This is a post-peer-review, pre-copyedit version of an article published in Artificial Intelligence and Soft Computing.
The final authenticated version is available online at: http://www.doi.org/10.1007/978-3-642-38610-7_48

A new approach to designing interpretable models of dynamic systems

Krystian Łapa[1], Andrzej Przybył[1], Krzysztof Cpałka[1]

[1]Częstochowa University of Technology,
Institute of Computational Intelligence, Poland
{krystian.lapa,andrzej.przybyl,krzysztof.cpalka}@iisi.pcz.pl

Abstract. In the process of designing automatic control system it is very important to have an accurate model of the controlled process. Approaches to modelling dynamic systems presented in the literature are often approximate, uninterpretable (acting as a black box), not appropriate to work in real-time, so it is not possible to create a hardware emulator on the basis of these approaches. The paper presents a new method to create model of nonlinear dynamic systems which gives a real opportunity for the interpretation of accumulated knowledge. By combining methods of control theory with fuzzy logic rules a good accuracy of the model can be achieved with use of a small number of fuzzy rules. Our method is based on the evolutionary strategy (μ, λ).

1 Introduction

Modelling of the systems is a widely developed area. In the literature many topics connected with modelling issue are considered. In the last years, besides classic solutions, modelling solutions based on the artificial neural networks and fuzzy logic rules are presented in e.g. [11], [16], [21]-[24], [31]-[34], [39]. It should be noted that most of the papers which use above-mentioned methods relate to the phenomena that can be simply described by "input-output" type transposition

$$\mathbf{y} = F(\mathbf{u}), \tag{1}$$

where $\mathbf{y}$ and $\mathbf{u}$ are vectors of input and output signals. In reality most of physical phenomena are dynamic and state of them is not only depended on input of current signals, but also on their prior states. Inclusion of historical data in the input vector $\mathbf{u}$ (i.e. prior values of inputs and/or outputs) enables consideration of dynamic dependences in the designed model. However, in the general case that model may be too complex and uninterpretable, what makes it usefulness in the practice. This disadvantage is a characteristic of modelling by artificial neural networks and also modelling based on the fuzzy rules which input vector was enlarged by the historic data. Another way of modelling is the use of the state variables technique. State of the dynamic object model may be comprehensively described by vector of state variables which has an appropriate size [17]. Vector of state variables describes completely a state of the object, what means that knowledge about it at time t, and knowledge about input signals (vector $\mathbf{u}$) at the next moments, with known model of the object, gives a complete knowledge about the object behaviour. The model in a linear case is described as follows

$$\frac{d\mathbf{x}}{dt} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \tag{2}$$

where $\mathbf{A}$ and $\mathbf{B}$ are system output matrices of an appropriate size. In the non-stationary models a coefficients of matrix which describe the model change in the time function. In the stationary models the coefficients are constant. In a more general case - i.e. for non-linear objects, equation (2) takes a form

$$\frac{d\mathbf{x}}{dt} = F\left(\mathbf{x}, \mathbf{u}\right), \tag{3}$$

where F is a non-linear dependency in the function of state variables and input signals. Modelling of the non-linear objects and phenomena is much more complicated. However, it should be noted that many physical phenomena may be described by local linear approximation (2) of non-linear dependency (3) about an operating point [17], [28]. Operating point changes over time during the process. However, a local re-determination of linear approximation in any new point is possible. For the discretization with the suitable short time step T that solution is enough accurate, even if the first order approximation is used, i.e.

$$\mathbf{x}\left(k+1\right) = \left(\mathbf{I} + \mathbf{A}\left(k\right)\right)\mathbf{x}\left(k\right)\cdot T + \mathbf{B}\left(k\right)\mathbf{u}\left(k\right)\cdot T, \tag{4}$$

where $\mathbf{I}$ is the identity matrix with the appropriate size. Designed models will refer to continuous objects noted as discrete form with time step T, connected with the current time t by the dependency $t = kT$, where $k = 1, 2, ...$ Modelling with use of the dynamic phenomena description as state variables and fuzzy rules will be based on the canonical form of the state equations [17] how it was explained in the previous paper [28].

