IJESMR
International Journal OF Engineering Sciences & Management Research

NEURO-FUZZY SYSTEMS: A HYBRID INTELLIGENT APPROACH

Manoj Sharma[*1]
[*1] Department of Electronics & Communication Engineering Dronacharya College of Engineering, Gurgaon, Haryana, India.

Keywords: Neural Networks, Fuzzy Logic, Neuro-fuzzy modelling, ANFIS, back propagation algorithms.

ABSTRACT
Integration of Artificial Neural Networks (ANN) and Fuzzy Inference Systems (FIS) have attracted the growing interest of researchers in various scientific and engineering areas due to the growing need of adaptive intelligent systems to solve the real world problems. ANN learns from scratch by adjusting the interconnections between layers. FIS is a popular computing framework based on the concept of fuzzy set theory, fuzzy if-then rules, and fuzzy reasoning. The advantages of a combination of ANN and FIS are obvious. There are several approaches to integrate ANN and FIS and very often it depends on the application. Neuro-fuzzy modelling can be regarded as a gray-box technique on the boundary between neural networks and qualitative fuzzy models. The tools for building neuro-fuzzy models are based on combinations of algorithms from the fields of neural networks, pattern recognition and regression analysis. In this paper, an overview of neuro-fuzzy modelling given.

INTRODUCTION
The techniques of artificial intelligence based in fuzzy logic and neural networks are frequently applied together. The reasons to combine these two paradigms come out of the difficulties and inherent limitations of each isolated paradigm. Generically, when they are used in a combined way, they are called Neuro-Fuzzy Systems. This term, however, is often used to assign a specific type of system that integrates both techniques. This type of system is characterised by a fuzzy system where fuzzy sets and fuzzy rules are adjusted using input output patterns. The modern techniques of artificial intelligence have found application in almost all the fields of the human knowledge. However, a great emphasis is given to the accurate sciences areas, perhaps the biggest expression of the success of these techniques is in engineering field. These two techniques neural networks and fuzzy logic are many times applied together for solving engineering problems where the classic techniques do not supply an easy and accurate solution. The neuro-fuzzy term was born by the fusing of these two techniques. As each researcher combines these two tools in different way, then, some confusion was created on the exact meaning of this term. Still there is no absolute consensus but in general, the neuro-fuzzy term means a type of system characterized for a similar structure of a fuzzy controller where the fuzzy sets and rules are adjusted using neural networks tuning techniques in an iterative way with data vectors (input and output system data). Such systems show two distinct ways of behaviour. In a first phase, called learning phase, it behaves like neural networks that learns its internal parameters off-line. Later, in the execution phase, it behaves like a fuzzy logic system.

Neuro-fuzzy modeling has been recognized as a powerful tool which can facilitate the effective development of models by combining information from different sources, such as empirical models, heuristics and data. Neuro-fuzzy models describe systems by means of fuzzy if-then rules, such as ‘If x is small then y is large’ represented in a network structure, to which learning algorithms known from the area of artificial neural networks can be applied. Thanks to this structure, neuro-fuzzy models are to a certain degree transparent to interpretation and analysis, i.e., can be better used to explain solutions to users than completely black-box models such as neural networks. Both neural networks and fuzzy systems are motivated by imitating human reasoning processes. In fuzzy systems, relationships are represented explicitly in the form of if–then rules. In neural networks, the relations are not explicitly given, but are ‘coded’ in the network and its parameters. In contrast to knowledge-based techniques, no explicit knowledge is needed for the application of neural nets. Neuro–fuzzy systems combine the semantic transparency of rule-based fuzzy systems with the learning capability of neural networks.

NEURO-FUZZY SYSTEMS
Since the moment that fuzzy systems become popular in industrial application, the community perceived that the development of a fuzzy system with good performance is not an easy task. The problem of finding membership functions and appropriate rules is frequently a tiring process of attempt and error. This leads to the idea of applying learning algorithms to the fuzzy systems. The neural networks, that have efficient learning algorithms, had been

ISSN 2349-6193
Impact Factor (PIF): 2.243

IJESMR

International Journal OF Engineering Sciences & Management Research

presented as an alternative to automate or to support the development of tuning fuzzy systems. The first studies of the neuro-fuzzy systems date of the beginning of the 90's decade, with Jang, Lin and Lee in 1991, Berenji in 1992 and Nauck from 1993, etc. The majority of the first applications were in process control. Gradually, its application spread for all the areas of the knowledge like, data analysis, data classification, imperfections detection and support to decision-making, etc. Neural networks and fuzzy systems can be combined to join its advantages and to cure its individual illness. Neural networks introduce its computational characteristics of learning in the fuzzy systems and receive from them the interpretation and clarity of systems representation. Thus, the disadvantages of the fuzzy systems are compensated by the capacities of the neural networks. These techniques are complementary, which justifies its use together.

