University of Ostrava

Institute for Research and Applications of Fuzzy Modeling

LEARNING RULE BASE IN LINGUISTIC EXPERT SYSTEMS

Radim Bělohlávek and Vilém Novák

Research report No. 16

January 29, 1999

Submitted/to appear:

Soft Computing

Supported by:

grant No. 201/96/0985 of the GAČR

University of Ostrava
Institute for Research and Applications of Fuzzy Modeling
Bráfova 7, 701 03 Ostrava 1, Czech Republic

tel.: +420-69-6160 233 fax: +420-69-22 28 28
e-mail: radim.belohlavek@osu.cz, vilem.novak@osu.cz

LEARNING RULE BASE IN LINGUISTIC EXPERT SYSTEMS [1]

Radim Bělohlávek

Institute for Research and Applications of Fuzzy Modeling
University of Ostrava
Bráfova 7, 701 03 Ostrava, Czech Republic
e-mail: belohlav@osu.cz

Vilém Novák

Institute for Research and Applications of Fuzzy Modeling
University of Ostrava
Bráfova 7, 701 03 Ostrava, Czech Republic
e-mail: novakv@osu.cz

Abstract. Linguistic fuzzy control is introduced in a pure logical framework. The problem of learning of the linguistic rule base from the data obtained by monitoring of successful control and the problem of learning the linguistic context are studied. The methods are demonstrated by results of experiments.

Key words: linguistic oriented fuzzy control, rule base, linguistic context, learning

1 Introduction

The original idea of the fuzzy controller proposed by L. A. Zadeh [15] and E. H. Mamdani [10] is to translate knowledge of the human control operator into mathematical description such that would mimic a successful course of his/her control. Since it can be expected that such a knowledge is expressed in linguistic terms, the main problem consists in translation of the linguistic terms into mathematical relations. The way how it is done became now a kind of folklore, which appears in hundreds of papers. Let us mention only briefly that this mostly assumes up to 7 terms of the kind "positively big", "negatively small", etc. The linguistic meaning of such terms is characterized as fuzzy sets in some universe of discourse, and from the linguistic point of view it is oversimplified only to linear membership functions forming mostly simple triangles, which are then tuned by modifying the membership functions (shifting, extending or narrowing their basement). In practice, the linguistic motivation is not taken into account. Of course, the result may be far from meaning of the original linguistic description, which may cause difficulties especially when it is necessary to modify the description.

A possible way out of this situation stems from analyses, presented, e.g. in [5, 6, 7]. They demonstrate that in this case, we are interpolating some classical function, not precisely known to us. The fuzzy sets in concern are most naturally interpreted as groupings of objects being similar (fuzzy equal) to certain element. Such a procedure is then quite natural. Let us stress, however, that the IF-THEN rules using which the fuzzy control is realized are not logical implications but conjunctions of imprecise propositions of the kind "X is approximately equal to x_i".

In few papers (see, e.g. [12]) we turned attention to another possibility, which is a translation of the operator's free linguistic description into a set of linguistically characterized logical implications (IF-THEN rules). Such implications are formed by linguistic terms containing a quite wide variety of adverbs taking the role of modifiers, which are translated into fuzzy logic to fit the linguistic feeling in the best possible way. Of course, this is not an easy task since among others, we have to cope with the specific problems of the so called intension and extension, and linguistic context. Without going into details, let us only briefly mention that the intension covers a property given by the linguistic expression and extension covers elements, which fall into its meaning (for more information, see the book [14]). The linguistic context must be taken into account when trying to distinguish the meaning, e.g. of "small ball" from "small tree".

[1] Supported by grant No. 201/96/0985 of the GAČR.

Sets of linguistically characterized logical implications are called linguistic descriptions. Using formal theory of the fuzzy predicate logic (cf. [14]), we can develop the theory of logical inference which is a formalization of the generalized modus ponens as outlined by L. A. Zadeh (cf. [16]).

In this paper, we will deal with linguistic descriptions and logical inference based on them. Our goal is to develop a theory of their learning, which uses the information comming form monitoring of the successful control of a process. The learning covers two basic problems, namely learning of the linguistic context given a linguistic description, and learning of the linguistic description itself. To solve this task, we employ two fundamental tools, namely neural nets and the concept of a typical linguistic term, given a value in the given context.

We will work with the set L of truth values (membership degrees) which form the interval of reals $L = [0, 1]$. Besides the standard lattice operations $\vee, \wedge$, we will work also with the Łukasiewicz implication $\rightarrow$ (residuation) given by

$$a \rightarrow b = 1 \wedge (1 - a + b), \qquad a, b \in [0, 1] \tag{1}$$

and the Łukasiewicz conjunction (product) $\otimes$, which is adjoined with the implication and which is given by

$$a \otimes b = 0 \vee (a + b - 1), \qquad a, b \in [0, 1]. \tag{2}$$

The negation $\neg$ is defined in a logical way by

$$\neg a = a \rightarrow 0 = 1 - a, \qquad a \in [0, 1].$$

These operations are interpretations of the logical connectives, namely *disjunction* $\boldsymbol{\vee}$, *conjunction* $\boldsymbol{\wedge}$, *implication* $\boldsymbol{\Rightarrow}$, *Łukasiewicz conunction* $\boldsymbol{\&}$ and *negation* $\boldsymbol{\neg}$.

A fuzzy set $A \underset{\sim}{\subset} U$ in U is a function $A : U \longrightarrow L$. By $\mathcal{F}(U)$ we denote the set of all the fuzzy sets in U (i.e., $\mathcal{F}(U) = L^U$).

2 Characterization of linguistic descriptions and their use in fuzzy control

As stated, the linguistic description is a set of IF-THEN rules, which are linguistically expressed implications whose parts are linguistic terms. The latter are linguistic expressions of the form

$$\langle \text{noun} \rangle \ \mathsf{is}\ \mathcal{A} \tag{3}$$

where $\mathcal{A}$ is a simple evaluating expression defined further. The $\langle$noun$\rangle$ is usually substituted by some variable X, for example "temperature, pressure, height", etc. whose values are elements from an ordered scale — usually real numbers.

