FUZZY MODEL IDENTIFICATION BY MEANS OF MULTIOBJECTIVE GENETIC PROGRAMMING

B. Bica, A. J. Chipperfield, and P. J. Fleming

Department of Automatic Control & Systems Engineering,
The University of Sheffield
Mappin Street, S1 3JD, Sheffield, UK, e-mail: Beatrice@acse.shef.ac.uk

Abstract - Fuzzy model identification of high-dimension, non-linear systems is a complex, non-trivial task. Current techniques for designing Takagi-Sugeno (TS) fuzzy systems deal with the choice of input-output variables and the number and shape of membership functions whilst assuming that predefined linear structures will locally represent the system under investigation. A method for the derivation of optimal fuzzy system structures based upon non-linear local system representations is proposed. This technique also aims to reduce model complexity. In order to tackle the issue of structural optimality of the fuzzy system, a multiobjective genetic programming approach is used in this study. *Copyright © 2000 IFAC.*

Keywords: fuzzy modelling, multiobjective optimisation, genetic programming.

1. INTRODUCTION

Most real systems are non-linear and their behaviour often cannot be captured satisfactorily by linear systems. The increasing interest in non-linear dynamic systems identification is therefore justified. However, the issue of non-linear modelling is more difficult than its linear counterpart and research is still active in the field to find feasible solutions to the problem. This paper seeks the development of a methodology for non-linear system identification by means of fuzzy logic. Furthermore, it is shown that fuzzy modelling can be more effective if local non-linear approximations of the plant are used, rather than linear approximations. Previous work [1,2] has demonstrated that the choice of the fuzzy controller structure (polynomial order, number of terms) can affect the system performance. The present study attempts the extension of previous work for modelling problems.

Typically, the Takagi-Sugeno (TS) system expresses the outputs y_i as linear combinations of the inputs x_i, e.g.:

Rule i: IF x_1 is A_{i1} AND x_2 is A_{i2} THEN $y_1 = a_{01} + a_{11}x_1 + a_{21}x_2$ and $y_2 = a_{01} + a_{11}x_1 + a_{21}x_2$ (1)

where A_i is the linguistic label associated with a membership function. With this formulation, a non-linear system can be represented as a piece-wise linearized system, around specific operating points. However, it should be emphasised that for high dimension non-linear systems this kind of linear approximation may be unable to reflect the true non-linear character of the system. Even if the fuzzy system variables undergo optimisation, the original system/controller may not be accurately approximated by linear functions. Moreover, by partitioning the input parameter space into more regions in an attempt to increase the accuracy of the mapping, the complexity of the fuzzy rule-base and of the inference algorithm will also increase. The designer is confronted with the well-known phenomenon of "rule explosion". With an augmented rule-base to be handled, and a larger number of partitions of the space, the prospect of successfully optimising the fuzzy system parameters is less likely. Additionally, it is important to note that it is not certain that all the inputs have a contribution towards each output of the fuzzy system, i.e. the structure selection aspect. A solution to the problem, which simultaneously caters for the "curse of dimensionality" issue, is to alter the degree and the structure of the polynomial describing the system outputs, whilst maintaining a minimal number of design parameters (rules and membership functions). To represent complex systems correctly at both local and global level, higher-order polynomials in the consequent TS rules, comprising only the significant terms, might be required to capture the non-linearity of the system.

A genetic programming approach appears to be an appropriate tool for search and optimisation of the model configuration, particularly due to its ability to manipulate structures, rather then simply parameters [3]. In order to accommodate multiple objectives, the optimisation is treated in a Pareto-optimal fashion. The approach is exemplified on a chaotic dynamic system, the glycolytic oscillator [4]. Experimental results indicate that the structure derived via genetic programming exhibits very good modelling accuracy.

2. FUZZY MODEL IDENTIFICATION

A few methodologies for fuzzy system identification have already been established in the fuzzy research community, including clustering methods, neural networks and genetic algorithms.

