Biogeography Based Optimization Approach for Rule extraction for a Battery Charger System

Shakti Kumar[1], Parul Narula[2] & Tazeem Ahmed[3]

[1]Computational Intelligence, Laboratory, Institute of Science & Technology Klawad, Haryana, shakti@istk.org

[2] Department of Electronics & instrumentation, Apeejay College of Engg. , Sohna, parul.narula@gmail.com

[3] Department of Electronics & Communication, Alfalah School of Engg. & Technology, khan_taz@yahoo.com

Abstract- **Fuzzy rule based systems are a class of knowledge based systems. The intelligence of a fuzzy logic based system lies in its rule base. Two approaches can be found in the literature which are used for rule base generation. In the knowledge driven fuzzy models, the requisite rule base is provided by domain experts and knowledge engineers. In the data driven models, the rule base is generated from the available numerical data. As, the domain experts are difficult to find and the requisite knowledge extraction from the experts itself is a difficult task, the data driven modeling assumes significance. Fuzzy systems are used to model highly complex and highly nonlinear systems and under the circumstances, the rule base extraction problem becomes NP hard problem. When the problem is very complex, application of classical methods turns out to be very expensive computationally. One has to apply soft computing based methodology to extract a rule base from data. Neural networks, genetic algorithms, ant colony optimization and particle swarm optimization are some of the approaches, which can be found in the literature.**
In this paper, we present biogeography based optimization (BBO) for the rule base generation of Mamdani type fuzzy logic based systems. Biogeography is the study of the geographical distribution of biological organisms. It is a burgeoning nature inspired technique to find the optimal solution of the problem. In BBO, habitats represent the problem solutions, and species migration represents the sharing of features between solutions according to the fitness of the habitats. The results indicate that the BBO is a very promising optimizing algorithm for evolving fuzzy logic based systems.

Keywords —**Biogeography, Biogeography Based Optimization, Fuzzy membership function, Mamdani system, Rule generation.**

I INTRODUCTION

The identification of dynamic nonlinear systems is a difficult task as demonstrated by the great effort devoted by many researchers during the last decade. Many identification techniques based on fuzzy interference have been proposed that can explain the behavior of an unknown system for which only a set of input output data is available. Fuzzy modeling approach for system identification from numerical data has a distinguishing feature in that it can express complex nonlinear system linguistically using fuzzy inference rules. The premise part of a fuzzy rule defines a local fuzzy region while the consequent part describes the behavior within this region. The consequent can be a fuzzy set, a constant, or a linear equation. That is, different consequents result in different fuzzy inference systems, but their premise parts are always the same. One of the most outstanding fuzzy inference models is the one suggested by Mamdani and his associates [1] in the early 1970s whose rule-base consequent part is a fuzzy set. It is easy to implement in a digital computer and is intuitively persuasive toward human beings. Since then, various fuzzy modeling techniques for the design of controllers have been developed [2], [3]. Traditionally, system modeling using fuzzy rules has been obtained through discussions with domain experts but this approach has many problems and shortcomings the interviews are generally long, inefficient and frustrating for both the domain experts and knowledge engineers, especially so in domains where experts make decisions based on incomplete or imprecise information. This knowledge acquisition phase is often the main bottleneck within the knowledge engineering process and therefore considerable effort has been expended in designing algorithms that automatically induce fuzzy rules from historical data [4]. With this requirement in mind, a large number of methods have been proposed to automatically generate fuzzy rules from numerical data making use of different soft computing techniques [5-14], such as neural networks, genetic algorithms, swarm intelligence, ant colony optimization etc.

In this paper a novel way of facing the fuzzy rule base problem is proposed making use of biogeography based optimization algorithms. The science of biogeography can be traced to the work of century nationalists such as Alfred Wallace [15] and Charles Darwin [16].

The paper is divided into five sections. Section II introduces the biogeography based optimization (BBO). Section III presents a brief of fuzzy rule based systems. Section IV presents the BBO algorithm and methodology for rule base generation using BBO. Finally section V of this paper presents with an example the application of BBO algorithm to the design of a rule base for a fuzzy logic based battery charger, where operator's control actions for charging are linguistically modeled to design a fuzzy controller. The data for the battery

1

charger has been obtained through experimentation with an objective to charge the batteries as fast as possible.

