Introduction: Genetic Fuzzy Systems

Francisco Herrera*
Department of Computer Science and Artificial Intelligence, E.T.S. Ingeniería Informática, University of Granada, 18071-Granada, Spain

Luis Magdalena†
Department of Applied Mathematics, E.T.S.I. de Telecomunicación, Politechnic University of Madrid, 28040-Madrid, Spain

This special issue encompasses eight papers devoted to genetic fuzzy systems. All of them are revised and expanded versions of papers presented in a series of two invited sessions organized by the Guest Editors of this special issue at the Seventh International Fuzzy Systems Association World Congress (IFSA’97) that was held in Prague, Czech Republic, June 25–29, 1997.

First, we briefly introduce genetic fuzzy systems as a research area that combines genetic algorithms and fuzzy systems, focusing this introduction on three points that are the keys to the use of genetic algorithms for designing fuzzy systems. After that, we provide a short presentation of the papers in this issue.

I. GENETIC FUZZY SYSTEMS

In a very broad sense, a fuzzy system (FS) is any fuzzy logic-based system where fuzzy logic can be used either as the basis for the representation of different forms of system knowledge or to model the interactions and relationships among the system variables. FSs have proven to be an important tool for modeling complex systems in which, due to complexity or imprecision, classical tools are unsuccessful.

Genetic algorithms (GAs) are search algorithms that use operations found in natural genetics to guide the trek through a search space. GAs are theoretically and empirically proven to provide robust search capabilities in complex spaces, offering a valid approach to problems requiring efficient and effective searching.

Recently, numerous papers and applications combining fuzzy concepts and GAs have appeared, and there is increasing concern about the integration of these two topics. In particular, a great number of publications explore the use of GAs for designing fuzzy systems. These approaches have been given the general name genetic fuzzy systems (GFSs).

*Author to whom correspondence should be addressed. E-mail: herrera@decsai.ugr.es
†E-mail: llayos@mat.upm.es

© 1998 John Wiley & Sons, Inc. CCC 0884-8173/98/100887-04
The automatic design of FSs can be considered in many cases as an optimization or search process on the space of potential solutions (FSs). GAs are the best known and most widely used global search technique with an ability to explore and exploit a given operating space using available performance measures. *A priori* knowledge in the form of linguistic variables, fuzzy membership function parameters, fuzzy rules, number of rules, etc., may be incorporated easily into the genetic design process. The generic code structure and independent performance features of GAs make them suitable candidates for incorporating *a priori* knowledge. Over the last few years, these advantages have extended the use of GAs in the development of a wide range of approaches for designing fuzzy systems.

As in the general case of FSs, the main application area of GFSs is system modeling and control. Regardless of the kind of optimization problem, i.e., given a system to be modeled/controlled, the involved design/tuning/learning process will be based on evolution. Three points are the keys to a genetic process: the population of potential solutions, the pair evolution operators/code, and the performance index. In the following paragraphs, we analyze them briefly.

- **The population of potential solutions.** The learning (search) process works on a population of potential solutions to the problem. The individuals of the population are called chromosomes. Different approaches have been considered, but the most widely used is the so-called Pittsburgh approach. In this case, each chromosome represents a complete potential solution, an FS. From this point of view, the learning process will work on a population of FSs. FSs are knowledge-based systems with a processing structure and a knowledge base, and considering that all the systems use an identical processing structure, the individuals in the population will be reduced to rule bases, knowledge bases, etc. In some cases, the process starts off with an initial population obtained from available knowledge, while in other cases the initial population is randomly generated.

- **The evolution operators/code.** The second question is the definition of a set of evolution operators that search for new and/or better potential solutions. The search reveals two different aspects: the exploitation of the best solution and the exploration of the search space. The success of evolutionary learning is specifically related to obtaining an adequate balance between exploration and exploitation, that finally depends on the selected set of evolution operators. The new potential solutions are obtained by applying the evolution operators to the members of the population; each one of these members is referred to as an individual in the population. Basically, there are three evolution operators that work with a code (chromosome) representing the FS: selection, crossover, and mutation. Since these evolution operators work in a coded representation of the FSs, a certain compatibility between the operators and the structure of the chromosomes is required. This compatibility is stated in two different ways: work with chromosomes coded as binary strings (adapting the problem solutions to binary code) using a set of *classical* (defined for binary-coded chromosomes) genetic operators or adapt the operators to obtain compatible evolution operators using chromosomes with a nonbinary code. Consequently, the question of defining a set of evolution operators involves defining a compatible couple of evolution operators and chromosome coding.