A *simple linguistic evaluating expression* is a linguistic expression of the form

$$\langle \text{sign} \rangle \langle \text{linguistic modifier} \rangle \langle \text{atomic term} \rangle \tag{4}$$

where $\langle$sign$\rangle$ is one of the words "positively" or "negatively", $\langle$atomic term$\rangle$ is one of the words "small, medium, big", or "zero" and a $\langle$linguistic modifier$\rangle$ is an intensifying adverb such as "very, roughly", etc.

The linguistic description is a set of rules

$$\mathcal{R} := \{\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_m\} \tag{5}$$

where the rules $\mathcal{R}_i$, $i = 1, \ldots, m$ have the form

$$\mathcal{R}_i := \ \mathsf{IF}\ \mathcal{A}_{1i}\ \mathsf{AND}\ \cdots\ \mathsf{AND}\ \mathcal{A}_{ni}\ \mathsf{THEN}\ \mathcal{B}_i \tag{6}$$

and $\mathcal{A}_{ji}, \mathcal{B}_i$ are linguistic terms (3).

In the sequel, we will confine to the rules containing two antecedent variables, namely

$$\mathcal{R} := \ \mathsf{IF}\ X\ \mathsf{is}\ \mathcal{A}\ \mathsf{AND}\ Y\ \mathsf{is}\ \mathcal{B}\ \mathsf{THEN}\ Z\ \mathsf{is}\ \mathcal{C} \tag{7}$$

where $\mathcal{A}, \mathcal{B}, \mathcal{C}_i$ denote simple evaluating expressions (4).

To give an explanation of the meaning of the rules (7), we fix some language J of the many-sorted predicate fuzzy logic and a set F_J of all the well-formed formulas of J. Each linguistic evaluating expression (4) is assigned a formula $A \in F_J$. Furthermore, we suppose a finite set of sorts ι and the existence of a set of closed terms M_ι to each sort ι.

The syntax of fuzzy logic deals with so called *evaluated formulas.* These are couples $^a/A$ where $A \in F_J$ is a formula and $a \in L$ is its syntactic evaluation. If $A(x)$ is a formula with a free variable x of the sort ι and $t \in M_\iota$ is a term of the same sort then $A_x[t]$ is a formula in which all the free occurrences of x are replaced by t. Since it is not the purpose of this paper to elaborate the linguistic semantics in details (the interested reader is refered to [13, 14]) we will not specify J and F_J more precisely.

The linguistic modifiers from (4) are of two basic kinds, namely those with narrowing and extending effect. Narrowing modifiers are, for example, "extremely, significantly, very" and widening ones are "more or less, roughly, quite roughly, very roughly". We will take these modifiers as canonic. Note that narrowing modifiers make the meaning of the whole expression more precise while widening ones do the opposite. Thus, "very small" is more precise than "small", which, on the other hand, is more precise than "roughly small". A significant role is also played by the sign, which implies the necessity to distinguish between elements being, in some sense, negative from those being positive. We will usually think of negative and positive numbers but in general, this may not always be the case.

In the logical calculus, linguistic modifiers are represented by unary connectives $\lhd$ which induce an ordering of the simple evaluating expressions (4). These are assigned certain formulas of the form $\lhd Q$ where $Q \in F_J$ is an atomic formula. A canonical ordering of the evaluating expressions together with the formulas assigned to them is the following:

$$\begin{array}{rll}
\text{"negative extremely big"} & \text{is assigned} & \lhd_{-p} Q, \\
\text{"negative significantly big"} & \text{is assigned} & \lhd_{-p+1} Q, \\
 & \vdots & \\
 & \vdots & \\
\text{"negative extremely small"} & \text{is assigned} & \lhd_{-2} Q, \\
\text{"negative roughly zero"} & \text{is assigned} & \lhd_{-1} Q, \\
\text{"zero"} & \text{is assigned} & \lhd_{0} Q, \\
\text{"positive roughly zero"} & \text{is assigned} & \lhd_{1} Q, \\
\text{"positive extremely small"} & \text{is assigned} & \lhd_{2} Q, \\
\vdots & \vdots & \\
\text{"positive extremely big"} & \text{is assigned} & \lhd_{q} Q
\end{array} \tag{8}$$

where $\lhd_{-p}, \ldots, \lhd_q$ are various specific unary connectives.

Let $\mathcal{A} := \lhd_i Q$ and $\mathcal{B} := \lhd_j Q$ and $i \leq j$. we say that $\mathcal{A}$ is *narrower* than $\mathcal{B}$. Vice-versa, $\mathcal{B}$ is *wider* than $\mathcal{A}$.

The assignements in (8) are not sufficient to characterize the meaning of the corresponding linguistic expressions. As discussed in [7, 12, 14] and elsewhere, the meaning of the linguistic term (3) is a, so called *multiformula*, which is a set of evaluated formulas

$$\mathbf{A}_x := \{^{a_t}/A_x[t] \mid t \in M_\iota\}. \tag{9}$$

Each multiformula being assigned to some linguistic term $\mathcal{A}$ can be interpreted in a model $\mathcal{D}$. This interpretation is a fuzzy set

$$\mathcal{D}(\mathbf{A}_x) = \{^{\mathcal{D}(A_x[t])}/\mathcal{D}(t) \mid t \in M_\iota\} \underset{\sim}{\subset} D. \tag{10}$$

In (10), D is a support of the model $\mathcal{D}$, $\mathcal{D}(t) \in D$ is an element assigned to the term t and

$$a_t \leq \mathcal{D}(A_x[t]) \tag{11}$$

holds for every term $t \in M_\iota$. In the sequel, we will call the multiformula (9) the *intension* and its interpretation (fuzzy set) (10) the *extension* of the corresponding linguistic term $\mathcal{A}$.

The intension of the rules (7) is then the multiformula

$$\mathbf{R}_{xyz} := (\mathbf{A}_x \wedge \mathbf{B}_y) \Rightarrow \mathbf{C}_z. \tag{12}$$

In the details, (12) can be rewritten as

$$\mathbf{R}_{xyz} := \{(a_t \wedge b_s) \to c_r/((A_x[t] \wedge B_y[s]) \Rightarrow C_z[r]) \mid t \in M_1, s \in M_2, r \in M_3\}. \tag{13}$$

Furthermore, we will work with some model $\mathcal{D}_0 = \langle U, V, W, \dots \rangle$ taken as *canonical*, where $U, V, W \subseteq \mathbb{R}$ ($\mathbb{R}$ is a set of real numbers) are bounded sets, each consisting of two symmetric parts denoted by $U = U^- \cup U^+$, $V = V^- \cup V^+$, $W = W^- \cup W^+$ where $U^- = [-u^{max}, 0]$ $U^+ = [0, u^{max}]$, $V^- = [-v^{max}, 0]$ $V^+ = [0, v^{max}]$, $W^- = [-w^{max}, 0]$ and $W^+ = [0, w^{max}]$, respectively.

