ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 199-204

The synergy between multideme genetic algorithms and fuzzy systems

I.Rojas, J.L.Bernier, E.Ros, F.J.Rojas, C.G. Puntonet
Department of Architecture and Computer Technology. University of Granada. Spain.

Abstract

In this article, a real-coded genetic algorithm (GA) is proposed capable of simultaneously optimizing the structure of a system (number of inputs, membership functions and rules) and tuning the parameters that define the fuzzy system. A multideme GA system is used in which various fuzzy systems with different numbers of input variables and with different structures are jointly optimized. Communication between the different demes is established by the migration of individuals presenting a difference in the dimensionality of the input space of a particular variable. We also propose coding by means of multidimensional matrices of the fuzzy rules such that the neighborhood properties are not destroyed by forcing it into a linear chromosome. The effectiveness of the proposed approach is verified and is compared with other fuzzy, and neuro-fuzzy approaches in terms of the root mean squared error (RMSE).

I. GENETIC ALGORITHMS AND FUZZY SYSTEM

Since the introduction of the basic methods of fuzzy reasoning by Zadeh and the success of their first application to fuzzy control, fuzzy logic has been widely studied [5][7] and [11]. However, certain important problems still remain, including: 1) the selection of the fuzzy rule base; 2) the subjective definitions of the membership functions; 3) the selection of the variables of the system. The design of a fuzzy system involves the structure of the rules of the system, and the membership function parameters. GAs have the potential to be used to evolve both the fuzzy rules and the corresponding fuzzy set parameters [9]. Some of the work of fuzzy systems and GAs concentrates exclusively on tuning of membership functions [6] or on the selecting an optimal set of fuzzy rules [8], while others attempt to derive rules and membership functions together [2]. To obtain optimal rule sets and optimal sets of membership functions, it is preferable that both are acquired simultaneously [4]. To optimize the whole fuzzy system simultaneously, two structures will be used: one to encode the membership functions and the other for the fuzzy rules.

A. Membership function coding

The membership functions are encoded within an "incomplete" matrix in which each row represents one of the variables of the system, and where the columns encode the parameters of the membership functions (Fig.1). Because each of the input variables of the system has a different number of membership functions, the chromosome structure used to store the membership functions is not a "complete" matrix, as each of the m rows has a different number of columns n_m. As we have selected a triangular partition (TP), the only parameter that needs to be stored is the centre of the function [12].

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 199-204

B. Fuzzy rules codification

To encode fuzzy rules, rather than a string or vector where the numerical consequents of the conclusions will appear, we carried out spatial encoding in the form of a n_1 x...x n_n matrix, noting that n_m is the number of membership functions contained within each input variable. By using string linear encoding, rules that are close together within the antecedent and which, when fuzzy inference is performed are activated simultaneously, can be distantly encoded. Thus, in a planar structure, the neighborhood properties are destroyed when it is forced into a linear chromosome. In the behaviour of GAs, it is preferable for fuzzy rules that are similar in the antecedent to be encoded as neighbors. Therefore, and as is implicit in encoding, rules that are neighbors in the rule table create interference with each other. Fig.1 shows the complete fuzzy systems codification. Note that the genetic operators described in the following section take into account the spatial structure of the fuzzy rules.

C. Fitness function

To evaluate the fuzzy system obtained, we have used the error approximation criterion, but to take into account the parsimony principle, that is, the number of parameters to be optimized in the system, we add a new term to describe the complexity of the derived fuzzy system. In the approach presented, GAs are used to search for an optimized subset of rules (both number of rules and the rule values) from a given knowledge base to achieve the goal of minimizing the number of rules used while maintaining the system performance. If we have various models based on the same set of examples, the most appropriate one is determined as that with the lowest description length. Another more flexible alternative is to define the fitness function as a linear combination of the error committed by the system and the number of parameters defining [3]:

$$fitness = W_E \cdot Error + W_C \cdot Complexity \quad (1)$$

II. MULTIPLE-DEME GAS

The theme of this article is that different structures of fuzzy systems may evolve and compete with each other, in such a way that even information obtained by fuzzy systems with different numbers of input variables may be shared. In general, for identification purposes, no a priori information about the structure of the fuzzy system is always obtained. Even the number of inputs (for example, in time-series prediction problems) is not always known. For this purpose, a multiple-population (or multiple-deme) GA configuration is used [1], in which each deme has a different number of input variables; within each deme there are fuzzy systems with different numbers of membership functions and rules. Basically, the configuration consists of the existence of several sub-populations which occasionally exchange individuals. Therefore it is necessary for there to exist intercommunication between the various demes that comprise the total genetic population. This exchange of individuals is called migration and is controlled by several parameters.