NEURO-FUZZY SYSTEM ARCHITECTURES

In this section the different architectures will be discussed to show how different approaches managed to combine ANNs with Fuzzy Systems.

A. ANFIS Architecture

One of the first Neuro-Fuzzy Systems was introduced by Jang ([1], [2]). This architecture is called ANFIS (Adaptive-Network-based Fuzzy Inference System) and it uses the Takagi-Sugeno inference system. Figure 1 show the ANFIS architecture which consists of six layers.

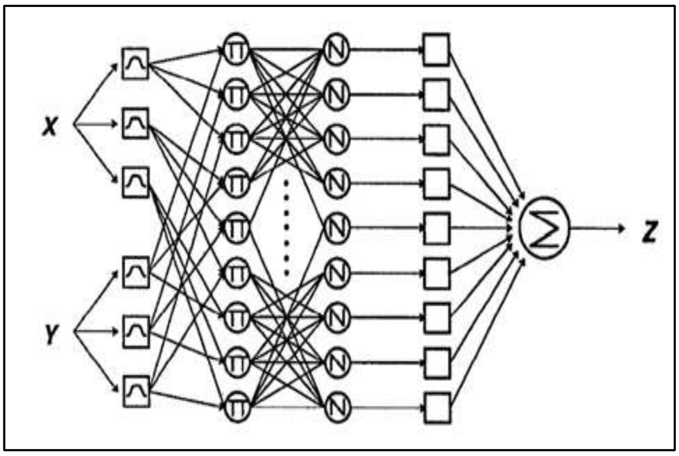

Fig. 1. The ANFIS Architecture

The first layer contains two nodes for input x and y, the second layer is responsible for mapping input values to the membership functions. The nodes of the third layer correspond to the fuzzy rules in the form of production functions; their output values are the firing strengths of each rule while the nodes in the fourth layer calculate the ratio to the sum of all rules' firing strengths. Defuzzyfication happens in the fifth layer and the sixth layer's output nodes sum their input values. Iterative learning of ANFIS is composed of two stages. In the first stage the parameters of the consequent functions (in the fifth layer) are tuned via a least mean square method. During the second stage the parameters of the premise functions (in the second layer) are adjusted by a back propagation algorithm. These two stages are repeated iteratively for training of the system.

B. Cooperative Neuro-Fuzzy Systems

In a cooperative system the neural networks are only used in an initial phase. In this case, the neural networks determines sub-blocks of the fuzzy system using training data, after this, the neural networks are removed and only the fuzzy system is executed. In the cooperative neuro-fuzzy systems, the structure is not total interpretable what can be considered a disadvantage.

ISSN 2349-6193
Impact Factor (PIF): 2.243

IJESMR

International Journal OF Engineering Sciences & Management Research

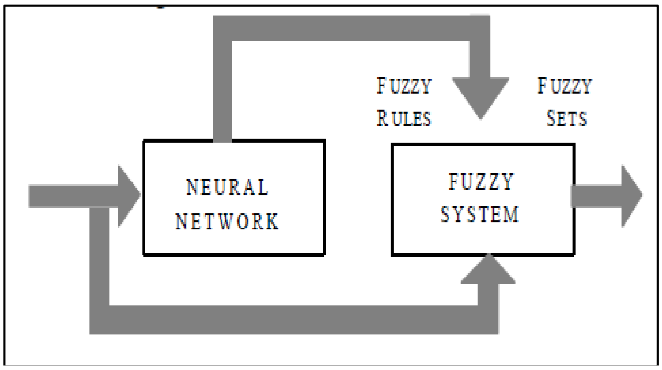

Figure 2. Cooperative Systems

C. Concurrent Neuro-Fuzzy Systems

A concurrent system is not a neuro-fuzzy system in the strict sense, because the neural network works together with the fuzzy system. This means that the inputs enters in the fuzzy system, are pre-processed and then the neural network processes the outputs of the concurrent system or in the reverse way. In the concurrent neuro-fuzzy systems, the results are not completely interpretable, what can be considered a disadvantage.

