Proceedings of the 9th International Conference on Neural Information Processing (ICONIP'02), Orchid Country Club, Singapore, November 18–22 (2002)
12. Rutkowski, L., Cpałka, K.: Neuro-fuzzy systems derived from quasi-triangular norms. In: IEEE International Conference on Fuzzy Systems, Budapest, July 26–29, vol. 2, pp. 1031–1036 (2004)
13. Soleimani, H., Lucas, K., Araabi, B.N.: Recursive Gath-Geva clustering as a basis for evolving neuro-fuzzy modelling. Evolv. Syst. **1**(1), 59–71 (2010)
14. Sugeno, M., Yasukawa, T.: A fuzzy logic based approach to qualitative modeling. IEEE Trans. Fuzzy Syst. **1**, 7–31 (1993)
15. Thrun, S.: Explanation-Based Neural Network Learning: A Lifelong Learning Approach. Kluwer Academic Publishers, Boston (1996)
16. Ward, J.H.: Hierarchical grouping to optimize an objective function. J. Am. Statist. Assoc. **58**, 236–244 (1963)
17. Waugh, S.: Extending and benchmarking Cascade-Correlation. PhD thesis, Computer Science Department, University of Tasmania (1995)

Springer

http://www.springer.com/978-3-319-46582-1

Information Systems Architecture and Technology: Proceedings of 37th International Conference on Information Systems Architecture and Technology – ISAT 2016 – Part I

Borzemski, L.; Grzech, A.; Świątek, J.; Wilimowska, Z. (Eds.)

2017, XV, 263 p. 72 illus., 32 illus. in color., Softcover
ISBN: 978-3-319-46582-1