Building rule base of fuzzy controller from empirical data

P. Vysoký, Z. Pechal
Department of Control Engineering, Faculty of Electrical Engineering
Czech Technical University, Technická 2, 166-27 Praha 6, Czech Republic
e-mail: vysoky@lab.felk.cvut.cz, pechal@electra.felk.cvut.cz

March 28, 1996

1 Introduction

The kernel of any fuzzy controller (both Mamdani and Sugeno type) is a rule base. There are several methods for determining of appropriate rules in common use. Methods which enable exact synthesis of rules in order to obtain prescribed behaviour of the controller (to fulfill quality requirements, dynamic behaviour etc.) like analytical controller design for conventional controllers are in the stage of theoretical considerations yet. For practical applications the use of empirical methods is necessary. We usually use the fuzzy regulator to substitute a human operator controlling a system with unknown mathematical model.

In this case, rules may be obtained by the application of different methods of knowledge acquisition. An experienced expert or skilled operator is interviewed and asked which rules he uses in a specific situation to obtain a successful decision. But even a successful operator is often not able to describe his decision processes satisfactorily. He is not able to formulate all rules explicitly. One possible method is to observe the operator's activity, to store values of all variables observed by the operator and his corresponding decisions and then to try to find appropriate rules by analysing the data.

The most popular technique for this purpose is the use of neural nets. The suitable neural net can learn operator's behaviour by means of a training set containing measured data and decisions. We do not find specific rules. They are distributed as well as membership functions in connections (weights) into the whole net. From the point of view of controller design or maintenance there are two serious problems in this approach.

The neural net can learn all rules only with help of a training set. We have usually a little information on the structure of the training set, if all possible states of the controller are represented in this set. Similarly we have little knowledge if all areas of the controller state space are occupied by any rules. The nonexistence of rules in specific areas is manifested by the malfunction or failures of the controller. Correction needs new learning with additional training set. It leads us to the second problem. In industrial applications we usually have some prior information. (An operator can formulate many rules explicitly.) Due to the distributed structure of the neural net, integration of prior information with information obtained by learning, is very limited. It is a great disadvantage for tuning of industrial controllers.

Measurements of pertinent variables are usually noisy. It limits the use of methods like [1], based on specific weighting of possible rules. This and the above-mentioned facts are main reasons leading to the attempt to utilize statistical methods for the search for rules using empirical data. But this way is also not straightforward.

Our first problem is the fact, that the dimension of the controller's state space need not correspond with the number of variables measured or observed by an operator. Sometimes the operator's decision depends on variables that are measurable indirectly. For instance decisions of an operator controlling a biotechnological process depend on colour and smell of the mixture in the reactor and these variables are manifested as concentrations of specific liquid and gaseous components [2]. From this point of view it is difficult to distinguish whether all variables necessary for decision are available. On the other hand, the controller is a dynamical system and usually it is necessary to consider previous values of these variables. Usually it is impossible to assume, that the operator would be familiar with differential equations. His decision depends on history of these variables often subconsciously and he is usually unable to express these dependencies explicitly in the rules. To obtain these missing delayed variables we have to identify the structure of the model describing dynamical behaviour of the operator, the structure of the rules.

The information necessary for decision is carried by certain variables and their specific delayed values. Having no knowledge on a mathematical model of the controlled system, we try to measure all variables potentially carrying information for control. It is possible that in the set of measured variables there are variables carrying negligible or zero information and/or any variables are redundant. Our aim is to choose all variables relevant for control decisions and to exclude all others. The number of all relevant variables determines dimension of controller state space. On this initial level we deal with different kinds of variables. An illustrative example is on table 1.

Table 1: Data carrying information for operator's decision

k	v_1	v_2			v_3	v_4	v_5	v_6
		A	B	C				
1	0.13	0.1	0.5	0.4	Medium	FIL	135	0.15
2	0.08	0.2	0.4	0.4	High	FIL	86	0.75
3	0.26	0.3	0.5	0.2	High	CH_4	32	0.48

The table contains data necessary for control of a biotechnological process, which consist of values of six variables: v_1 is, e.g. the concentration of the substrate. This is a crisp metric variable exactly measurable. v_2 is the colour of the batch subjectively perceived by operator. This variable is a fuzzy variable.

The following variable v_3 (level of foam) is taken as ordinal crisp variable. Next variable, v_4 (occurrence of bacterial colonies in filament form and occurrence of CH_4 bubbles) is a crisp nominal variable. Remaining two variables v_5 and v_6 are also crisp metric variables. As we will see later, a general tool for processing so different kinds of data provide, information measures especially entropy and mean mutual information.

