ORIGINAL RESEARCH PAPER

Computer Science

INTEGRATING DESIGN PHASES OF FUZZY SYSTEMS USING EVOLUTIONARY ALGORITHMS

KEY WORDS: Fuzzy System,Genetics Algorithms,TSK,FRBS,Genetic Fuction.

Kuldeep Kumar Katiyar Department of Computer Science & Engineering, Faculty of Engineering & Technology,Rama University Uttar Pradesh,Kanpur,India

Sunil Awasthi* Department of Computer Science & Engineering, Faculty of Engineering & Technology, Rama University Uttar Pradesh,Kanpur,India *Corresponding Author

ABSTRACT

This paper initiate an automatic fuzzy system design method that uses a genetic algorithm and integrates three design stages; our method determines membership functions, the number of fuzzy rules, and the rule consequent parameters at the same time because these design stages may not be independent, it is important to consider e method includes a genetic algorithm and a penalty strategy that favors systems with fewer rules. The proposed method is applied to the classic inverted pendulum control problem . Now a-days researchers are taking keen interest towards integrating fuzzy systems with learning and adaptation capabilities. The two well known methodologies to augment fuzzy systems along with learning and adaptation procedures are neural fuzzy systems and genetic fuzzy systems.

1.INTRODUCTION

Fuzzy systems have become popular components of consumer products because they are inexpensive to implement, able to solve difficult non-linear control problems, and exhibit robust behavior. Designers are especially attracted to fuzzy systems because fuzzy systems allow them to capture domain knowledge quickly using rules that contain fuzzy linguistic terms.These attributes allow products with embedded fuzzy systems to be both cost effective and high performance. Fuzzy systems have been successfully implemented to problems related to classification, modeling and control and wide range of applications. In most of the cases the key for the success is the ability of the fuzzy systems to incorporate human expert knowledge.

While it is easy to describe human knowledge with fuzzy linguistic terms, it is not easy to define the terms by membership functions. In addition, fuzzy system design requires two other stages: determining the number of rules and determining the rule-consequent parameters.

This paper proposes an automatic fuzzy system design method that uses a Genetic Algorithm and integrates three design stages; our method determines membership functions ,the number of fuzzy rules, and the rule-consequent parameters at the same time. As a sample fuzzy system, we use the Takagi-Sugeno-Kang (TSK) fuzzy model [1]. Rules in a TSK fuzzy model use traditional fuzzy variables for antecedents. However, the consequent values are computed by summing weighted combinations of the input values. We have formulated a TSK fuzzy model representation that parameterizes membership function shape and position and rule- consequent parameters. By combining our representation with the target application's boundary conditions, we can represent fuzzy systems with different numbers of rules. A genetic algorithm operation, this representation and optimizes the fuzzy system parameters with respect to performance and resource requirements. We chose a genetic algorithm optimization technique because genetic algorithms because genetic algorithms are robust, search any points simultaneously, and able to avoid local minima.In the following sections we briefly review genetic algorithms and automatic fuzzy system design research. Next we discuss our TSK fuzzy model, how we incorporated genetic algorithms into the design process, and parameters of our design method.We demonstrate our method by deriving a four rule fuzzy system that balances an inverted pendulum. We conclude by comparing the performance of our controller with a controller derived.

2. REVIEW

2.1 Genetic algorithms

A genetic algorithm is a probabilistically guided optimization technique modeled after the mechanics of genetic evolution. Unlike many classical optimization techniques, genetic algorithms do not rely on computing local derivatives to guide the search process. Genetic algorithms also include random elements,which helps avoid getting trapped in local minima.Genetic algorithms explore a population of solutions in parallel.The size of the population is a free parameter, which trades off coverage of the search space against the time required to compute the next generation. Each solution in the population is coded as a binary string or gene, and a collection of genes forms a generation. A new generation evolves by performing genetic operations, such as reproduction, crossover, and mutation, on genes in the current population and then placing the products into the new generation.