The clustering methods involve the organization of the design points in a number of classes, variable in number or location in the search space. Each identified cluster is then associated with a fuzzy rule (or metarule, as named by Pedrycz [5]), which describes a linear approximation of the overall fuzzy hypersurface. The fuzzy clustering methods reported in the literature generate the fuzzy antecedent and consequent parameters, but the manner in which they are inferred varies. For instance, Pedrycz [5] uses c-mean clustering to derive the antecedent membership functions, and a fuzzy relational matrix to determine the consequence. In contrast, Babuška and Verbruggen [6] derive the consequent parameters for each cluster and through fuzzy matrix projection infer the antecedent membership functions. Park *et al* [7] proposed an alternative approach, where the identification process consists of two distinct phases. Initially, the coarse tuning of both consequent and antecedent parameters is performed, via hyper-plane c-means clustering (a variation of the original c-means algorithm proposed by Bezdek, [8]). The second step involves fine tuning with a gradient descent method of the antecedent and consequent parameters found at the former step. The approach proposed by Park is similar to the Babuska-Verbruggen technique in the sense that estimates first the consequent parameters and then infers the antecedent parameters. Various other clustering approaches have been reported by Zhao *et al* [9], Pal *et al* [10], Delgado *et al* [11].

The neuro-fuzzy synergism still remains very appealing for system approximation, particularly due to the excellent learning capabilities of the hybrid structure, combined with their inherent "universal approximator" character. From a fuzzy system identification standpoint, neural networks can be used to perform parameter training and rule generation. Wang, Hollatz, Guély *et al* [12-14] etc. employ feedforward neural networks to model fuzzy systems and use gradient descent for parameter tuning. The approach allows the alteration of the input-output membership functions until an optimal solution is found. Neural networks can be also used for the data clustering, which lead to the membership function realisation [15].

With regard to fuzzy model identification, a wide range of genetic algorithm based techniques has been presented in the literature [16-18]. GAs, as search and optimisation algorithms manipulating a population of potential solution simultaneously, are sought as the appropriate mechanism for a more effective and robust exploration of the parameter space. In many cases, the genetic identification of the fuzzy system is synonymous with parameter tuning. Indeed, most of the proposed methods aim at the tuning of the fuzzy system parameters (scaling factors, membership function, rule base), usually by minimising an error function.
A number of the approaches deal with multiobjective genetic optimisation in order to obtain more precise fuzzy models [19-20]. Few though, have employed the Pareto-optimality paradigm for system tuning [21].

The genetic programming concept, as a powerful branch of GAs, presents a huge potential for system identification in general [3,22], and for fuzzy system identification in particular. However, this research area is relatively unexplored. Only one paper has been found that discusses the use of GP in conjunction with fuzzy systems [23], and the study uses GP for fuzzy rule-base extraction.

3. FUZZY MODELS OF DYNAMIC SYSTEMS

Wang defines a dynamic fuzzy system as the fuzzy system in which the output appears as one of its inputs, i.e. the system comprises recurrent rules in the form:

If $y(k)\,is\,A_{1,i}$ *and* $y(k-1)\,is\,A_{2,i}$ *and* ... *and* $y(k-n_y+1)\,is\,A_{n_y,i}$ *and* $u(k)\,is\,B_{1,i}$ *and* $u(k-1)\,is\,B_{2,i}$ *and* ... *and* $u(k-n_u+1)\,is\,B_{n_u,i}$ ***then***

$$\hat{y}(k+1)=\sum_{j=1}^{n_y} a_{i,j}\,y(k-j+1)+\sum_{j=1}^{n_u} b_{i,j}\,u(k-j+1)+c_i \tag{2}$$