In general, it is not possible to assure the equality in (11). However, it is possible for the case of the terms (3) based on the simple evaluating terms (4). Then, due to the completeness theorem, we are able to mix syntactical and semantical considerations, similarly as in classical logic.

Given a canonical model $\mathcal{D}_0$, the extension of (12) is a fuzzy set

$$R = \mathcal{D}_0(\mathbf{R}_{xyz}) = \mathcal{D}_0((\mathbf{A}_x \wedge \mathbf{B}_y) \Rightarrow \mathbf{C}_z) \underset{\sim}{\subset} U \times V \times W. \tag{14}$$

The extensions of the meaning of the simple evaluating expressions (8) in the canonical model are fuzzy sets of the shape depicted on Fig. 1. The extensions of the meaning of the rules (7) are fuzzy relations $R_i \underset{\sim}{\subset} U \times V \times W$.

Let us now discuss inference in fuzzy logic. The fundamental inference rule is that of *modus ponens*

$$r_{MP} : \frac{a/A,\ c/A \Rightarrow B}{a \otimes b/B}.$$

Note that this rule makes possible to derive both a resulting formula as well as its syntactic truth evaluation. Certain modifications of this rule are possible. Among them, more important is the following one:

$$r_{MMP} : \frac{a/\lhd_i A,\ c/A \Rightarrow B}{a \otimes b/\lhd_j B}$$

where $i \leq j$. This rule can be used where working with the linguistic modifiers.

Our task now is to describe the generalized modus ponens scheme based on the set of linguistic IF-THEN rule and a slightly modified antecedent. L. A. Zadeh and E. H. Mamdani (cf. [10]) initiated a method, which is now widely used in fuzzy control and whose goal is to approximate an unknown function being imprecisely described using a set of conjunctions. As analyzed, e.g. in [7], this method introduces certain additional assumptions in the calculus of fuzzy logic. Another straightforward possibility is to introduce the set of logical implications and looking for a conclusion from them. This possibility is briefly described below.

The linguistic description (5) with the rules (7) represents a set of special axioms of some fuzzy theory T. More specifically, we consider a fuzzy theory T given by a set of set of special axioms (recall that each multiformula is a set)

$$\begin{aligned} \mathbf{R}_{xyz,1} &:= (\mathbf{A}_{x,1} \wedge \mathbf{B}_{y,1}) \Rightarrow \mathbf{C}_{z,1} \\ &\vdots \\ \mathbf{R}_{xyz,m} &:= (\mathbf{A}_{x,m} \wedge \mathbf{B}_{y,m}) \Rightarrow \mathbf{C}_{z,m} \end{aligned} \tag{15}$$

Let now multiformulas $\mathbf{A}'_x, \mathbf{B}'_y$ be given. These are intensions of some linguistic terms being slight modification of the antcedent of some of the rules (7). The modification, however, cannot be quite arbitrary. Essentially, there are three possibilities: either $\mathbf{A}'_x, \mathbf{B}'_y$ are multiformulas $\mathbf{A}_{x,i}, \mathbf{B}_{y,i}$ for some $1 \leq i \leq m$ which differ from the latter only in the syntactic truth evaluation inside them, or they contain some unary connective $\lhd$ which is narrower than that inside $\mathbf{A}_{x,i}, \mathbf{B}_{y,i}$, or both cases happen. The inference then uses one of the rules r_{MP} or r_{MMP} and consists in the following formal proof:

$$\mathbf{A}'_x,\ \mathbf{B}'_y,\ (\mathbf{A}_{x,i} \wedge \mathbf{B}_{y,i}) \Rightarrow \mathbf{C}_{z,i},\ \mathbf{C}'_z. \tag{16}$$

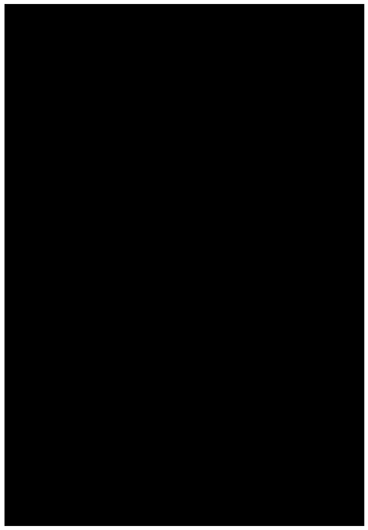

Figure 1: Shapes of the simple evaluating expressions

Note that (16) used only one of the rules (15), namely the i-th rule considered above.

A specific case occurs when

$$\mathbf{A}'_x = \{\mathbf{1}/A_{x,i}[t_0]\}, \tag{17}$$

$$\mathbf{B}'_y = \{\mathbf{1}/B_{y,i}[s_0]\} \tag{18}$$

for some special terms $t_0 \in M_1, s_0 \in M_2$ and $1 \leq i \leq m$. This is usual in fuzzy control where the precise measured values are used in the inference. Using again the proof (16), we obtain a multiformula $\mathbf{C}'_z$.

The final problem is to find a term r_0, which would, in some sense, represent $\mathbf{C}'_z$. In fuzzy control, this problem is known as *defuzzification*. Several defuzzification algorithms are described in the literature. Among them, most often is used the center of gravity method. However, this method does not give good results when applied with the linguistic terms considered above. Therefore, another method called Defuzzification of Evaluating Expressions (DOEE) has been developed in the University of Ostrava (cf. [12]). This method classifies the resulting fuzzy set into one of the three types, namely the "small, medium, big" ones and then realizes the defuzzification accordingly so that small values are shifted to the left and big values are shifted to the right. From the logical point of view, given a multiformula $\mathbf{A}_x$ assigned to some formula $A \in F_J$ then the defuzzification means finding a crisp formula $\overline{A} \in F$ such that $T \vdash \overline{A}_x[r_0]$ for exacly one term r_0, i.e. $T \vdash_0 \overline{A}_x[t]$ holds for all terms $t \neq r_0$. Consequently, in each model $\mathcal{D}$ there is exactly one element $d \in D$ such that $\mathcal{D}(\overline{A}_x[r_0]) = \mathbf{1}$ and $\mathcal{D}(r_0) = d$.

