EXTRACTION OF FUZZY RULES USING DETERMINISTIC ANNEALING INTEGRATED WITH ε-INSENSITIVE LEARNING

ROBERT CZABAŃSKI

Institute of Electronics
Silesian University of Technology
ul. Akademicka 16, 44–100 Gliwice, Poland
e-mail: robert.czabanski@polsl.pl

A new method of parameter estimation for an artificial neural network inference system based on a logical interpretation of fuzzy if-then rules (ANBLIR) is presented. The novelty of the learning algorithm consists in the application of a deterministic annealing method integrated with ε-insensitive learning. In order to decrease the computational burden of the learning procedure, a deterministic annealing method with a "freezing" phase and ε-insensitive learning by solving a system of linear inequalities are applied. This method yields an improved neuro-fuzzy modeling quality in the sense of an increase in the generalization ability and robustness to outliers. To show the advantages of the proposed algorithm, two examples of its application concerning benchmark problems of identification and prediction are considered.

Keywords: fuzzy systems, neural networks, neuro-fuzzy systems, rules extraction, deterministic annealing, ε-insensitive learning

1. Introduction

A fundamental problem while designing fuzzy systems is the determination of their rule bases, which consist of sets of fuzzy conditional statements. Because there is no standard method of expert knowledge acquisition in the process of determining fuzzy if-then rules, automatic methods of rule generation are intensively investigated. A set of fuzzy conditional statements may be obtained automatically from numerical data describing input/output system characteristics. A number of fuzzy rules extraction procedures use the learning capabilities of artificial neural networks to solve this task (Mitra and Hayashi, 2000). The integration of neural networks and fuzzy models leads to the so-called neuro-fuzzy systems. Neuro-fuzzy systems can be represented as radial basis function networks because of their mutual functional equivalence (Jang and Sun, 1993). This quality resulted in the construction of the Adaptive Network based Fuzzy Inference System (ANFIS) (Jang, 1993), which is equivalent to the Takagi-Sugeno-Kang (TSK) type of fuzzy systems. A way of improving the interpretability of TSK fuzzy models by combining the global and local learning processes was presented by Yen *et al.* (1998). A similar approach was described by Rose *et al.* (Rao *et al.*, 1997; 1999; Rose, 1991; 1998). They proposed a deterministic annealing (DA) optimization method that makes it possible to improve the estimation quality of radial function parameters. Another fuzzy inference system which is equivalent to a radial basis function network, i.e., the Artificial Neural Network Based on Fuzzy Inference System (ANNBFIS) was presented by Czogała and Łęski (1996; 1999). Its novelty consisted in using parameterized consequents in fuzzy if-then rules. The equivalence of approximate reasoning results using logical and conjunctive interpretations of if-then rules which occurs in some circumstances was shown in (Czogała and Łęski, 1999; 2001). This observation led to a more generalized structure of the ANNBFIS–ANBLIR (Artificial neural Network Based on Logical Interpretation of fuzzy if-then Rules), a computationally effective system with parameterized consequents based on both conjunctive and logical interpretations of fuzzy rules (Czogała and Łęski, 1999). The ANBLIR system can be successfully applied to solve many practical problems such as classification, control, digital channel equalization, pattern recognition, prediction, signal compression and system identification (Czogała and Łęski, 1999). Its original learning procedure uses a combination of steepest descent optimization and the least-squares method. However, it may produce a local minimum in the case of a multimodal criterion function. Therefore, several modifications of the learning algorithm were proposed (Czabański, 2003). One of them uses a deterministic annealing method adopted in the ANBLIR system instead of the steepest descent procedure.

Neuro-fuzzy modeling has an intrinsic inconsistency (Łęski, 2003b): it may perform inference tolerant of imprecision but its learning methods are intolerant of imprecision. An approach to fuzzy modeling tolerant of imprecision, called ε-insensitive learning, was described in (Łęski, 2002; 2003a; 2003b). It leads to a model with a minimal Vapnik-Chervonenkis dimension (Vapnik, 1999), which results in an improved generalization ability of the neuro-fuzzy system (Łęski, 2002; 2003a; 2003b). Moreover, ε-insensitive learning methods lead to satisfactory learning results despite the presence of outliers in the training set (Łęski, 2002; 2003a; 2003b).

In this work, a new learning procedure of the ANBLIR is proposed. Its novelty consists in the application of a deterministic annealing method integrated with ε-insensitive learning. In order to reduce the computational burden of the learning procedure, a deterministic annealing method with a "freezing" phase (DAF) and ε-insensitive Learning by Solving a System of Linear Inequalities (εLSSLI) are employed. To show the validity of the proposed algorithm, two benchmark examples of its application are shown. We consider the system identification problem based on Box and Jenkins's data (1976), and the prediction example using Weigend's sunspots data (Weigend *et al.*, 1990).

The structure of the paper is as follows: In Section 2, the ANBLIR neuro-fuzzy system is presented. Section 3 introduces a deterministic annealing method adopted to the neuro-fuzzy modeling problem. In Section 4, a description of ε-insensitivity learning of the neuro-fuzzy system with parameterized consequents is given. The ε-insensitivity learning problem can be solved by means of the εLSSLI method. In Section 5, a hybrid learning algorithm that integrates the DAF method with the εLSSLI procedure is shown. The numerical examples are given in Section 6. Section 7 concludes the paper.

2. Neuro-Fuzzy System with Parameterized Consequents

The ANBLIR is a fuzzy system with parameterized consequents that generates inference results based on fuzzy if-then rules. Every fuzzy conditional statement from its rule base may be written down in the following form (Czogała and Łęski, 1999):

$$R^{(i)}: \textbf{ if } \underset{j=1}{\overset{t}{\textbf{and}}} \left(x_{0j} \textbf{ is } A_j^{(i)}\right) \textbf{ then } Y \textbf{ is } B^{(i)}\left(y, \underline{x}_0\right), \quad \forall\, i = 1, 2, \ldots, I, \quad (1)$$

where I denotes the number of fuzzy if-then rules, t is the number of inputs, x_{0j} is the j-th element of the input vector $\underline{x}_0 = \left[x_{01}, x_{02}, \ldots, x_{0t}\right]^T$, Y is the output linguistic variable of the system, $A_j^{(i)}$ and $B^{(i)}\left(y, \underline{x}_0\right)$ are linguistic values of fuzzy sets in antecedents and consequents, respectively.

The fuzzy sets of linguistic values in rule antecedents have Gaussian membership functions, and the linguistic connective "**and**" of multi-input rule predicates is represented by the algebraic product t-norm. Consequently, the firing strength of the i-th rule of the ANBLIR system can be written in the following form (Czogała and Łęski, 1999):

$$F^{(i)}\left(\underline{x}_0\right) = \prod_{j=1}^{t} A_j^{(i)}\left(x_{0j}\right) = \exp\left[-\frac{1}{2}\sum_{j=1}^{t}\left(\frac{x_{0j}-c_j^{(i)}}{s_j^{(i)}}\right)^2\right], \quad \forall\, i = 1, 2, \ldots, I, \quad (2)$$

where $c_j^{(i)}$ and $s_j^{(i)}$ for $i = 1, 2, \ldots, I$, and $j = 1, 2, \ldots, t$ are membership function parameters, centers and dispersions, respectively.

The consequents of ANBLIR fuzzy rules have symmetric triangular membership functions. They can be defined using two parameters: the width of the triangle base $w^{(i)}$ and the location of the gravity center $y^{(i)}\left(\underline{x}_0\right)$, which can be determined on the basis of linear combinations of fuzzy system inputs:

$$y^{(i)}\left(\underline{x}_0\right) = p_0^{(i)} + p_1^{(i)} x_{01} + \cdots + p_t^{(i)} x_{0t} = \underline{p}^{(i)T} \underline{x}_0', \quad (3)$$

where $\underline{x}_0' = \left[1, x_{01}, x_{02}, \ldots, x_{0t}\right]^T$ is the extended input vector. The above dependency formulates the so-called parameterized (moving) consequent (Czogała and Łęski, 1996; 1999).

The kind of operations executed during the inference process and therefore the shapes of membership functions of the conclusions obtained after the inference process depend on the chosen way of interpreting if-then rules. The ANBLIR permits both conjunctive and logical interpretations of fuzzy rules. Consequently, the general form of the resulting conclusion of the i-th rule can be written down as (Czogała and Łęski, 1999):

$$B^{(i)\prime}\left(y, \underline{x}_0\right) = \Psi\left[F^{(i)}\left(\underline{x}_0\right), B^{(i)}\left(y, \underline{x}_0\right)\right], \quad (4)$$

where Ψ stands for a fuzzy implication (for the logical interpretation of if-then rules) or a t-norm (for the conjunctive interpretation of if-then rules). The final output fuzzy set of the neuro-fuzzy system is derived from the aggregation process. Throughout the paper, we use the normalized arithmetic mean as the aggregation,

$$B'\left(y\right) = \frac{1}{I}\sum_{i=1}^{I} B^{(i)\prime}\left(y, \underline{x}_0\right). \quad (5)$$

The resulting fuzzy set has a non-informative part, i.e., there are elements of s fuzzy set $y \in \mathbb{Y}$ whose membership values are equal in the whole space $\mathbb{Y}$. Therefore, the following modified indexed center of the gravity defuzzifier (MICOG) is used (Czogała and Łęski, 1999):

$$y_0 = \frac{\displaystyle\int y\ \left(B'\left(y\right)-\alpha\right)\mathrm{d}y}{\displaystyle\int \left(B'\left(y\right)-\alpha\right)\mathrm{d}y}, \tag{6}$$

where y_0 denotes the crisp output value and $\alpha \in [0,1]$ describes the uncertainty attendant upon information. Consequently, the final crisp output value of the fuzzy system with parameterized consequents can be evaluated from the following formula:

$$\begin{aligned} y_0 &= \frac{\displaystyle\int \frac{y}{I}\sum_{i=1}^{I}\left(B^{(i)\prime}\left(y,\underline{x}_0\right)-\alpha_i\right)\mathrm{d}y}{\displaystyle\int \frac{1}{I}\sum_{i=1}^{I}\left(B^{(i)\prime}\left(y,\underline{x}_0\right)-\alpha_i\right)\mathrm{d}y} \\ &= \frac{\displaystyle\sum_{i=1}^{I}\int y\ \left(B^{(i)\prime}\left(y,\underline{x}_0\right)-\alpha_i\right)\mathrm{d}y}{\displaystyle\sum_{i=1}^{I}\int \left(B^{(i)\prime}\left(y,\underline{x}_0\right)-\alpha_i\right)\mathrm{d}y}. \end{aligned} \tag{7}$$