Each such rule describes a local ARX (AutoRegressive with eXogenous input) model. The typical approach used in fuzzy dynamic modelling is to predefine the structure, so that linear approximations are described through each local ARX model. However this is a very rigid, maybe inaccurate, definition of the structure, which would lead to a poor representation of the system. Furthermore, it is recognized that model structure selection plays a very important part in identification and hence this aspect should not be overlooked in the design of fuzzy systems. Regarding model selection, the identification of the consequent Takagi-Sugeno structure should follow the same principles as traditional system identification (SI), where the non-linear models have the ability to reflect better the system behaviour. However, similar to the classical SI techniques, the difficulty in finding suitable non-linear models increases accordingly.
Another common dynamic system representation is given by the state-space modelling, where each rule represents a local discrete-time model:

If $x(k)$ *is* A_i *and* $u(k)$ *is* B_i *then*

$$\begin{cases} x(k+1)=A_i\cdot x(k)+B_i\cdot u(k) \\ y(k)=C_i\cdot x(k) \end{cases} \tag{3}$$

where $x(k)$ and $u(k)$ are the state and the input of the system at the k time instant.

A chaotic system is the subject of the current investigation into the fuzzy model identification. The choice of a chaotic system is explained by the fact that the technique presented in this work aims at the investigation of fuzzy modelling for highly non-linear systems, with complicated behaviour, where an appropriate structure selection could be of paramount importance.

4. OPTIMISATION USING MULTIOBJECTIVE GENETIC PROGRAMMING (MOGP)

In general terms, evolutionary algorithms replicate the Darwinian process of natural evolution by progressively improving populations of potential solutions according to the philosophy of "survival of the fittest". Perhaps the most familiar representatives of evolutionary algorithms are the genetic algorithms (GAs) proposed by Holland in 1975.

Genetic programming (GP) [3] represents a branch of evolutionary computation, more specifically a subclass of genetic algorithms. The most prominent feature that differentiates GP from GA is the type of genotype (individuals), which they handle. Whilst a GA uses a binary or real-valued encoding of the individuals, computer programs represent the GP's genotype. The genetic operators manipulated by GA or GP approaches are virtually the same. However, they are different in the way they act and in the results they produce - dependent on the individual's type.
For the traditional GA, selection, crossover and mutation preserve the individual's structure (length, content).
Since the main entity of the GP is a program, the outcome of genetic operators acting on a population of individuals is new programs, structurally different from the parental population. The GP operates with a terminal set, comprising variables, and a function set, consisting in mathematical or logical operators.

The Pareto-optimality scheme is frequently employed to solve multiobjective optimisation problems. The Pareto-optimal philosophy has been demonstrated to be a very realistic and versatile approach for tackling multiobjective optimisation problems. It clearly outperforms traditional non-linear programming methods, such as epsilon-constraint, weighted sum or goal attainment. Additionally, Pareto-optimal approaches are able to handle multimodality and discontinuities in the function space, a deficiency in other non-linear programming techniques.

In most cases there will not be one ideal 'optimal' solution, rather a set of Pareto-optimal solutions for which an improvement in one of the design objectives will lead to a degradation in one or more of the remaining objectives. Such solutions are also known as non-inferior or non-dominated solutions to the multiobjective optimisation problem. Similar to the multiobjective genetic algorithms approaches, goals and priorities information may also be embedded into the GP structure, in order to distinguish the solutions corresponding to certain demands.
For this study, the MOGP is deemed the ideal environment for the simultaneous consideration of multiple objectives, modelling several objective functions. The resulting trade-off solutions should be able to minimise some performance measures, usually functions of residuals: variance, mean of squared error (MSE) etc., and simultaneously to satisfy the principle of parsimony. Often, in identification problems the notorious bias-variance trade-off is addressed by employing the Akaike Information Criterion (AIC) as objective function. However the presented approach does not use the AIC function, but sets as distinctive objectives, model size and residual variance, since in this problem the emphasis is placed on more accurate representations of the system. From a conceptual standpoint, this means that residual variance has a higher priority in this application.