It is specific for the inference method described above that it is purely logical and gives results in accordance with the intuition. To convince the reader, let us, for example, consider two rules

$$\mathcal{R}_1 := \textsf{IF } X \textsf{ is small AND } Y \textsf{ is small THEN } Z \textsf{ is big}$$
$$\mathcal{R}_2 := \textsf{IF } X \textsf{ is big AND } Y \textsf{ is big THEN } Z \textsf{ is small} .$$

These rules are interpreted by multiformulas

$$(\mathbf{Sm}_x \wedge \mathbf{Sm}_y) \Rightarrow \mathbf{Bi}_z \tag{19}$$
$$(\mathbf{Bi}_x \wedge \mathbf{Bi}_y) \Rightarrow \mathbf{Sm}_z . \tag{20}$$

Furthermore, let X, Y, Z be interpreted in a model $\mathcal{D}_0$ with the support $U = V = W = [0, 1]$. Then small values are some values among 0.3 (and smaller) and big ones some values among 0.7 (and bigger). Of course, given the input, e.g. $X = 0.3$ and $Y = 0.25$ then we expect the result $Z \approx 0.7$ due to the rule $\mathcal{R}_1$. Similarly, for $X = 0.75$ and $Y = 0.8$ we expect the result $Z \approx 0.25$ due to the rule $\mathcal{R}_2$. In the first case, we have

$$\mathbf{Sm}'_x = \{\mathbf{1}/Sm_x[t_0]\}, \tag{21}$$
$$\mathbf{Sm}'_y = \{\mathbf{1}/Sm_{y,i}[s_0]\} \tag{22}$$

where t_0 is a term representing the value 0.3 in $\mathcal{D}_0$, i.e. $\mathcal{D}_0(t_0) = 0.3$ and similarly s_0 is a term representing the value 0.25, i.e. $\mathcal{D}_0(t_0) = 0.25$. Then the inference rule r_{GMP} is applied on (21), (22) and the implication (19). The result is a multiformula $\mathbf{Bi}'_z$ which then must be defuzzified. The DOEE method finds a term representing satisfactory value being intuitively big, as expected.

The Zadeh-Mamdani fuzzy interpolation method with the center of gravity defuzzification in general cannot assure such a result. The reason is not in the wrong work of it, but in its different goal, which is approximation of a function, while the logical inference tries to find a logical conclusion purely on the basis of the given information, i.e. on the set of logical implications. We call the algorithm realizing the described inference the *inference with preselection of rules.*

To conclude this section, we also mention the problem of *linguistic context*, which plays an important role in the sequel. It is clear that the same word, for example "small", is used in various situations to denote completely different objects. Of course, "small man" is a man about 160 cm while "small mountain" may have about 600 m. Yet the general shape of the membership function, as demonstrated in many papers, is apparently the same since people understand this concept in the same way but apply it in different context. From the above discussed theory, this can well be understood as a difference between intension and various extensions. Consequently, different contexts mean the interpretation of the same intension $\mathbf{A}_x$ in different models $\mathcal{D}, \mathcal{D}', \ldots$ and obtaining different extensions $\mathcal{D}(\mathbf{A}_x), \mathcal{D}'(\mathbf{A}_x), \ldots$. A concrete model $\mathcal{D}$ will be called *a linguistic context.* Note that when applied to fuzzy control, the linguistic context is called *scaling* (see [4]).

3 Learning of linguistic descriptions for fuzzy control

An interesting question arises whether it is possible to learn the linguistic description. This problem has been solved by various methods, which mostly rely on the application of some clustering method and deriving fuzzy sets from the resulting clusters. This works with certain limitations, among which crucial is the use of the Zadeh-Mamdani fuzzy interpolation with the center of gravity defuzzification. The fuzzy sets used have mostly triangular shape and thus, they should be interpreted as fuzzy numbers. Other kind of interpretation of such fuzzy sets, for example "big, small" does not fit the linguistic feeling. Tuning of the rules then consists in modification of the position and shape of the fuzzy sets. Such a procedure is quite natural when accepting the interpretation of the triangular fuzzy sets as fuzzy numbers using which a certain function is (imprecisely) interpolated. However, a consistent linguistic interpretation requires predefied shapes which should not be changed. But then, the Zadeh-Mamdani interpolation with center of gravity defuzzification does not give convincing results.

Predefined shapes of the membership functions have several advantages. Among them, most important is good understanding to the linguistic description and closeness to the idea that fuzzy control should realize the description of the control provided by a human operator. A logical deduction seems to be most prefarable for fulfilling this goal.

In the sequel, we will suppose that the interpretation of the linguistic terms and the inference are those described in the previous section. Based on this, we describe a learning method which is able to provide good results for the control of stable processes.

The learning problem can be divided into two subproblems:

1. Learning of the linguistic description, i.e. finding a set of linguistic IF-THEN rules.

2. Learning of the linguistic context.

In the first case we suppose that we are given data from monitoring of the successful control of some process. Furthermore, we assume that the controlled process is stable. In the second case, we suppose to be given a linguistic description and a process to be controlled. The task is to find a linguistic context for all the variables such that the process were successfully controlled by the given linguistic description.

3.1 Learning of the linguistic description

The initial situation can be described in our terms as follows. Given is a dynamic process P, which is controlled manually so that the precise description of P is not known. The only known general characteristics of P is that it is a stable process.