A. Migration between neighbour demes

In this paper, two different situations of migration between demes are considered: the migration towards demes with a lower dimensionality and that towards those with a higher dimensionality. Fig.2 illustrates the case in which the exchange of individuals

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 199-204

between demes only occurs between near neighbours, which is equivalent to say that the exchange occurs between fuzzy systems that differ by one in their input space dimensionality. The migration of a fuzzy system with a particular number of input variables towards a system with a lower dimensionality requires the previous, and random, selection of the variable to be suppressed (we term this variable m). The second step is then to determine, again in random fashion, one of the membership functions of this variable (termed j) and to construct the new, lower dimensionality, fuzzy system that only has the rules corresponding to the membership function j that has been selected (Fig.3). Thus the set of membership functions of the new fuzzy system is identical to that of the donor system, except that the variable m has been removed. The rules are determined by the following expression:

$$R^{Offpring}_{i_1 i_2 \ldots, i_{m-1}, i_{m+1} \ldots, i_N} = R_{i_1 i_2 \ldots i_{m-1}, j, i_{m+1} \ldots, i_N};\ j \in [1, n_m] \quad (2)$$

In the second case, the new fuzzy system (the new offspring) proceeds from a donor fuzzy system with a lower number of input variables (Fig.4). Here, it is not necessary to determine any donor system input variable, as in the migration described above, because the new offspring is created on the basis of the information obtained from the donor system, with the increase of a new variable; which is randomly selected from the set of variables in the higher dimensionality deme that are different to the deme with lower dimensionality. This new variable initially has a random number of homogeneously distributed membership functions, and its rules are an extension of the donor fuzzy system, taking the form: $R^{Offpring}_{\substack{i_1 i_2 \ldots, i_N, i_{N+1} \\ \forall i_{N+1}}} = R_{i_1 i_2 \ldots, i_N}$

III. GENETIC OPERATORS.

To perform the crossover of the individuals within the same subsystem, we distinguish between the crossover of the membership functions and that of the rules.

A. Crossover of the membership functions.

When two individuals have been selected (which could be termed, i and i') within the same subsystem in order to perform the crossover of the membership functions, the following steps are taken:

1.-One of the input variables of the system (for example, m) is randomly selected.

2.-Let n^i_m and $n^{i'}_m$ be the number of membership functions of system i and i' for the randomly selected variable m. Assume that n^i_m # $n^{i'}_m$. Then from system i we randomly select two crossover points, p1 and p2, such that: 1#p1#p2# n^i_m. The membership functions that belong to the interval [p1,p2] of individual i are exchanged for the membership functions of individual i' that occupy the same position.

B. Crossover of the rules.

To achieve the crossover of the consequents of the membership functions, we substitute N-dimensional submatrices within the two individuals selected to carry out the operation. One of the individuals is termed the receptor, R, whose matrix is to be modified, and the other is the donor, D, which will provide a randomly selected submatrix of itself. The crossover operation consists of selecting a submatrix S from the rule matrix of the donor individual such that a matrix S^* of equal dimensions and located at the same place within the receptor individual is replaced by the new rules

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 199-204

specified by matrix S. In other words, the new offspring O is equal to R except in the submatrix of the rules given by matrix S, located at the point vector (A_1, A'_1, A_2,...,A_N). Therefore, the following steps are taken:

1.- Select two individuals R and D
2.- In order to perform the (N+1) points crossover operator, select a vector (A_1,A'_1, A_2,...,A_N), such as the submatrices S and S^* fulfill $S \subset R$ and $S^* \subset D$.
3.- Create an offspring interchanging the submatrices S and S^* in R.