5. MOGP FUZZY SYSTEM IDENTIFICATION: IMPLEMENTATION AND RESULTS

The chaotic glycolytic oscillator can be described by the following differential equations:

$$\begin{cases} \dot{x}_1(t) = f_1(x) + g_1(x)\cdot u_1 \\ \dot{x}_2(t) = f_2(x) + g_2(x)\cdot u_2 \end{cases} \qquad (4)$$

where $f_1(x) = -x_1 \cdot x_2^2 + 0.999$, $f_2(x) = x_1 \cdot x_2^2 - x_2$, $g_1(x) = 0.42$, $g_2(x) = 0$, $u1 = cos(1.75 \cdot t)$.

A chaotic system is very sensitive to initial conditions, which can influence significantly the trend of the trajectory. For $x_1(0) = x_2(0) = 1.5$, the trajectory of the system is presented in Fig.1. Both of the system states are observable and bounded.

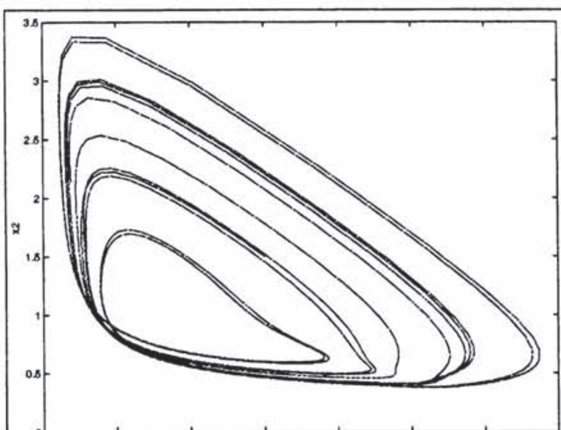

Figure 1. The trajectory of the glycolytic oscillator

The goal of the modelling approach is to build a fuzzy NARX replica of the chaotic system, that is, to approximate with a high level of accuracy both of the system outputs, x_1 and x_2. (The dynamic NARX representation is perhaps the most commonly used in fuzzy modelling approaches, as well as in most neural network identification routines [24].) The system can thus be decomposed into two SISO systems, where the excitation signal, *u*, is the common input. Each of these SISO systems is equivalent to a MISO fuzzy system, as each past instant of the variables *y* and *u* defines a single fuzzy input [12].

Moreover, the study targets the development of a p-steps-ahead predictor ($p = 6$ in this application). In terms of fuzzy system, this prediction of $x(k+p)$ is normally formulated as an input-output mapping [12], where the inputs are past and present data in the following form:

$$[x(k-(i-1)\cdot p), x(k-(i-2)\cdot p),..,x(k),$$
$$u(k-(i-1)\cdot p), u(k-(i-2)\cdot p),..,u(k)] \mapsto x(k+p),$$
$$k = 1,2,3...; i \geq 2; p \geq 1 \tag{5}$$

The proposed approach seeks to improve the precision of the mapping without increasing substantially the rule base size or the overall number of consequent parameters. The search for potential solutions to the problem is realised by employing a MOGP optimiser. The choice of MOGP over a simple single-objective GP is justified by the need to achieve at least three objectives: the minimization of the residuals, number at terms and polynomial degree. These objectives will ensure a good fitting of data to the output surfaces and will also guarantee a parsimonious structure. The optimisation objectives and their goal values are presented in Table 1. A priority level of 1 for objective 3 (residual of variance) means that objective 3 is treated as a constraint, as opposed to a standard objective. Constraints can have priority levels 1, 2 or 3 whilst 0 denotes a standard objective. Constraints are optimised first according to their relative priority level and once their goal values are achieved, the remainder of the objectives are optimised. A higher number indicates a higher priority.

