Proceedings of the 9th International Conference on Neural Information Processing (ICONIP'02), Vol. 4
Lipo Wang, Jagath C. Rajapakse, Kunihiko Fukushima, Soo-Young Lee, and Xin Yao (Editors)

EVOLUTIONARY FUZZY SYSTEM DESIGN AND IMPLEMENTATION

J. F. M. Amaral[1], *R.. Tanscheit*[2], *M. A C. Pacheco*[2], *M. M. R.. Vellasco*[2]

[1]Dept. of Electronics and Telecommunications, Rio de Janeiro State University, Brazil
[2]Dept. of Electrical Engineering, Catholic University of Rio de Janeiro, Brazil
E-mails: franco@uerj.br, [ricardo, marco, marley]@ele.puc-rio.br

ABSTRACT

This work proposes a methodology for the design of fuzzy systems based on evolutionary computation techniques. A three-stage evolutionary algorithm that uses Genetic Algorithms (GAs) evolves the knowledge base of a Fuzzy System – rule base and parameters. The evolutionary aspect makes the design more simple and efficient, especially when compared with traditional trial and error methods. The method emphasizes interpretability so that the resulting strategy is clearly stated. An Evolvable Hardware (EHW) platform for the synthesis of analog electronic circuits is proposed. This platform, which can be used for the implementation of the designed fuzzy system, is based on a Field Programmable Analog Array (FPAA). The performance of a fuzzy system in the control of both a linear and a nonlinear plant is evaluated. The results obtained with these two plants show the applicability of this hybrid model in the design of fuzzy control systems.

1. INTRODUCTION

Fuzzy systems have already shown their ability to solve problems in various fields and there has been an increasing interest in providing them with learning capabilities, especially through computational intelligence techniques. The resulting models are commonly called hybrid systems, examples of which are neuro-fuzzy systems [1]. A different and more recent approach [2] couples fuzzy systems with evolutionary algorithms, giving origin to genetic fuzzy systems, which constitute the main interest of this work.

Hybrid systems may either optimise membership functions parameters or have a limited capability of creating and adjusting a rule-base. Other important features that could be evaluated are the number of linguistic values, expressed by membership functions, and the choice of t-norm and t-conorm operators.

This paper proposes a new evolutionary design of fuzzy systems. The method comprises three stages: partitioning of input and output spaces, structure identification and parameter adjustments. Initially the type and number of membership functions (partitioning) are chosen. In a second stage, a GA searches for a rule-base that leads to the lowest input-output mapping error. Eventually another GA is used for parameter optimisation. In the second and third stages the fitness of each individual is calculated in terms of the total input-output mapping error and the system evolves until a suitable performance may be obtained. Once the design of the fuzzy system is completed, this can be implemented.

Fuzzy Logic Controllers (FLCs) have been used successfully in numerous control systems. The most common implementations of FLCs rely on micro-processors; however, since FLCs admit a high degree of parallelism, an analog solution is also suitable. Unfortunately, analog circuit design is not always an easy task. Recently, genetic algorithms have been used as agents that program a reconfigurable platform in order to carry out circuit design, promoting a novel area of interest known as Evolvable Hardware (EHW) [3].

Analog hardware implementations of FLCs [4] have usually followed circuit design common rules. Projects are based on experience and intuition. However, synthesis of unconventional electronic circuits for which there are no textbook guidelines are particularly appealing to EHW [5].

Reconfigurable platforms [5][6], which have recently been employed in the synthesis of analog circuits, consist of integrated circuits of which the internal connections can be programmed by the user. Field Programmable Analog Arrays (FPAAs) constitute the state of the art in the technology of reconfigurable analog platforms.

This paper is divided into five additional sections. Section 2 presents some aspects of Hybrid Systems. Section 3 describes the main concepts about Evolvable Hardware, FLCs and hardware implementations. Section 4 presents the proposed evolutionary fuzzy system and the platform for future evolution of the fuzzy hardware. Section 5 presents experimental test results and section 6 concludes the work.

2. HYBRID SYSTEMS

In fuzzy systems, the procedure of adjusting rule-bases and membership functions in an empirical way can be successful when there is a good description of the system to be modeled. When input-output data are available and a linguistic representation is important, rules can be extracted from those data through various techniques.

A fuzzy hybrid system is an adaptive system where the structure can be built and parameters can be adjusted by a learning algorithm. Examples of this approach are neural fuzzy systems such as the NEFCON [1], well suited for control applications. In a previous work [7], a genetic algorithm has been used to tune some parameters in the NEFCON model; the resulting neuro-fuzzy-genetic system was called NEFCON-GA.