II BIOGEOGRAPHY BASED OPTIMIZATION

Biogeography Based Optimization (BBO) is a population based evolutionary algorithm (EA) motivated by the migration mechanisms of ecosystems. It is based on the mathematics of biogeography. In BBO, problem solutions are represented as islands, and the sharing of features between solutions is represented as emigration and immigration. The idea of BBO was first presented by D. Simon [17] in December 2008. It is an example of natural process that can be modeled to solve general optimization problems. One characteristic of BBO is that the original population is not discarded after each generation; it is rather modified by migration. Also for each generation, BBO uses the fitness of each solution to determine its emigration and immigration rate [17]. In BBO, each individual is considered as a habitat with a habitat suitability index (HSI), which is similar to the fitness of EAs, to measure the individual. Also, an SIV (suitability index variable) that characterizes the habitability of an island is used. A good solution is analogous to an island with a high HSI, and a poor solution indicates an island with a low HSI. High HSI solutions tend to share their features with low HSI solutions. Low HSI solutions accept a lot of new features from high HSI solutions [17]. As with every other evolutionary algorithm, each solution might also have some probability of mutation, although mutation is not an essential feature of BBO.

III FUZZY RULE-BASED SYSTEMS

Fuzzy rule based systems are an extension of classical rule based systems. Fuzzy rules are linguistic IF-THEN constructions that have the general form "IF A, THEN B" where A and B are (collections of) propositions containing linguistic variables. In effect, the use of linguistic variables and fuzzy IF-THEN rules exploits the tolerance for imprecision and uncertainty. In this respect, fuzzy logic mimics the crucial ability of the human mind to summarize data and focus on decision-relevant information.

A fuzzy rule based system consists of four major modules: fuzzification, inference engine, knowledge base and defuzzification module [18]. The fuzzification module transforms the crisp input(s) into fuzzy values. These values are then processed in fuzzy domain by inference engine based on the knowledge base supplied by the domain expert(s). The knowledge base is composed of the Rule Base (RB), characterizes the control goals and control policy of the domain expert by a set of linguistic control rules, and of the Data Base (DB), containing the term sets and the membership functions defining their semantics. Finally, the processed output is transformed from fuzzy domain to crisp domain by defuzzification module. The structure of a linguistic Fuzzy rule-based system is shown in Figure 1.
The structure of a rule base can be stated as follows:

$$R_i : \text{if } X_i \text{ is } A_{i1} \ldots X_n \text{ is } A_{in} \text{ then } Y \text{ is } B_j$$

Where A_{in} and B_j are fuzzy sets defined on the input and output domains respectively. $X_1 \ldots X_n$ and Y are input and output
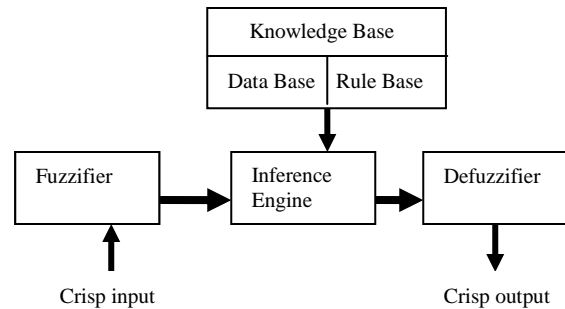

Figure 1 Structure of fuzzy rule-based system

linguistic variables, respectively, and $A_{i1} \ldots A_{in}$ and B_j linguistic labels, each one of them having associated a fuzzy set defining its meaning.

A. Design steps for Fuzzy based rule system-
The design steps for fuzzy rule base designing [19] are described as -
1. Identify the input and output variables.
2. For these variables, generate membership functions and decide their shapes, such as, triangular, Z-type, S-type etc.
3. Generate rule base for the system.
4. Select the type of inference are rule composition operator, implication and aggregation operators.
5. Decide on the defuzzification technique and generate a crisp control action (defuzzification).