In a simple genetic algorithm, operations are performed in the following order; reproduction, crossover, and then mutation. Reproduction involves selecting two parent genes from the current population. Selection is based probabilistically on a gene's fitness value; the higher the fitness of a gene, the more likely it can reproduce. After selecting two parents, crossover is performed according to a crossover probability. If crossover is to be performed, offspring are constructed by copying portions of parent genes designated by random crossover points (single point crossover shown in Figure 1).

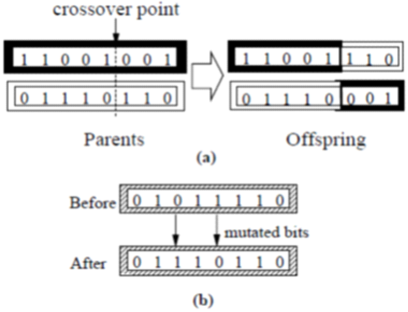

Figure 1: Genetic operations: (a) cross over (b) mutation

Otherwise, an offspring copies its entire gene from one of the parents. As each bit is copied from parent to offspring, the bit has the probability of flipping, or mutating. Mutation is believed to help reinject any information that may have been lost in previous generations [3].Variations of these operators are discussed in [2].

2.2 Genetic-based learning approaches considering different model structures:

Improvements in linguistic fuzzy modeling can be accomplished to make learning and/or model structure more Fexible.Three possibilities to relax the model structure using a GFS are as follows:

- Use of double-consequent fuzzy rules,that allows the model to present rules such that each combinations of antecedents may have two consequents associated when it improves the model accuracy. The GA acts as a genetic method to get a cooperative and compact set of fuzzy rules.
- Consideration of weighted fuzzy rules in which an importance factor (weight) is considered for each rule. By means of an evolutionary technique, the way in which these rules interact with their neighbor ones.
- Genetic selection with hierarchical knowledge bases , the structure of the KB of FRBSs is extended in a hierarchical way. Linguistic rules defined over linguistic partitions of different granularity levels provide additional flexibility, and thus improve the model accuracy in those regions in which the usual non-hierarchical models demonstrate poor performance. This type of improvement is the starting point for the development of different hierarchical system of linguistic rules learning methodologies, which are considered as a refinement of the basic linguistic fuzzy models. These methodologies have been thought as a refinement of simple linguistic models which, preserving their descriptive power, introduces small changes to increase their accuracy. A GA is used to get a compact set of hierarchical rules.

2.3 Genetic-based machine learning approaches:

GFSs with specific combination of evolution and bio-inspired models have been developed. For instance, genetic schemes inspired on the virus theory of evolution have been derived to learn TSK fuzzy rule sets , including genetic recombination in bacterial genetics and DNA coding schemes .

2.4 Analyzing the Evolutionary Fuzzy Systems' models:

The essential part of FRBSs is a set of IF-THEN fuzzy rules (traditionally linguistic values), whose antecedents and consequents are composed of fuzzy statements, related to with the dual concepts of fuzzy implication and the compositional rule of inference.Specifically, an FRBS is composed of a knowledge base (KB), that includes the information in the form of those IF-THEN fuzzy rules,i.e. the RB, and the correspondence of the fuzzy values, known as DB. It also comprises of an inference engine module that includes a fuzzification interface, an inference system, and a defuzzification interface.EFSs are a family of approaches that are built on top of FRBSs, whose components are improved by means of an evolutionary learning/optimization process as depicted in **Fig.2**. This process is designed for acting or tuning the elements of a fuzzy system in order to improve its behavior in a particular context. Traditionally ,this was carried out by means of GAs, leading to the classical term of Genetic Fuzzy Systems. In this paper, we consider a generalization of the former by the use of EAs [4]. Taking this into account, the first step in designing an EFS is to decide which parts of the fuzzy system are subjected to optimization by the EA coding scheme. Hence, EFS approaches can be mainly divided into two types of processes: tuning and learning. Additionally, we must make a decision whether to just improve the accuracy/precision of the FRBS or to achieve a tradeoff between accuracy and interpretability (and/or other possible objectives) by means of a MOEA. Finally, we must stress that new fuzzy set representations have been designed, which implies a new aspect to beevolved in order to take the highest advantage of this approach.