The idea of our method consist in taking advantage of computational intelligence in the modelling of nonlinear systems. In the commonly used modelling method which models the industrial nonlinear dynamic systems, the applied intelligent systems were used as a "black box" or as a "grey box" models. It should be noted that the systems in many cases replaced commonly used and reputable solutions from the classic control theory. The authors would like to create a new hybrid method which will be used to support that solutions from control theory, but it will also make available the knowledge which describes a mechanism of the model work. The task of our intelligent systems is not modelling input-output dependencies. In this approach a large number of rules are required. This implies that the rules become illegible. In our approach an intelligent system is used to generate the coefficients of the matrices of the state-vector equation. For learning the intelligent system we use an evolutionary strategy (μ, λ) and data obtained from the observation of the real object.

The proposed approach has not yet been described in the literature by other authors. In paper [27] only a little similar conception of the authors has been presented. The conception was successfully used in the project of adaptive observer of state variables of induction motor. Received results confirm very well qualities of this solution. The results confirm also rightness of the presumptions of new algorithm development. In a previous paper [28] we simplified considered the problem, i.e. we assumed that we know the values of some coefficients of the system output matrices (i.e. matrix $\mathbf{A}$). In this paper we continue the investigations, however, we analyse the more generalized case.

This paper is organized into six sections. In the next section an idea of the proposed modelling method is presented. In Section 3 we describe neuro-fuzzy system for nonlinear modelling. Section 4 shows the evolutionary generation of the interpretable models of dynamic systems and Section 5 presents experimental results. Conclusions are drawn in Section 6.

2 Idea of the proposed modelling method

Comments on the proposed method can be summarized as follows:

- Our method allows to use of a lot of methods designed for analysis and design of linear model, also in use of the non-linear objects.
- Our method is based on the both described techniques: numeric modelling by the local linear approximation (about an current operating point) of non-linear object and fuzzy rules allow to practical interpretation of new received knowledge. In the proposed solution obtainment of accurate model of non-linear object (performed by fuzzy rules) and analysis of the model by techniques developed for linear models are possible, so the solution allows on automatic building of the object model based on the fuzzy rules.
- Our method based on the fuzzy rules enables interpretation of knowledge which was automatically obtained by an experiment during the observation of the real object. Interpretability of the model of any object is the result of the knowledge obtainment possibility about physical processes of modelled object. This is an attribute of the objects based on the fuzzy rules which are described as linguistic form.
- Our method (based on the algebraic equations with support of small number of fuzzy rules) allows for hardware realization, e.g. in the FPGA structures, and allows to real use of high potential of the novel soft-computing methods.

The approach proposed by us is illustrated in the Fig. 1 and uses fuzzy rules to generate the coefficients of matrices which are in algebraic notation of equations (3). The coefficients define the model. If the model is non-linear or non-stationary, the coefficients of matrices will change over time function or selected state variables function. The dependency between selected matrix coefficients will be described by the fuzzy rules in an interpretable way (Fig. 1). Form and number of the fuzzy rules will be automatically selected in the procedure of minimization of properly defined objective function with use of the evolutionary algorithm. If the approximate linear model is available, it will be used as initial model which will be improved in the next steps the of algorithm.

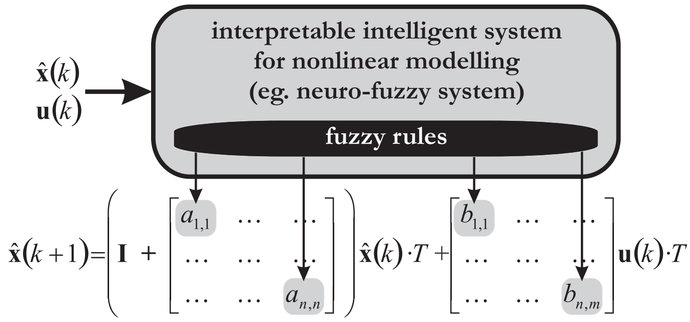

Fig. 1. Idea of the modelling method based on the fuzzy logic rules and modelling technique with use of dynamic state object variables.

3 Neuro-fuzzy system for nonlinear modelling

We propose the neuro-fuzzy system for nonlinear modelling, because the knowledge contained in it is interpretable. In literature various neuro-fuzzy systems have been developed (see e.g. [4], [9], [10], [15], [19]-[26], [30], [34]-[41]). They combine the natural language description of fuzzy systems and the learning properties of neural networks (see e.g. [1]-[3], [5]-[8], [33]).