2 Structure identification of the rule base

For the choice of relevant variables we will use an approach known as General Systems Problem Solver (GSPS) according to [3]. It is out of frame of this paper to describe GSPS in the whole. We try to outline only the specific part useful for our purpose. Let us consider that we have a table containing values of variables potentially describing an operator's behaviour. For simplicity let us assume initially that all variables are crisp metric variables as on the table 2.

As we can see later our considerations may be extended for all kinds of previously mentioned variables. Let us consider further that a requisite (from the statistical point of view) amount of data is available and that the variables are optimally sampled in time. (One method for determining an optimal sampling frequency for nonmetric variables is in [4]). We have also a valuable prior information – we know which variable represents operator's decisions, it is the output of our identified system. Our task is now to distinguish which variables and their delayed states carry the necessary information.

Table 2: Activity matrix

k	x_1	x_2	x_3	$\cdots$	x_n	y
1	0.15	13.2	18.5	$\cdots$	24.8	31.7
2	12.5	133	4.3	$\cdots$	8.6	25.6
3	8.6	2.5	27.6	$\cdots$	13.2	8.1
4	33.9	8.6	0.76	$\cdots$	6.3	2.2
5	3.2	45.7	87.6	$\cdots$	0.37	8.9
6	25.6	0.18	15.6	$\cdots$	3.2	2.7
7	2.87	8.17	0.05	$\cdots$	34.3	7.6

The matrix as on the table 2 is called the activity matrix of the system. If the interpretation of all variables is available (the number of quantising levels, units for their measurements, the domain of their possible values etc.) then both activity matrix and interpretation provide data system in Klir's terminology. To obtain a generative system – mathematical relation among variables, which allows to generate the same data as in the activity matrix the variables carrying maximum information for the examined variable must be chosen. An actual value of the output variable is the generated element and actual or delayed values of relevant variables are generating elements.

The generated element and generating elements are connected by the translation rule for instance as

$$x_6(k) = f(x_6(k-1), x_3(k), x_3(k-1)) \tag{1}$$

The translation rule corresponds with a specific matrix called mask. Generally the mask is a matrix $v(d+1)$ where v is the number of variables and d is the depth of memory.

The mask corresponding with translation rule 1 is on the table 3.

The elements of a generative mask according to [5] are zero, negative or positive, meaning "neutral element", "generating element" and "generated element" respectively. There are many possible masks. Among these masks there are one or several masks that choose all necessary variables for description of operator's behaviour considered as optimal masks. There are different approaches to find an optimal mask. For the overview we refer [3], [5], [6].

Table 3: Mask corresponding to translation rule 1

x_1	x_2	x_3	$\cdots$	x_n	y	
0	0	0	$\cdots$	0	0	$k=2$
0	0	-1	$\cdots$	0	-2	$k=1$
-3	0	-4	$\cdots$	0	$+1$	$k=0$

In our case, we apply for simplicity the "classical" method [7], [8]. In principle the method is the following. We start with actual value of variable y and we find which actual or delayed values of any other variable in the mask carry maximal information for y. Information carried by variable x for output y is measured with help of mean mutual information

$$T(X:Y) = H(X) + H(Y) - H(X,Y) \tag{2}$$

where

$$H(X) = -\sum_{j=1}^{q} P_j \log_2 P_j \tag{3}$$

is the entropy, q is the number of quantising levels for the appropriate variable. Entropy is usually estimated from frequencies as

$$\hat{H}(X) = \log_2 N - \frac{1}{N}\sum_{j=1}^{q} N_j \log_2 N_j \tag{4}$$

where N_j are frequencies of occurrence of individual values of relevant variables.

The algorithm for the search of variables constituting the optimal mask is the following. We find $\max\{T(Y:X_i)\}$, where $i = 1,2,\ldots,v(d-1)$. Let us imagine, that the maximum will be obtained for x_{i0}. Now we find maximum information carried by the pair x_{i0}, x^j, where $j = 1, 2, \ldots, v(d-1), j \neq i_0$. We obtain x_{j0}. In the following step we find maximum information carried by the triplet x_{i0}, x_{j0}, x_k for $k = 1, 2, \ldots, v(d-1), k \neq i_0, k \neq j_0$, etc. More information on the practical application of the presented method is in [7], [8].