For the systems of small complexity the experts can perform Step 1 by including all the available inputs. For the systems of higher complexity, it is not possible to take into account all the inputs and one may be constrained to select only those inputs, which have significant contribution to the overall output of the system. Some of the suggested procedures in literature are forward selection procedure, backward elimination procedure, best subset method and few other statistical selection procedures [14]. Redundant variables may be removed using exhaustive search technique. Step 2 can be performed with the help of domain expert(s) if they are available, from the common sense, or from the available numerical data. In case numerical data is available for these variables the membership functions generating using techniques like, FCM, neural networks, GA etc can be used. As far as Step 4, is concerned there may be hundreds of combinations of composition, implication and aggregation operators. For Step 5, a large number of defuzzification techniques are available in the literature. Some of the commonly used defuzzification techniques are, centre of gravity (COG), centre of sum methods, first/last of maxima, Mean of maxima (MOM).

Step 3 involves the development of rule base. In the case of a knowledge based model development, Step 3 is performed by a knowledge engineer with the help of domain experts whereas in the case of data driven modeling, certain computerized techniques are used to develop the rule base. Rule base generation techniques are classified into four categories. The first category will be called as classical methods. This consists of methods like the one proposed by Wang and Mendel [20]; the second category consists of all the methods that employ genetic algorithm (GA) based methods [6,9,10]. The third category contains neural networks based methods for rule base generation [5,8]. The methods employing the swarm intelligence techniques (PSO and ACO) will be placed into category four [5,11,12]. This paper proposes another technique based on the mathematics of biogeography, *i.e.*, biogeography based optimization (BBO) algorithm. This technique can be placed in soft computing based algorithms, *i.e.*, genetic algorithms and evolutionary algorithms.

B. Rule Base Generation Problem Formulation

Figure 2 represents Mamdani fuzzy systems. It is clear from figure that such system consists of four major modules, *i.e.*, fuzzifier, rule composition module (fuzzy 'MIN' operators), implication module (fuzzy 'MIN' operators in this case), and defuzzification module.

The overall computed output, in the case of a Mamdani type system, could be written as follows:

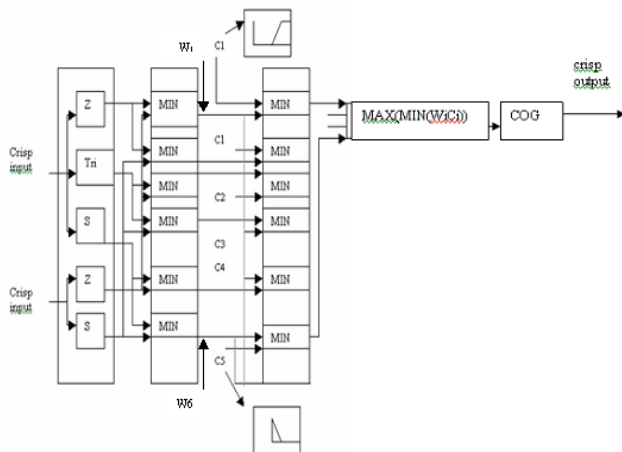$$\text{Computed output} = COG (Max (Min (W_i, C_i))) \quad (1)$$

Figure 2 Mamdani fuzzy system

Where COG is centre of gravity defuzzification, which is expressed as

$$X^* = [\sum \mu_A (x). x] / [\sum \mu_A (x)]$$

Where x is a running point in a discrete universe of discourse and $\mu_A (x)$ is its membership value in the membership function.

W_i and C_i are weights or matching degree and consequents respectively. Total numbers of rules are R, but these rules are incomplete as we are aware of only antecedent parts and consequent part is yet to be determined. Because for any set of inputs W_i are easily computed by fuzzifier and rule composing modules, the right hand side of output expression (1) can be evaluated if we could choose the proper values for C_is.

For a given data set of a system, W_is are known. We have to find the appropriate values of C_i, such that, the difference between the computed output and the actual output as given in data is minimum.