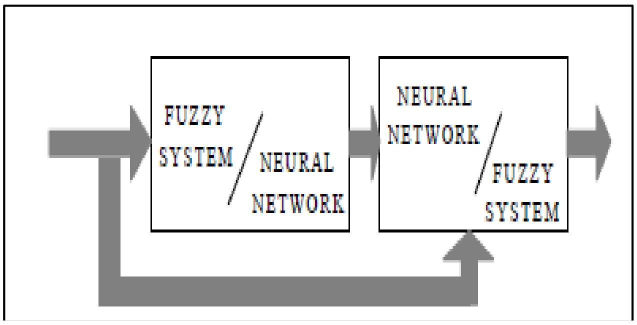

Figure 3. Concurrent Systems

D. GARIC Architecture

The GARIC (Generalized Approximate Reasoning based Intelligence Control) system is composed of three components: the ASN (Action Selection Network), the AEN (Action Evaluation Network) and the SAM (Stohastic Action Modifier) [3]. Figure 4 shows the ASN component. The ASN is a five layer network which is responsible for selecting an action based on the current state of the system using fuzzy inference. Input nodes are in the first layer and the second one holds the membership functions. Each node in the third layer represents a fuzzy rule and nodes of the forth layer correspond to consequent labels, e.g. if a consequent label is in a rule then there is a link between the label's node and the rule's node. The fifth layer's nodes calculate the real output values based on the rules' firing strength and the forth layer's outputs. AEN is a simple feedforward network which predicts reinforcements based on the state variables of the system.

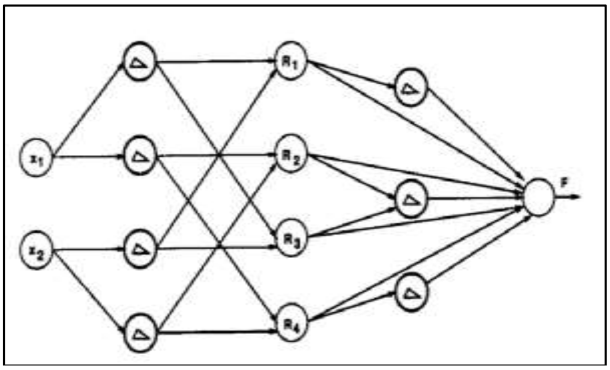

Fig. 4. The GARIC Architecture

ISSN 2349-6193
Impact Factor (PIF): 2.243

IJESMR
International Journal OF Engineering Sciences & Management Research

GARIC uses gradient descending and reinforcement learning to adjust its internal parameters.

E. SONFIN Architecture

SONFIN (Self-Constructing Neural Fuzzy Inference Network) is a Takagi-Sugeno-Kang-type fuzzy rule-based model which consists of six layers [4]. Figure 5 shows the SONFIN architecture which, in fact, is similar to the ANFIS. Layer 1-4 and 6 are functioning as they are in the ANFIS architecture. The fifth, consequent layer can hold two types of nodes. The first type represents the fuzzy sets by membership functions while the second type is optional and gains its inputs from the first and fourth layer. Constructing of SONFIN happens concurrently by a structure and a parameter learning method. The structure learning identifies both the precondition and consequent parts of the rules by minimizing the number of rules and membership functions for the input and by optimally generating new membership functions for the output variables. Parameter learning uses LMS or RLS algorithms to adjust consequent parameters and back propagation for precondition parameters.

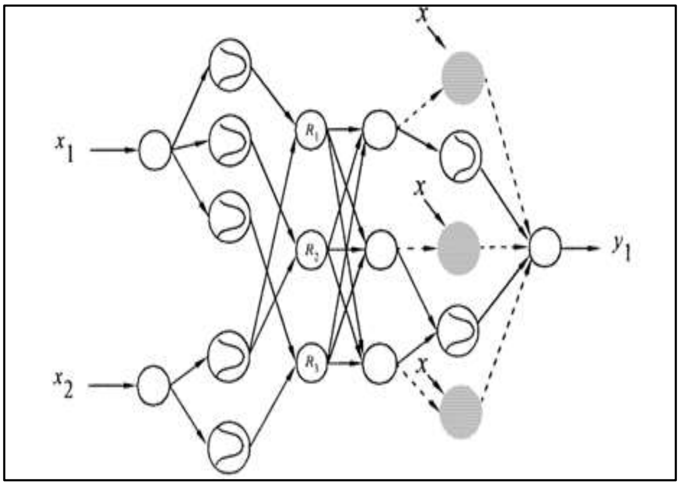

Fig. 5. The SONFIN Architecture

F. NEFCON Architecture

The Neural Fuzzy Controller NEFCON [5] was drawn to implement a Mamdani type inference fuzzy system as illustrated in figure 6.