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

$$R^k : \left[\text{IF} x_1 \text{is} A_1^k \text{AND} \ldots \text{AND} x_n \text{is} A_n^k \text{THEN} y_1 \text{is} B_1^k, \ldots, y_m \text{is} B_m^k \right], \tag{5}$$

where $\mathbf{x} = [x_1, \ldots, x_n] \in \mathbf{X}$, $\mathbf{y} = [y_1, \ldots, y_m] \in \mathbf{Y}$, $A_1^k, A_2^k, \ldots, A_n^k$ are fuzzy sets characterized by membership functions $\mu_{A_i^k}(x_i)$, $i = 1, \ldots, n$, $k = 1, \ldots, N$, whereas B_j^k are fuzzy sets characterized by membership functions $\mu_{B_j^k}(y_j)$, $j = 1, \ldots, m$, $k = 1, \ldots, N$.

Each of N rules (5) determines fuzzy sets $\bar{B}_j^k \subset \mathbf{Y}$ given by

$$\mu_{\bar{B}_j^k}(y_j) = \mu_{A_1^k \times \ldots \times A_n^k \to B_j^k}(\bar{\mathbf{x}}, y_j) = \mu_{\mathbf{A}^k \to B_j^k}(\bar{\mathbf{x}}, y_j) = T\left\{\tau_k(\bar{\mathbf{x}}), \mu_{B_j^k}(y_j)\right\}, \tag{6}$$

where $T\{\}$ is a t-norm (see e.g. [18]) and $T\left\{\tau_k(\bar{\mathbf{x}}), \mu_{B_j^k}(y_j)\right\}$ denotes an inference operator in the Mamdani-type system (see e.g. [34]). As a result of aggregation of the fuzzy sets $\bar{B}_j^k$, we get set B_j' with membership function given by

$$\mu_{B_j'}(y_j) = \overset{N}{\underset{k=1}{S}} \left\{\mu_{\bar{B}_j^k}(y_j)\right\}. \tag{7}$$

The defuzzification is realized by the COA method defined by the following formula

$$\bar{y}_j = \frac{\sum\limits_{r=1}^{N} \bar{y}_{j,r}^{B} \cdot \mu_{B_j'}\left(\bar{y}_{j,r}^{B}\right)}{\sum\limits_{r=1}^{N} \mu_{B_j'}\left(\bar{y}_{j,r}^{B}\right)}, \tag{8}$$

where $\bar{y}_{j,r}^{B}$ are centers of the membership functions $\mu_{B_j^r}(y_j)$, i.e. for $j = 1, \dots, m$ and $r = 1, \dots, N$, we have

$$\mu_{B_j^r}\left(\bar{y}_{j,r}^{B}\right) = \max_{y_j \in \mathbf{Y}} \left\{\mu_{B_j^r}(y_j)\right\}. \tag{9}$$

In our investigation we used the neuro-fuzzy system (8) to produce the values of coefficients of matrix $\mathbf{A}$ of the equation (3) as shown in the Fig. 1. Detailed description of the system (8) can be found in [10].

4 Evolutionary generation of the interpretable models of dynamic systems

We used the evolutionary strategy (μ, λ) in the process of creating the interpretable model of the dynamic systems. The purpose of this is to obtain the parameters of neuro-fuzzy system described in the previous section. In the process of evolution we assumed that:

- In a single chromosome $\mathbf{X}_{ch}^{\text{par}}$, according to the Pittsburgh approach, a complete linguistic model is coded in the following way

$$\begin{array}{l} \mathbf{X}_{ch}^{\text{par}} = \begin{pmatrix} \bar{x}_{1,1}^{A}, \sigma_{1,1}^{A}, \dots, \bar{x}_{n,1}^{A}, \sigma_{n,1}^{A}, \bar{y}_{1,1}^{B}, \sigma_{1,1}^{B}, \dots, \bar{y}_{m,1}^{B}, \sigma_{m,1}^{B}, \dots \\ \bar{x}_{1,N}^{A}, \sigma_{1,N}^{A}, \dots, \bar{x}_{n,N}^{A}, \sigma_{n,N}^{A}, \bar{y}_{1,N}^{B}, \sigma_{1,N}^{B}, \dots, \bar{y}_{m,N}^{B}, \sigma_{m,N}^{B} \end{pmatrix} \\ \qquad = \left(X_{ch,1}^{\text{par}}, X_{ch,2}^{\text{par}}, \dots, X_{chL^{\text{par}}}^{\text{par}}\right) \end{array}, \tag{10}$$

  where $ch = 1, \dots, \mu$ for parent population or $ch = 1, \dots, \lambda$ for the temporary population and $L^{\text{par}} = 2 \cdot N \cdot (n + m)$ denotes length of the chromosome (10).
- In a single binary chromosome $\mathbf{X}_{ch}^{\text{red}}$ is coded information about which the input and output fuzzy sets and the rules in the system (8) are redundant (their reduction does not affect the accuracy of the system (8))