The gravity center of the rule consequents is defined as

$$y^{(i)}\left(\underline{x}_0\right) = \frac{\displaystyle\int y\ \left(B^{(i)\prime}\left(y,\underline{x}_0\right)-\alpha_i\right)\mathrm{d}y}{\displaystyle\int \left(B^{(i)\prime}\left(y,\underline{x}_0\right)-\alpha_i\right)\mathrm{d}y}. \tag{8}$$

Substituting (8) into (7) yields (Czogała and Łęski, 1999):

$$y_0 = \frac{\displaystyle\sum_{i=1}^{I}\left[\int \left(B^{(i)\prime}\left(y,\underline{x}_0\right)-\alpha_i\right)\mathrm{d}y\right] y^{(i)}\left(\underline{x}_0\right)}{\displaystyle\sum_{i=1}^{I}\int \left(B^{(i)\prime}\left(y,\underline{x}_0\right)-\alpha_i\right)\mathrm{d}y}. \tag{9}$$

The integral $\int\left(B^{(i)\prime}\left(y,\underline{x}_0\right)-\alpha_i\right)\mathrm{d}y$ defines the area of the region under the curve corresponding to the membership function of the i-th rule consequent after removing the non-informative part. For a symmetric triangular function it is a function of the firing strength of the rule $F^{(i)}\left(\underline{x}_0\right)$ and the width of the triangle base $w^{(i)}$:

$$\int \left(B^{(i)\prime}\left(y,\underline{x}_0\right)-\alpha_i\right)\mathrm{d}y = g\left(F^{(i)}\left(\underline{x}_0\right),w^{(i)}\right). \tag{10}$$

The function $g\left(F^{(i)}\left(\underline{x}_0\right),w^{(i)}\right)$ depends on the interpretation of fuzzy conditional statements we use. The respective formulas for selected fuzzy implications are tabulated in Table 1. For notational simplicity, we use $B \triangleq B^{(i)}\left(y,\underline{x}_0\right), F \triangleq F^{(i)}\left(\underline{x}_0\right)$ and $w \triangleq w^{(i)}$. It was proven (Czogała and Łęski, 1999; 2001) that the neuro-fuzzy system with parameterized consequents based on Łukasiewicz and Reichenbach's implications produces inference results equivalent to those obtained from Mamdani and Larsen's fuzzy relations, respectively.

Finally, the crisp output value of the fuzzy system can be written in the following form:

$$y_0 = \sum_{i=1}^{I} G^{(i)}\left(\underline{x}_0\right)\ y^{(i)}\left(\underline{x}_0\right), \tag{11}$$

where

$$G^{(i)}\left(\underline{x}_0\right) = \frac{g\left(F^{(i)}\left(\underline{x}_0\right),w^{(i)}\right)}{\sum\limits_{k=1}^{I} g\left(F^{(k)}\left(\underline{x}_0\right),w^{(k)}\right)}. \tag{12}$$

The fuzzy system with parameterized consequents can be treated as a radial basis function neural network (Czogała and Łęski, 1999). Consequently, unknown neuro-fuzzy system parameters can be estimated using learning algorithms of neural networks. Several solutions to this problem were proposed in the literature (Czabański, 2003; 2005; Czogała and Łęski, 1996; 1999; Łęski, 2002; 2003a; 2003b). In this work, a new hybrid learning procedure which connects a deterministic annealing method and the ε-insensitive learning algorithm by solving a system of linear inequalities is presented. In the following, we assume that we have N examples of the input vectors $\underline{x}_0\left(n\right) \in \mathbb{R}^t$ and the same number of the known output values $t_0\left(n\right) \in \mathbb{R}$ which form the training set.

3. Deterministic Annealing

Our goal is the extraction of a set of fuzzy if-then rules that represent the knowledge of the phenomenon under consideration. The extraction process consists in the estimation of membership function parameters of both antecedents and consequents $\underline{\zeta} = \{c_j^{(i)}, s_j^{(i)}, p_j^{(i)}, w^{(i)}\}$, $\forall i = 1,2,\ldots,I$, $\forall j = 1,2,\ldots,t$. The number of rules I is also unknown. We assume that it is preset arbitrarily. The number of antecedents t is defined by the size of the input training vector directly. To increase the ability to avoid many local minima that interfere with the steepest descent method used in the original ANBLIR learning algorithm, we employ the technique of deterministic annealing (Rao *et al.*, 1997; 1999; Rose, 1991; 1998) adapted for training the neuro-fuzzy system with parameterized consequents. However, it is not guaranteed that a global optimum of the cost will be found (Rao *et al.*, 1999). Deterministic annealing (DA) is a simulated annealing (Metropolis *et al.*, 1953; Kirkpatrick *et al.*, 1983) based method which replaces the computationally intensive stochastic simulation by a straightforward deterministic optimization of the modeled system error energy (Rao *et al.*, 1997). The algorithm consists in the minimization

Table 1. Function $g\left(F^{(i)}\left(\underline{x}_0\right), w^{(i)}\right)$ for selected fuzzy implications.

Fuzzy implication $\Psi\left[F, B\right]$	α	$g\left(F, w\right)$
Fodor $\begin{cases} 1, & \text{if } F \le B, \\ \max\left(1-F, B\right), & \text{otherwise,} \end{cases}$	$1-F$	$\begin{cases} \frac{w}{2}\left(1-2F+2F^2\right), & F \ge \frac{1}{2}, \\ wF\left(1-F\right), & F < \frac{1}{2}, \end{cases}$
Gödel $\begin{cases} 1, & \text{if } F \le B, \\ B, & \text{otherwise,} \end{cases}$	0	$\frac{w}{2}\left(2-2F+F^2\right),$
Gougen $\min\left(\frac{B}{F}, 1\right), F \neq 0,$	0	$\frac{w}{2}\left(2-F\right),$
Kleene-Dienes $\max(1-F, B),$	$1-F$	$\frac{w}{2}F^2,$
Łukasiewicz $\min(1-F+B, 1),$	$1-F$	$\frac{w}{2}F\left(2-F\right),$
Reichenbach $1-F+FB,$	$1-F$	$\frac{w}{2}F,$
Rescher $\begin{cases} 1, & \text{if } F \le B, \\ 0, & \text{otherwise,} \end{cases}$	0	$w\left(1-F\right),$
Zadeh $\max\{1-F, \min(F, B)\},$	$1-F$	$\begin{cases} \frac{w}{2}\left(2F-1\right), & F \ge \frac{1}{2}, \\ 0, & F < \frac{1}{2}. \end{cases}$

of the squared-error cost

$$E = \sum_{n=1}^{N} E_n = \sum_{n=1}^{N} \frac{1}{2}\left(t_0\left(n\right) - y_0\left(n\right)\right)^2, \qquad (13)$$

while simultaneously controlling the entropy level of a solution.

Equation (11) defines the neuro-fuzzy system as a mixture of experts (models). Its global output is expressed as a linear combination of I outputs $y^{(i)}\left(\underline{x}_0\right)$ of the local models, each represented by a single fuzzy if-then rule. The weight $G^{(i)}\left(\underline{x}_0\right)$ may be interpreted as the possibility of the association of the i-th local model with the input data $\underline{x}_0$. For every local model we have to determine a set of its parameters $\underline{p}^{(i)}$as well as assignments $G^{(i)}\left(\underline{x}_0\right)$ that minimize the criterion (13). The randomness of the association can be quantified using the Shannon entropy:

$$S = -\sum_{n=1}^{N}\sum_{i=1}^{I} G^{(i)}\left(\underline{x}_0\left(n\right)\right) \log G^{(i)}\left(\underline{x}_0\left(n\right)\right). \qquad (14)$$

In the deterministic annealing method the objective is the minimization of the cost E with an imposed level of entropy S_0:

$$\min E \text{ subject to } S = S_0. \qquad (15)$$

Constrained optimization is equivalent to the unconstrained minimization of the Lagrangian (Rao *et al.*, 1997):

$$L = E - T\,\left(S - S_0\right), \qquad (16)$$

where T is the Lagrange multiplier.

A connection between the above equation and the annealing of solids is essential here. The quantity L can be identified as the Helmholtz free energy of a physical system with the "energy" E, entropy S and "temperature" T (Rao *et al.*, 1997).

The DA procedure involves a series of iterations while the randomness level is gradually reduced. To achieve the global optimum of the cost, the simulated annealing method is used. The algorithm starts at a high level of the pseudotemperature T and tracks the solution for continuously reduced values of T. For high values of the pseudotemperature, the minimization of the Lagrange function L amounts to entropy maximization of associating data and models. In other words, we seek a set of local models that are equally associated with each input data point—the set of local models which cooperate to produce a desired output. It can be noticed that, as $T \to \infty$, we get the uniform distribution of $G^{(i)}\left(\underline{x}_0\right)$ and therefore, identical local models. As the pseudotemperature is lowered, more emphasis is placed on reducing the square error. It also results in a decrease in entropy. We get more and more competitive local models, each associated with given data more closely. We cross gradually from cooperation to competition. Finally, at $T = 0$, the optimization is conducted regardless of the entropy level and the cost is minimized directly.

The pseudotemperature reduction procedure is determined by the annealing schedule function $q\left(T\right)$. In the sequel, we use the following decremental rule:

$$T \leftarrow q\,T, \tag{17}$$

where $q \in (0, 1)$ is a preset parameter.

The deterministic annealing algorithm can be summarized as follows (Rao *et al.*, 1997):

1. Set parameters: an initial solution $\underline{\zeta}$, an initial pseudotemperature $T_{\max}$, a final pseudotemperature $T_{\min}$ and an annealing schedule function $q\left(T\right)$. Set $T = T_{\max}$.
2. Minimize the Lagrangian L:
$$\frac{\partial L}{\partial \underline{\zeta}} = \frac{\partial E}{\partial \underline{\zeta}} - T\frac{\partial S}{\partial \underline{\zeta}}. \tag{18}$$
3. Decrement the pseudotemperature according to the annealing schedule.
4. If $T < T_{\min}$, then STOP. Otherwise, go to Step 2.

At each level of the pseudotemperature, we minimize the Lagrangian iteratively using the gradient descent method in the parameter space. The parameters of the neuro-fuzzy system are given by

$$\underline{\zeta}\left(k+1\right) = \underline{\zeta}\left(k\right) - \eta\left.\frac{\partial L}{\partial \underline{\zeta}}\right|_{\underline{\zeta}=\underline{\zeta}(k)}, \tag{19}$$

where k denotes the iteration index and η is the learning rate, which can be further expressed using the formula proposed by Jang (1993):

$$\eta = \frac{\eta_{\text{ini}}}{\sqrt{\sum\limits_{i=1}^{n_i}\left(\dfrac{\partial L}{\partial \zeta_i}\right)^2_{\zeta_i=\zeta_i(k)}}}. \tag{20}$$

Here η_{ini} denotes the initial (constant) stepsize, n_i is the number of optimized parameters: for the parameters of the membership function of fuzzy sets in the antecedents $n_i = 2It$, for the parameters of the linear function in the consequents $n_i = I\left(t+1\right)$, and for the triangle base widths $n_i = I$.