Alternatively, a fuzzy hybrid system may include only a genetic approach for learning – membership functions parameters optimisation and rule-base creation and tuning in the case of this work.

3. EVOLVABLE HARDWARE

Synthesis of a circuit consists of the conception of its topology and selection of the components for its implementation, which are normally based on the designer's experience and creativity. An automatic approach for circuit generation – called Evolutionary Electronics or Evolvable Hardware (EHW) – makes synthesis less dependent on human ability and is very useful for finding new and unconventional topologies. Results of various applications [8] show the potential of this technique as compared to the traditional ones.

In Evolvable Hardware a genetic algorithm is used in the search for the circuit, among all possible circuits in the searching space, that satisfies best a given fitness criterion – a circuit configuration presenting a pre-defined output, for example. Circuits are generally represented by a binary code, the chromosome. An initial population is created and all individuals (circuits) are evaluated. The best individuals have a higher chance of being selected for reproduction in order to form the next population; this is performed until the objective is reached.

4. EVOLUTIONARY FUZZY SYSTEM DESIGN

The design of the fuzzy system through evolutionary computation comprises the fuzzy system conception and the implementation of the electronic circuit that represents that fuzzy system.

Important aspects to be considered in development of a fuzzy hybrid system are:

1. interpretability in terms of linguistic (fuzzy) rules;
2. capability of adjustment to a given set of data;
3. rule-base creation and parameters adjustment

Rules are obtained from m numerical data, which are arranged as shown below:

$(x_1(1), x_2(1),..., x_n(1); y(1))$,
$(x_1(2), x_2(2),..., x_n(2); y(2))$,....,
$(x_1(m), x_2(m),..., x_n(m); y(m))$

where x_i are the n inputs and y is the single output.

Since good interpretability is assured by not too large a number of linguistic values and, in consequence, a reasonable number of fuzzy rules, the method allows the designer to decide whether or not the number of membership functions can be increased. Therefore, there is a compromise between precision and interpretability.

The proposed fuzzy system has the following characteristics:

1. Type: Mamdani
2. Input and output partitioning: adaptive fuzzy grid.
3. Membership functions: triangular, trapezoidal, gaussian, bell shape and singleton
4. Implication: min or product
5. Rule aggregation: max or sum
6. Defuzzification: centre of gravity or mean of maxima

The learning algorithm is illustrated in Figure 1. Initially the type of membership functions, fitness evaluation and stopping criterion are defined. The policy of alteration in the number of membership functions is also defined.

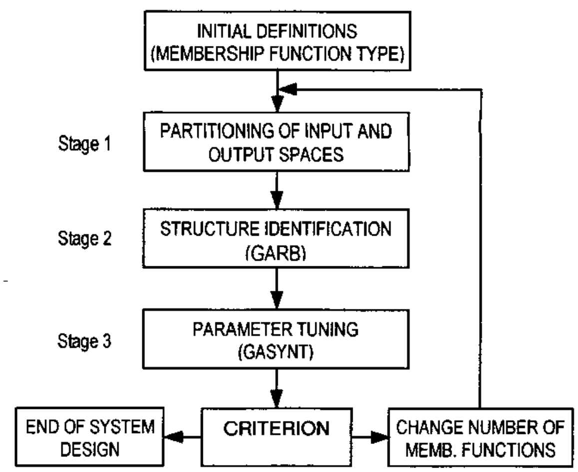

Figure 1. Learning algorithm

The learning algorithm is based on the concept of *evolutionary cycle*; each cycle consists of three stages.

In the first stage, partitioning of input and output spaces takes place. This is in accordance with the number of membership functions assigned to each variable in the cycle.

In the second stage a GA searches for the best set of rules among all possible ones. Implication and aggregation operators and defuzzification methods are also taken into account here. Eventually, structure identification is concluded: definition of the rule-base, choice of implication and aggregation operators and defuzzification method.

In the third stage parameter tuning takes place. Membership functions are adjusted so that the most satisfactory fuzzy system for the given numbers of membership functions is obtained. If the requirements are met, the fuzzy system design is completed. If not, the number of membership functions may be altered and a new cycle begins.

Two GAs are employed: one in the second stage for rule-base generation (GARB), and another (GASYNT), in stage 3, for the adjustment of membership functions.