$$\begin{array}{l} \mathbf{X}_{ch}^{\text{red}} = \begin{pmatrix} A_1^1, \dots, A_n^1, B_1^1, \dots, B_m^1, \text{rule}_1, \dots, \\ A_1^N, \dots, A_n^N, B_1^N, \dots, B_m^N, \text{rule}_N \end{pmatrix} \\ \qquad = \left(X_{ch,1}^{\text{red}}, X_{ch,2}^{\text{red}}, \dots, X_{ch,L^{\text{red}}}^{\text{red}}\right) \end{array}, \tag{11}$$

  where $ch = 1, \dots, \mu$ for parent population or $ch = 1, \dots, \lambda$ for the temporary population and $L^{\text{red}} = N \cdot (n + m + 1)$ denotes length of the chromosome (11). The genes in the chromosome $\mathbf{X}_{ch}^{\text{red}}$ take values from the set $\{0, 1\}$.

- Each neuro-fuzzy system has a number of outputs equal to the number of matrix $\mathbf{A}$ coefficients. Fitness function is based on the differences between output signals $\hat{\mathbf{x}}(\cdot)$ generated by the created model at step $k+1$ and corresponding reference values $\mathbf{x}(\cdot)$. Starting values for the model are the reference values at step k

$$\mathrm{ff}\left(\mathbf{X}_{ch}\right)=\sqrt{\frac{1}{2Z}\sum_{z=1}^{Z}\begin{pmatrix}\left(x_1(z+1)-\hat{x}_1(z+1)\right)^2+\\ \left(x_2(z+1)-\hat{x}_2(z+1)\right)^2\end{pmatrix}}. \tag{12}$$

- Genes in chromosome $\mathbf{X}_{ch}^{\mathrm{par}}$ which correspond to the input fuzzy sets $A_i{}^k$, $k=1,\ldots,N$, $i=1,\ldots,n$, ($\bar{x}_{i,k}^{A}$ and $\sigma_{i,k}^{A}$) and genes which correspond to the output fuzzy sets B_j^k, $k=1,\ldots,N$, $j=1,\ldots,m$ ($\bar{y}_{j,k}^{B}$ and $\sigma_{j,k}^{B}$) were initialized on the basis of the method described in [10] and [14].
- Genes in chromosome $\mathbf{X}_{ch}^{\mathrm{red}}$ were chosen as random numbers.

For more details on the evolutionary strategy (μ,λ) used to modify the parameters and structure of the neuro-fuzzy system (8) please see [13], [14], [33].

5 Experimental results

In our work we considered the van der Pol oscillator [42] which is used in the medicine as the model of the heartbeat. We have attempted to identify that model on the basis of the reference data which were generated using the adequate differential equation

$$\frac{d^2x}{dt^2}+\alpha\left(x^2-1\right)\frac{dx}{dt}+\omega^2x=0, \tag{13}$$

where α, ω are oscillator parameters and $x(t)$ is a reference value of the modelled process as a function of time (in the simulations we assumed that $\alpha=10$ and $\omega=1$). We used the following state variables: $x_1(t)=x(t)$, $x_2(t)=dx(t)/dt$. In such a case the system matrix $\mathbf{A}$ described by the formula (2) takes the following form

$$\mathbf{A}=\begin{bmatrix}0 & 1\\ -\omega^2 & -\alpha\left(x_1^2-1\right)\end{bmatrix}=\begin{bmatrix}a_{11}(\mathbf{x}) & a_{12}(\mathbf{x})\\ a_{21}(\mathbf{x}) & a_{22}(\mathbf{x})\end{bmatrix}. \tag{14}$$

The goal of the modelling was to recreate the unknown parameters $a_{11}(\mathbf{x})$, $a_{12}(\mathbf{x})$, $a_{21}(\mathbf{x})$, $a_{22}(\mathbf{x})$ in such a way that the model reproduces the reference data as accurately as possible. Of course in a general case analytical dependencies which were used to generate the reference data are not known. However, the proposed method allows us to reconstruct these dependencies in the form of a set of interpretable fuzzy rules (5). This is possible on the basis of the analysis of measurable outputs of the modelled process. In our case the measurable output signals are $x_1(k)$ and $x_2(k)$. They are used as reference values for the outputs. Output signals generated by the created model $\hat{x}_1(k)$ and $\hat{x}_2(k)$ are compared with their reference values and the error of the model is calculated (12).

