Designing Unsupervised Hierarchical Fuzzy Logic Systems

M. Mohammadian
University of Canberra, Australia

INTRODUCTION

Systems such as robotic systems and systems with large input-output data tend to be difficult to model using mathematical techniques. These systems have typically high dimensionality and have degrees of uncertainty in many parameters. Artificial intelligence techniques such as neural networks, fuzzy logic, genetic algorithms and evolutionary algorithms have created new opportunities to solve complex systems. Application of fuzzy logic [Bai, Y., Zhuang H. and Wang, D. (2006)] in particular, to model and solve industrial problems is now wide spread and has universal acceptance. Fuzzy modelling or fuzzy identification has numerous practical applications in control, prediction and inference. It has been found useful when the system is either difficult to predict and or difficult to model by conventional methods. Fuzzy set theory provides a means for representing uncertainties. The underlying power of fuzzy logic is its ability to represent imprecise values in an understandable form. The majority of fuzzy logic systems to date have been static and based upon knowledge derived from imprecise heuristic knowledge of experienced operators, and where applicable also upon physical laws that governs the dynamics of the process.

Although its application to industrial problems has often produced results superior to classical control, the design procedures are limited by the heuristic rules of the system. It is simply assumed that the rules for the system are readily available or can be obtained. This implicit assumption limits the application of fuzzy logic to the cases of the system with a few parameters. The number of parameters of a system could be large. The number of fuzzy rules of a system is directly dependent on these parameters. As the number of parameters increase, the number of fuzzy rules of the system grows exponentially.

Genetic Algorithms can be used as a tool for the generation of fuzzy rules for a fuzzy logic system. This automatic generation of fuzzy rules, via genetic algorithms, can be categorised into two learning techniques, supervised and unsupervised. In this paper unsupervised learning of fuzzy rules of hierarchical and multi-layer fuzzy logic control systems are considered. In unsupervised learning there is no external teacher or critic to oversee the learning process. In other words, there are no specific examples of the function to be learned by the system. Rather, provision is made for a task-independent measure of the quality or representation that the system is required to learn. That is the system learns statistical regularities of the input data and it develops the ability to learn the feature of the input data and thereby create new classes automatically [Mohammadian, M., Nainar, I. and Kingham, M. (1997)].

To perform unsupervised learning, a competitive learning strategy may be used. The individual strings of genetic algorithms compete with each other for the "opportunity" to respond to features contained in the input data. In its simplest form, the system operates in accordance with the strategy that 'the fittest wins and survives'. That is the individual chromosome in a population with greatest fitness 'wins' the competition and gets selected for the genetic algorithms operations (cross-over and mutation). The other individuals in the population then have to compete with fit individual to survive.

The diversity of the learning tasks shown in this paper indicates genetic algorithm's universality for concept learning in unsupervised manner. A hybrid integrated architecture incorporating fuzzy logic and genetic algorithm can generate fuzzy rules for problems requiring supervised or unsupervised learning. In this paper only unsupervised learning of fuzzy logic systems is considered. The learning of fuzzy rules and internal parameters in an unsupervised manner is performed using genetic algorithms. Simulations results have shown that the proposed system is capable of learning the control rules for hierarchical and multi-layer fuzzy logic systems. Application areas considered are, hierarchical control of a network of traffic light control and robotic systems.

D

A first step in the construction of a fuzzy logic system is to determine which variables are fundamentally important. Any number of these decision variables may appear, but the more that are used, the larger the rule set that must be found. It is known [Raju, S., Zhou J. and Kisner, R. A. (1990), Raju G. V. S. and Zhou, J. (1993), Kingham, M., Mohammadian, M, and Stonier, R. J. (1998)], that the total number of rules in a system is an exponential function of the number of system variables. In order to design a fuzzy system with the required accuracy, the number of rules increases exponentially with the number of input variables and its associated fuzzy sets for the fuzzy logic system. A way to avoid the explosion of fuzzy rule bases in fuzzy logic systems is to consider Hierarchical Fuzzy Logic Control (HFLC) [Raju G. V. S. and Zhou, J. (1993)]. A learning approach based on genetic algorithms [Goldberg, D. (1989)] is discussed in this paper for the determination of the rule bases of hierarchical fuzzy logic systems.