For the notational simplicity of the gradient formulas, we introduce the following symbols:

$$\begin{aligned}\Xi^{(i)}\left(\underline{x}_0\left(n\right)\right) &= \left[y_0\left(n\right) - t_0\left(n\right)\right]\, y^{(i)}\left(\underline{x}_0\left(n\right)\right)\\ &\quad + T\,\log G^{(i)}\left(\underline{x}_0\left(n\right)\right),\end{aligned} \tag{21}$$

$$\overline{\Xi}\left(\underline{x}_0\left(n\right)\right) = \sum_{i=1}^{I} G^{(i)}\left(\underline{x}_0\left(n\right)\right)\,\Xi^{(i)}\left(\underline{x}_0\left(n\right)\right). \tag{22}$$

Then the partial derivatives $\partial L/\partial \underline{\zeta}$ with respect to the unknown parameters may be expressed as

$$\begin{aligned}\frac{\partial L}{\partial c_j^{(i)}} &= \frac{1}{\left(s_j^{(i)}\right)^2}\sum_{n=1}^{N}\left[x_{j0}\left(n\right) - c_j^{(i)}\right]\\ &\quad\times \frac{F^{(i)}(\underline{x}_0(n))}{g(F^{(i)}(\underline{x}_0(n)), w^{(i)})}\frac{\partial g(F^{(i)}(\underline{x}_0(n)), w^{(i)})}{\partial F^{(i)}(\underline{x}_0(n))}\\ &\quad\times G^{(i)}\left(\underline{x}_0\left(n\right)\right)\left[\Xi^{(i)}\left(\underline{x}_0\left(n\right)\right) - \overline{\Xi}\left(\underline{x}_0\left(n\right)\right)\right],\end{aligned} \tag{23}$$

$$\begin{aligned}\frac{\partial L}{\partial s_j^{(i)}} &= \frac{1}{\left(s_j^{(i)}\right)^3}\sum_{n=1}^{N}\left[x_{j0}\left(n\right) - c_j^{(i)}\right]^2\\ &\quad\times \frac{F^{(i)}\left(\underline{x}_0\left(n\right)\right)}{g\left(F^{(i)}\left(\underline{x}_0\left(n\right)\right), w^{(i)}\right)}\frac{\partial g\left(F^{(i)}\left(\underline{x}_0\left(n\right)\right), w^{(i)}\right)}{\partial F^{(i)}\left(\underline{x}_0\left(n\right)\right)}\\ &\quad\times G^{(i)}\left(\underline{x}_0\left(n\right)\right)\left[\Xi^{(i)}\left(\underline{x}_0\left(n\right)\right) - \overline{\Xi}\left(\underline{x}_0\left(n\right)\right)\right],\end{aligned} \tag{24}$$

$$\begin{aligned}\frac{\partial L}{\partial p_j^{(i)}} &= \frac{\partial E}{\partial p_j^{(i)}}\\ &= \begin{cases}\left[y_0\left(n\right) - t_0\left(n\right)\right]\sum\limits_{n=1}^{N} G^{(i)}\left(\underline{x}_0\left(n\right)\right)\, x_{j0}\left(n\right) & \text{for } j \neq 0,\\ \left[y_0\left(n\right) - t_0\left(n\right)\right]\sum\limits_{n=1}^{N} G^{(i)}\left(\underline{x}_0\left(n\right)\right) & \text{for } j = 0,\end{cases}\end{aligned} \tag{25}$$

$$\begin{aligned}\frac{\partial L}{\partial w^{(i)}} &= \sum_{n=1}^{N}\frac{1}{g\left(F^{(i)}\left(\underline{x}_0\left(n\right)\right), w^{(i)}\right)}\\ &\quad\times \frac{\partial g\left(F^{(i)}\left(\underline{x}_0\left(n\right)\right), w^{(i)}\right)}{\partial w^{(i)}}\\ &\quad\times G^{(i)}\left(\underline{x}_0(n)\right)\left[\Xi^{(i)}\left(\underline{x}_0(n)\right) - \overline{\Xi}\left(\underline{x}_0(n)\right)\right].\end{aligned} \tag{26}$$

In the original ANBLIR learning method, the parameters of the consequents $\underline{p}^{(i)}$ were estimated using the least-squares (LS) method (Czogała and Łęski, 1999). It accelerates the learning convergence (Czogała and Łęski, 1999). A novel, impecision-tolerant method for estimating the parameters of consequents (ε-insensitive learning) was presented in (Łęski, 2002; 2003a; 2003b). It improves the generalization ability of the neuro-fuzzy system compared with the LS algorithm. Three different approaches to solve the ε-insensitive learning problem were proposed in (Łęski, 2002; 2003a; 2003b) as well. In this work we use ε-insensitive Learning by Solving a System of Linear Inequalities (εLSSLI) because of its lower computational burden which is approximately three times higher in comparison with imprecision-intolerant learning with LS (Łęski, 2003b). εLSSLI can be solved globally and locally (Łęski, 2003b). In what follows, we assume the local solution. This enables us to tune every local model (rule) independently. Its integration with the deterministic annealing procedure is described in the sequel.

4. ε-Insensitive Learning with εLSSLI Solution

Neuro-fuzzy systems usually have an intrinsic inconsistency (Łęski, 2003b): they may perform approximate reasoning but simultaneously their learning methods are intolerant of imprecision. In a typical neuro-fuzzy learning algorithm, only the perfect match of the fuzzy model and the modeled phenomenon results in the zero error value. Additionally, the zero loss is usually obtained through a high complexity of the model. However, according to statistical learning theory (Vapnik, 1998), we should find the simplest model from among all which accurately represent the data. It is inspired by the well-known principle of Occam's razor, which essentially states that the simplest explanation is best. An imprecision-tolerant approach with the control of model complexity called ε-insensitive learning was presented in (Łęski, 2002; 2003a; 2003b). It is based on the ε-insensitive loss function (Vapnik, 1998):

$$
\begin{aligned}
E_n &= \rceil t_0(n) - y_0(n) \lceil_{\varepsilon} \\
&= \begin{cases} 0 & \text{if } |t_0(n) - y_0(n)| \le \varepsilon, \\ |t_0(n) - y_0(n)| - \varepsilon & \text{if } |t_0(n) - y_0(n)| > \varepsilon. \end{cases}
\end{aligned}
\tag{27}
$$

The symbol ε represents the limiting value of imprecision tolerance. If the difference between the modeled and desired outputs is less than ε, then the zero loss is obtained. As was shown in (Łęski, 2002; 2003a; 2003b), ε-insensitive learning may be used for estimating the parameters of the consequents of the ANBLIR system.

ε-Insensitive learning with the control of model complexity may be formulated as the minimization of the following ε-insensitive criterion function (Łęski, 2003b):

$$
\mathcal{I}^{(i)}\left(\underline{p}^{(i)}\right) = \left\rceil \underline{t}_0 - \underline{X}_0^{'} \underline{p}^{(i)} \right\lceil_{\varepsilon, \underline{G}} + \frac{\tau}{2} \underline{p}^{(i)T} \underline{\widetilde{I}} \underline{p}^{(i)}, \tag{28}
$$

where $\underline{t}_0 = [t_0(1), t_0(2), \dots, t_0(N)]^T$, $\underline{X}_0^{'} = [\underline{x}_0^{'}(1), \underline{x}_0^{'}(2), \dots, \underline{x}_0^{'}(N)]^T$, $\underline{\widetilde{I}} = \mathrm{diag}([0, \underline{1}_{t\times 1}^T])$, $\underline{1}_{t\times 1}$ is a $(t \times 1)$-dimensional vector with all entries equal to 1, $\underline{G} = [G^{(i)}(\underline{x}_0(1)), G^{(i)}(\underline{x}_0(2)), \dots, G^{(i)}(\underline{x}_0(N))]^T$ and $\rceil \cdot \lceil_{\varepsilon, \underline{G}}$ denotes the weighted Vapnik loss function defined as

$$
\begin{aligned}
&\left\rceil \underline{t}_0 - \underline{X}_0^{'} \underline{p}^{(i)} \right\lceil_{\varepsilon, \underline{G}} \\
&\quad = \sum_{n=1}^{N} G^{(i)}\left(\underline{x}_0(n)\right) \left\rceil t_0(n) - \underline{p}^{(i)T} \underline{x}_0'(n) \right\lceil_{\varepsilon}.
\end{aligned}
\tag{29}
$$

The second term in (28) is associated with the minimization of the Vapnik-Chervonenkis dimension (Vapnik, 1998) and, therefore, the minimization of model complexity. The regularization parameter $\tau \ge 0$ controls the trade-off between model matching to the training data and the model generalization ability (Łęski, 2003b). Larger τ results in an increase in the model generalization ability. The above formula is called the weighted (or fuzzy) ε-insensitive estimator with complexity control (Łęski, 2003b).

The ε-insensitive learning error measure $\rceil \underline{t}_0 - \underline{X}_0^{'} \underline{p}^{(i)} \lceil_{\varepsilon}$ can be equivalently rewritten using two systems of inequalities (Łęski, 2003b): $\underline{X}_0^{'} \underline{p}^{(i)} + \varepsilon \underline{1}_{N\times 1} > \underline{t}_0$ and $\underline{X}_0^{'} \underline{p}^{(i)} - \varepsilon \underline{1}_{N\times 1} < \underline{t}_0$. In practice, not all inequalities from this system are satisfied for every datum from the learning set (i.e., not all data fall into the insensitivity region). The solution method that enables us to maximize the fulfilment degree of the system of inequalities was presented in (Łęski, 2003b).