Details of the simulations can be summarized as follows:

- The neuro-fuzzy system (8) used in the simulations is characterized by Gaussian membership functions, min/max triangular norms, $N = 3$ and $m = 4$.
- The evolutionary strategy (μ, λ) used for the learning of the intelligent system (8) is characterized by the following parameters: $\mu = 10$, $\lambda = 500$, $p_m = 0.077$, $p_c = 0.770$, and the number of generations $= 100000$ (for details see e.g. [10]).

The results of simulations, depicted in Table 1 and presented in the Fig. 3 and Fig. 4, can be summarized as follows:

- Neuro-fuzzy system obtained in evolutionary learning is characterized by two rules ($N = 2$), two inputs ($\hat{x}_1(k)$ and $\hat{x}_2(k)$) and four outputs ($a_{11}(k)$, $a_{12}(k)$, $a_{21}(k)$ and $a_{22}(k)$).
- A small number of rules in the obtained system allows on their interpretation. Equation (5) can be written as follows

$$\begin{cases} R^1 : \left[\mathrm{IF}\, x_1 \,\mathrm{is}\, A_1^1 \,\mathrm{AND}\, x_2 \,\mathrm{is}\, A_2^1 \,\mathrm{THEN}\, y_1 \,\mathrm{is}\, B_1^1, y_3 \,\mathrm{is}\, B_3^1, y_4 \,\mathrm{is}\, B_4^1 \right] \\ R^2 : \left[\mathrm{IF}\, x_1 \,\mathrm{is}\, A_1^2 \,\mathrm{AND}\, x_2 \,\mathrm{is}\, A_2^2 \,\mathrm{THEN}\, y_2 \,\mathrm{is}\, B_2^2, y_4 \,\mathrm{is}\, B_4^2 \right] \end{cases} . \tag{15}$$

Equation (15) can be written as follows

$$\begin{cases} R^1 : \begin{bmatrix} \mathrm{IF}\, x_1(k) \,\mathrm{is}\, \mathbf{low} \,\mathrm{AND}\, x_2(k) \,\mathrm{is}\, \mathbf{low} \\ \mathrm{THEN}\, a_{11}(\mathbf{x}) \,\mathrm{is}\, \mathbf{medium}, a_{21}(\mathbf{x}) \,\mathrm{is}\, \mathbf{medium}, a_{22}(\mathbf{x}) \,\mathrm{is}\, \mathbf{low} \end{bmatrix} \\ R^2 : \begin{bmatrix} \mathrm{IF}\, x_1(k) \,\mathrm{is}\, \mathbf{high} \,\mathrm{AND}\, x_2(k) \,\mathrm{is}\, \mathbf{high} \\ \mathrm{THEN}\, a_{12}(\mathbf{x}) \,\mathrm{is}\, \mathbf{medium}, a_{22}(\mathbf{x}) \,\mathrm{is}\, \mathbf{high} \end{bmatrix} \end{cases} , \tag{16}$$

where "low", "medium" and "high" are the values of the linguistic variables $x_1(k)$, $x_2(k)$, $a_{11}(\mathbf{x})$, $a_{12}(\mathbf{x})$, $a_{21}(\mathbf{x})$ and $a_{22}(\mathbf{x})$ and are represented by fuzzy sets shown in the Fig. 2.
- The coefficients of the matrix $\mathbf{A}$, generated by neuro-fuzzy system (8) as a function of $x_1(k)$ and $x_2(k)$ are presented in the Fig. 3. We can see that the values of the coefficient $a_{11}(\mathbf{x})$ takes the value like 0 and $a_{12}(\mathbf{x})$ takes the value like 1. We can also see that the values of $a_{21}(\mathbf{x})$ and $a_{22}(\mathbf{x})$ vary according to the theoretical model described by (13) and (14).
- Accuracy of the nonlinear modelling obtained in simulations is shown in Table 1 and presented in the Fig. 4. The evolutionarily selected neuro-fuzzy system (8) works with good accuracy. It should be emphasized that our goal was not obtaining the best performance of neuro-fuzzy systems, but we wanted to show that it is possible to automatically design neuro-fuzzy structures and to find all parameters of such structures characterized by a good performance. In addition, our goal was to show that our method of determining the coefficients of the matrix $\mathbf{A}$ can be described by transparent rules. These goals have been fully achieved.
- In the simulations a simple approach to nonlinear modelling were also considered. In this approach, the system (8) was directly used for modelling according to formula (2). The value of RMSE was equal 0.437 for the three rules ($N = 3$). Thus, modelling accuracy was lower compared to that obtained using the method proposed in this paper.