The computed output is compared with actual output as given in data set and finds the error. Let the error be defined as follows:

Error (E) = Actual output (as given in data set) – computed Output (as given in eq^n (1)).

Now the whole problem of rule base generation boils down to a minimization problem as stated below:

Minimize objective function : E = Oactual – Ocomputed

Where Oactual = Actual output, Ocomputed = Computed output.

Subject to the constraint that C_i ε {specified set of Consequents}. (2)

IV BBO ALGORITHM

1. Initialize the BBO parameters: maximum species count Smax, the maximum migration rates *E* and *I*, the maximum mutation rate m_{max}, an elitism parameter and number of iterations.
2. Initialize a random set of habitats, each habitat corresponding to a potential solution to the given problem.
3. For each habitat, map the Habitat Suitability Index (HSI) to the number of species *S*, the immigration rate 1 and the emigration rate μ.
4. Probabilistically use immigration and emigration to modify each non-elite habitat, then recompute each HSI.
5. For each habitat, update the probability of its species count given by equation (4). Then, mutate each non-elite habitat based on its probability and recompute each HSI.

$$\dot{P}_s = \begin{cases} -(\lambda_s + \mu_s)P_s + \mu_{s+1}P_{s+1} & S = 0 \\ -(\lambda_s + \mu_s)P_s + \lambda_{s-1}P_{s-1} + \mu_{s+1}P_{s+1} & S = 1, \cdots, S_{max} - 1 \\ -(\lambda_s + \mu_s)P_s + \lambda_{s-1}P_{s-1} & S = S_{max} \end{cases}$$

(3)

where λs and μs are the immigration and emigration rates, when there are S species in the habitat.
6. Is acceptable solution found? If yes then go to Step 8.
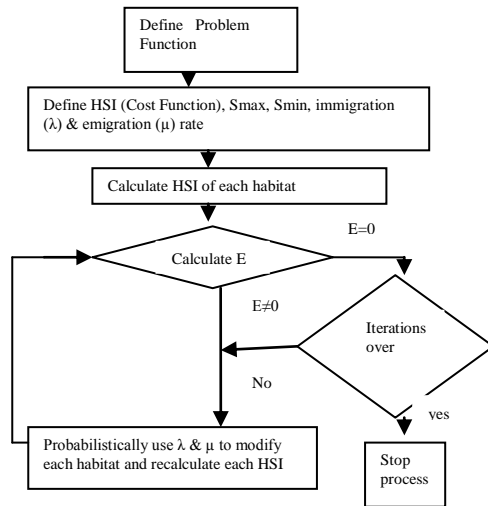7. Number of iterations over? If no then go to Step 3 for the next iteration.
8. Stop

Figure 3 Flow Chart

The algorithm discussed above is illustrated in flow diagram (figure 3)

V RULE BASE GENERATION USING BBO

Here, the algorithm that extracts rules from data set is Discussed [22].

Step 1: Divide Input and Output Spaces into Fuzzy Regions

For each of the input and output variable its range or interval from starting value to end value is identified. Each of the input and output spaces are partitioned into K regions (membership functions). The number K can be different for each of the variables. Neural networks, GAs and Fuzzy C-means clustering (FCM) etc. can be used to partition input and output variable ranges.

Step 2: For Each Training Example Considering the Input Values Generate Fuzzy Rule(s)

For the steps which follow, let x_1 and x_2 be the inputs and y be the output variable. In order to implement this step, that is followed as:

  (a) Determine the membership grade μ_i of a given input x_i in to the different fuzzy membership functions.

  (b) Enumerate the composed values or depth of firing *Wis*.

  (c) Call BBO algorithm to choose appropriate value of *Ci* from the specified set, such that,

   $error_i$ = (actual output$_i$ – computed output$_i$) is minimized.

  (d) Create a combined fuzzy rule base.