If we introduce the extended versions of $\underline{X}_0^{'}$ and $\underline{t}_0$ defined as $\underline{X}_{0e}^{'} = [\underline{X}_0^{'T} \vdots - \underline{X}_0^{'T}]^T$ and $\underline{t}_{0e} = [t_0(1) - \varepsilon,\ t_0(2) - \varepsilon, \dots, t_0(N) - \varepsilon, -t_0(1) - \varepsilon, -t_0(2) - \varepsilon, \dots, -t_0(N) - \varepsilon]^T$, then the above systems of two inequalities can be written down as one, namely, $\underline{X}_{0e}^{'} \underline{p}^{(i)} - \underline{t}_{0e} > \underline{0}$. We can solve it using the system of equalities (Łęski, 2003b): $\underline{X}_{0e}^{'} \underline{p}^{(i)} - \underline{t}_{0e} = \underline{b}$, where $\underline{b} > \underline{0}$ is an arbitrary positive vector. Now we can define the error vector (Łęski, 2003b): $\underline{e} = \underline{X}_{0e}^{'} \underline{p}^{(i)} - \underline{t}_{0e} - \underline{b}$. If the n-th datum falls in the insensitivity region, then the n-th and $2n$-th error components are positive. Accordingly, they can be set to zero by increasing the respective components of $\underline{b}$. If the n-th datum falls outside the insensitivity region, then the n-th and $2n$-th error components are negative. In this case, it is impossible to set the error values to zero by changing (decreasing) the respective components b_n

(b_{2n}) because they have to fulfil the conditions $b_n > 0$ ($b_{2n} > 0$). Hence, the non-zero error values correspond only to data outside the insensitivity region. Now, we can approximate the minimization problem (28) with the following one (Łęski, 2003b):

$$\begin{aligned}\min_{\underline{p}^{(i)}\in\mathbb{R}^{t+1},\,\underline{b}>0} &\mathcal{I}^{(i)}\left(\underline{p}^{(i)},\underline{b}\right)\\ &= \left(\underline{X}_{0e}^{'}\underline{p}^{(i)} - \underline{t}_{0e} - \underline{b}\right)^T \underline{G}_e \left(\underline{X}_{0e}^{'}\underline{p}^{(i)} - \underline{t}_{0e} - \underline{b}\right)\\ &\quad + \frac{\tau}{2}\,\underline{p}^{(i)T}\,\underline{\widetilde{I}}\,\underline{p}^{(i)}, \end{aligned} \qquad (30)$$

where $\underline{G}_e = \mathrm{diag}([\underline{G}^T, \underline{G}^T]^T)$.

The above criterion is an approximation of (28) because the square error is used rather than the absolute one. It is due to mathematical simplicity. A learning algoritm for the absolute error can be obtained by selecting the following diagonal weight matrix: $\underline{D}_e = \mathrm{diag}(G^{(i)}(\underline{x}_0(1))/|e_1|,\ G^{(i)}(\underline{x}_0(2))/|e_2|, \ldots, G^{(i)}(\underline{x}_0\,(N))/|e_N|,\quad G^{(i)}(\underline{x}_0\,(1))/|e_{N+1}|, \ldots, G^{(i)}\left(\underline{x}_0\,(N)\right)/|e_{2N}|)$, where e_i is the i-th component of the error vector, instead of $\underline{G}_e$.

The optimal solution is given by differentiating (30) with respect to $\underline{p}^{(i)}$ and $\underline{b}$, and equating the result to zero. After introducing the absolute error criterion, we get the following system of equations (Łęski, 2003b):

$$\begin{cases}\underline{p}^{(i)} = \left(\underline{X}_{0e}^{'T}\underline{D}_e\underline{X}_{0e}^{'}+\frac{\tau}{2}\underline{\widetilde{I}}\right)^{-1}\underline{X}_{0e}^{'T}\underline{D}_e\left(\underline{t}_{0e}+\underline{b}\right),\\ \quad\underline{e} = \underline{X}_{0e}^{'}\,\underline{p}^{(i)} - \underline{t}_{0e} - \underline{b} = \underline{0}.\end{cases} \qquad (31)$$

The vector $\underline{b}$ is called the margin vector (Łęski, 2003b) because its components determine the distances between the data and the insensitivity region. From the first equation of (31), we can see that the solution vector $\underline{p}^{(i)}$ depends on the margin vector. If a datum lies in the insensitivity region, then the zero error can be obtained by increasing the corresponding distance. Otherwise, the error can be decreased only by decreasing the corresponding component of the margin vector. The only way to prevent the margin vector $\underline{b}$ from converging to zero is to start with $\underline{b} > \underline{0}$ and not allow any of its components to decrease (Łęski, 2003b). This problem can be solved using the procedure of ε-insensitive Learning by Solving a System of Linear Inequalities (εLSSLI) (Łęski, 2003b), which is an extended version of Ho and Kashyap's (1965; 1966) iterative algorithm. In εLSSLI, margin vector components are modified by the corresponding error vector components only if the change results in an increase in the margin vector components (Łęski, 2003b):

$$\underline{b}^{[k+1]} = \underline{b}^{[k]} + \rho\left(\underline{e}^{[k]} + \left|\underline{e}^{[k]}\right|\right), \qquad (32)$$

where $\rho > 0$ is a parameter and $[k]$ denotes the iteration index. The $\underline{p}^{(i)}$ vector is obtained from the first equation of (31) (Łęski, 2003b):

$$\underline{p}^{(i)[k]} = \left(\underline{X}_{0e}^{'T}\underline{D}_e^{[k]}\underline{X}_{0e}^{'} + \frac{\tau}{2}\underline{\widetilde{I}}\right)^{-1}\underline{X}_{0e}^{'T}\underline{D}_e^{[k]}\left(\underline{t}_{0e} + \underline{b}^{[k]}\right), \qquad (33)$$

and the error vector $\underline{e}$ from the second equation of (31):

$$\underline{e}^{[k]} = \underline{X}_{0e}^{'}\,\underline{p}^{(i)[k]} - \underline{t}_{0e} - \underline{b}^{[k]}. \qquad (34)$$

Consequently, the εLSSLI algorithm can be summarized as follows (Łęski, 2003b):

1. Set the algorithm parameters $\varepsilon \geq 0, \tau \geq 0, 0 < \rho < 1$ and the iteration index $k = 1$. Calculate $\underline{D}_e^{[1]}$ and initialize the margin vector $\underline{b}^{[1]} > \underline{0}$.
2. Calculate $\underline{p}^{(i)[k]}$ according to (33).
3. Calculate $\underline{e}^{[k]}$ on the basis of (34).
4. Update $\underline{D}_e^{[k+1]}$ using $\underline{e}^{[k]}$.
5. Update the margin vector components according to (32).
6. If $\|\underline{b}^{[k+1]} - \underline{b}^{[k]}\| > \kappa$, where κ is a preset parameter or $k < k_{\varepsilon\,\max}$, then $k = k + 1$ and go to Step 2. Otherwise, STOP.

This procedure is based on the postulate that near an optimal solution the consecutive vectors of the minimizing sequence differ very little. It was proven (Łęski, 2003b) that for $0 < \rho < 1$ the above algorithm is convergent for any matrix $\underline{D}_e$.

5. Hybrid Learning Algorithm

The integration of the εLSSLI procedure with the deterministic annealing method leads to a learning algorithm where the parameters of fuzzy sets from the antecedents and consequents of fuzzy if-then rules are adjusted separately. The antecedent parameters $c_j^{(i)}, s_j^{(i)},\ i = 1, 2, \ldots, I,\ j = 1, 2, \ldots, t$, as well as the triangle base widths $w^{(i)},\ i = 1, 2, \ldots, I$ of fuzzy sets in the consequents are estimated by means of a deterministic annealing method, whereas the parameters of linear equations from the consequents $\underline{p}^{(i)},\ i = 1, 2, \ldots, I$, are adjusted using ε-insensitive learning and then tuned using the deterministic annealing procedure. We called the method "hybrid" as we used a mixture of two methods to estimate the $\underline{p}^{(i)}$ values. For decreasing the computational burden of the learning procedure, the deterministic annealing method with the "freezing" phase (DAF) can be applied (Rao *et al.*, 1999; Czabański, 2003). The "freezing" phase consists in the calculation of $\underline{p}^{(i)}$ using the εLSSLI procedure after every decreasing step of the pseudotemperature value while keeping $c_j^{(i)}, s_j^{(i)}$ and $w^{(i)}$ constant. Hybrid learning can be summarized as follows:

1. Set the parameters: an initial solution $\underline{\zeta}$, an initial pseudotemperature $T_{\max}$, a final pseudotemperature $T_{\min}$ and an annealing schedule function. Set $T = T_{\max}$.
2. Minimize the Lagrangian L using the steepest descent method (18).
3. Check the equilibrium

   $$|\delta S| = \left| \frac{S^{[k-1]} - S^{[k]}}{S^{[k-1]}} \right| > \delta$$

   or the iteration stopping condition $k \leq k_{\max}$, where k denotes the iteration index, δ is a preset parameter and $k_{\max}$ denotes the maximum number of iterations at a given level of the pseudotemperature. If one of them is fulfilled, go to Step 2.
4. Lower the pseudotemperature according to the annealing schedule.
5. Perform the "freezing" phase, i.e., estimate the parameters of linear equations from the consequents for all rules by means of the εLSSLI procedure.
6. If $T \geq T_{\min}$, go to Step 2.
7. Stop the algorithm.

Another problem is the estimation of the initial values of membership functions for antecedents. It can be solved by means of preliminary clustering of input training data (Czogała and Łęski, 1999). We use the fuzzy $c(I)$-means (FCM) (Bezdek, 1982) method for this task. The center and dispersion parameters of Gaussian membership functions can be initialized using the final FCM partition matrix (Czogała and Łęski, 1999):

$$c_j^{(i)} = \frac{\sum\limits_{n=1}^{N} (u_{in})^m \, x_{0j}(n)}{\sum\limits_{n=1}^{N} (u_{in})^m}, \qquad \forall\, 1 \leq i \leq I, \quad \forall\, 1 \leq j \leq t, \tag{35}$$

and

$$\left(s_j^{(i)}\right)^2 = \frac{\sum\limits_{n=1}^{N} (u_{in})^m \left(x_{0j}(n) - c_j^{(i)}\right)^2}{\sum\limits_{n=1}^{N} (u_{in})^m}, \qquad \forall\, 1 \leq i \leq I, \quad \forall\, 1 \leq j \leq t, \tag{36}$$

where u_{in} is the FCM partition matrix element and m is a weighted exponent ($m \in [1, \infty)$, usually $m = 2$).

6. Numerical Experiments

To validate the introduced hybrid method of extracting fuzzy if-then rules, two numerical experiments using benchmark databases were conducted. The first concerns a problem of system identification and the second deals with the prediction of sunspots. The purpose of these experiments was to verify the influence on the generalization ability of the neuro-fuzzy system with parameterized consequents of learning based on a combination of deterministic annealing with the "freezing" phase and the εLSSLI method.

The example of system identification is based on data originating from Box and Jenkins' work (1976). It concerns the identification of a gas oven. An input signal consists of measuring samples of methane flow $x(n)$ [ft/min]. Methane is delivered into the gas oven together with air to form a mixture of gases containing carbon dioxide. The samples of CO_2 percentage content form an output signal $y(n)$. The sampling period was 9 s. The data set consisting of 290 pairs of the input vector $[y(n-1), \ldots, -y(n-4), x(n), \ldots, x(n-5)]^T$, and the output value $y\,(n)$ was divided into two parts: the training one and the testing one. The training set consists of the first 100 pairs of the data and the testing set contains the remaining 190 pairs.