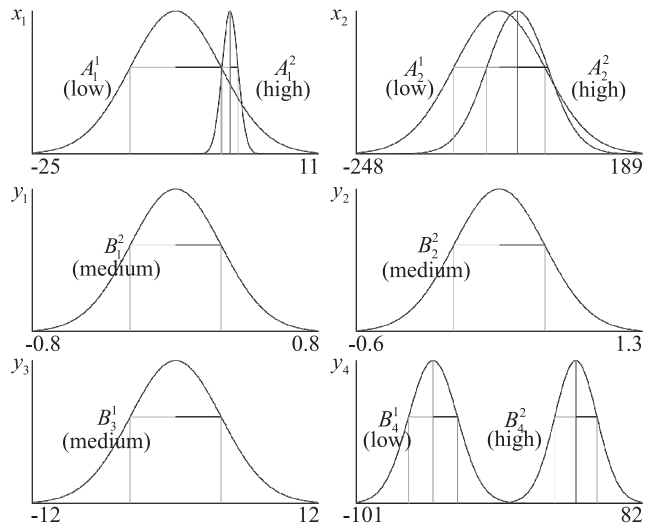

Fig. 2. Input and output fuzzy sets of the neuro-fuzzy system (8).

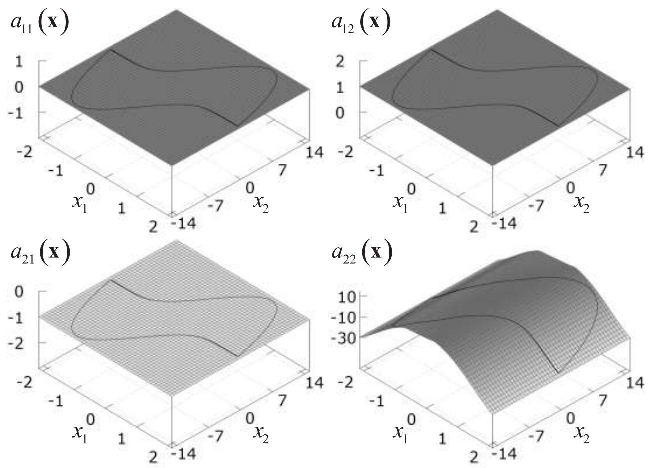

Fig. 3. The coefficients of the matrix $\mathbf{A}$, generated by neuro-fuzzy system (8) as a function of $x_1(k)$ and $x_2(k)$.

6 Summary

In the paper a new method to create model of nonlinear dynamic systems was proposed. The novelty was the combination of the method of control theory and fuzzy rules, which enables to obtain good accuracy of the model using a small number of fuzzy rules. Our method will be able to work in the real-time and it will be enough accurate for mapping in details many nonlinear dynamic systems,

Table 1. Accuracy of the nonlinear modelling obtained in simulations.

Method	Number of rules	Average RMSE
A. Przybył, K. Cpałka ([28])	$N=5$	0.0007
A. Przybył, K. Cpałka ([28])	$N=3$	0.0035
our result	$N=2$	**0.0145**

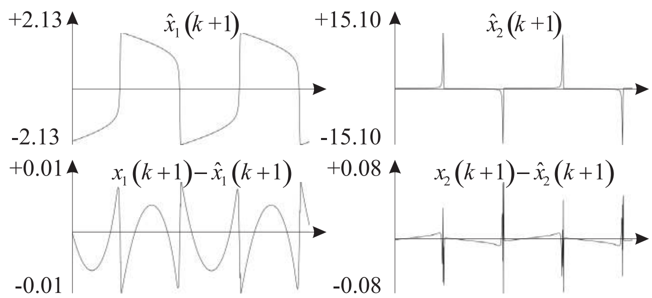

Fig. 4. Output and error signals obtained in the simulations.

e.g. industrial processes or natural phenomena. It will be able to obtain the model by the non-invasive observation, transparent for the monitored process. In a case of industrial process modelling, data will be collected by the analysis of the packets which are sent in the real-time in the Ethernet network (see e.g. [29]). The proposed method is based on the evolutionary strategy (μ, λ) and allows to find both the structure and parameters of the used neuro-fuzzy system in the process of evolution. That model gives the potential possibility to the interpretation of accumulated knowledge. The simulation shows the fully usefulness of the proposed method.