In GARB, the chromosome – a vector with N+5 positions – codes the consequent of each of the N rules, t-norm and t-conorm operators, implication and aggregation operators and defuzzification method (5 parameters in all). Although the fitness function may be defined by the designer, it is generally considered as one of the traditional metrics used in evolutionary algorithms (MSE – Mean Square Error; RMSE – Root Mean Square Error; etc). Codification of rules for a system with two inputs and one output variables, each of them with 7 linguistic values, is shown in Figure 2.

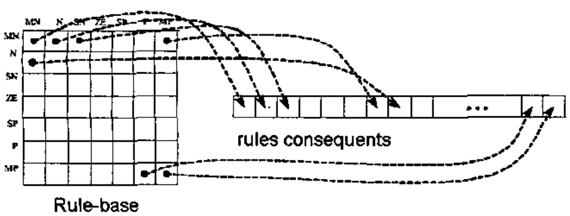

Figure 2. Rule codification in GARB

The chromosome used in GASYNT, an example of which is shown in Figure 3, is formed by the parameters of each membership function of the input and output variables. In this example, the parameters identify the vertices of triangular membership functions.

a_1	b_1	c_1	a_2	b_2	c_2	....	c_n

Figure 3. Example of a chromosome in GASYNT

The GA is used to search for the best membership function parameters a_i , b_i and c_i. These characterize the individual to be evaluated – through, for example, an error metric.

The stopping criterion can be a time restriction (fixed number of generations) or a quality restriction (the solution is better than the desired one). Evolution may be halted when there is no decrease in error within a certain number of generations.

Once the system has been designed and validated, hardware implementation can be started.

There are two basic approaches to implement an analog fuzzy system: monolithic and modular.

The first considers a monolithic block implementation to map the inputs to the output. The design via a monolithic block (for implementation of a fuzzy controller, for example) makes use of less hardware than the modular approach does. On the other hand, the design may not be easy due to the complex nonlinear multi input-single output mapping.

In the modular approach, the implementation reproduces the traditional configuration of fuzzy logic systems; three main function blocks are considered: fuzzifier, fuzzy inference engine and defuzzifier. Thus, analysis and understanding of the entire system is made easier. On the other hand, many analog circuits are necessary for implementation (each membership function is implemented by a single circuit, for example).

In a future work a new evolutionary platform shall be used for implementation of the entire system. It is based on a PAMA (Programmable Analog Multiplexer Array) [6], which supplies an environment for the evolution of generic analog circuits based on discrete components.

The PAMA is an analog platform based on analog multiplexers/demultiplexers. These are fixed elements and are responsible for the interconnections of the different discrete components that can be plugged into the board. Evolution of analog circuits is achieved through a traditional Genetic Algorithm whose chromosomes are made up of genes (bit string). Each gene configures the select signals of a particular analog multiplexer. A multifunction I/O board connected to the PC bus is responsible for the A/D conversion and for the chromosome download. To evolve fuzzy circuits using PAMA, an eight-channel D/A converter must be added to the platform. The D/A converter allows the application of different input values to the fuzzy circuits being evolved, in order to perform the necessary input to output mapping.

5. EXPERIMENTS

A linear and a nonlinear plant have been used for evaluation of the EFS (Evolutionary Fuzzy System) model. Only the fuzzy system design is addressed here; implementation of the electronic circuit is still under development.

The use of previously obtained input-output data for evolution of the controller would mean a simple emulation of another system (a conventional controller, for example). In the experiments a more interesting – and useful – approach has been used: evolution of the controller (in closed loop control) by considering the minimization of the error to a step reference input The GA experimental setup consisted of a population of 40 individuals, evolved for 20 generations.

Since a comparison between the EFS and the already implemented NEFCON and NEFCON-GA was one of the objectives, the number of membership functions was kept the same as in these systems – three for each input and five for the output. Larger numbers of membership functions may be considered, but it should not be forgotten that interpretability is an important feature of the proposed system.

It should be mentioned that the GARB has been used only for rule-base creation; the other positions in the chromosome were kept unchanged (*min-max* composition and *centre-of-gravity* defuzzification) throughout evolution.

5.1. Experiment 1

A third order plant, defined by Eq. (1), was used to evaluate the EFS model in comparison with both the NEFCON and a PID controller.

$$G_1(s) = \frac{1}{s(s+1)(s+5)} \qquad (1)$$

The PID parameters were obtained through Ziegler-Nichols's method [9]: $K_p = 18.0$, $K_i = 12.8$ and $K_d = 6.32$. In the NEFCON and EFS three fuzzy sets (two trapezoidal and one triangular) have been used for each rule antecedent (error and change in error) and five fuzzy sets (triangular) for the rule consequent (controller output).