In an ideal case the process is terminated if any subsequent variable does not carry additional information. In a real case, especially for dynamical systems, the problem is not so simple. If for instance $y(k)$ depends on $y(k-1)$, then $y(k-1)$ depends on $y(k-2)$ and $y(k-2)$ carries some information for $y(k)$. We must examine, whether $y(k-2)$ carries some information for $y(k)$, excluding variability of $y(k-1)$. It may be accomplished with help of conditional mean mutual information.

$$T(Y(k) : Y(k-1), Y(k-2)) - T(Y(k) : Y(k-1)) = T(Y(k) : Y(k-2) | Y(k-1)) \tag{5}$$

If $y(k-2)$ is relevant with respect to $y(k)$, then conditional information (5) is zero. Because we do not deal with entropies but with estimates of entropies, we usually obtain a small nonzero value of the mean mutual information. Now it is necessary to distinguish whether this small value will be considered as zero or not. In other words it leads to testing of the hypothesis "both variables are independent" against the alternate hypothesis "they are not independent". The mean mutual information may be transformed into variable with approximately χ^2 distribution and the χ^2 test may be used for this purpose. The problematic of testing of entropies and information and calculation of degrees of freedom is also out of frame of this paper. For the overview of this topic we refer to [9].

Using this method, we obtain a suboptimal mask, [7]. Sometimes we obtain several equivalent masks. The choice of an appropriate mask depends on the experience of the designer and on the prior information, for instance on obstacles in the measurement of individual variables. Having the mask, we have all necessary variables or in other words we have the dimension of the controller state space.

3 Extension for fuzzy variables

It is clear from table 1 that we deal with different kinds of variables. For all kinds of crisp variables (metric, ordinal, nominal) the use of informational measures as entropy and mean mutual information is fully acceptable. Fuzzy variables need more careful approach. A fuzzy variable is often (especially in control engineering applications) an artificial construct. The original variable is a crisp, metric variable and it is secondary fuzzified. There are no problems with this kind of fuzzy variables. They may be considered for structure identification as crisp variables, especially if their quantification corresponds with the number of primary fuzzy sets used for their fuzzification. (For recommendations concerning quantification see [10]).

Their membership functions are usually estimated from empirical data with help of some clustering method (algorithms as ISODATA, fuzzy C-MEANS etc.), or an algorithm finding the "natural" number of clusters may be applied [11].

Let us note, that selected fuzzy sets have to satisfy the condition of ε-completnes. The union of their supports should cover the relevant universes in relation to some level set ε. This level corresponds with membership degree in crossover points of neighbour membership functions. (The level of the crossover point is usually chosen greater then 0.5). Crossover points coincide with boundaries among individual cells of the state space. Having crossover points, we have also the partition of the state space. Among crossover points (in the specific cell of the state space) a dominant rule always exists (fig. 2). These intervals may be considered as quantising levels of a relevant crisp variable and we can determine frequency of occurrence and calculate necessary entropies in the same manner as for crisp variables.

There are variables that have to be considered as fuzzy variables in their nature. The data available are as variable v_2 on table 1. Here a little different approach must be applied.

Let the standard probability space be (Ω,K,P). Here Ω is a sample space, K is the complete σ algebra of a subsets of the Ω, and P is a probability measure. For the fuzzy event F with the membership function $\mu_F(\omega)$ we obtain according to [12] for the probability

$$P(F) = \int_{\Omega} \mu_{\mathrm{F}}(\omega) dP \qquad \mu_{\mathrm{F}}(\Omega) : \Omega \rightarrow (0,1) \ . \tag{6}$$

If we have a discrete sample space $\Omega = \{\omega_1, \omega_2, \ldots, \omega_n\}$, then the probability will be:

$$P(F) = \sum_{i=1}^{n} \mu_{\mathrm{F}}(\omega_i) P(\omega = \omega_i) \ . \tag{7}$$

This is summation of probabilities $P(\omega = \omega_i)$ multiplied by the degree to which ω_i belong to the fuzzy event F. In realistic situations the probability is estimated by the category frequency of ω_i and we obtain the relative pseudofrequency of the event F as an estimate of the probability of the fuzzy event. The absolute pseudofrequency is a sum of all degrees of membership of elementary events belonging to one fuzzy set. E. g. for ω_1 and the fuzzy set A

$$N(\omega_1) = \sum_{i=1}^{n} \mu_A^i(\omega_1) \ .$$

Normalization with respect to the sum of all absolute pseudofrequencies provides a relative pseudofrequency as an estimate of the probability of a fuzzy event. To estimate the multidimensional probabilities it is necessary to aggregate relevant states. The membership degree of an aggregated state is determined from "marginal" memberships with help of an aggregation function. There are many possible aggregation functions [3]. We use a product for computing membership degree of aggregated state. Obtaining all necessary pseudofrequencies we can calculate relevant entropies and mean mutual information.