Acknowledgment

The project was financed by the National Science Center on the basis of the decision number DEC-2012/05/B/ST7/02138.

References

1. Bartczuk Ł., Rutkowska D., 2006. A new version of the Fuzzy-ID3 algorithm. *Artificial Intelligence and Soft Computing* vol.4029, pp. 1060-1070.
2. Bartczuk Ł., Rutkowska D., 2009. Medical diagnosis with type-2 fuzzy decision trees. *Computers in Medical Activity, Advances in Intelligent and Soft Computing*, vol. 65, pp. 11-21.
3. Bartczuk Ł., Rutkowska D., 2008. Type-2 fuzzy decision trees, *Lecture Notes in Artificial Intelligence Subseries of Lecture Notes in Computer Science*, vol. 5097, pp. 197-206.
4. Casillas J., Cordon, O., Herrera, F. and Magdalena, L., eds., 2003. *Interpretability Issues in Fuzzy Modelling*. Springer.

5. Cierniak R., 2008. A new approach to image reconstruction from projections problem using a recurrent neural network, emphApplied Mathematics and Computer Science, vol.18, no. 2, pp. 147-157.
6. Cierniak R., 2008. A novel approach to image reconstruction from fan-beam projections using recurrent neural network, *Lecture Notes in Artificial Intelligence*, vol. 5097, pp. 752-761.
7. Cierniak R., 2004. An image compression algorithm based on neural networks, *Lecture Notes in Artificial Intelligence*, vol. 3070, pp. 706-711.
8. Cierniak R., 2009. New neural network algorithm for image reconstruction from fan-beam projections, *Elsevier Science: Neurocomputing*, vol. 72, pp. 3238-3244.
9. Cpałka K., 2009. A New Method for Design and Reduction of Neuro-Fuzzy Classification Systems. *IEEE Transactions on Neural Networks*, vol. 20, no. 4, pp. 701-714.
10. Cpałka K., 2009. On evolutionary designing and learning of flexible neuro-fuzzy structures for nonlinear classification. *Nonlinear Analysis Series A: Theory, Methods and Applications*, Elsevier, vol. 71, issue 12, pp. e1659-e1672.
11. Cpałka K., Rutkowski L., 2006. A new method for designing and reduction of neuro-fuzzy systems. *2006 IEEE International Conference on Fuzzy Systems*, pp. 1851-1857.
12. Cpałka K. and Rutkowski L., 2005. Flexible Takagi-Sugeno Neuro-Fuzzy Structures for Nonlinear Approximation. *WSEAS Transactions on Systems*, vol. 4, no. 9, pp. 1450-1458.
13. Fogel D. B., 2006. *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence.* IEEE Press, Piscataway, NJ. Third Edition.
14. Gabryel M., Cpałka K. and Rutkowski L., 2005. Evolutionary strategies for learning of neuro-fuzzy systems. *I Workshop on Genetic Fuzzy Systems*, Genewa, pp. 119-123.
15. Gabryel M., Rutkowski L., 2008. Evolutionary methods for designing neuro-fuzzy modular systems combined by bagging algorithm, *Lecture Notes in Artificial Intelligence, Subseries of Lecture Notes in Computer Science*, vol. 5097, pp. 398-404.
16. Horzyk A., Tadeusiewicz R., 2004. Self-Optimizing Neural Networks. *Lecture Notes in Computer Science*, vol. 3173, pp. 150-155.
17. Katsuhiko Ogata, 2001. Modern Control Engineering. *Prentice Hall.*
18. Klement E. P., Mesiar R. and Pap E., 2000. *Triangular Norms.* Kluwer Academic Publishers.
19. Korytkowski M., Gabryel M., Rutkowski L., Drozda S., 2008. Evolutionary methods to create interpretable modular system, *Lecture Notes in Artificial Intelligence Subseries of Lecture Notes in Computer Science*, vol. 5097, pp. 405-413.
20. Korytkowski M., Rutkowski L., Scherer R., 2005. On combining backpropagation with boosting, *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, Vancouver, pp. 1274-1277.
21. Laskowski Ł., 2012. A novel hybrid-maximum neural network in stereo-matching process. *Neural Computing & Applications*, DOI 10.1007/s00521-012-1202-0.
22. Laskowski Ł., 2012. Objects auto-selection from stereo-images realized by self-correcting neural network. *Lecture Notes in Artificial Intelligence*, vol. 7268, pp. 119-125.
23. Laskowski Ł., 2010. A novel continuous dual mode neural network in stereo-matching process. *Lecture Notes in Artificial Intelligence*, vol. 6354, pp. 294-297.
24. Li X., Er M. J., Lim B. S. et al., 2010. Fuzzy Regression Modeling for Tool Performance Prediction and Degradation Detection, *International Journal of Neural Systems*, vol. 20, issue 5, pp. 405-419.

