Current Issues and Future Directions in Evolutionary Fuzzy Systems Research

B. Carse and A.G. Pipe
IAS Laboratory
Faculty of Computing,
Engineering and Mathematics
University of the West of England
Bristol, UK
Brian.Carse@uwe.ac.uk

I.Renners and A. Grauel
University of Applied Sciences
Soest Campus, Dept. Mathematics
Centre Computational Intelligence
and Cognitive Systems
Lübecker Ring 2, D-59494
Soest/Germany
Renners@fh-swf.de

A.F. Gómez-Skarmeta, F. Jiménez and G. Sánchez
Department of Communications
and Information Engineering
University of Murcia, Spain
Skarmeta@dif.um.es

O. Cordón and F. Herrera
Department of Computer Science
and Artificial Intelligence
University of Granada, Spain
herrera@decsai.ugr.es

F. Gomide and I. Walter
State University of Campinas
Faculty of Electrical and Computer
Engineering, Dept. of Computer
Engineering and Automation
13083-970 Campinas - SP, Brazil
gomide@dca.fee.unicamp.br

A. González and R. Pérez
Department of Computer Science
and Artificial Intelligence
University of Granada, Spain
fgr@decsai.ugr.es

Abstract

Contributors to the special track on Evolutionary Fuzzy Systems at the EUSFLAT 2003 conference were asked to record their thoughts and ideas on the current state of evolutionary fuzzy systems research, "burning issues" and future directions. This paper brings together these contributions.

Keywords: Evolutionary Algorithms. Fuzzy Systems.

1 B. Carse and A.G. Pipe

We envisage a big contribution from traditional and recent work in the field of discrete-valued Michigan-style classifier systems with particular application to real-time, reinforcement learning in a variety of applications. Some researchers [1,2,3] have begun to explore this area in the fuzzy case but much more work remains to be done. While casting these advances into the framework of fuzzy systems offers great potential, it also throws up many problems which remain to be solved. A major difference between discrete-valued and fuzzy classifier systems (apart from the obvious differences in encodings and inference mechanisms) is that, in discrete-valued systems, only one classifier "fires" at any one time whereas in the fuzzy case, several fuzzy classifiers fire simultaneously (indeed this is one of the strengths of fuzzy systems). This causes a number of problems when using reinforcement learning, namely (1) How to apportion environmental credit between simultaneously active fuzzy classifiers (2) How to learn and simultaneously maintain a population of optimally general and specific classifiers (3) How to take into account the variability of reward over the input space covered by active rules (4) How to deal with the fact that (particularly early in the GA search) an active rule might fire with different other rules at different times; clearly the rule will then receive different payoffs each time it fires depending on which other fuzzy rules it fires with.

As far as we are aware, no-one has yet successfully transferred the ideas in Wilson's XCS [4] (which uses payoff prediction accuracy as the basis of evolutionary fitness) to the fuzzy case. In the discrete-valued case, XCS can lead to evolution of optimally general and specific classifiers. This would obviously be desirable for the fuzzy classifier system but again, much work needs to be done to transfer accuracy based fitness to the fuzzy domain. Other issues which have been and still are being studied in the discrete-valued LCS, and worthy of study in the fuzzy LCS are rule chaining and internal message passing for the purpose of providing internal memory.

2 I. Renners and A. Grauel

From our point of view Fuzzy Rule-Based Systems (FRBSs) are more and more utilized to perform data-driven system identification. System identification describes the task of mapping a part of the real world to an analytical expression. For example the analytical expression may be a FRBS.

The nowadays regarded real world systems become more complicated. Expert knowledge is seldom available and completely data-driven approaches are tentatively used to model very complex systems. The application area of FRBSs has reached economic and ecological fields. Overall, the complexity of systems which are tackled by FRBSs rises dramatically, but the framework of FRBS hardly ever uses hierarchical structures, although the underlying real world problems must be assumed to be of hierarchical nature. Thus, it becomes more and more inevitable to deal with hierarchical FRBSs.