Furthermore, let the data have been obtained from the monitoring of a successful control of P. The data have the following form:

$$\begin{aligned} c_1 &= \langle u_1, y_1, v_1 \rangle \\ &\vdots \\ c_n &= \langle u_n, y_n, v_n \rangle \end{aligned} \tag{23}$$

where u_i is the input to P in the time moment i, v_i is the required value and y_i is the output $y_i = P(u_i)$. Furthermore, we suppose that the control has proceeded on the basis of errors and their derivations. The error is the value $e_i = v_i - y_i$ and its difference (derivation) is $\Delta e_i = e_i - e_{i-1}$. The data then can be transformed into the following form

$$\begin{aligned} c'_1 &= \langle e_1, \Delta e_1, u_1 \rangle \\ &\vdots \\ c'_n &= \langle e_n, \Delta e_n, u_n \rangle. \end{aligned} \tag{24}$$

Definition 1 *Given a process P and the set (24). Let $\vartheta = \{v_{j_1}, \ldots, v_{j_k}\}$ be a set of set-points (required values) such that $j_1 = 1$ and $\varepsilon > 0$ be a (small) value such that to each j_l, $l = 1, \ldots, k$ there is i fulfilling the condition*

$$(\forall j)((j \geq i \wedge v_j = v_{j_l}) \Rightarrow |e_j| < \varepsilon). \tag{25}$$

Then the data (24) are called the ε-successful control strategy for P with respect to ϑ.

In words, a ε-successful control strategy means that the process P outputs reached all the set-points from ϑ with the accuracy ε.

From the point of view of the logical theory presented in the previous section, the process P can be represented by some model $\mathcal{D}$ of the fuzzy theory to be applied and the data (24) are elements from its support, i.e. $c'_i \in D = V \times W \times U$. Each value $e_i \in V$, $\Delta e_i \in W$, $u_i \in U$ is an interpretation of some term t_i, s_i, r_i, respectively.

In general, if a set-point v is given then we may compute the values e, and Δe. They correspond to some multiformulas $\mathbf{A}'_x$, $\mathbf{B}'_y$ of the form (17), (18), respectively. Let us suppose the following general form of the linguistic rules (7)

$$\mathcal{R} := \ \mathsf{IF}\ e\ \mathsf{is}\ \mathcal{A}\ \mathsf{AND}\ \Delta e\ \mathsf{is}\ \mathcal{B}\ \mathsf{THEN}\ u\ \mathsf{is}\ \mathcal{C}. \tag{26}$$

The *learning task* is to find a set of multiformulas $\mathbf{R}_{xyz,1}, \ldots \mathbf{R}_{xyz,m}$, each being an intension of the linguistic rule (26), on the basis of the data (24), which fulfils the following condition.

Given a set ϑ' of set-points. Starting from the initial error $e_1 = v_1 - y_1$ where $v_1 \in \vartheta'$ is the first set-point and y_1 is the initial state of the process P then there is a control strategy

$$\begin{aligned} c''_1 &= \langle e'_1, \Delta e'_1, u'_1 \rangle \\ &\vdots \\ c''_n &= \langle e'_n, \Delta e'_n, u'_n \rangle. \end{aligned} \tag{27}$$

for P with respect to ϑ' such that each $e'_i, \Delta e'_i$ correspond to multiformulas $\mathbf{A}'_x$, $\mathbf{B}'_y$ of the form (17), (18), respectively and $u'_i = \mathcal{D}(r_0)$ is an interpretation of a term r_0, so that $T \vdash \overline{C}'_z[r_0]$ where $\overline{C}'(x)$ is a crisp formula obtained from the multiformula $\mathbf{C}'_z$ by defuzzification and the latter is a result of the proof (16) from the premises $\mathbf{A}'_x$, $\mathbf{B}'_y$.

Note that in principle, the learning task starts with a model $\mathcal{D}$ (determined by the process P) and tries to find an appropriate fuzzy theory T, i.e. a fuzzy theory whose special axioms are given by the multiformulas $\mathbf{R}_{xyz,1}, \ldots \mathbf{R}_{xyz,m}$ so that proving in it may lead to a successful control strategy for any set ϑ of set points and some (acceptable) ε. In the following section, we will demonstrate that this is indeed possible.

The learning procedure consists in two main steps. First step: The set (24) of data obtained by monitoring of successful control may be quite extensive. The idea is to reduce (24), i.e. to transform it into a smaller set of the same form by grouping similar data together. The reason is not only the reduction itself. Typically, the data (24) contain several triples $\langle e_i, \Delta e_i, u_i \rangle$ whhich correspond to a given particular situation encountered by the control. These triples are (usually) similar. As the monitored control might be performed by a human operator, some of the similar triples correspond to more successful, some of them to less successful actions. It seems therefore to be reasonable to take some representative point (e.g. center of gravity) of the cluster of the similar triples. Moreover, it can happen that (24) contains also triples which correspond to errorenous actions. These should be avoided. The indicator of this situation is that the cluster of similar triples whichh contains the errorenous triple contains only this (or a few more) triple.

To perform the above described task we used a kind of self-organizing neural net based on competitive learning [1, 8]. We briefly describe the architecture and the learning algorithm. The net consists of two layers. The first layer has three input places where the signals (real numbers) $e(t)$, $\Delta e(t)$, and $u(t)$ appear in discrete time t. The second layer consists of several neurons (referred to by their indicies j). Each of the neurons is connected to each of the three input places. The connections of neuron j are assigned weights (real numbers) e^j, Δe^j, and u^j. Thus, both the input signals and the neurons represent points in $\mathbb{R}^3$. Each input signal is "recognized" by a unique neuron (that one that is closest to the signal). The idea is that the neurons are distributed in the space of input signals in such a manner that for each cluster of the input signals it holds that they are "recognized" by the same neuron. In this sense, a neuron represents a cluster if it "recognizes" all the signals of that cluster (not each neuron does necessarily recognize some cluster). The learning task is to find an appropriate distribution of the neurons in the input space. This is done by a kind of iterative process called competitive learning (see below). After the net is learned (the clusters are found), the clusters which contain only a (predefined) small number of elements are excluded. For each of the remaining (non-excluded) clusters c, its center of gravity $\langle e_c, \Delta e_c, u_c \rangle$ is computed. The computed centers of gravity form then the reduced data of the same form as (24). The formal descriptioin of the above procedure follows.

1. Initialize the weights $\langle e^j, \Delta e^j, u^j \rangle$ of the neurons j (e.g. so that they form a uniform grid in the input space).

2. Put subsequently the triples of (24) on the input of the net (after the last triple, the first one is put again).

3. For the i-th triple put on the input, adapt as follows: Select neuron j such that $||\langle e^j, \Delta e^j, u^j \rangle - \langle e_i, \Delta e_i, u_i \rangle||$ is minimal ($||\ ||$ is an appropriate norm, e.g. Euclidean). Input i has been recognized by neuron j. Adapt j by

$$\langle e^j_{\text{new}}, \Delta e^j_{\text{new}}, u^j_{\text{new}} \rangle := \langle e^j, \Delta e^j, u^j \rangle + \eta(\langle e_i, \Delta e_i, u_i \rangle - \langle e^j, \Delta e^j, u^j \rangle)$$

   where η is a (small) learning rate.