Objective function	Parsimony (number of nodes)	Model order	Residual variance
Goal value	25	5	0.02
Priority	0	0	1

Table 1: Objective functions in MOGP

A set of 250 data points was used to build each of the two fuzzy models via MOGP optimisation. A further 500 points were then used to validate these models. Maximum lag values were set at: $i = n_u = n_y = 3$.

Therefore, for the prediction of each output $x1$ and $x2$, a 6-input/single-output fuzzy TS system was built. Three triangular fuzzy sets per input were used and their shape and position were kept fixed during MOGP optimisation, as the influence on the performance of the consequent structure alone was to be investigated.

Figure 2 presents a set of non-dominated solutions evolved from a MOGP with 50 individuals, which ran for 50 generations.

These solutions are trade-offs between the design objectives presented in Table 1. In Fig. 2, the objectives are displayed along the x-axis and the relative cost of the objectives on the y-axis. Each line corresponds to a particular fuzzy model; crossing lines indicate that a trade-off exists between adjacent objectives, while concurrent lines indicate that there is no conflict between objectives. The cross marks represent the design goals as indicated in Table 1. The further a line is below these marks the greater the over-attainment of that design objective.

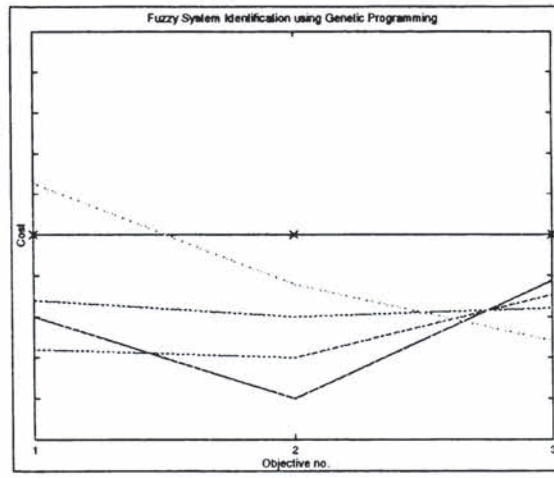

Figure 2. Set of trade-off solutions found by MOGP

Table 2 presents the performance of several models evolved from the MOGP optimiser for the (u,x1) mapping, and also provides the details of their structure.

Soln. no.	No. of nodes	Order	Residual variance	Total no. of consequent parameters
1	23	4	0.0162	210
2	17	3	0.0174	294
3	25	3	0.0179	294

Table 2: Performance of the MOGP solutions

After the model structure was found, the consequent parameters were estimated with a least squares approach. A model estimation is completed in 12 seconds of CPU. For each structure with 3 membership functions (mfs) per input, 42 rules were found.

Table 3 shows the best-case performance of the linear consequent structure with different fuzzy partitions. In order to evaluate the performance of the models built via MOGP, Tables 2 and 3 must be compared.

For the same number of fuzzy partitions and rules, from Tables 2 and 3 it can be seen that all the presented MOGP solutions are able to provide better fitting to the input-output data than their linear counterparts.

No. partitions / Performance	3mfs	5mfs	6mfs	7mfs
Var. of Residuals	0.0526	0.0192	0.0104	0.01
No. Rules Found	42	111	118	152
No Conseq. Param.	294	777	826	1064
Estimation Time	15sec	3min	8min	15min

Table 3: Best-case performance of the fuzzy system with linear consequent functions

This is also illustrated in Fig. 3, where the original output of the glycolytic oscillator is plotted against (a) the predicted non-linear (MOGP build) and (b) linear fuzzy response. This graph corresponds to 3 membership functions per input. Fig. 3 represents the system response for the training data. The original output is indicated by continuous line, whereas the predicted response is given be the dashed lines. It can be seen that the non-linear case outperforms the linear one for the same number of partitions.