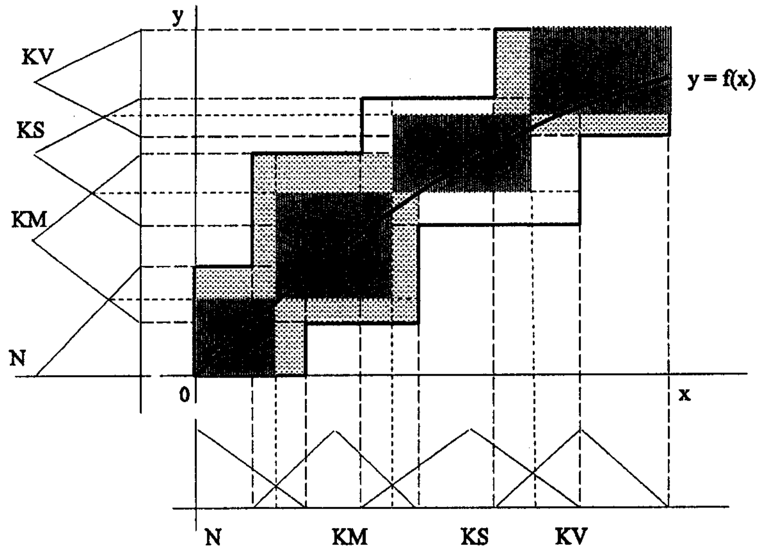

Figure 1: Areas corresponding to relevant rules

According to [13] it is also possible to use the χ^2 test for fuzzy data. Let us consider that $P(x)$ is assumed to be a theoretical probabilistic distribution.

Let $\Pi(k)$ be:

$$\Pi(k) = \int_x \mu_k(x) dP(x) \ . \tag{8}$$

This is Zadeh's distribution induced by theoretical distribution $P(x)$ on the k-th fuzzy set with membership function $\mu_x(x)$, defined on the universum x. $\gamma(k)$ are experimental pseudofrequencies for the fuzzy sets $k = 1, 2, \ldots, m$ and n is the number of measurements. In [12] it is proven that the statistics:

$$\sum_{k=1}^{n} \frac{[\gamma(k) - n\Pi(k)]^2}{n\Pi(k)} = \chi^2_\nu \ . \tag{9}$$

has an approximate χ^2 distribution with $r - 1 = \nu$, where r is a cardinality of X.

From these relations follows that there are no principal restrictions in the extension of the presented method for fuzzy data. We deal with pseudofrequencies instead of frequencies and we obtain estimates of probabilities as well as for crisp data.

4 Search for rules

Now we have the state space of the controller partitioned on cells and we can assign appropriate outputs to the individual cells and form rules. Due to the overlapping of the neighbour fuzzy sets and due to uncertainties in measurement, the assignment of the appropriate output to the specific cell is ambiguous.

It is natural to select as a rule the most frequented coincidence between specific cell and specific output. It leads to the multidimensional histogram or histogram depicting the pseudofrequencies.

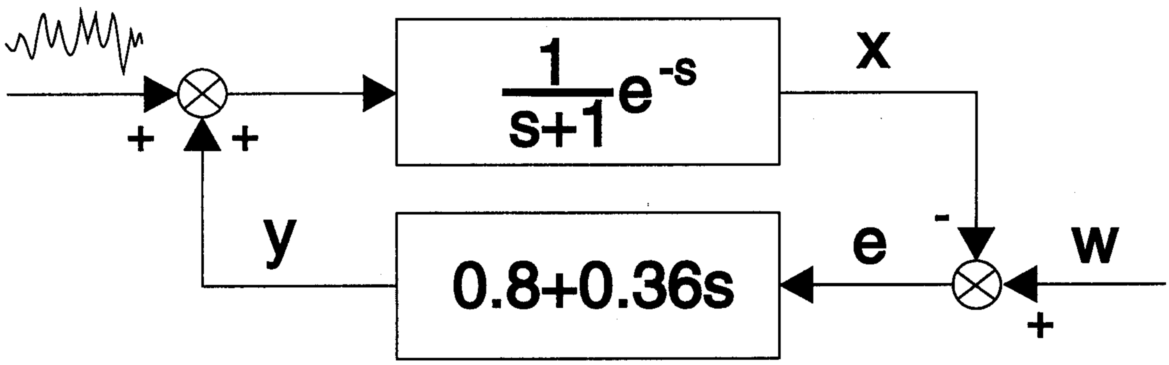

Figure 2: A simple control system for demonstration of the method

A technique can be shown with help of the following example. A simple control system according to fig. 2 has been simulated.