  (e) Rule Reduction Technique: Assign a degree to each rule as follows,

D (rule) = μ_A (x_1) × μ_B (x_2) (product of membership grade of input x_1 in fuzzy set A and membership grade of x_2 in fuzzy set B). At this point if an expert is available and he assigns his degree of belief in the correctness of particular data set then that degree of belief '*m*' must be multiplied

Table 1 Input and output variables for rapid Ni-Cd battery charger along with their universes of discourse

	Minimum value	Maximum value
Input variables		
Temperature (T),°C	0	50
Temperature gradient (dT/dt), °C/s	0	1
Output variable		
Charging rate (C_l), A	0	8C

with above expression. This will help to choose an appropriate rule when conflicts like two rules with same antecedent part but different consequent result. Then, the rule that has maximum degree is chosen. In case two rules are exactly similar then discard the one with lower degree. After discarding the redundant and contradicting rules, the complete set of rules that constitute the final compact rule base for the system is obtained.

VI DESIGN OF A FUZZY LOGIC BASED BATTERY CHARGER

A simple system of fuzzy battery charger is taken. Let the temperature range is 0° to 50°C (in terms of corresponding voltage) and the temperature gradient as obtained by taking time derivative of the conditioned signal as obtained from temperature sensor, varied from 0 to 1 mV/10 s. The input and output variables identified for rapid Ni-Cd battery charger along with their universes of discourse are listed in Table 1. Using Fuzzy Modified C-means clustering (FCM), the temperature range is fuzzified into three membership functions and temperature gradient into two membership functions. Same labels for these membership functions are chosen. Because there are three membership functions for the input 'temperature' and there are two inputs for the second input, *i.e.*, 'temperature gradient' there can be at most 3 × 2 = 6 combinations or antecedent parts of the rules. The rule base is yet incomplete as for each rule the consequent need to be found out. From the given dataset of Table 1 it is found that there are only five consequents that form the set of consequents from where one particular element is chosen as the consequent for a particular rule. The specified set of consequents in this case are $C1 = \mu_{ultrafast}$, $C2 = \mu_{high}$, $C3 = \mu_{medium}$, $C4 = \mu_{low}$ and, $C5 = \mu_{trickle}$. The parameters of antecedent and consequents are chosen in such a way so as to fulfill condition given by expression (2). Degree of compatibility of any input data set to rule represented by W_i can be easily computed using the following formula

$$W_1 = min (\mu_{LOW} (temperature), \mu_{LOW} (temp\_grad))$$

4

This way all the W_i are evaluated, the right hand side of output expression (1) can be evaluated if the proper values for C_i ε {ULTRAFAST, MED, LOW, HIGH, TRICKLE} can be chosen.

The BBO algorithm is implemented in MATLAB to select the values of consequents to satisfy the equation (2). It was observed that the algorithm was successfully able to generate the required rule base for the FLS shown in figure 4. With the application of rule reduction algorithm as given in [Step 2- (e)] BBO leads to the following rule set of only six rules.

Figure 4. Extracted Rulebase Using BBO

It has to be stated clearly, that this rule set is the best result obtained during a few runs of the BBO and it could not always be reached. It is very easy to interpret and has the nice property that the more relevant input 'temperature' is used for all rules, while the less relevant input 'temperature gradient' only appears in two rules in combination with the other input.

VII CONCLUSIONS

Rule extraction from numerical data is a high computational complexity problem. BBO was applied to successfully extract rule base from numerical data. All rules were extracted from the given data for Mamdani type system. The method appears to be very efficient. Its performance is being compared with other optimization approaches, *ie*, Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO) and Genetic Algorithms (GA) based optimization techniques. The preliminary results indicate that BBO appears to be most efficient approach to such NP hard problems.

REFERENCES

[1] Mamdani, E.H., Assilian, S. (1975). An experiment in linguistic synthesis with a fuzzy logic controller. *Intern. J. Man- Machine Studies*, *7*, 1-13.

[2] Sugeno, M.(1985). An introductory survey of fuzzy control. *Information Sciences*, *36*, 59-83.

[3] Yager, R.R., & Filev, D.P. (1994).Essentials of fuzzy modeling and control, *New York: Wiley*.