The learning was carried out in two phases. In both of them, the most popular fuzzy implications were applied (Fodor, Gödel, Gougen, Kleene-Dienes, Łukasiewicz, Reichenbach, Rescher and Zadeh). The learning results obtained from Łukasiewicz and Reichenbach's implications are equivalent to the inference results obtained on the basis of Mamdani and Larsen's fuzzy relations, respectively (Czogała and Łęski, 1999). The number of if-then rules I was changed from 2 to 6, and the initial values of membership functions of antecedents were estimated by means of FCM clustering. The partition process was repeated 25 times for different random initializations of the partition matrix, and results characterized by the minimal value of the Xie-Beni validity index (Xie and Beni, 1991) were chosen. The generalization ability was determined on the basis of root mean square error (RMSE) values on the testing set. All experiments were conducted in a MATLAB environment.

During the first phase of the learning only the εLSSLI algorithm was used (with the initial values of antecedent parameters calculated by means of the FCM method and the triangle base widths set to 1). We sought a set of parameters for which the best generalization ability of the neuro-fuzzy system was achieved. We set $\rho = 0.98$, $\underline{b}^{[1]} = 10^{-6}$, $\kappa = 10^{-4}$ and $k_{\varepsilon\max} = 1000$. The parameters τ and ε were changed from 0.01 to 0.1 with a step of 0.01. The lowest RMSE values for each number of if-then rules and each fuzzy implication used are shown in Tables 2–6. For comparison, RMSE results for imprecision-intolerant learning (the LS method) are shown, too. The best results are marked in bold.

Table 2. RMSE of identification—the first learning phase ($I = 2$).

Fuzzy implication	εLSSLI			LS
(relation)	RMSE	ε	τ	RMSE
Fodor	0.3507	0.01	0.01	0.3595
Gödel	**0.3453**	**0.01**	**0.01**	**0.3493**
Gougen	**0.3453**	**0.01**	**0.01**	**0.3493**
Kleene-Dienes	0.3516	0.01	0.01	0.3604
Łukasiewicz (Mamdani)	0.3507	0.01	0.01	0.3595
Reichenbach (Larsen)	0.3507	0.01	0.01	0.3595
Rescher	0.3455	0.01	0.01	0.3494
Zadeh	0.3458	0.01	0.01	0.3494

Table 3. RMSE of identification—the first learning phase ($I = 3$).

Fuzzy implication	εLSSLI			LS
(relation)	RMSE	ε	τ	RMSE
Fodor	0.3656	0.09	0.01	0.3776
Gödel	0.3457	0.01	0.01	0.3493
Gougen	**0.3456**	**0.01**	**0.01**	**0.3493**
Kleene-Dienes	0.3682	0.09	0.01	0.3793
Łukasiewicz (Mamdani)	0.3656	0.09	0.01	0.3776
Reichenbach (Larsen)	0.3656	0.09	0.01	0.3776
Rescher	0.3458	0.01	0.01	0.3493
Zadeh	0.3467	0.01	0.01	0.3497

Table 4. RMSE of identification—the first learning phase ($I = 4$).

Fuzzy implication	εLSSLI			LS
(relation)	RMSE	ε	τ	RMSE
Fodor	0.3935	0.02	0.02	0.4280
Gödel	0.3489	0.01	0.01	0.3493
Gougen	**0.3458**	**0.01**	**0.01**	**0.3493**
Kleene-Dienes	0.3928	0.01	0.03	0.4217
Łukasiewicz (Mamdani)	0.3935	0.02	0.02	0.4280
Reichenbach (Larsen)	0.3936	0.02	0.02	0.4280
Rescher	0.3460	0.01	0.01	0.3493
Zadeh	0.3468	0.01	0.01	0.3499

Table 5. RMSE of identification—the first learning phase ($I = 5$).

Fuzzy implication	εLSSLI			LS
(relation)	RMSE	ε	τ	RMSE
Fodor	0.4007	0.05	0.06	0.4156
Gödel	0.3462	0.01	0.01	0.3493
Gougen	**0.3461**	**0.01**	**0.01**	**0.3493**
Kleene-Dienes	0.3923	0.07	0.01	0.4146
Łukasiewicz (Mamdani)	0.4001	0.14	0.05	0.4158
Reichenbach (Larsen)	0.4000	0.14	0.05	0.4160
Rescher	0.3462	0.01	0.01	0.3493
Zadeh	0.3482	0.01	0.01	0.3504

Table 6. RMSE of identification—the first learning phase ($I = 6$).

Fuzzy implication	εLSSLI			LS
(relation)	RMSE	ε	τ	RMSE
Fodor	0.5186	0.57	0.03	0.5524
Gödel	0.3466	0.01	0.01	0.3493
Gougen	**0.3465**	**0.01**	**0.01**	**0.3493**
Kleene-Dienes	0.5190	0.03	0.21	0.5733
Łukasiewicz (Mamdani)	0.5122	0.59	0.02	0.5535
Reichenbach (Larsen)	0.5094	0.59	0.02	0.5544
Rescher	0.3467	0.01	0.01	0.3493
Zadeh	0.3469	0.01	0.01	0.3487

The obtained results confirm that ε-insensitive learning leads to a better generalization ability compared with imprecision-intolerant learning. The identification error for testing data increases with an increase in the number of fuzzy if-then rules for all implications used. This is due to the overfitting effect of the training set. However, a decrease in the generalization ability of εLSSLI is slower compared with imprecision-tolerant learning. Different methods of interpreting if-then rules lead to different learning results. Generally, the lowest values of the identification error during the first learning phase were achieved using a logical interpretation of fuzzy if-then rules based on Gougen's fuzzy implication. The best identification quality $(\text{RMSE} = 0.3453)$ was obtained using εLSSLI for $I = 2$ fuzzy conditional statements.

During the second phase of the learning, the proposed DAF + εLSSLI algorithm was employed. The parameters of the εLSSLI method were set using the results from the first learning phase. For the deterministic annealing procedure, the following parameter values were applied: $\eta_{\text{ini}} = 0.01$, $T_{\max} \in \{10^3, 10^2, \ldots, 10^{-3}\}$, $T_{\min} = 10^{-5}\,T_{\max}$, $\lambda = 0.95$, $\delta = 10^{-5}$ and $k_{\max} = 10$. As a reference procedure, we used the DAF method combined with the LS algorithm and the original ANBLIR learning method. Five hundred iterations of the steepest descent procedure combined with the least squares algorithm were executed. Moreover, two heuristic rules for changes in the learning rate were applied in the ANBLIR reference procedure (Jang *et al.*, 1997; Czogała and Łęski, 1999): (a) if in four successive iterations the value of the error function diminished for the whole learning set, then the learning parameter was increased (multiplied by 1.1), (b) if in four successive iterations the value of the error function increased and decreased consecutively for the whole learning set, then the learning parameter was decreased (multiplied by 0.9). The learning results are tabulated in Tables 7–11.

Table 7. RMSE of identification ($I = 2$).

Fuzzy implication	DAF + εLSSLI		DAF + LS		ANBLIR
(relation)	$T_{\max}$	RMSE	$T_{\max}$	RMSE	RMSE
Fodor	$\mathbf{10^1}$	**0.3430**	10^2	0.3553	**0.3609**
Gödel	10^{-3}	0.3431	10^{-3}	0.4449	0.4581
Gougen	10^0	0.3436	10^3	0.4573	0.4636
Kleene-Dienes	10^0	0.3436	10^{-1}	0.3583	0.3624
Łukasiewicz (Mamdani)	10^1	0.3434	10^1	0.3543	0.3609
Reichenbach (Larsen)	10^{-1}	0.3441	$\mathbf{10^1}$	**0.3491**	0.3608
Rescher	10^0	0.3431	10^1	0.4552	0.4791
Zadeh	10^1	0.3452	10^1	0.3532	0.3526

Table 8. RMSE of identification ($I = 3$).

Fuzzy implication	DAF + εLSSLI		DAF + LS		ANBLIR
(relation)	$T_{\max}$	RMSE	$T_{\max}$	RMSE	RMSE
Fodor	10^1	0.3528	10^3	0.3668	0.3786
Gödel	$\mathbf{10^0}$	**0.3445**	10^2	0.4156	0.4217
Gougen	10^0	0.3446	10^2	0.4229	0.4340
Kleene-Dienes	10^{-3}	0.3675	10^{-1}	0.3719	0.3785
Łukasiewicz (Mamdani)	10^1	0.3477	10^3	0.3705	0.3785
Reichenbach (Larsen)	10^1	0.3547	10^2	0.3669	0.3786
Rescher	$\mathbf{10^0}$	**0.3445**	10^1	0.4160	0.4353
Zadeh	10^{-1}	0.3451	$\mathbf{10^{-1}}$	**0.3485**	**0.3493**

Table 9. RMSE of identification ($I = 4$).

Fuzzy implication	DAF + εLSSLI		DAF + LS		ANBLIR
(relation)	$T_{\max}$	RMSE	$T_{\max}$	RMSE	RMSE
Fodor	10^2	0.3528	10^{-3}	0.4307	0.4298
Gödel	$\mathbf{10^1}$	**0.3448**	10^{-2}	0.4645	0.4671
Gougen	10^3	0.3458	10^2	0.4713	0.4737
Kleene-Dienes	10^2	0.3717	10^3	0.3755	0.4251
Łukasiewicz (Mamdani)	10^3	0.3729	10^2	0.4296	0.4298
Reichenbach (Larsen)	10^3	0.3560	10^2	0.4284	0.4299
Rescher	10^3	0.3449	10^1	0.4793	0.4855
Zadeh	10^3	0.3460	$\mathbf{10^2}$	**0.3501**	**0.3532**

Table 10. RMSE of identification ($I = 5$).

Fuzzy implication	DAF + εLSSLI		DAF + LS		ANBLIR
(relation)	$T_{\max}$	RMSE	$T_{\max}$	RMSE	RMSE
Fodor	10^3	0.3546	10^2	0.4310	0.4428
Gödel	$\mathbf{10^1}$	**0.3451**	10^{-2}	0.6279	0.6693
Gougen	10^3	0.3461	10^2	0.6359	0.7286
Kleene-Dienes	10^1	0.3764	10^0	0.3988	0.4366
Łukasiewicz (Mamdani)	10^3	0.3599	10^3	0.4268	0.4429
Reichenbach (Larsen)	10^3	0.3893	10^1	0.4282	0.4433
Rescher	10^3	0.3453	10^{-3}	0.7341	0.8061
Zadeh	10^3	0.3478	$\mathbf{10^3}$	**0.3516**	**0.3530**

Table 11. RMSE of identification ($I = 6$).