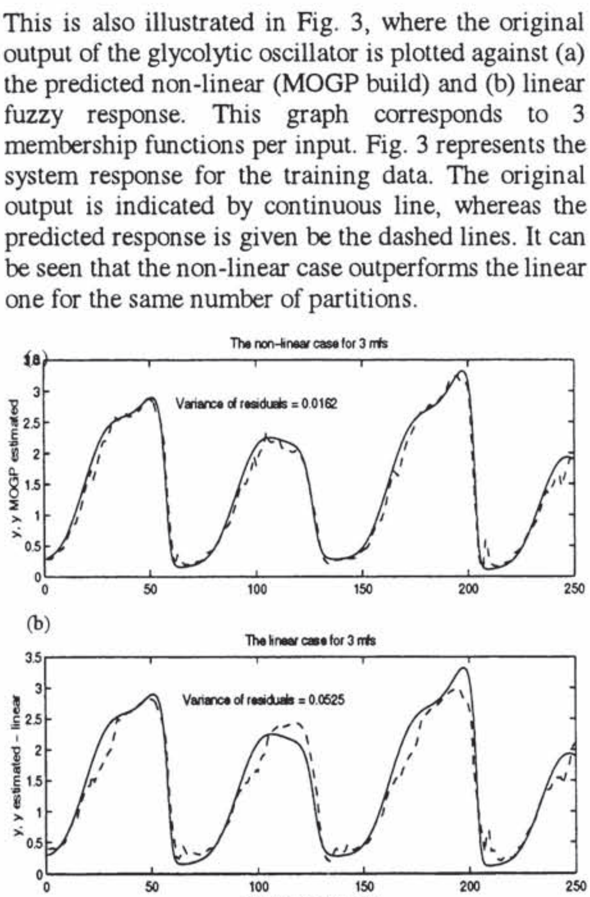

Figure 3. Original and predicted fuzzy responses for (a) non-linear and (b) linear consequent functions

In addition, from Tables 2 and 3 can also be seen that the fit provided by the MOGP solutions for 3 mfs are better than those provided by the linear functions with 5 fuzzy partitions. Simultaneously, the number of consequent parameters of the MOGP models is comparable to the number of parameters needed by the linear structures for the same number of partitions of the universe of discourse. Furthermore, if the structures with similar fitting are compared (3mfs non-linear, 5mfs linear) the number of coefficients to be estimated is over 3.5 times smaller in the non-linear case. The time required for the fuzzy model estimation is also a few times smaller for the MOGP solutions.

Very similar results were obtained for the second output of the system, *x2*.

However, the evaluation of the MOGP model is not complete until model validation is performed. As in SI, a few methods are well-known, such as statistical methods (long term prediction error, correlation function), model simulation on a new set of data, or indeed, as [6] indicates, time or frequency analysis. For chaotic systems, an additional technique is recognised. This method checks the ability of the predicted data to reconstruct the initial state trajectory. In this application the methods of the trajectory and of the validation on a different data set, were used to validate the models. The results of the validation techniques for a set of MOGP solutions *(x1_MOGP, x2_MOGP)* are given in Fig. 4. Figure 4 (a) represents the original output x1 (continuous line) *vs.* the estimated x1 (dashed line) on the new set of points. For the same checking data, the estimated trajectory is depicted in Fig. 4(b). It can be seen that it resembles the original graph from Fig.1.