The results shown in Fig. 4 demonstrate that the response obtained with the PID controller shows large overshoot and settling time. Both the EFS and NEFCON provide slower responses, but with acceptable overshoots.

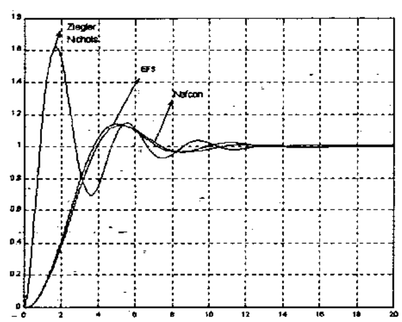

Figure 4. Third order system's closed loop responses

5.2. Experiment 2

The nonlinear plant consists of a DC servomotor [9] with a nonlinear feature – saturation – and can be described by a transfer function in cascade with a nonlinear block, as shown in Figure 5. This figure also shows the closed loop control scheme for this system. By assigning numerical values to the motor parameters, the transfer function is given by Eq. (2):

$$G_2(s) = \frac{C(s)}{E(s)} = \frac{1}{s(1{,}3s+1)} \qquad (2)$$

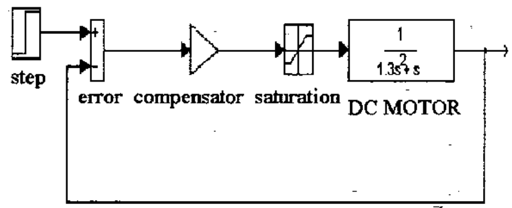

Figure 5. Control system scheme – nonlinear plant

The closed loop step responses for the nonlinear plant are shown in Figure 6. Three hybrid controllers have been used: EFS, NEFCON and NEFCON-GA. It can be noticed that the responses obtained with the hybrid genetic controllers (NEFCON-GA and EFS) have similar rise times and smaller overshoots than the one obtained with NEFCON.

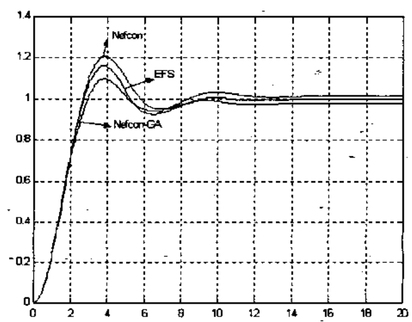

Figure 6. Nonlinear plant closed loop step responses

The NEFCON-GA has provided the best performance from the step response viewpoint. However, unusual support sets of the evolved membership functions may seriously affect interpretability, as shown in Figure 7 for the change in error input.

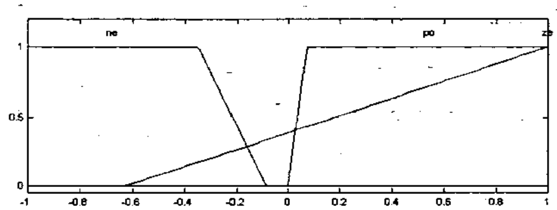

Figure 7. Fuzzy sets for change in error (NEFCON-GA)

In contrast, Figure 8 shows the fuzzy sets obtained with EFS, which considers interpretability during evolution. It must be recalled that there is a compromise between performance and interpretability.

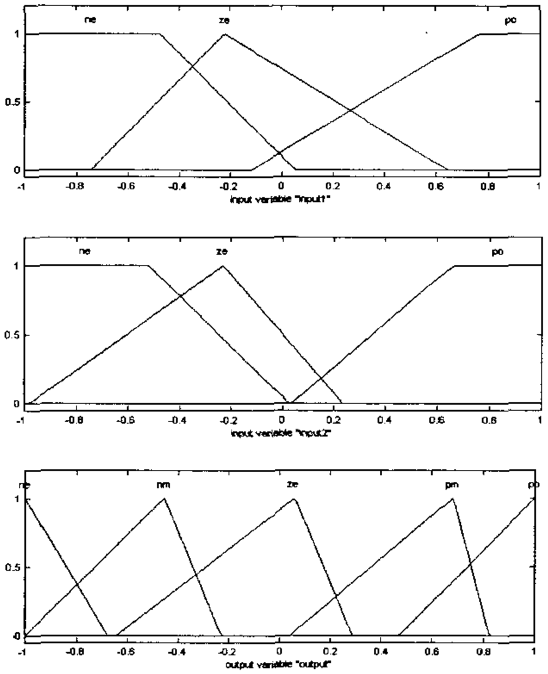

Figure 8. Membership functions for the EFS

6. CONCLUSIONS

The results obtained through simulated experiments have been satisfactory for the controllers tested. It must be stressed that the results are even more satisfactory in view of the number of fuzzy sets used for the controller inputs and output.