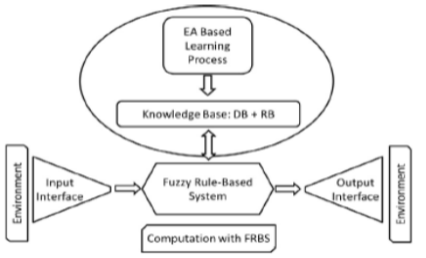

Fig. 2. Integration of an EFS on top of an FRBS [5].

3. EXPERIMENTAL ENVIRONMENT

The goal of our work is to develop an automatic fuzzy system design that uses minimal knowledge of the system to be controlled. As a sample fuzzy system, we chose the TSK fuzzy model, which is widely used in actual applications. In this section we first introduce our TSK fuzzy model representation used in our experiments. Second we present the inverted pendulum application used to illustrate our technique. Lastly we present our method for evaluating a fuzzy system's performance in our application context.

The system proposed by Takagi, Sugeno, and Kang [6] (shortly called TSK) differs from the linguistic one in the use of a different consequent structure. While linguistic rules consider a linguistic variable in the consequent, TSK-type fuzzy rules are based on representing the output variables as polynomial functions of the input variables, i.e. IF X_1 is A_1 and……… and X_n is A_n THEN $Y_1 = p_1(X_1,……,X_n)$ and………. and $Y_m = p_m(X_1,………,X_n)$ with $p_j(.)$ being the polynomial function defined for the j^{th} output variable. Using this fuzzy rule structure, the human interpretation on the action suggested by each rule is garbled but, on the contrary, the approximation capability is significantly increased. For this reason, TSK-type FRBSs are very useful in PFM.

4. EXPERIMENTAL RESULTS

Our method combines a genetic algorithm, a penalty strategy, and unconstrained membership function overlap to automatically design fuzzy systems. In this section, we present results of our method applied to the inverted pendulum problem.In our experiments, we used a genetic algorithm with two point crossover and mutation operators. In this proposal the major focus would be on the following issues:

1. Effective Interpretable Encoding Scheme for the Fuzzy systems in Evolutionary Environment.
2. Consideration of Interpretability enhancement maintaining competitive accuracy.

Fuzzy reasoning builds this understanding into the process rather than tacking it onto the end. Fuzzy logic can model nonlinear functions of arbitrary complexity. You can create a fuzzy system to match any set of input-output data. Fuzzy logic can be blended with conventional control techniques.

5. CONCLUSIONS AND FURTHER RESEARCH

We have proposed a method for automatically designing complete fuzzy systems. Our method uses a genetic algorithm and a penalty strategy to determine membership function shape and position, number of fuzzy rules, and consequent parameters simultaneously. Our experimental results demonstrate the practicality of our method, by producing systems that perform comparably to a system produced by another method.Other extensions to this work that need to be explored include applying this method to more complex tasks, directly comparing results with a sequential method, applying this method to other types of fuzzy systems, and eliminating unnecessary rules by considering overlap. This work presents a new methodology (HILK) for building knowledge bases with a good balance between accuracy and interpretability.

REFERENCES

[1] Takagi, T. and Sugeno, M., "Fuzzy Identification of Systems and Its Applications to Modeling and Control", IEEE Trans. SMC-15-1, 1995, pp. 116-132

[2] Davis, L., ed. Handbook of Genetic Algorithms, Van Nostrand, Reinhold, 1991

[3] Goldberg, D., Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, 1989

[4] A.E. Eiben, J.E. Smith, Introduction to Evolutionary Computation, Springer-Verlag, Berlin, Germany, 2003.

[5] A. Fernandez, M.J. del Jesus, F. Herrera, On the 2-tuples based genetic tuning performance for fuzzy rule based classification systems in imbalanced datasets, Inf. Sci. 180 (8) (2010) 1268–1291.

[6]. P.P. Bonissone, R. Subbu, K.S. Aggour, Evolutionary optimization of fuzzy decision systems for automated insurance underwriting, in Proc. of IEEE Int. Conf. on Fuzzy Systems FUZZ-IEEE, 2006, pp. 1003–1008.