C. Mutation

In mutation, the parameters of the fuzzy system are altered in a different way from what occurs within a binary-coded system. As the individual is not represented by binary numbers, the random alteration of some of the system's bits does not occur. Instead of this, there are perturbations of the parameters that define the fuzzy system. Firstly, when the fuzzy system that will be mutated has been selected, a parameter defining the fuzzy system (membership functions or rules) is randomly selected with a probability of P_m. Secondly, the parameter is modified according to the following expression:

$$\begin{aligned} c_m^j &= c_m^j + random(-\Delta c_m^{j-1}, \Delta c_m^{j+1}) \\ R_{i_1 i_2 \ldots i_N} &= R_{i_1 i_2 \ldots i_N} + random(-\Delta R, \Delta R) \end{aligned} \quad (3)$$

where the values that perturb the membership functions are given by $\Delta c_m^{j-1} = \frac{c_m^j - c_m^{j-1}}{b}$ and $\Delta c_m^{j+1} = \frac{c_m^{j+1} - c_m^j}{b}$. The active radius '*b*' is the maximum variation distance and is used to guarantee that the order of the membership function locations remains unchanged (a typical value is *b*=2, meaning that, at most, a centre can be moved as far as the midpoint between it and its neighbour). The parameter ΔR is the maximum variation of the conclusion of the rules.

IV. SIMULATION RESULTS

A two-variable target function was selected to demonstrate the ability of the proposed algorithms to construct approximations to highly non-linear target functions. We have selected the function presented in [10] where a perturbation has been added by means of a second variable X_2:

$$F(X_1, X_2) = 3 \cdot \exp(-X_1^2) \cdot sin(\pi X_1) + 0.2\cos(X_2) \quad (4)$$

N^{max} is two. If the number of membership functions of the input variables is small, only the most important features of the output can be expressed in the fuzzy rules, since small local disturbances of the output would demand a greater partition of the input subspace. It is even possible that if the perturbation of an input variable on the output is small then such a variable might be removed or considered as noise. In this GAs, the system with the best fitness is the one that has a single input variable, the variable X_1, and 8 membership functions. The approximation error (RMSE) obtained with this structure is 0.0268. Fig 5.a and Fig.5.b present the original and the obtained approximation with 8 fuzzy rules. Table 1 compares the results obtained with our algorithm and those obtained with [10].

V. CONCLUSIONS

The design problems in modeling a fuzzy system are structure identification and parameter tuning. The decision for the former task is usually based on the behavioral knowledge of an expert. Therefore, the final fuzzy system obtained from this ad hoc approach is sometimes far from the optimum. While the bibliography describes many methods that have been developed for the adjustment or fine tuning of the parameters of a fuzzy system with partially or totally known structures, few have been dedicated to achieving both simultaneous and joint structure and parameter adjustment. The goal of this paper is to find a design method of optimal fuzzy system modeling that takes both tasks into account simultaneously and without the help of experts in the domain.

Acknowledgement
This work has been partially supported by the CICYT Spanish Project TAP97-1166.

REFERENCES

[1] E.Cantú-Paz, "Topologies, migration rates, and multi-population parallel genetic algorithms". Proceedings of the Genetic and Evolutionary Computation Conference. San Francisco, CA: Morgan Kaufmann. pp. 91-98, 1999.

[2] A. Homaifar, E. McCormick, "Simultaneous design of membership functions and rule sets for fuzzy controllers using genetic algorithms", IEEE Trans. Fuzzy Systems, 3, pp.129-139, 1995.

[3] H. Ishibuchi, T.Murata, I.B.Türksen, "A genetic-algorithm-based approach to the selection of linguistic classification rules", Proc. EUFIT'95, Aachen, Germany, pp.1414-1419, 1995.

[4] I.Jagielska, C. Matthews, T. Whitfort, "An investigation into the application of neural networks, fuzzy logic, genetic algorithms, and rough sets to automated knowledge acquisition for classification problems" *Neurocomputing*, vol.24 pp.37-54, 1999.

[5] J.S.R.Jang, C.T.Sun, E.Mizutani, "Neuro-Fuzzy and soft computing", Prentice Hall, ISBN 0-13-261066-3, 1997.

[6] C. Karr, E. Gentry, “Fuzzy control of pH using genetic algorithms”, IEE Trans. Fuzzy Systems, 1, pp.46-53, 1993

[7] D.Kim, C. Kim, "Forecasting time series with genetic fuzzy predictor ensemble", IEEE Transactions on Fuzzy Systems, vol.5, no.4, pp.523-535, November 1997.

[8] K.Kropp, U.G.Baitinger, "Optimization of fuzzy logic controller inference rules using a genetic algorithm", Proc. EUFIT'93, Aachen, Germany, pp.1090-1096, 1993.

[9] J.Liska, S.S.Melsheimer, "Complete design of fuzzy logic systems using genetic algorithm", Proc. FUZZ-IEEE'94, pp.1377-1382, 1994.