There exist adaptive hierarchical approaches (e.g. ASMOD) and real hierarchical approaches (e.g. NetFAN). Why it is not common to use them? First the necessity is only given for more complex problems and many, still not considered, low-complex problems have the potential to be solved by non-hierarchical FRBSs. Secondly the demands to model hierarchical systems with FRBSs are far more challenging and thus the availability of tools tends to zero.

Using FRBSs is an accepted approach and widely used in industry and thus, the scientific focus should (and also will) move to the modeling of high-complexity problems. Implementing hierarchical models needs many pieces and yet these pieces are not easily available in a bundle. Furthermore, by now there is no common used set of artificial benchmark data describing different hierarchical systems. We think that hierarchical modeling is a "burning" issue, but it is an issue where still some premises are missing to become popular.

3 A. F. Gómez-Skarmeta, F. Jiménez and G. Sánchez

In recent years, many techniques have appeared to cope with the task of data-driven fuzzy modeling. Two approaches can be distinguished: one is treating the problem as an optimization one, solved with soft computing techniques, and the other is trying to obtain the target model as a system identification process.

The two approaches have one characteristic in common: they both solve the problem in a hybrid fashion. The first approach is hybrid in the sense of the soft computing hybrid nature (neuro-fuzzy, fuzzy-genetic, etc.). These can be considered tightly-coupled hybrids. The second kind are loosely-coupled hybrids in which some techniques are combined to do each one of the identification phase (input selection, rules generation and parameter identification).

These facts and the modular nature of the system identification process, give rise to a need of tools to easily and flexibly work with hybridization. Investigation on how capabilities and preconditions of techniques must be specified is the natural starting point, and the design of multi-agent systems to solve hybrid fuzzy modeling tasks could be a promising goal.

Multi-objective Optimization (MO) is a "burning" line within Evolutionary Computation because population-based algorithms are capable of capturing a set of non-dominated solutions in a single run of the algorithm. In Evolutionary Fuzzy Systems (EFS), MO appears in multiple fields. We emphasize the following:

1) Accuracy and interpretability of fuzzy models can be treated as two objectives to optimize in fuzzy modeling. Mean squared error is used for accuracy criteria and similarity measures can be defined for interpretability.

2) In Mobile Robotics, behaviour fusion is basically a MO optimization process in which multiple objectives are considered for task performance.

3) Solutions in Fuzzy Nonlinear Optimization can be approximated as non-dominated solutions of a two-objective nonlinear programming problem where one objective represents a fuzzyness degree.

4 O. Cordón and F. Herrera

There are different questions that the contributers will emphasise in this short contribution, such as the lack of systematic procedures to design and develop fuzzy systems, the experimentation for evaluating GFS models, new areas of application, etc.

In our short communication we would like to introduce the discussion on the actual experimentation that justifies the majority of the publications. Currently there is no systematic evaluation methodology for GFSs. We will point out different questions related to this problem:

- Usually, for getting conclusions, the majority of papers only use one data set and one partition or a specific control problem. An n-fold cross validation is necessary for avoiding the bias of a specific partition for data learning, and different problems/cases must be used in the experimentation of a paper.

- Genetic algorithms are probabilistic problem-solving techniques, therefore it would be necessary to run them several times for analyzing their robutsness.

- For a new approach/proposal/genetic-learning-algorithm, it would be necessary to compare its performance with other learning proposals widely published in the literature on learning fuzzy systems, for analyzing the real behaviour of the new proposal with respect to well known ones (with good behaviour according to a specific criterion, interpretability, precision, ...).

Therefore, it is necessary to develop an acceptable evaluation method. Some suggestions are:

- To manage adequate and unified sets of benchmark problems for evaluating the learning methods for fuzzy systems for different kinds of problems, classification, data mining, regression and control problems.

- To design experimental analysis models using statistical analysis techniques (see [6]), comparing the new proposal with others considered as the departure point for every kind of approach (learning rules bases, tuning or learning membership functions, learning knowledge bases, ...).