Fuzzy implication	DAF + εLSSLI		DAF + LS		ANBLIR
(relation)	$T_{\max}$	RMSE	$T_{\max}$	RMSE	RMSE
Fodor	10^2	0.3620	10^{-3}	0.5427	0.5427
Gödel	$\mathbf{10^1}$	**0.3455**	10^3	0.5887	0.6343
Gougen	10^1	0.3464	10^2	0.6146	0.6341
Kleene-Dienes	10^3	0.4040	10^3	0.5049	0.5437
Łukasiewicz (Mamdani)	10^3	0.3584	10^{-3}	0.5410	0.5412
Reichenbach (Larsen)	10^{-1}	0.3590	10^3	0.5291	0.5390
Rescher	10^2	0.3464	10^2	0.6922	0.7041
Zadeh	10^3	0.3468	$\mathbf{10^{-1}}$	**0.3441**	**0.3441**

Clearly, the ε-insensitive learning based method demonstrates a consistent improvement in the generalization ability. It can be noticed that the proposed hybrid algorithm leads to better identification results in comparison with both imprecision-intolerant reference procedures and εLSSLI performed individually. Only in one example ($I = 6$, Zadeh's implication) we did not obtain a decrease in the identification error. A decrease in the generalization ability of DAF + εLSSLI for all fuzzy implications used is much slower in comparison with imprecision-intolerant learning using DAF + LS and the original ANBLIR too. Again, different methods of interpreting if-then rules lead to different learning results. Nevertheless, it is hard to qualify one of them as best. Generally, the lowest values of the identification error were achieved using a logical interpretation of fuzzy if-then rules based on Gödel's implication. However, the best identification quality $(\text{RMSE} = 0.3430)$ was obtained using the DAF + εLSSLI procedure for Fodor's implication, $I = 2$ and $T_{\max} = 10$. Figures 1, 2 and 3 show the input signal, the output signal (original—a continuous line, modeled—a dotted line) and the identification error, respectively.

The proposed procedure was also tested for robustness to outliers. For this purpose, we added one outlier to the training set: the minimal output sample $y\,(43)$ equal to 45.6 was set to the doubled value of the maximal output sample $2\,y\,(82)$ equal to 116.8. Then we performed

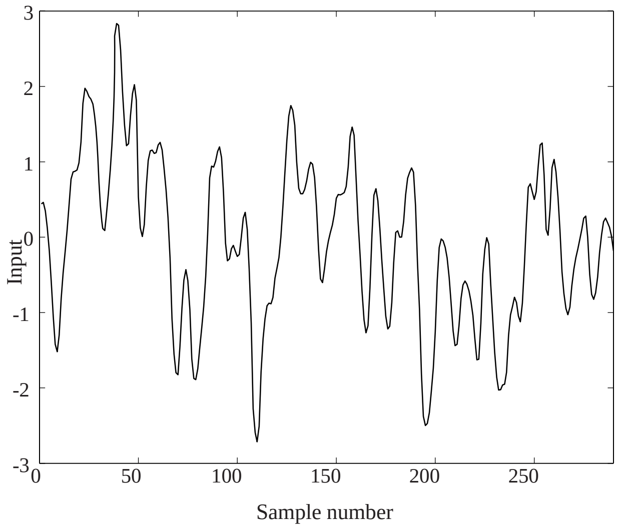

Fig. 1. Input signal for system identification data.

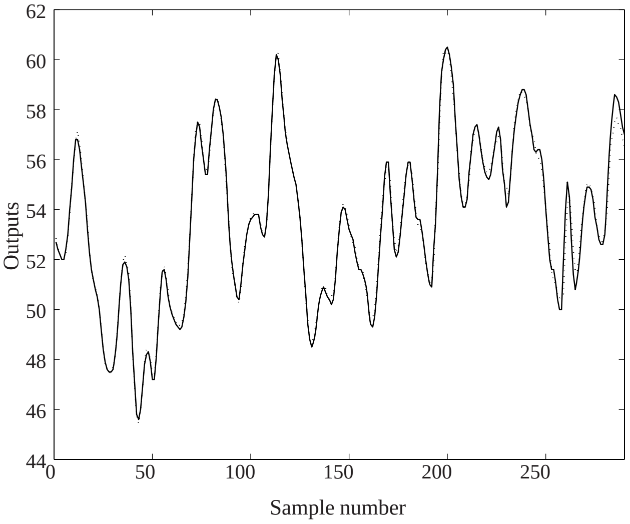

Fig. 2. Output signals for system identification data: original (a continuous line) and modeled (a dotted line) ($I = 2$, Fodor implication, $T_{\max} = 10$).

the second learning stage for two fuzzy if-then rules using the parameters $(\varepsilon, \tau, T_{\min})$ for which we obtained the best generalization ability without outliers. The results are shown in Table 12. We can see that the DAF $+\varepsilon$LSSLI approach improves the generalization ability in the presence of outliers in the training set over the reference algorithms. For Reichenbach's fuzzy implication (and the same conjunctive interpretation based on Larsen's fuzzy relation) we obtained the best learning quality $(\text{RMSE} = 0.3649)$.

The second numerical experiment concerned the benchmark prediction problem of sunspots (Weigend *et al.*, 1990). The data set consists of 280 samples $x(n)$ of sunspot activity measured within a one-year period from 1700 to 1979 A.D. The goal is the prediction of the number of sunspots (the output value) $y(n) = x(n)$ using past values combined in the embedded input vector $[\ x(n-1),\ \ x(n-2),\ \ \ldots,\ \ x(n-12)\ ]^T$. The training set consists of the first 100 input-output pairs of the data and the testing set contains the remaining 168 pairs.

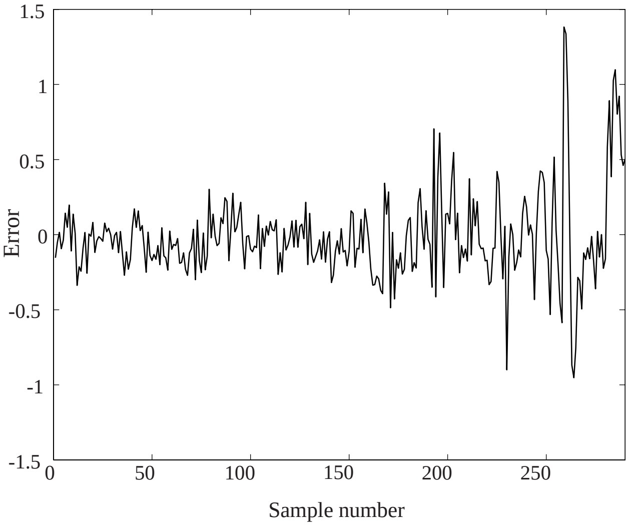

Fig. 3. Error signal for system identification data ($I = 2$, Fodor implication, $T_{\max} = 10$).

Table 12. RMSE of identification in the presence of outliers ($I = 2$).

Fuzzy implication	DAF + εLSSLI	DAF + LS	ANBLIR
(relation)	RMSE	RMSE	RMSE
Fodor	0.6605	**1.0599**	1.5973
Gödel	0.5351	2.1271	4.6723
Gougen	0.5281	4.5499	4.6242
Kleene-Dienes	0.5263	3.1560	4.5197
Łukasiewicz (Mamdani)	0.8167	2.4337	**1.5758**
Reichenbach (Larsen)	**0.3649**	2.1698	1.5878
Rescher	0.5283	4.6511	4.7096
Zadeh	0.5333	4.2039	4.4558

Analogously to the previous example, the whole learning process was split into two phases. The specification of the learning algorithms was the same. The results obtained from the first learning phase are tabulated in Tables 13–17.

Again, in this case the εLSSLI method leads to a better generalization ability than LS imprecision-tolerant learning for all fuzzy implications used. We observe a consistent decrease in the overfitting effect accompanying an increase in the number of fuzzy if-then rules for

Table 13. RMSE of prediction—the first learning phase ($I = 2$).

Fuzzy implication	εLSSLI			LS
(relation)	RMSE	ε	τ	RMSE
Fodor	0.0845	0.09	0.09	0.0933
Gödel	0.0843	0.16	0.19	**0.0917**
Gougen	0.0843	0.16	0.19	**0.0917**
Kleene-Dienes	0.0867	0.09	0.11	0.0962
Łukasiewicz (Mamdani)	**0.0838**	**0.11**	**0.19**	0.0933
Reichenbach (Larsen)	0.0846	0.09	0.10	0.0933
Rescher	0.0843	0.16	0.19	0.0917
Zadeh	0.0843	0.16	0.19	0.0917

Table 14. RMSE of prediction—the first learning phase ($I = 3$).

Fuzzy implication	εLSSLI			LS
(relation)	RMSE	ε	τ	RMSE
Fodor	0.0785	0.09	0.03	**0.0845**
Gödel	0.0843	0.16	0.12	0.0917
Gougen	0.0843	0.16	0.12	0.0917
Kleene-Dienes	0.0800	0.08	0.03	0.0858
Łukasiewicz (Mamdani)	0.0784	0.09	0.05	0.0845
Reichenbach (Larsen)	**0.0783**	**0.09**	**0.05**	0.0846
Rescher	0.0843	0.16	0.12	0.0917
Zadeh	0.0843	0.16	0.12	0.0919

Table 15. RMSE of prediction—the first learning phase ($I = 4$).

Fuzzy implication	εLSSLI			LS
(relation)	RMSE	ε	τ	RMSE
Fodor	0.0794	0.03	0.06	**0.0900**
Gödel	0.0843	0.16	0.09	0.0917
Gougen	0.0843	0.16	0.09	0.0917
Kleene-Dienes	0.0811	1.00	0.01	0.0963
Łukasiewicz (Mamdani)	0.0791	0.03	0.06	0.0900
Reichenbach (Larsen)	**0.0786**	**0.03**	**0.06**	0.0900
Rescher	0.0843	0.16	0.09	0.0918
Zadeh	0.0843	0.16	0.09	0.0916

εLSSLI in comparison with the LS procedure, too. Analogously to the first numerical experiment, we obtained different learning results from different methods of interpreting if-then rules. All implications lead to a satisfactory identification quality and it is difficult to qualify one of them as best. The lowest value of the prediction error $(\text{RMSE} = 0.0783)$ was achieved for $I = 3$, using a logical interpretation of fuzzy if-then rules based on Reichenbach's fuzzy implication and the same conjunctive interpretation based on Larsen's fuzzy relation.

Table 16. RMSE of prediction—the first learning phase ($I = 5$).