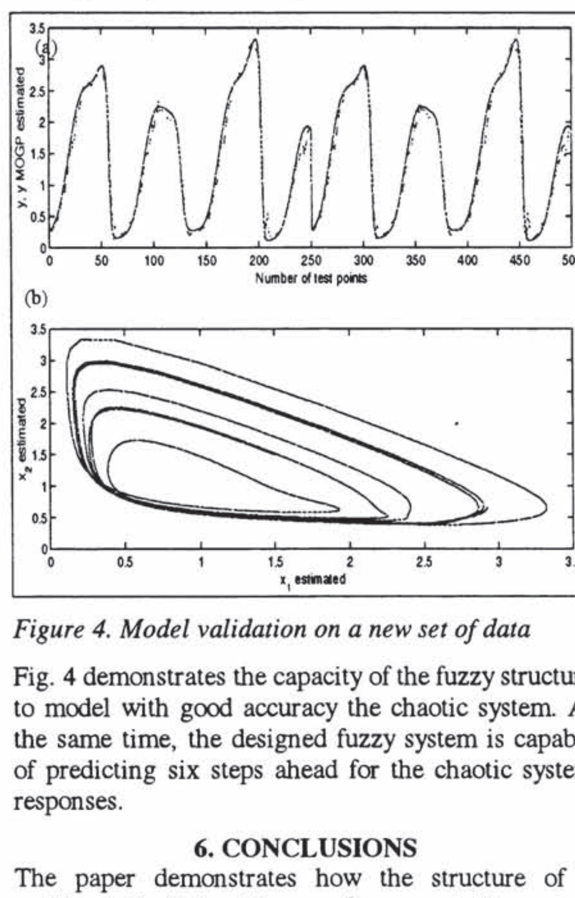

Figure 4. Model validation on a new set of data

Fig. 4 demonstrates the capacity of the fuzzy structure to model with good accuracy the chaotic system. At the same time, the designed fuzzy system is capable of predicting six steps ahead for the chaotic system responses.

6. CONCLUSIONS

The paper demonstrates how the structure of a multivariable Takagi-Sugeno fuzzy controller output can affect the system behaviour and proposes a method for the derivation of optimal fuzzy system structures based upon non-linear local system representations.

In order to tackle the issue of structural optimality of the system and simultaneously to ensure that the principle of parsimony is satisfied, a multiobjective genetic programming is used in this study. The choice of this optimisation approach is mainly justified by its ability to manipulate structures, rather then simply parameters, in a Pareto-optimal manner.

Experiments were carried out on a chaotic glycolytic oscillator. The use of higher-order polynomials in the consequent TS rules, comprising only the significant terms, proved to be able to capture with greater accuracy the systems non-linearity and to model its behaviour, without increasing the overall complexity. The technique is particularly useful for modelling complex, multivariable systems, with a very high degree of non-linearity. This is often the case in the biological and physiological systems, but also in chemical or aeronautical industry.

7. ACKNOWLEDGEMENTS

The first author gratefully acknowledges the financial support of a grant from the CVCP and an Overseas Research Studentship from the University of Sheffield.

8. REFERENCES

[1] Bica, B., Ceconvciuc, C., Chipperfield, A. J. and Fleming, P. J., "Multivariable Fuzzy Controller Design Using Multiple Response Surfaces", to be published, *UKACC, Control 2000*.

[2] Bica, B., Chipperfield, A. J., Fleming, P. J. and MacKenzie, S., "Enhancing The Performance Of A Multivariable Fuzzy Controller By Means Of Multiobjective Genetic Programming And Statistical Analysis", submitted to, *IECON 2000*, Japan, Mar. 2000.

[3] Koza, J. "Genetic Programming: On the Programming of Computers by Means of Natural Selection", *MIT Press*, 1992.

[4] Holden, A.V., "Chaos", *Princeton University Press*, 1986.

[5] Pedrycz, W., "Fuzzy Control and Fuzzy Systems", 2nd edition, *John Wiley & Sons*, 1993.

[6] Babuška, R. and Verbruggen, H.B, "Fuzzy Set Methods for Local Modelling and Identification", in Murray-Smith, R. and Johansen, T.A., editors, "Multiple Model Approaches to Modelling and Control", pp.75-100, *Taylor and Francis*, 1997.

[7] Park, M.-K., Kim, E.-T. and Park, M., "Identification of Takagi-Sugeno Fuzzy Models via Clustering and Hough Transform", in Hellendoorn, H. and Driankov, D., editors, "Fuzzy Model Identification", Springer-Verlag, pp.91-120, 1997.