25. Nowicki R., Pokropińska A., 2004. Information criterions applied to neuro-fuzzy architectures design, *Lecture Notes in Artificial Intelligence*, vol. 3070, pp. 332-337.
26. Nowicki R, Scherer R. and Rutkowski L., 2002. A method for learning of hierarchical fuzzy systems. Peter Sincak, Jan Vascak, Vladimir Kvasnicka, Jiri Pospichal editors, *Intelligent Technologies - Theory and Applications*, IOS Press, pp. 124-129.
27. Przybył A., 2003. Doctoral dissertation: Adaptive observer of induction motor using artificial neural networks and evolutionary algorithms (in Polish). *Poznan University of Technology.*
28. Przybył A. and Cpałka K., 2012. A new method to construct of interpretable models of dynamic systems. *Lecture Notes in Artificial Intelligence*, vol. 7268, pp. 697-705.
29. Przybył A., Smoląg J. and Kimla P., 2010, Real-time Ethernet based, distributed control system for the CNC machine (in Polish). *Electrical Review 2010-2.*
30. Rutkowska D., Nowicki R. and Rutkowski L., 2000. Neuro-fuzzy architectures with various implication operators. Peter Sincak et al. editors, *The State of the Art in Computational Intelligence*, pp. 214-219.
31. Rutkowski L., PrzybyłA. and Cpałka K., 2012. Novel Online Speed Profile Generation for Industrial Machine Tool Based on Flexible Neuro-Fuzzy Approximation. *IEEE Transactions on Industrial Electronics*, vol. 59, no. 2, pp. 1238-1247.
32. Rutkowski L. and Cpałka K., 2002. Flexible weighted neuro-fuzzy systems. *Proceedings of the 9th Neural Information Processing*, pp. 1857-1861.
33. Rutkowski L., 2007. *Computational Intelligence*, Springer.
34. Rutkowski L. and Cpałka K., 2003. Flexible neuro-fuzzy systems. *IEEE Trans. Neural Networks*, vol. 14, no. 3, pp. 554-574.
35. Scherer R., 2006. Boosting Ensemble of Relational Neuro-fuzzy Systems. *Lecture Notes in Artificial Intelligence*, vol. 4029, pp. 306-313.
36. Scherer R., Rutkowski L., 2003. A fuzzy relational system with linguistic antecedent certainty factors, *6th International Conference on Neural Networks and Soft Computing, Zakopane, Poland. Book Series Advances In Soft Computing*, pp. 563-569.
37. Scherer R., Rutkowski L., 2005. Connectionist fuzzy relational systems, *9th International Conference on Neural Information and Processing; 4th Asia-Pacific Conference on Simulated Evolution and Learning; 1st International Conference on Fuzzy Systems and Knowledge Discovery, Singapore. Computational Intelligence for Modelling and Prediction Book Series: Studies In Computational Intelligence*, vol. 2, pp.35-47.
38. Scherer R., Rutkowski L., 2004. Neuro-fuzzy relational classifiers, *Lecture Notes in Artificial Intelligence*, vol. 3070, pp. 376-380.
39. Szaleniec M., Goclon J., Witko M., Tadeusiewicz R., 2006. Application of artificial neural networks and DFT-based parameters for prediction of reaction kinetics of ethylbenzene dehydrogenase. *Journal of Computer-Aided Molecular Design*, vol. 20, no. 3, pp. 145-157.
40. Zalasiński M., Cpałka K., 2011, *A new method of on-line signature verification using a flexible fuzzy one-class classifier*, Selected Topics in Computer Science Applications, EXIT, pp. 38-53.
41. Zalasiński M., Cpałka K., 2012, *Novel algorithm for the on-line signature verification*, Lecture Notes in Artificial Intelligence, vol. 7268, pp. 362-367.
42. Żebrowski J. and Grudziński K., 2005, Observations and modelling of unusual patterns in human heart rate variability. *Acta Physica Polonica B*, vol. 36, no. 5, pp. 1881-1895.