THE GENETIC FUZZY RULE GENERATOR ARCHITECTURE

In this section we show how to learn the fuzzy rules in a fuzzy logic rule base using a genetic algorithm. The full set of fuzzy rules is encoded as a single string in the genetic algorithm population. To facilitate this we develop the genetic fuzzy rule generator whose architecture consists of five basic steps

1. Divide the input and output spaces of the system to be controlled into fuzzy sets (regions),

2. Encode the fuzzy rules into bit-string of 0 and 1,
3. Use a genetic algorithm as a learning procedure to generate set of fuzzy rules,
4. Use a fuzzy logic controller to assess the set of fuzzy rules and assign a value to each generated set of fuzzy rules,
5. Stop generating new sets of fuzzy rules once some performance criteria is met,

Figure 1 shows the genetic fuzzy rule generator architecture graphically. Suppose we wish to produce fuzzy rules for a fuzzy logic control with two inputs and single output. This simple two-input u_1, u_2 single-output y case is chosen in order to clarify the basic ideas of our new approach. Extensions to multi-output cases are straightforward. For more information on multi-output cases refer to Mohammadian et al [Mohammadian, M. and Stonier, R J., (1998)].

As a first step we divide the domain intervals of u_1, u_2 and y into different fuzzy sets. The number of the fuzzy sets is application dependent. Assume that we divide the interval for u_1, u_2 and y into 5, 7 and 7 fuzzy sets respectively. For each fuzzy set we assign a fuzzy membership function. Therefore a maximum of 35 fuzzy rules can be constructed for this system. Now the fuzzy rule base can be formed as a 5×7 table with cells to hold the corresponding actions that must be taken given the condition corresponding to u_1, u_2 are satisfied.

In step 2 we encode the input and output fuzzy sets into bit-strings (of 0 and 1). Each complete bit-string consists of 35 fuzzy rules for this example and each

Figure 1. Genetic fuzzy rule generator architecture
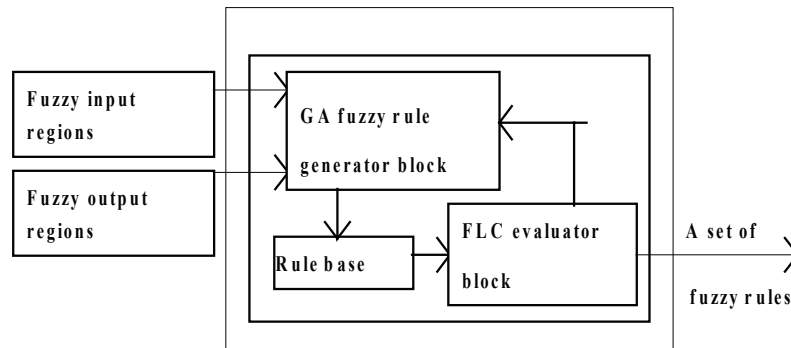
Related Content

The Role of Analytics Within the Higher Education Institutions
Hamid Mahroeian (2022). *Strategy, Policy, Practice, and Governance for AI in Higher Education Institutions (pp. 23-47).*
www.irma-international.org/chapter/the-role-of-analytics-within-the-higher-education-institutions/304100

Towards a Conceptual Framework for Customer Intelligence in the Era of Big Data
Nguyen Anh Khoa Dam, Thang Le Dinhand William Menvielle (2021). *International Journal of Intelligent Information Technologies (pp. 1-17).*
www.irma-international.org/article/towards-a-conceptual-framework-for-customer-intelligence-in-the-era-of-big-data/289968

Predicting Healthcare Readmissions Using Artificial Intelligence
Manu Banga (2023). *Handbook of Research on AI and Machine Learning Applications in Customer Support and Analytics (pp. 276-298).*
www.irma-international.org/chapter/predicting-healthcare-readmissions-using-artificial-intelligence/323125

Hardware and Software Cyber Security Tools
Tamalika Das, Nabonita Nath, Kshounish Acharyya, Shirsa Chakrabortyand Parag Chatterjee (2023). *AI Tools for Protecting and Preventing Sophisticated Cyber Attacks (pp. 109-131).*
www.irma-international.org/chapter/hardware-and-software-cyber-security-tools/328596

Risks in Sustainable Food Supply Chain Management
Yogesh Kumar Sharma, Sachin Kumar Manglaand Pravin P. Patil (2019). *Advanced Fuzzy Logic Approaches in Engineering Science (pp. 117-131).*
www.irma-international.org/chapter/risks-in-sustainable-food-supply-chain-management/212332