Table 4: Frequencies of coincidences between cells and output values for estimation of the rules

cell		abs. freq.			rel. freq.			rule
$e(k)$	$\Delta e(k)$	N	Z	P	N	Z	P	
N	N	18	0	0	1.00	0.00	0.00	P
N	Z	40	5	0	0.89	0.11	0.00	P
N	P	2	8	0	0.20	0.80	0.00	Z
Z	N	5	14	0	0.26	0.74	0.00	Z
Z	Z	0	34	3	0.00	0.92	0.08	Z
Z	P	0	6	9	0.00	0.40	0.60	Z
P	N	0	6	3	0.00	0.67	0.33	Z
P	Z	0	6	32	0.00	0.16	0.84	N
P	P	0	0	9	0.00	0.00	1.00	N

A conventional PD controller has been used and the system was stimulated by the white noise. The sampled corresponding inputs and outputs of the controller have been stored. The data set consists of 1000 samples of input and output. Analysis of the mask showed, that for $y(k)$ we need $e(k)$ and $e(k-1)$. The variables $y(k)$, $e(k)$ and $e(k)-e(k-1)$ were quantised into three levels (assigned with linguistic terms "positive"=P, "zero"=Z, "negative"=N). Frequencies of coincidences between cells and output are in the table 4.

Choosing the most frequented rule as a typical rule for the specific cell of the phase space, we obtain the following rule base, table 5.

Table 5: Final form of the rule base

		Δe		
		N	Z	P
e	N	P	P	Z
	Z	Z	Z	Z
	P	Z	N	N

5 Conclusion

The method submitted enables to find the control rules if any controller able to control a given system exists. For instance, if an expert who can control the given system exists, but is unable to formulate the rules explicitly. The method is important especially in situations where more complicated controllers are applied, a little information on dynamical properties of controlled system is available and measurements are indirect and noisy. For instance by the control of biotechnological processes connected with water purification. The method has little significance for the design of fuzzy PI, PD, PID controllers. This is the domain of the template based methods.

Having all relevant variables after the structure identification, the neural nets approach may be used for finding the rules and the membership functions. Nevertheless a simple method for the choice of rules on the base of their frequency (or pseudofrequency) provides information on statistical properties of the training set if it contains data corresponding with all necessary states. Using neural nets, these properties are usually not known, what leads sometimes to erroneous results.

References

[1] Wang. L. X., Mendel J. M.: Generating fuzzy rules by learning from examples. IEEE trans. SMC, 1992, 22, 6, 1414-1427.

[2] Flaus.J.M.,Cheruy A., Engasser J.M., Poch M., Sola C.: Estimation of the state and parameters in bioprocesses from indirect measurement. Proc. IFAC European control conf. Grenoble, 1991, 1642-1647.

[3] Klir G.J.: Architecture of systems problem solving. Plenum Press, New York, 1985.

[4] Vysoky P.: Sampling qualitative and fuzzy variables as a tool for reconstructability analysis. In: Cybernetics and systems, Trappl R. ed., World Scientific, Singapore, 1994.

[5] Cellier F.E.: General system problem solving paradigm for qualitative modeling. In: Qualitative simulation, modeling and analysis, Fishwick P.A. and Luker P.A. eds., Springer, New York, 1991.

[6] Klir G.J., Way E.C.: Reconstructability analysis: Aims, results, open problems. Systems Research 1985, vol. 2., no. 2, 141-163

[7] Connant R.C.: Structural modeling using a simple information measure. Int J. System Sci. 1980, vol. 11., no.6.,721-730.

[8] Connant R.C.: Exteded dependency analysis of large systems. Part I-II. Int. J. Gen. Syst. 1988, vol.14., no.2., 97-141.

[9] Reinke W., Diekmann V.: Uncertainty analysis of human EEG spectra: A multivariate information theoretical method for the analysis of the brain activity. Biological Cybernetics, 1987, vol. 57, 379-387.

[10] Valdes-Perez R.E., Connant R.C.: Information loss due to data quantization in reconstructability analysis. Int. J. Gen .Syst. 1983, vol.9., no.4., 235-247.

[11] Yaeger R.R., Filev D.P.: Essentials of fuzzy modeling and control. J. Willey, New York, 1994.

[12] Zadeh L. A. : Probability measures of fuzzy events. J. Math. Anal. Appl. 1968, 23, 421- 427.

[13] Gill M.A., Corral N., Cassals M.R. : The likelihood ratio test for goodness of fit with fuzzy experimental observations. IEEE trans SMC, 1989, vol. 19., no. 4., 771-779.