4. Stop learning when it stabilizes (i.e. the same inputs are repeatedly recognized by the same neurons).

5. For each neuron, if the relevance (the number of inputs recognized by the neuron) is sufficiently high, compute the center of gravity of all the recognized inputs.

6. The reduced data set consists of all computed centers of gravity.

The original data (24) is replaced by the reduced data. Since the form of both of the data is the same, in what follows we denote by (24) the reduced data.

Second step: A crucial idea consists in the linguistic analysis of the meaning of the linguistic terms and the model of their meaning considered above. In principle, given a value x, it may always be considered in a certain linguistic context. More specifically, the value can be small or big, or something in between. At the same time, we can always specify, what are the minimal and maximal sensible values, for example, when measuring body temperature then the minimal sensible value is 35°C and the maximal one is 42°C. Consequently, the linguistic context is a model $\mathcal{D}$ with the support $D = [d^{min}, d^{max}]$. But then there exists a most suitable linguistic term which characterizes the given value x (of course, with respect to the model $\mathcal{D}$). Stemming form this, learning of the rules (26) from the data (24) consists in finding suitable linguistic terms $\mathcal{A}_i$, $\mathcal{B}_i$, $\mathcal{C}_i$ to each value e_i, Δe_i and u_i, respectively for all $i = 1, \dots, n$. We have developed such a procedure.

From the logical point of view, to each multiformula $\mathbf{A}$ being intension of some linguistic term $\mathcal{A}$ there exists a set of terms $S \subset M_\iota$ being taken as typical for the meaning of $\mathcal{A}$. The mentioned procedure finds an assignement $\mathcal{D}(t) \in D$ for all the terms $t \in S$, given a model (i.e. a linguistic context) $\mathcal{D}$. We refer to this procedure as follows: given a term $t \in S$ and a term $\mathcal{A}$ above. Then t is a *typical term* for $\mathcal{A}$ and $\mathcal{A}$ is a *suitable linguistic term* for t.

Learning procedure. The learning will proceed on the basis of the data (24). We will generate the rules either of the form (26), or with the modification that the succedent is Δu is $\mathcal{C}$. The linguistic description with the rules (26) is usually called PD-fuzzy controller and the latter PI-fuzzy controller. In the case of PI one, the data have the form $c'_i = \langle e_i, \Delta e_i, \Delta u_i \rangle$ where $\Delta u_i = u_i - u_{i-1}$. Since the learning procedure is the same in both cases, we will stick to (24).

The learning procedure has the following steps:

1. Reduce the data (24) by the above described algorithm and replace (24) by the reduced data.

2. Given the data item $c'_i = \langle e_i, \Delta e_i, u_i \rangle$, we take it as an interpretation of three terms t, s, r, and find suitable linguistic terms $\mathcal{A}_i$, $\mathcal{B}_i$, $\mathcal{C}_i$ to $\mathcal{D}e_i$, respectively. The result is a rule $\mathcal{R}_i$ of the form (26).

3. Repeat the item 1. for all $i = 1, \dots, n$ and generate the linguistic description $\mathcal{R}_1, \dots, \mathcal{R}_n$.

4. Reduce the generated linguistic description due to item 2. as follows.

   (a) Replace all the identical rules by one only.

   (b) Let $\mathcal{R}_i$ and $\mathcal{R}_j$ be two generated rules such that all terms have the same sign and their succedents be identical. Let the antecedent of $\mathcal{R}_i$ be wider than that of $\mathcal{R}_j$. Then exclude the latter rule.

   (c) Let $\mathcal{R}_i$ and $\mathcal{R}_j$ be two generated rules such that all terms have the same sign and their antecedents be identical. Let the succedent of $\mathcal{R}_i$ be narrower than that of $\mathcal{R}_j$. Then exclude the latter rule.

3.2 Learning of the linguistic context

We suppose that we are given a process P, a linguistic description (5) whose rules have the intension (7). Learning a linguistic context means finding a model $\mathcal{D}$ so that the given linguistic description yields a ε-successful control strategy for some acceptable ε to each set ϑ of set-points.

Consider first the case of the context of the error e and its change Δe. The idea behind learning of e and Δe is very simple. We start with the assumption that *any non-zero error is big in the beginning.* Hence, the context is defined according to the value of initial error

$$e_0 = v - y_0$$

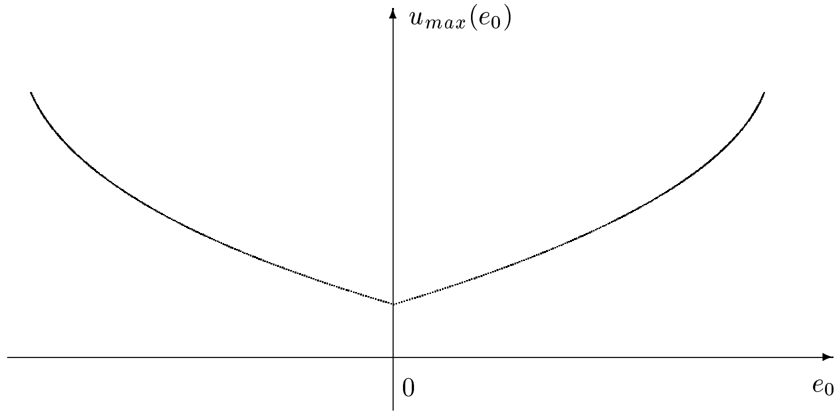

Figure 2: Dependence of optimal context on initial error

where y_0 is the initial process output and v is the required value. The experience shows that it works quite well if we put $V^- = [-k_e e_1, 0]$, $V^+ = [0, k_e e_1]$ and $W^- = [-k_{\Delta e} e_1, 0]$, $W^+ = [0, k_{\Delta e} e_1]$ where k_e and $k_{\Delta e}$ are constants not very far from 1.