- **The performance index.** Finally, the third question is that of designing an evaluation system capable of generating an appropriate performance index related to each individual in the population and in such a way that a better solution will obtain a higher performance index. This performance index will drive the search/optimization process.
In summary, the points that characterize a specific design/tuning/learning process are: the initial population of solutions (obtained randomly or from some initial knowledge), the coding scheme for FSs (chromosomes representing the structure according to the design process, as rule bases, membership functions clustering centers for genetic fuzzy clustering, etc.), the set of evolution operators, and the evaluation function. In addition to these four points, each evolutionary learning process is characterized by a set of parameters such as the dimension of the population (fixed or variable), the parameters regulating the activity of the operators or even their effect, and the parameters or conditions defining the end of the process or the time when a qualitative change in the process occurs.

II. THE PAPERS IN THE SPECIAL ISSUE

The papers in this issue represent different approaches to GFSs, including several kinds of coding and evolution operators, multiple optimization objectives using suitable evaluation functions, etc.

The first paper, entitled “Genetic-Based On-Line Learning for Fuzzy Process Control,” by J.R. Velasco, is based on the Michigan approach, and deals with the problem of continuous learning in process control, using GAs for learning rules and testing them before their insertion into the knowledge base, as a function of Limbo—the place where the rules are tested. The paper shows how Limbo can be used to improve continuous learning.

The paper “Artificial Evolution of Fuzzy Rule Bases Which Represent Time: A Temporal Fuzzy Classifier System,” by B. Carse et al., proposes a genetic fuzzy system for learning a temporal fuzzy classifier system that explicitly represents time in the classifier syntax by augmenting individual classifiers with temporal tags. The genetic learning algorithm explores and exploits temporal features of the environment in which the classifier system might be expected to operate.

The paper “Context Adaptation in Fuzzy Processing and Genetic Algorithms,” by R. Gudwin et al., introduces the use of contextual transformation functions to adjust membership functions in fuzzy systems. A genetic algorithm is used to find a nonlinear transformation function, given the base membership functions and a set of data extracted from an environment classified by means of fuzzy concepts.

M.M. Chowdhury and Y. Li present the combination of messy genetic algorithms and dynamic programming-based reinforcement learning techniques to form an unsupervised learning scheme for designing autonomous optimal fuzzy logic control systems in the paper “Learning Fuzzy Control by Evolutionary and Advantage Reinforcements.”

The paper “Fuzzy Clustering with Evolutionary Algorithms,” by F. Klawonn and A. Keller, investigates the possibility of making use of evolutionary algorithms in fuzzy clustering. Experiments and theoretical investigations show that the application of evolutionary algorithms to shell clustering, where the clusters are in the form of geometric contours, is not very promising due to the shape of the objective function, whereas they can be helpful in finding solid clusters that
are not smooth, for example, rectangles or cubes. These types of clusters play an important role in fuzzy rule extraction from data.

In the paper “Crossing Unordered Sets of Rules in Evolutionary Fuzzy Controllers,” L. Magdalena proposes a genetic learning process of knowledge bases for fuzzy logic controllers with a new crossover operator, working with lists (sets) of rules. The operator is designed in such a way that maintaining the advantage of working with a reduced set of rules incorporates the characteristics of an easy crossover by using a virtual decision table structure.

J. Kacprzyk presents the use of genetic algorithms as an alternative to dynamic programming for solving the classical Bellman and Zadeh multistage control problem under fuzzy constraints imposed on applied controls and fuzzy goals imposed on attained states. Kacprzyk discusses a stochastic system under control that is assumed to be a Markov chain in the paper “Multistage Control of a Stochastic System in a Fuzzy Environment Using a Genetic Algorithm.”

The paper “Genetic Learning of Fuzzy Rule-Based Classification Systems Cooperating with Fuzzy Reasoning Methods,” by O. Cordón et al., presents a multistage genetic learning process for obtaining linguistic fuzzy rule-based classification systems that integrates fuzzy reasoning methods cooperating with the fuzzy rule base and learns the best set of linguistic hedges for the linguistic variable terms.

Finally, as Guest Editors of this special issue, we would like to thank all the authors for their contributions and the referees for their outstanding cooperation. We sincerely thank R. Mesiar, Program Committee Chair of IFSA’97, and R.R. Yager, Editor of the International Journal of Intelligent Systems, for providing us with the opportunity to edit this issue.