5 F. Gomide and I. Walter

Currently, there is no systematic procedure to design and develop fuzzy rule-based systems. A common approach is to define fuzzy systems using background and expert knowledge and test the design decisions to verify if system requirements are achieved. In general, when there is a lack of expert knowledge or when large amounts of data must be processed and analyzed, the knowledge-based design approach becomes limited. Machine learning is a useful alternative in these circumstances, but it also has limitations. For instance, neural networks can be used, but the linguistic meaning of fuzzy rules and transparency may be lost [7]. Neural fuzzy networks attempt to surpass this problem, but the designer still has to choose major design parameters such as antecedent aggregation operators; rule semantics, rule aggregation operators and defuzzification methods [8, 9, 10, 11, 12]. Automatic methods based on fuzzy clustering and rule induction from large collections of learning data are attractive alternatives [10], but they suffer the same limitations, as do the fuzzy neural network-based alternatives.

GA-based approaches have been suggested to learn (i) membership functions with fixed fuzzy rules [13], (ii) fuzzy rules with fixed membership functions [14], (iii) fuzzy rules and membership functions using (i) and (ii) in alternate steps [15], (iv) membership functions and membership functions simultaneously [16], (v) rule and rule base structure and parameters (granulation, rule antecedent aggregation operator, rule semantics, rule base aggregation operator, defuzzification, membership function shape and parameters) simultaneously [17, 18]. Supervised, hybrid learning schemes to increase design efficiency have also been suggested as an alternative. Genetic neural learning algorithms combined with least squares and singular value decomposition is an example [20]. Contrary to neural network, clustering and rule induction approaches, a GA provides a means to encode and to evolve antecedent aggregation operators, rule semantics, rule aggregation operators and defuzzification methods. In addition, GA-based fuzzy systems can effectively integrate multiple sources of fuzzy knowledge into a single knowledge base.

Fuzzy knowledge integration is a key issue that still challenges machine learning methods [19]. Increasing chromosome complexity to encode operators, rule semantics and defuzzification methods enlarge the search space considerably, but design efforts can be reduced via judicious exploration of system requirements and designer judgment. Smaller search spaces are expected when the rule structure is given and the problem is to find membership parameters' values, contrary to the case where rule structure, membership functions and parameter values must be learned simultaneously. But we should note that, despite the chance the GA brings to design more complex fuzzy systems, efficient chromosome representation of rules, rule bases and system structure still needs to be found. Optimal coding schemes are not known yet, and approaches other than suggested by Michigan and Pittsburgh schemes may be needed. For instance, Holland's principle of minimal alphabets [21] suggests that, given two possible encodings, the one with the lowest cardinality alphabet gives a greater number of schema sampled by a given population. Actually, there is indication of the opposite. Given two encodings, the one with higher cardinality alphabet gives a greater number of schema sampled by a given population [22, 23].

Another critical issue concerns the fundamental theorem of genetic algorithms: short, low-order, above average schemata receive exponentially increasing trials in subsequent generations [21]. Recently, this statement has been challenged and a growing number of criticisms reported [24, 25]. Another, maybe more evident issue once it has been often discussed in the literature, surfaces from research on general search problems. Known as *no free lunch theorems*, they pose a key question since they show that a GA, in average, is no better than random search [26]. In view of these critical points, we are left with the question whether fuzzy set theory may bring a contribution to evolutionary computation foundations and applications. A very important notion behind of fuzzy systems concerns information granulation and information compression. The use of linguistic variables as an encoding mechanism for GA may provide an effective ingredient to achieve acceptable tradeoffs to obtain *optimal* representations and search meaning. Currently, no genetic fuzzy system theory is known. Despite being effective and very useful in practice, genetic fuzzy systems still are viewed as hybridization schemes only.

6 A. González, R. Pérez and E. Aguirre

In the first stages of Artificial Intelligence, intelligent system design worked in the knowledge incorporation, based on the information given by the expert in the field of application of the system. It was soon realized that extraction of this kind had significant problems (difficulties finding experts in the field, difficulties in representing the given information, etc.)

Learning algorithms were seen as an acceptable way for an easier extraction of expert knowledge. For this reason, the main objective of the field for researchers in machine learning, consisted in designing algorithms to automate this process.