[4] Mendel, J.M., & Mouzouris, G.C. (1997). Designing Fuzzy Logic Systems. *IEEE Transactions on Circuits and Systems II, Analog and Digital Signal Processing*, *44*(11), 1414-1427.

[5] He, Z., Wei, C., Yang, L., Gao, X., Yao, S., Eberhart, R.C., & Shi, Y. (1998). Extracting Rules from Fuzzy Neural Network by Particle Swarm Optimization. *IEEE International Conference on Evolutionary Computation*, Anchorage, Alaska, USA.

[6] Lee, M.A., & Takagi, H. (1993). Integrating Design Stages of Fuzzy Systems using Genetic Algorithms. *Proceedings Second IEEE International Conference on Fuzzy Systems*, 612-617.

[7] Mahfouf, M., Jamei, M., & Linkens, D.A. (2001). Rule-Base Generation via Symbiotic Evolution for a Mamdani-Type Fuzzy Control System. *IEEE International Fuzzy Systems Conference*, 396-399.

[8] Fu, L. (1994). Rule Generation from Neural Networks. *IEEE Transactions on Systems, Man and Cybernetics*, *24*(8), 1114-1124.

[9] Ashwani, K., Agarwal, D.P., & Joshi, S.D. (2003). A GA-based Method for Constructing TSK Fuzzy Rules from Numerical Data. *IEEE International Conference on Fuzzy Systems*, 131-136.

[10] Pal, T., & Pal, N.R. (2003). SOGARG: A Self Organized Genetic Algorithm based Rule Generation Scheme for Fuzzy Controllers. *IEEE Transactions on Evolutionary Computation*, *7*(4).

[11] Dorigo, M., & Stutzle, T. (2005). Ant Colony Optimization. *Eastern Economy Edition*, *PHI*.

[12] Casillas, J., Cordon, O. and Herrera, F. (2000). Learning Fuzzy Rules using Ant Colony Optimization Algorithms. *Proceedings the Second International Workshop Ant Algorithms*, 13-21.

[13] Kumar, S. (2005). Automatic Fuzzy Rule base Generation. *Proceedings of the One Week Workshop on Applied Soft Computing CAT, Haryana Engineering College*, Jagadhri, Haryana, 26-42.

[14] Kumar, S., Kaur, P., & Singh, A. (2009) Soft Computing Approaches to Fuzzy System Identification: A Survey. *The Third International Conference on Intelligent Systems and Networks (IISN), ISTK, Jagadhri, Haryana*, 402-411.

[15] Wallace, A. (2005). The Geographical Distribution of Animals (Two volumes, Boston, *MA: Adamant Media Corporation*.

[16] Darwin, C. (1995). The Origin of Species. *New York: Gramercy Books*.

[17] Simon, D. (2008). Biogeography-Based Optimization. *IEEE Transaction on Evolutionary Computation*, *12* (6), 702-713.

[18] Abe, S., & Lan, M.S. (1995). Fuzzy Rule Extraction Directly from Numerical Data for Function Approximation. *IEEE Transactions on Systems, Man and Cybernetics*, *25*(1), 119-129.

[19] Kumar, S., Bhalla, P., Singh, A.P. (July, 2009). Fuzzy Rule base Generation from Numerical Data using Biogeography-based Optimization. *IE (I) Journal–ET*, *90*.

[20] Wang, L.X., & Mendel, J.M. (1992). Generating Fuzzy Rules from Examples. *IEEE Transactions on Systems, Man and Cybernetics*, *22*(6), 1414-1427.

[21] Panchal, V.K, Singh, P., Kaur, N., & Kundra, H. (2009). Biogeography based satellite image classification. *(IJCSIS) International Journal of Computer Science and Information Security,. 6(2)*, 269-274.

[22] Panchal, V.K., Goel, S., Bhatnagar, M. (2009). Biogeography based land cover feature extraction. *IEEE World Congress on Nature & Biologically Inspired Computing*.

[23] Setnes, M., Babuska, R., & Verbruggen, H.B. (1998). Rule-based Modelling: Precision and Transparency. *IEEE Transactions on Systems, Man and Cybernetics*, *28*(1), 165-169.