Fuzzy implication	εLSSLI			LS
(relation)	RMSE	ε	τ	RMSE
Fodor	**0.0810**	**0.77**	**0.01**	0.0948
Gödel	0.0843	0.16	0.07	0.0917
Gougen	0.0843	0.16	0.07	0.0961
Kleene-Dienes	0.0865	0.39	0.01	0.1025
Łukasiewicz (Mamdani)	0.0810	0.76	0.01	0.0948
Reichenbach (Larsen)	0.0810	0.76	0.01	0.0949
Rescher	0.0843	0.16	0.07	**0.0917**
Zadeh	0.0843	0.16	0.07	0.0917

Table 17. RMSE of prediction—the first learning phase ($I = 6$).

Fuzzy implication	εLSSLI			LS
(relation)	RMSE	ε	τ	RMSE
Fodor	0.0856	0.02	0.16	0.0984
Gödel	0.0843	0.16	0.06	**0.0917**
Gougen	**0.0842**	**0.16**	**0.06**	**0.0917**
Kleene-Dienes	0.0877	0.01	0.15	0.1159
Łukasiewicz (Mamdani)	0.0856	0.02	0.16	0.0984
Reichenbach (Larsen)	0.0857	0.01	0.10	0.0984
Rescher	0.0843	0.16	0.06	**0.0917**
Zadeh	0.0840	0.13	0.02	0.0922

Table 18. RMSE of prediction ($I = 2$).

Fuzzy implication	DAF + εLSSLI		DAF + LS		ANBLIR
(relation)	$T_{\max}$	RMSE	$T_{\max}$	RMSE	RMSE
Fodor	10^{-2}	0.0743	10^{-2}	0.0840	0.0881
Gödel	10^{-3}	0.0838	10^{3}	0.0910	0.1034
Gougen	10^{-3}	0.0838	10^{3}	0.0913	0.1032
Kleene-Dienes	10^{-2}	0.0750	10^{1}	0.0860	0.0942
Łukasiewicz (Mamdani)	10^{-3}	0.0756	$\mathbf{10^{0}}$	**0.0833**	**0.0880**
Reichenbach (Larsen)	$\mathbf{10^{-2}}$	**0.0728**	10^{-2}	0.0843	0.0882
Rescher	10^{-3}	0.0813	10^{3}	0.0910	0.1039
Zadeh	10^{3}	0.0843	10^{1}	0.0844	0.0892

The clearer superiority of the ε-insensitive learning method over imprecision-tolerant learning can be observed in the second stage of the experiment (Tables 18–22). Taking into account the obtained learning results, it can be concluded that the combination of the DAF and εLSSLI procedures leads to an improved generalization ability of sunspot prediction. For all fuzzy implications used, analogously to the first numerical experiment, a decrease in the generalization ability with an increase in the number of fuzzy rules for DAF + εLSSLI is much

Table 19. RMSE of prediction ($I = 3$).

Fuzzy implication	DAF + εLSSLI		DAF + LS		ANBLIR
(relation)	$T_{\max}$	RMSE	$T_{\max}$	RMSE	RMSE
Fodor	10^{-2}	0.0749	$\mathbf{10^{-2}}$	**0.0843**	0.0920
Gödel	10^{-3}	0.0835	10^{3}	0.0885	0.1104
Gougen	10^{-1}	0.0842	10^{3}	0.0885	0.1126
Kleene-Dienes	10^{0}	0.0786	10^{-1}	0.0864	**0.0912**
Łukasiewicz (Mamdani)	10^{1}	0.0764	10^{-2}	0.0846	0.0920
Reichenbach (Larsen)	10^{0}	0.0760	10^{-2}	0.0847	0.0921
Rescher	10^{-2}	0.0840	10^{1}	0.0889	0.1153
Zadeh	$\mathbf{10^{2}}$	**0.0748**	10^{-1}	0.0855	0.0922

Table 20. RMSE of prediction ($I = 4$).

Fuzzy implication	DAF + εLSSLI		DAF + LS		ANBLIR
(relation)	$T_{\max}$	RMSE	$T_{\max}$	RMSE	RMSE
Fodor	$\mathbf{10^{-3}}$	**0.0751**	10^{3}	0.0985	0.1110
Gödel	10^{-3}	0.0841	10^{2}	0.0898	0.1334
Gougen	10^{-3}	0.0841	10^{2}	0.0911	0.1237
Kleene-Dienes	10^{1}	0.0781	10^{-1}	0.0979	0.1188
Łukasiewicz (Mamdani)	10^{-2}	0.0775	10^{1}	0.0922	0.1111
Reichenbach (Larsen)	10^{0}	0.0765	10^{2}	0.0990	0.1112
Rescher	10^{-1}	0.0829	10^{3}	0.0898	0.1422
Zadeh	10^{1}	0.0809	$\mathbf{10^{3}}$	**0.0870**	**0.0886**

Table 21. RMSE of prediction ($I = 5$).

Fuzzy implication	DAF + εLSSLI		DAF + LS		ANBLIR
(relation)	$T_{\max}$	RMSE	$T_{\max}$	RMSE	RMSE
Fodor	10^{1}	0.0779	10^{-3}	0.1116	0.1124
Gödel	10^{0}	0.0842	10^{2}	0.0893	0.1298
Gougen	10^{1}	0.0843	10^{2}	0.0914	0.1230
Kleene-Dienes	$\mathbf{10^{-2}}$	**0.0766**	10^{-1}	0.1077	0.1142
Łukasiewicz (Mamdani)	10^{1}	0.0766	10^{-3}	0.1116	0.1123
Reichenbach (Larsen)	10^{0}	0.0776	10^{0}	0.1083	**0.1122**
Rescher	10^{-2}	0.0840	10^{2}	0.0885	0.1349
Zadeh	10^{3}	0.0793	$\mathbf{10^{3}}$	**0.0861**	0.0891

Table 22. RMSE of prediction ($I = 6$).

Fuzzy implication	DAF + εLSSLI		DAF + LS		ANBLIR
(relation)	$T_{\max}$	RMSE	$T_{\max}$	RMSE	RMSE
Fodor	10^{1}	0.0772	10^{2}	0.1346	0.1435
Gödel	10^{-3}	0.0840	10^{3}	0.0874	0.1426
Gougen	10^{-3}	0.0843	10^{2}	0.0883	0.1199
Kleene-Dienes	10^{0}	0.0765	10^{-1}	0.1175	0.1453
Łukasiewicz (Mamdani)	10^{-2}	0.0778	10^{3}	0.1296	0.1431
Reichenbach (Larsen)	$\mathbf{10^{1}}$	**0.0763**	10^{-2}	0.1042	0.1428
Rescher	10^{-1}	0.0831	10^{2}	0.0890	0.1536
Zadeh	10^{-2}	0.0800	$\mathbf{10^{3}}$	**0.0870**	**0.0913**

Table 23. RMSE of prediction in the presence of outliers ($I = 2$).

Fuzzy implication	DAF + εLSSLI	DAF + LS	ANBLIR
(relation)	RMSE	RMSE	RMSE
Fodor	0.0940	0.1218	0.1295
Gödel	0.0992	0.1229	0.1313
Gougen	0.0998	0.1233	0.1316
Kleene-Dienes	**0.0870**	**0.0999**	0.1257
Łukasiewicz (Mamdani)	0.0967	0.1291	0.1405
Reichenbach (Larsen)	0.0900	0.1210	0.1411
Rescher	0.0953	0.1235	0.1318
Zadeh	0.1015	0.1171	**0.1194**

slower in comparison with the reference procedures. Different methods of interpreting if-then rules lead to different learning results. It is hard to find one fuzzy implication that gives the best prediction quality irrespective of the number of fuzzy if-then rules. Generally, the highest values of the identification error and the worst generalization ability were obtained using a logical interpretation based on the Gödel, Gougen and Rescher implications. The best identification quality $(\text{RMSE} = 0.0728)$ was achieved using DAF + εLSSLI for Reichenbach's implication (equivalent to Larsen's fuzzy relation) with $I = 2$ and $T_{\max} = 10^{-2}$. Figures 4 and 5 show the output signal (a continuous line), the predicted values (a dotted line) and the prediction error, respectively.

To test robustness to outliers for the prediction problem, we added one outlier to the training set: the minimal output sample $y\,(1)$ equal to zero was set to the doubled value of the maximal output sample $2\,y\,(67)$ equal to 1.6150. Then, analogously to the previous example, we performed the second learning stage for $I = 2$ using parameters characterized by the best generalization ability without outliers. The obtained results are shown in Table 23. From these results we can see significant improvements in the generalization ability in the presence of outliers when we use methods tolerant of imprecision. The best learning result $(\text{RMSE} = 0.0870)$ was achieved for the Kleene-Dienes fuzzy implication.

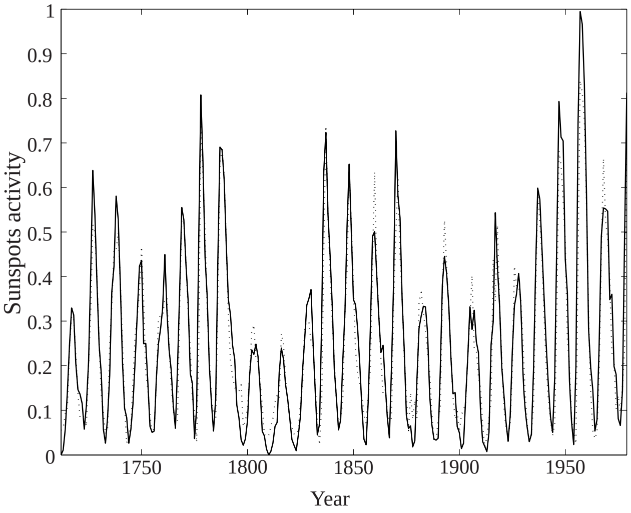

Fig. 4. Sunspots activity: original (a continuous line) and predicted values (a dotted line) ($I = 2$, the Reichenbach implication, $T_{\max} = 10^{-2}$).

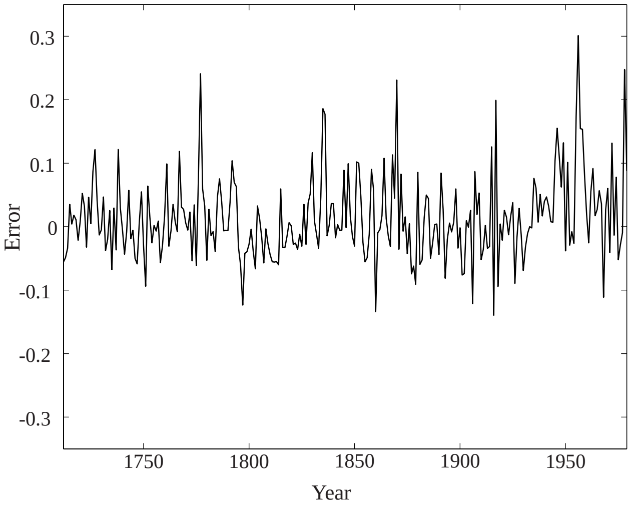

Fig. 5. Error values of sunspots prediction ($I = 2$, the Reichenbach implication, $T_{\max} = 10^{-2}$).