The main problems that these researchers had to deal with was noise treatment, inconsistencies and unknown values that frequently appear in real problems. However, there were not too many restrictions over the kind of knowledge that was extracted, basically because the objective of this knowledge was its incorporation to external software. In this sense, it was only needed that the knowledge had an easy coding representation in a computer program and that this representation had an associated mechanism of known inference and a good behavior.

Even though those designed algorithms gave acceptable solutions to these restrictions, nowadays the objectives have changed. Basically, this change follows two factors: on one side, the necessity of applying learning algorithms to big databases where data have a great number of variables and examples; and on the other side, the need for obtained knowledge to be legible and comprehensible form a human being perspective, that is, the receiver may not be software but a person.

With regards to the first of this factors, we can find proposals for feature selection, examples selection, new attributes construction, etc. and it is a live topic in the machine learning field.

However, the second factor is much less explored, even though we have to take into account that it is greatly related with the first factor and many of its proposals can be used as a base for the objective achievement. Undoubtedly, the development of fuzzy logic has helped in the improvement of the comprehensibility of knowledge bases that learning algorithms extract. However, from our point of view it is not enough, particularly for modelling problems.

In this way, the adaptation of the second necessity has to do with a change in the way knowledge is represented. It is known that human beings do not use only one way for representing knowledge. Even though we are not familiar with some of them, it is also true that there are others we do know, and there are computational representations for them. In this sense, we think that in a short period of time, researchers will have to deal with the design of learning algorithms capable of obtaining knowledge bases composed of different representations (trees, rules, semantic networks, etc.) with computationally more complex models, but with easier interpretation and a definition of new mechanisms of inference from multiple sources that let them reason with the different representations simultaneously. In this session, we give a little step in this way, proposing a learning algorithm that uses a fuzzy rule model in the knowledge representation that allows the incorporation of fuzzy binary relations between the variables involved in the problem. A priori, this new rule model improves the interpretability of knowledge, in the sense that it obtain as a result a reduced set of comprehensible rules.

Evidently, in the development of the two new necessities, bio-inspired algorithms play an important role, basically, for two reasons:

- the flexibility these algorithms add in the representation of solutions can be a good way for the codification of multiple models of knowledge representation, and
- the search power of these algorithms allows the search in more complex spaces of solutions.

7 Summary

This section provides a brief summary of the main points raised by the contributors to this paper. We have, on purpose, avoided calling this a “Conclusions” section since the intention of the article is to raise issues and ask questions rather than draw conclusions. That is for further work. With this in mind, we summarise the main points in the paper as follows:

- For real-time, reinforcement learning of fuzzy controllers, there are potentially many lessons to be learned from the discrete-valued Learning Classifier Systems (LCS) community.

- Further research is required in the area of hierarchical Evolutionary Fuzzy Systems to reflect the inherent hierarchical nature of real-world problems.

- Evolutionary fuzzy systems should be viewed in a multi-objective optimization (MO) framework: there are many trade-offs to be made between possibly conflicting goals. A deeper study of Evolutionary Fuzzy systems as MO problem solvers needs to be carried out.

- Many novel Evolutionary Fuzzy approaches are regularly proposed but there is little in the way of performance comparison between different approaches. There is a clear need for development of benchmark problems and statistical comparisons of the efficacy of different techniques.

- There is a clear lack of underlying theory regarding the Evolutionary Fuzzy field and this issue needs serious consideration. Representations and operators, and their interplay, need to be more clearly understood on a sound scientific basis.

- The design of evolved fuzzy systems based on different representation of the knowledge with computationally more complex models, but with easier interpretation will be an important issue.

References

[1] A. Bonarini (2000). An Introduction to Learning Fuzzy Classifier Systems. In P.L. Lanzi, W. Stolzmann and S.W. Wilson (Eds.), Learning Classifier Systems- from Foundations to Applications, Lecture Notes in Artificial Intelligence, pp. 83-104. Springer Verlag Berlin Heidelberg, Germany.

[2] M. Valenzuela-Rendón (1998). Reinforcement learning in the fuzzy classifier system. Expert Systems with Applications 14: 237-247.