With respect to the new evolutionary model, it should be emphasized that it is not a completely automatic learning fuzzy system; it requires monitoring by the user, who must be able to interpret the results and interact with the algorithm. Nevertheless, it is a powerful tool for the development of fuzzy controllers. The quality of control depends upon the algorithm's capability of generating an adequate set of rules and the restrictions imposed during the parameter adjustment stage.

The main contribution of this work is a method to evolve fuzzy systems emphasizing interpretability so that the resulting strategy is clearly stated. The evolutionary aspect makes the design simpler and more efficient, especially when compared with traditional trial and error methods. The experiments have shown that this leads to good responses when comparing to existing systems, maintaining interpretability.

The implementation of Fuzzy Logic Controllers using an FPAA and a genetic algorithm to achieve intrinsic evolution have also been proposed. In a first test, a single membership function has been evolved [10]. Other fuzzy building blocks are currently being implemented.

Further work includes the implementation of more complex fuzzy analog circuits.

7. REFERENCES

[1] A. Nürnberger, D. Nauck, and R. Kruse, "Neuro-fuzzy control based on the NEFCON-model: recent developments," *Soft Computing* Abstract, Vol. 2, No. 4, pp. 168-182, 1999.

[2] O. Cordon, F. Herrera, F. Hoffmann, L. Magdalena, and F. Gomide,"Ten Years of Genetic Fuzzy Systems: Current Framework and New Trends," *Proc. of IFSA/NAFIPS 2001*, Vancouver, Canada, pp.1241-1246, 2001.

[3] R. S. Zebulum, M. A. C. Pacheco, M. M. B. R. Vellasco, *Evolutionary Electronics: Automatic Design of Electronic Circuits and Systems by Genetic Algorithms*, CRC Press, 2001.

[4] T. Yamakawa, "A Fuzzy Inference Engine in Nonlinear Analog Mode and Its Application to Fuzzy Logic Control". *IEEE Trans. on Neural Networks* , Vol. 4 , No. 3, pp. 496-522, 1993.

[5] A. Stoica, D. Keymeulen, R. Zebulum, A. Thakoor, T. Daud, G. Klimeck, Y. Jin, R. Tawel, and V. Duong, "Evolution of analog circuits on Field Programmable Transistor Array", *Proc. 2nd NASA/DoD Workshop on Evolvable Hardware*, Palo Alto, CA, USA, pp. 99-108, 2000.

[6] C. C. Santini,R. Zebulum, M. Pacheco, M. Vellasco, M. Szwarcman, "PAMA – Programmable Analog Multiplexer Array", *Proc. 3rd NASA/DoD Workshop on Evolvable Hardware*, Long Beach, CA, USA, pp. 36-43, 2001.

[7] J. F. M. Amaral, M. M. Vellasco, R. Tanscheit, and M. A. C. Pacheco, "A Neuro-Fuzzy-Genetic System for Automatic Setting of Control Strategy", *Proc. IFSA/NAFIPS 2001*, Vancouver, Canada, pp. 1553-1558, 2001.

[8] T. Higuchi, M. Iwata, D. Keymeulen, H. Sakanashi, M. Murakawa, I. Kajitani, E. Takanashi, K. Toda, M. Salami, N. Kajihara, and N. Otsu, "Real World Applications of Analog and Digital Evolvable Hardware", *IEEE Transactions on Evolutionary Computation*, Vol. 3, No. 3, 1999.

[9] K. Ogata, *Modern Control Engineering*, 3rd Ed., Prentice-Hall, 1997.

[10] J. F. M. Amaral, J. L. M. Amaral, C. Santini, R. Tanscheit, M. M. R. Vellasco, and M. A. C. Pacheco, "Towards Evolvable Analog Fuzzy Logic Controllers", *Proc. 2002 NASA/DoD Conference on Evolvable Hardware*, Alexandria, USA, pp. 123-128, 2002.