To summarize, the combination of the deterministic annealing method and ε-insensitive learning leads to an improvement in fuzzy modeling results. However, it must be noted that the performance enhancement is achieved through a decrease in the computational effectiveness of the learning procedure. The computational burden of the deterministic annealing procedure is approximately two times greater compared with the gradient descent method used in the original ANBLIR learning algorithm, and the εLSSLI computational burden is approximately three times greater than that of the least-squares method. To make a precise comparison of the computational effort needed by the proposed method, we checked computational times of the training procedures considered. All experiments were run in the MATLAB 6.5 environment using a PC equipped with an Intel Pentium IV 1.6 GHz processor. The obtained results (time in seconds) are tabulated in Tables 24 (for Reichenbach's implication and the identification problem) and 25 (for Reichenbach's implication and the prediction problem). The training time using the proposed hybrid algorithm is approximately three to six times longer than the genuine training of the ANBLIR procedure (with 500 learning epochs), however similar (or even lower) in comparison with the DAF + LS procedure. This is because the precision criterion of the εLSSLI procedure was usually satisfied earlier than the criterion based on the maximum number of iterations.

Table 24. Computation times (in seconds) of learning algorithms for the identification of the gas oven (the Reichenbach implication).

I	DAF + εLSSLI	DAF + LS	ANBLIR
2	136	138	22
3	141	142	26
4	113	144	31
5	94	158	40
6	187	167	47

Table 25. Computation times (in seconds) of learning algorithms for the prediction of sunspots (the Reichenbach implication).

I	DAF + εLSSLI	DAF + LS	ANBLIR
2	133	133	23
3	140	140	27
4	143	146	30
5	130	160	39
6	144	167	52

Another drawback of the DAF procedure is the necessity of an arbitrary selection of the learning parameters. The value of the initial stepsize η_{ini} was determined on the basis of a trial-and-error procedure. Too small values of η_{ini} slow down the learning convergence and lead to unsatisfactory learning results. Too high η_{ini} values may worsen the learning quality as well since they may lead to an insufficient precision in the gradient descent "searching" in the parameter space.

Further parameters having considerable influence on the learning results are the initial pseudotemperature $T_{\max}$, the final pseudotemperature $T_{\min}$, the annealing schedule parameter q and the number of iterations at each level of the pseudotemperature $k_{\max}$. The initial pseudotemperature should be sufficiently high to ensure entropy maximization at the beginning of the optimization procedure. Too small values of the initial pseudotemperature may lead to unsatisfactory learning results as the influence of the entropy maximization factor may not be strong enough to ensure the appropriate range of the gradient descent search in the parameter space. The final pseudotemperature should be low enough to assure the minimization of the square error at the end. Too high values of the final pseudotemperature may lead to a decrease in the learning quality as well since we may not have a suitable square error minimization phase at the end of the deterministic procedure. Very high $T_{\max}$ values and similarly, very small $T_{\min}$ values, lead to an increase in the number of iterations needed. In our experiments a trial-and-error method was used to set their values. We attempted to get satisfactory modeling results and as small the number of iterations as possible.

The formula for the calculation of the annealing schedule parameter that guarantees finding the global minimum of the cost for the simulated annealing method was given in (German and German, 1984). However, there is no such confirmation for the deterministic annealing procedure. This method of determining the annealing schedule parameter leads to a significant increase in the number of steps needed to find optimal system parameters. Therefore its value was set arbitrarily. Again, we tried to obtain an acceptable modeling quality and a low number of iterations.

The number of iterations at each level of the pseudotemperature was determined on the basis of the entropy variation level (Rao *et al.*, 1999). To ensure faster convergence, the criterion of the maximum number of iterations $k_{\max}$ was added.

The parameters ε and τ of εLSSLI were also set using a trial-and-error procedure. We cannot give clear rules for the selection of their values. Too high values of ε and τ may lead to a decrease in the solution precision. On the other hand, too small values of ε and τ result in a decrease in the generalization ability.

The parameters κ and $k_{\varepsilon \max}$ control the solution precision and the computational cost of LSSLI training. The selection of their values was based on the trade-off between the learning quality and the number of iterations needed to get satisfactory learning results.

7. Conclusions

In this paper, a new learning algorithm of the ANBLIR neuro-fuzzy system was presented. In the proposed procedure, the parameters of fuzzy sets from the antecedents and consequents of fuzzy if-then rules are adjusted separately by means of deterministic annealing with a “freezing” phase and ε-insensitive learning by solving a system of linear inequalities, respectively. Experimentation shows the usefulness of the method in the extraction of fuzzy if-then rules for system identification and signal prediction problems. The obtained results indicate an improvement in the generalization ability and robustness to outliers compared with imprecision-intolerant learning. However, the performance enhancement is achieved through an increase in the computational burden of the learning procedure. Another problem is the necessity of an arbitrary selection of learning parameters. The determination of automatic methods for their selection constitutes a principal direction of future investigations.

References

Bezdek J.C. (1982): *Pattern Recognition with Fuzzy Objective Function Algorithms*. — New York: Plenum Press.

Box G.E.P. and Jenkins G.M. (1976): *Time Series Analysis. Forecasting and Control*. — San Francisco: Holden-Day.

Czabański R. (2003): *Automatic Fuzzy If-Then Rules Extraction from Numerical Data*. — Ph.D. thesis, Silesian University of Technology, Gliwice, (in Polish).

Czabański R. (2005): *Neuro-fuzzy modeling based on deterministic annealing approach*. — Int. J. Appl. Math. Comput. Sci., Vol. 15, No. 4, pp. 125–134.

Czogała E. and Łęski J. (1996): *A new fuzzy inference system with moving consequents in if-then rules. Application to pattern recognition*. — Bull. Polish Acad. Science, Vol. 45, No. 4, pp. 643–655.

Czogała E. and Łęski J. (1999): *Fuzzy and Neuro-Fuzzy Intelligent Systems*. — Heidelberg: Physica-Verlag.

Czogała E. and Łęski J. (2001): *On equivalence of approximate reasoning results using different interpretations of if-then rules*. — Fuzzy Sets Syst., Vol. 117, No. 2, pp. 279–296.

German S. and German D. (1984): *Stochastic relaxation, Gibbs distribution and the Bayesian restoration in images*. — IEEE Trans. Pattern Anal. Mach. Intell., Vol. 6, pp. 721–741.

Ho D. and Kashyap R.L. (1965): *An algorithm for linear inequalities and its applications*. — IEEE Trans. Elec. Comp., Vol. 14, No. 5, pp. 683–688.

Ho Y.C. and Kashyap R.L. (1966): *A class of iterative procedures for linear inequalities*. — SIAM J. Contr., Vol. 4, No. 2, pp. 112–115.

Jang J.S.R. (1993): *ANFIS: Adaptive-network-based fuzzy inference system*. — IEEE Trans. Syst. Man Cybern., Vol. 23, No. 3, pp. 665–685.

Jang J.S.R. and Sun J.S.R. (1993): *Functional equivalence between radial basis function networks and fuzzy inference systems*. — IEEE Trans. Neural Netw., Vol. 4, No. 1, pp. 156–159.

Jang J.S.R., Sun C.T. and Mizutani E. (1997): *Neuro-Fuzzy and Soft Computing. A Computational Approach to Learning and Machine Intelligence*. — Upper Saddle River: Prentice-Hall.

Kirkpatrick S., Gelatt C. and Vecchi M. (1983): *Optimization by simulated annealing*. — Science, Vol. 220, No. 4598, pp. 671–680.

Łęski J. (2002): *Improving generalization ability of neuro-fuzzy systems by ε-insensitive learning*. — Int. J. Appl. Math. Comput. Sci., Vol. 12, No. 3, pp. 437–447.

Łęski J. (2003a): *Neuro-fuzzy system with learning tolerant to imprecision*. — Fuzzy Sets Syst., Vol. 138, No. 2, pp. 427–439.

Łęski J. (2003b): *ε-Insensitive learning techniques for approximate reasoning systems*. — Int. J. Comput. Cognit., Vol. 1, No. 1, pp. 21–77.

Metropolis N., Rosenbluth A.W., Rosenbluth M.N., Teller A.H. and Teller E. (1953): *Equation of state calculation by fast computing machines*. — J. Chem. Phys., Vol. 21, No. 6, pp. 1087–1092.

Mitra S. and Hayashi Y. (2000): *Neuro-fuzzy rule generation: Survey in soft computing framework*. — IEEE Trans. Neural Netw., Vol. 11, No. 3, pp. 748–768.

Rao A.V., Miller D., Rose K. and Gersho A. (1997): *Mixture of experts regression modeling by deterministic annealing*. — IEEE Trans. Signal Process., Vol. 45, No. 11, pp. 2811–2820.

Rao A.V., Miller D., Rose K. and Gersho A. (1999): *A deterministic annealing approach for parsimonious design of piecewise regression models*. — IEEE Trans. Pattern Anal. Mach. Intell., Vol. 21, No. 2, pp. 159–173.

Rose K. (1991): *Deterministic Annealing, Clustering and Optimization*. — Ph.D. thesis, California Inst. Tech., Pasadena.

Rose K. (1998): *Deterministic annealing for clustering, compression, classification, regression and related optimization problems*. — Proc. IEEE, Vol. 86, No. 11, pp. 2210–2239.

Vapnik V. (1998): *Statistical Learning Theory*. — New York: Wiley.

Vapnik V. (1999): *An overview of statistical learning theory*. — IEEE Trans. Neural Netw., Vol. 10, No. 5, pp. 988–999.

Weigend A.S., Huberman B.A. and Rumelhart D.E. (1990): *Predicting the future: A connectionist approach*. — Int. J. Neural Syst., Vol. 1, No. 2, pp. 193–209.

Xie X.L. and Beni G. (1991): *A validity measure for fuzzy clustering*. — IEEE Trans. Pattern Anal. Mach. Intell., Vol. 13, No. 8, pp. 841–847.

Yen J., Wang L. and Gillespie C.W. (1998): *Improving the interpretability of TSK fuzzy models by combining global learning and local learning*. — IEEE Trans. Fuzzy Syst., Vol. 6, No. 4, pp. 530–537.

Received: 4 November 2005
Revised: 28 April 2006