[3] A.G. Pipe and B. Carse (2000). Autonomous acquisition of fuzzy rules for mobile robot control: first results from two evolutionary computation based approaches. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'00), pp 849-856, Morgan-Kaufman, San Francisco, CA.

[4] S.W. Wilson (1995). Classifier fitness based on accuracy. Evolutionary Computing 3(2): 149-175.

[5] T. Kovacs (1996). Evolving optimal populations with XCS classifier systems. Master's Thesis, School of Computer Science, University of Birmingham, Birmingham, UK.

[6] S.E. Maxwell, H.D. Delaney. (2000). Designing experiments and analyzing data. A model comparison perspective. Lawrence Erlbaum Associates Pub., London.

[7] H. Takagi and I. Hayashi (1991). NN-Driven approximate reasoning. International Journal of Approximate Reasoning 5(3), pp191-212.

[8] J. Castro, J. Benitez and C. Mantas (2002). Interpretation of artificial neural networks by means of fuzzy rules. IEEE Transactions on Neural Networks, 13(1): 101-116.

[9] J. Benitez, J. Castro, I. Requena (1997). Are artificial neural networks black boxes? IEEE Transactions on Neural Networks, 8(5): 1156-1164.

[10] J. Jang, C. Sun, E. Mizutani (1997). Neuro-fuzzy and Soft Computing. Prentice Hall, New Jersey.

[11] C. Lin and C. Lee (1991). Neural-network-based fuzzy logic control and decision system. IEEE Transactions on Computers, 40: 1320-1336.

[12] W. Caminhas, H. Tavares, F. Gomide and W. Pedrycz (1999). Fuzzy set based neural networks: strucutre, learning and application. Journal of Advanced Computational Intelligence, v. 3(3):151-157.

[13] C. Karr (1991). Genetic Algorithms for fuzzy controllers. AI Expert, 6(2): 26-33.

[14] P. Thrift (1991). Fuzzy logic synthesis with genetic algorithms. Proc. 4th Int. Conference Genetic Algorithms. Morgan Kaufmann, Los Altos, pp. 509-513.

[15] J. Kinzel, F, Klawonn and R. Kruse (1994). Modification of genetic algorithms for designing and optimizing fuzzy controllers. Proc. IEEE Int. Conf. Evolutionary Computation, IEEE Press, pp. 28-33.

[16] B. Carse, T. Fogarty, and A. Munro (1996). Evolving fuzzy rule based controllers using genetic algorithms. Fuzzy Sets and Systems, 80 :273-293.

[17] M. Delgado, F. Von Zuben and F. Gomide (2001). Hierarchical genetic fuzzy systems. Information Sciences, 136(1-4): 29-52.

[18] M. Regatieri Delgado (2002). Automatic Design of Fuzzy Systems: A Co-evolutionary Approach. PhD Thesis, State University of Campinas, Campinas, Sao Paulo, Brazil (in Portuguese).

[19] C. Wang, T. Hong and S. Tseng (1998). Integrating fuzzy knowledge by genetic algorithms. IEEE Transactions on Evolutionary Computation, 2(4):138-149.

[20] M. Russo (2000). Genetic Fuzzy Learning. IEEE Transactions on Evolutionary Computation, 4(3):259-273.

[21] J. Holland (1975). Adaptation in Natural and Artificial Systems. University of Michigan Press. Ann Harbor.

[22] J. Antonisse (1989). A new interpretation of schema notation that overturns the binary encoding constraint. Proc. 3rd Int. Conf. Genetic Algorithms, J.

Schaffer (Ed.), Morgan Kaufmann, San Francisco, 86-97.

[23] L. Davis (1991). Handbook of Genetic Algorithms. Van Nostrand Reinhold, New York.

[24] H. Muhlenbein (1991). Evolution in time and space – the parallel genetic algorithm, in Foundations of Genetic algorithms, Morgan Kaufmann, San Francisco, 316-338.

[25] M. Vose (1993). A critical examination of the schema theorem. Technical Report CS-93-212, University of Tennessee.

[26] D. H. Wolpert and W. G. Macready (1997). No free lunch theorems for optimization. IEEE Transactions on Evolutionary Computing, 1:67--83.