Consider now the problem of finding the context for the control action. The automatic setting of the context for the control action is a difficult task. The context depends namely on the nature of the process to be controlled, and also on the technical device used for the control. Therefore, the context is expected to be set by an expert. To set the context for the (change of) control action, we can employ two approaches. First, the context is given by the maximal technically possible position of the "control cap". The second approach can be used when the simulation of the control is available. Then the context can be determined experimentally. We propose a modification of the second approach originally developed in [2]. We find experimentally the optimal context for two initial error values. Then, given the setpoint and the initial process output, new context value is determined which is used for the control. Let us suppose that the context for the control action is given by the intervals $U^- = [-u_{max}, 0]$, $U^+ = [0, u_{max}]$. In the case of change of the control action we prefer Δu_{max} instead of u_{max}. Thus, the context is fully specified by u_{max}. Clearly, different contexts lead to different behavior of the control. We try to find the optimal context value, i.e. the context value that leads to the best control given initial conditions (initial process output and setpoint). It follows from experiments and observations that there is a dependence of the optimal context value on the initial error $e_0 = v - y_0$. This dependence is usually of the form depicted in Fig. 2.

The optimal context of the (change of) control action can be therefore seen as a function $u_{max}(e_0)$ of e_0. We will call it *the optimal context value for* e_0.

We will adopt the following procedure. By experiment we find some significant points $e_{01}, \ldots, e_{0n}$ and the corresponding optimal context values $u_{max\,1}, \ldots, u_{max\,n}$. Due to symmetry we can restrict ourselves to positive initial error values. Interpolating these values we get the function $u_{max}(e_0)$. Linear interpolation seems to be sufficient, i.e. we have to set two initial errors. Given, e.g., $e_{01} = 0$, $e_{02} = e^1 > 0$, and the corresponding $u_{max}(0)$ and $u_{max}(e^1)$, we get

$$u_{max}(e_0) = u_{max}(0) + |e_0| \frac{u_{max}(e^1) - u_{max}(0)}{e^1} ,$$

the well-known linear interpolation formula. The absolute value $|e_0|$ assures the same values both for positive as well as negative values of e_0. As will be demonstrated, this method for determination of the optimal context yields both good raise time and small overshoot.

This approach seems to be quite natural. The linguistic contexts for the control action used by an operator are different for different initial errors. For example, by small initial error, the term *big control action* has certainly other meaning than by big initial error — the meaning of it stands for a "bigger" control action in the case of big error than in the case of small error.

4 Results of experiments

In this section we demonstrate the capability of our methods. They are demonstrated on the simulation of the control in the closed feedback loop. The simulation has been performed using the software LFLC (Linguistic Fuzzy Logic Controller) developed at the University of Ostrava. All the described methods are implemented in LFLC.

In Fig. 3, a succesfull PI control of a process described by $y'' + y' + y = u(t)$ is depicted. Monitoring this control, a data set of the form (24) has been obtained. Two methods of learning the rule base from the data have

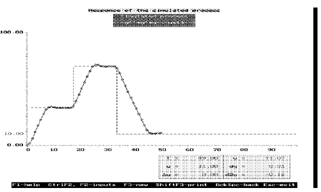

Figure 3: The succesful control performed by human operator.

been employed. First, the method of reducing the data by neural net and then the generation and simplification as described above have been used. In the second approach, the reduction by neural net has been omitted. The first approach yielded 10 rules, the second one 12 rules. An example of the control by the corresponding rule bases is depicted in Fig. 4. In the next example, PI control of the process described by the differential

Figure 4: Control of $y'' + y' + y = u(t)$. Left: rule base learned by using the neural net. Right: rule base learned without the neural net.

equation $y' + y = u(t)$ has been simulated. The values of further parameters were the following ones — $y_{min} = 0$, $y_{max} = 100$, sample period $t = 1$. This example demonstrates the capability of our first method. The contexts of the independent variables have been determined automatically by $K_e = 0.7$, $K_{\Delta e} = 2.0$. The context value of the change of the control action has been set experimentally to $\Delta u_{max} = 12$. The simulation results are depicted in Fig. 5. The left picture shows the control for the setpoint $v = 10$, the right picture shows the control for $v = 90$.

It can be said in general that using our methods we can exploit the same linguistic description for different controlled processes of similar nature. The difference between them is taken into account by employing of

Figure 5: Control of $y' + y = u(t)$ by learning context of independent variables.

different linguistic contexts. Fig. 6 shows the simulation of the control of the process described by $5y' + y = u(t)$, and $y_{min} = 0$, $y_{max} = 100$, $t = 1$. The same linguistic description as in the previous example has been used.

Figure 6: Control of $5y' + y = u(t)$ by only changing the context of the dependent variable. The same linguistic description as in the previous example is used.

The contexts of the independent variables have been set automatically again, $K_e = 0.7$, $K_{\Delta e} = 2.0$. The context value of the dependent variable has been set manually to $\Delta u_{max} = 4$. The context of the change of the control action is thus the only difference between the two controllers. The simulations of the control for $v = 10$, and $v = 90$ are in the left and in the rightpart of Fig. 6, respectively.

The capability of learning of the contexts of the independent variables can be significantly improved by tuning of the context of the dependent variable — our second method. Consider the process described by $1.5y'' + y' + y = u(t)$, $y_{min} = 0$, $y_{max} = 100$, $t = 1$. In the first case, we have employed learning of the context of the independent variables only ($K_e = 0.7$, $K_{\Delta e} = 2.0$). The context of the dependent variable has been set to $\Delta u_{max} = 2$. It was the best choice under the condition that the process has to be controlable to all setpoints $v \in [y_{min}, y_{max}]$. Simulation results for the setpoints $v = 10$, $v = 50$, and $v = 90$, are showed in the left part of Fig. 7. In the right part of Fig. 7, there are depicted the corresponding simulations performed by employing both learning of the context of e and Δe and learning of the context of Δu. Here we have set $K_e = 0.7$, $K_{\Delta e} = 2.0$, $\Delta u_{max}(0) = 0$, $\Delta u_{max}(e^1) = 9$, $e^1 = 50$.

References

[1] Arbib, M. A. (ed.): **The Handbook of Brain Theory and Neural Networks.** MIT Press, Cambridge (MA), London, 1995.

[2] Bělohlávek, R. and V. Novák: *Learning Linguistic Context for Linguistic Oriented Fuzzy Control.* Proc. FUZZ-IEEE'97, Barcelona, 1167–1172.