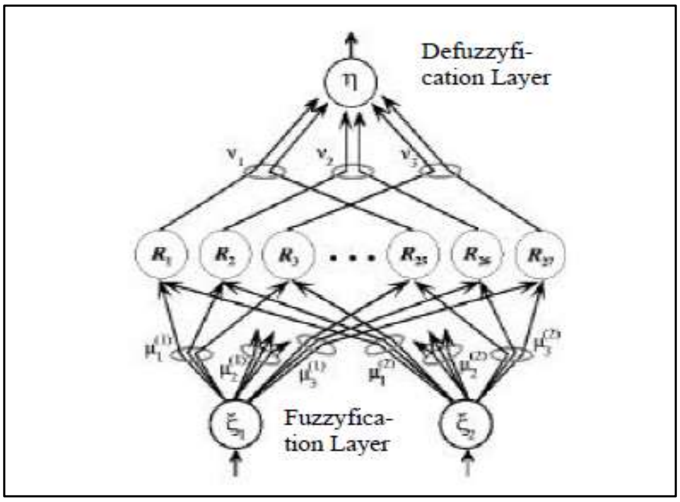

Fig. 6. The NEFCON Architecture

The connections in this architecture are weighted with fuzzy sets and rules using the same antecedents (called shared weights), which are represented by the drawn ellipses. They assure the integrity of the base of rules. The

ISSN 2349-6193
Impact Factor (PIF): 2.243

IJESMR

International Journal OF Engineering Sciences & Management Research

input units assume the function of fuzzyfication interface, the logical interface is represented by the propagation function and the output unit is responsible for the defuzzyfication interface. The process of learning in architecture NEFCON is based in a mixture of reinforcement learning with backpropagation algorithm. This architecture can be used to learn the rule base from the beginning, if there is no à priori knowledge of the system, or to optimise an initial manually defined rule base. NEFCON has two variants NEFPROX (for function approximation) and NEFCLASS (for classification tasks) [6].

CONCLUSION

Different applications of Neuro-Fuzzy Systems were discussed to show their high potential in technical diagnostics. These systems are successful because of their nature that they reveal the nature of the important interdependence between the parameters of the modelled system while they are, in fact, powerful approximators. The paper briefly reviewed the concept of Artificial Neural Networks and Fuzzy Systems as computational models and how they inspired the creation of Neuro-Fuzzy Systems. As it was discussed this fusion can unite the generalization capabilities of Neural Networks with the easy interpretability and high expressive power of fuzzy rules in an effective way. Six different architectures were presented and it can be concluded that these are the most important ones although there are other structure variations, too. Usually each architecture organizes its nodes a slightly different way and consequently they use specific learning algorithms which are adapted to the different structures.

REFERENCES

[1] R. Jang, "Fuzzy Modeling Using Generalized Neural Networks and Kalman Filter Algorithm", *Proc. Ninth Nat. Conf. Artificial Intell.*, pp. 762-767, 1991.
[2] R. Jang, "ANFIS: Adaptive-Network-Based Fuzzy Inference System", *IEEE Transactions on systems, man, and cybernetics*, vol. 23: (3), pp. 665-685, 1993.
[3] H. R. Berenji, P. Khedkar, "Learning and Tuning Fuzzy Logic Controllers Through Reinforcements", *IEEE Transactions on Neural Networks*, vol. 3, pp. 724-740, 1992.
[4] F. C. Juang, T. C. Lin, "An On-Line Self Constructing Neural Fuzzy Inference Network and its applications", *IEEE Transactions on Fuzzy Systems*, vol. 6, pp. 12-32, 1998.
[5] D. Nauck, R, Kurse, "Neuro-FuzzySystems for Function Approximation", 4th International Workshop Fuzzy-Neuro Systems, 1997.
[6] S. Sulzberger, N. Tschichold e S. Vestli, "FUN: Optimization of Fuzzy Rule Based Systems Using Neural Networks", Proceedings of IEEE Conference on Neural Networks, San Francisco, March 1993, pp. 312-316.