[8] Bezdek, J. C., Coray, C., Gunderson, R. and Watson, J., "Detection and Characterization of Cluster Substructure: I. Linear Structure: Fuzzy c-lines, II. Fuzzy c-varieties and convex combinations thereof., *SIAM Jour. Appl. Math.*, Vol. 40(2), pp.339-372, 1981.

[9] Zhao, J., Wertz, V. and Gorez, R., "A Fuzzy Clustering Method for the Identification of Fuzzy Systems", *Trans. of 9th IEEE Int. Symposium on Intelligent Control*, Ohio, USA, 1994.

[10] Pal. N. R. and Bezdek, J. C., "On Cluster Validity for the Fuzzy c-means Model", IEEE Trans. Fuzzy Systems, Vol. 3(3), pp.370-379, 1995.

[11] Delgado, M, Gómez-Skarmeta, A. F., Martín, F., "Generating Fuzzy Rules Using Clustering Based Approach", *Proc. EUFIT'95*, pp.810-814, 1995.

[12] Wang, L-X., " A course in Fuzzy Systems and Control", *Prentice Hall*, 1997.

[13] Guély, F. and Siarry, P., "Gradient Descent Method for Optimizing Various Fuzzy Rules, *IEEE Proc. of Int. Conf. on Fuzzy Systems*, pp. 1241-1246, 1993.

[14] Hollatz, J., "Fuzzy Identification Using Methods Of Intelligent Data Analysis", in Hellendoorn, H. and Driankov, D., editors, "Fuzzy Model Identification", *Springer-Verlag*, pp.165-191, 1997.

[15] Adeli, H. and Jung, S.L., "Machine Learning – Neural Networks, Genetic Algorithms and Fuzzy Systems, *John Wiley & Sons*, 1995.

[16] Linkens, D. A. and Nyongesa, H. O., "Genetic Algorithms And Fuzzy Control. Part I-II: Offline System Development And Application; Online System Development And Application", *IEE Proc.-Control Theory Appl.*, Vol. 142, No. 3, pp. 161-185,1995.

[17] Herrara, F., Lozano, M. and Verdegay, J.L., Tuning Fuzzy Logic Controllers by Genetic Algorithms", *Int. Jour. of Approx. Reasoning*, Vol.12, pp.299-315, 1995.

[18] Nakoula, Y., Galichet S. and Foulloy, L., "Learning of a Fuzzy Symbolic Rule Base", *Proc. EUFIT'95*, pp. 594-598, 1995

[19] Ishibuchi, H. and Murata, T., "Minimizing the Fuzzy Rule-Base and Maximizing its Performance by a Multi-Objective Genetic Algorithm, *FUZZ-IEEE*, pp.259-264, 1997.

[20] Lee, M.A. and Esbensen H., "Automatic Construction of Fuzzy Controllers for evolutionary multiobjective optimisation algorithms", *IEEE Int. Conf. on Fuzzy Systems*, Vol.2, pp.1518-1523, 1996.

[21] Bica, B., Akat, G., Chipperfield, A. J. and Fleming, P. J., "Multiobjective design of a fuzzy controller for a gas turbine aero-engine", *Proc. of International Conf. on Control '98*, No.455, Vol. II, pp. 901-906, 1998.

[22] Rodriguez-Vázquez, K. and Fleming, P. J., "Multiobjective Genetic Programming for Gas Turbine Engine Model Identification", *Proc. UKACC*, Swansea, pp. 1385-1390,1998.

[23] Alba, E. Cotta, C. and Troya, J.M.,"Type-Constrained Genetic Programming for Rule-Base Definition in Fuzzy Logic Controllers", pp.255-260, *Proc. of First Annual Conf. On Genetic Programming*, No.455, *MIT Press*, pp. 255-260, 1996.

[24] Harris, C.J., Moore, C.G. and Brown, M., "Intelligent Control: Aspects of Fuzzy Logic and Neural Nets", *World Scientific*, 1993.