[3] Bělohlávek, R., Dvořák, A., Jedelský, D. and V. Novák: *Object Oriented Implementation of Fuzzy Logic Systems.* In: Camarinha-Matos, L. M., Afsarmanesh, H. and Marik, V. (eds.): **Intelligent Systems for Manufacturing: Multi-Agent Systems and Virtual Organizations**, Kluwer, Boston/Dordrecht/London 1998, pp. 589–594.

[4] Driankov, D., Hellendoorn, H., Reinfrank, M.: *An Introduction to Fuzzy Control.* Springer–Verlag, Berlin 1993.

[5] Klawonn, F., Kruse, R., *Equality Relations as a Basis for Fuzzy Control*, Fuzzy Sets and Systems **54**(1993), 147–156.

[6] Klawonn, F., Kruse, R., *Fuzzy Control as Interpolation on the Basis of Equality Relations*, Proc. IEEE Intern. Conf. on Fuzzy Systems, San Francisco, 1993, 1125–1130.

[7] Klawonn, F. and V. Novák: *The Relation between Inference and Interpolation in the Framework of Fuzzy Systems.* Fuzzy Sets and Systems **81**(1996), 331–354.

[8] Kohonen, T.: **Self-organization and Associative Memory.** Springer-Verlag, Heidelberg, 1984.

[9] Lee, C. C., *Fuzzy Logic in Control Systems: Fuzzy Logic Controller – Part I, Part II.*, Trans. on Systems, Man, and Cybern. **20**(1990), pp. 404–418; pp. 419–433.

[10] Mamdani, E.H.: *Applications of fuzzy algorithms for simple dynamic plant.* Proc. IEE 121(1974), 1585--1588.

[11] Novák, V.: **The Alternative Mathematical Model of Linguistic Semantics and Pragmatics**. Plenum Press, New York 1992.

[12] Novák, V.: *Linguistically Oriented Fuzzy Logic Controller and Its Design.* Int. J. of Approximate Reasoning 1995, **12**(1995), 263–277.

[13] Novák, V. and I. Perfilieva: Evaluating Linguistic Expressions and Functional Fuzzy Theories in Fuzzy Logic. In: L. A. Zadeh and J. Kacpryk (eds.): **Computing with Words in Systems Analysis**. Springer-Verlag, Heidelberg 1998 (to appear).

[14] Novák, V., Perfilieva I. and J. Močkoř: **Mathematical Principles of Fuzzy Logic**. Kluwer, Dordrecht 1999 (to appear).

[15] Zadeh, L. A., *A Rationale for the Fuzzy Control.*

[16] Zadeh, L. A., *The concept of a linguistic variable and its application to approximate reasoning I, II, III*, Inf. Sci. **8**(1975), 199– 257, 301–357; **9**(1975), 43–80.

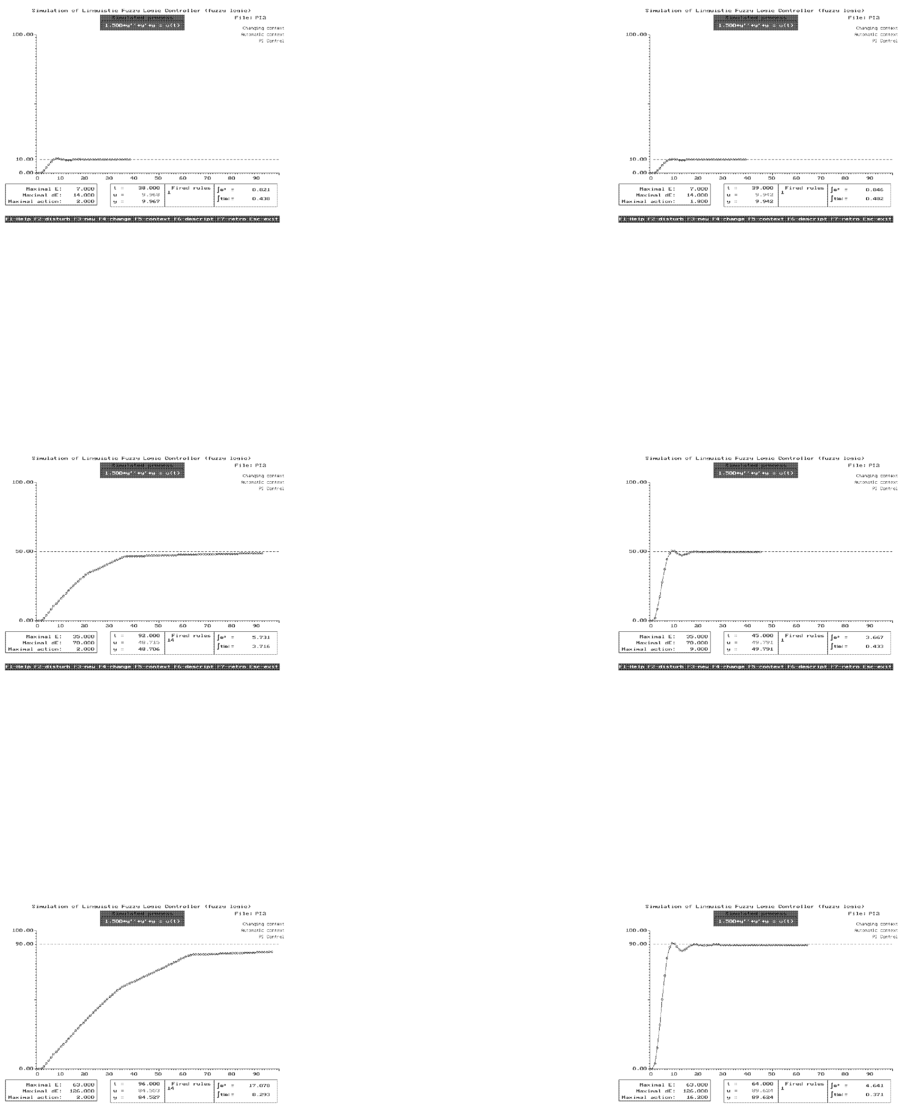

Figure 7: Control of $1.5y'' + y' + y = u(t)$ to setpoints $v = 10, 50, 90$. The control in the right part employs learning of the change of the conrol action, the control in the left part does not.