[10] J.H.Nie, T.H.Lee, “Rule-Based Modelling: Fast construction and optimal manipulation”, IEEE Trans. Syst. Man and Cyber. Part A, vol.26, no.6, pp.728-738, 1996.

[11] I.Rojas, H.Pomares, F.J. Pelayo, M.Anguita, E.Ros, A.Prieto, "New methodology for the development of adaptive and self-learning fuzzy controllers in real time", *International Journal of Approximate Reasoning* vol. 21, no 2, pp. 109-136, 1999.

[12] I.Rojas, J.J. Merelo, H.Pomares, A.Prieto "Genetic algorithms for optimum designing fuzzy controller", *Proceedings of the 2nd. Int. ICSC Symposium on Fuzzy Logic and Applications (ISFL'97), International Computer Science Conventions*, February 12-14, Zurich, ICSC Academic Press, Canada, ISBN 3-906454-03-7, pp. 165- 170, 1997.

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 199-204

	Different Algorithms	Nº Rules	MSE	Max. Error
J.H.Nie & T.H.Lee [10]	P1/γ1	30	0.0114	0.354
	P1/γ2		0.0078	0.353
	P2/γ1		0.0071	0.278
	P2/γ2		0.0068	0.278
	P3/γ1		0.0081	0.388
	P3/γ2		0.0080	0.388
Our Approach		8	0.0060	0.2195

Table 1: Comparison of the proposed algorithm with other fuzzy methods for direct synthesis of fuzzy systems

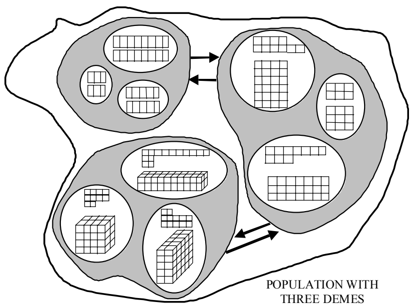

Fig.1Fuzzy system codification. Rules and membership functions

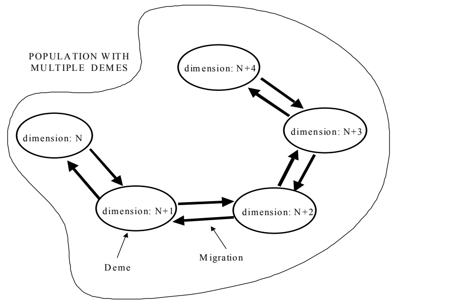

Fig.2 A schematic of multideme GA, in which the exchange of individual between demes only occurs between near neigbours

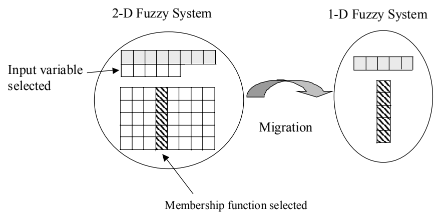

Fig.3 Migration towards a fuzzy system with a lower dimensionality (2 input variables towards 1 input

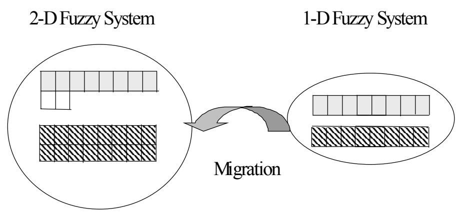

Fig.4 Migration towards a fuzzy system with a higher dimensionality (1 input variable towards 2 inputs)

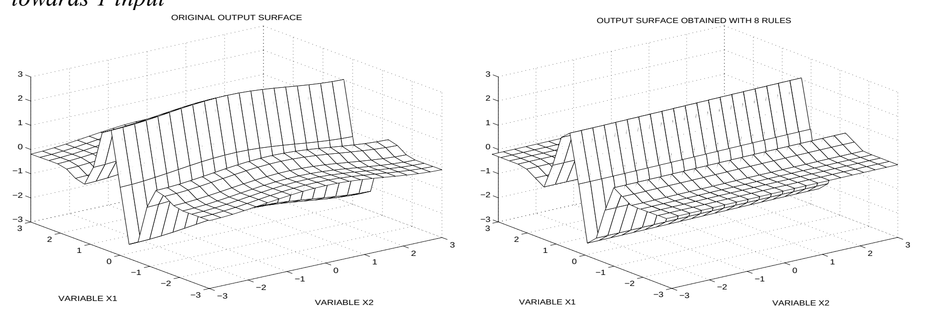

Fig.5 (a) Original output surface b) Output surface obtained by the proposed algorithm with 8 rules