

Solving Uncertain Problems using ANFIS

Dr G.S.V.P.Raju
Andhra University
Visakhapatnam, Andhra
Pradesh

V.Mary Sumalatha
D.I.E.T College
Anakapalle, Andhra
Pradesh India

K.V.Ramani
D.I.E.T College
Anakapalle, Andhra
Pradesh India

K.V.Lakshmi
D.I.E.T College
Anakapalle, Andhra
Pradesh India

ABSTRACT

Uncertain problems are problems that have no definitive way of solving. Many of the uncertain problems come under intelligence systems that exhibit the characteristics we associate with intelligence in human behavior. Soft Computing^[6] techniques which have drawn their inherent characteristics from biological systems, present an effective method for solving of even difficult inverse problems. The guiding principle of soft computing is to exploit the tolerance for imprecision, uncertainty and partial truth to achieve tractability, robustness and low cost solution, employment of soft computing for the solution of machine learning problems lead to high machine intelligence quotient. Hybrid intelligent systems deal with the integration of two or more of the technologies. The combined use of technologies has resulted in effective problem solving in comparison with each technology used individually and exclusively. The purpose of the paper is to solve an engineering problem, power failures in personal computers using neuro fuzzy modeling system ANFIS.

Keywords

Soft Computing, Hybrid Intelligent Systems, Robustness, Neuro-Fuzzy model, ANFIS.

1. INTRODUCTION

Artificial intelligence is an area of computer science related with designing, intelligent computer systems. These characteristics we associate with intelligence in human behavior. The term intelligence is less understood. The artificial intelligence tasks are learning, intuition, creativity, inference... and all are partially understood. Some technologies such as expert systems, neural networks, fuzzy logic, cellular automata, probabilistic reasoning exist to solve intelligent problems. Out of these neural networks, fuzzy logic and probabilistic reasoning are known as soft computing. Soft computing was introduced by Lotfi A.Zadeh. Probabilistic reasoning subsumes genetic algorithms, chaos and parts of learning theory. According to zadeh, soft computing differs from hard computing in its tolerance to imprecision, uncertainty and partial truth. Hard computing methods are based on mathematical approaches and therefore demand a high degree of precision and accuracy in their requirements. In most engineering problems the input parameters are used for obtaining solution to the problem. On the other hand Soft Computing techniques which have drawn their inherent characteristics from biological systems, present effective methods for the solution of even difficult inverse problems. The guiding principle of soft computing is to exploit the tolerance for imprecision, uncertainty and partial truth to achieve tractability, robustness and low cost solution, employment of soft computing for the solution of machine

learning problems lead to high machine intelligence quotient. Hybrid intelligent systems deal with the integration of two or more of the technologies. The combined use of technologies has resulted in effective problem solving in comparison with each technology used individually and exclusively.

So far many techniques are described for solving certain problems. And in this paper we explore techniques for solving problems which are incomplete and uncertain. In certain problems at any point a particular fact is believed to be true, believed to be false, or not considered. But now we consider two types of problems. The first one contains problems in which there is genuine randomness in real world. In these type of problems there a possibility of uncertainty but some knowledge about the likelihood of various outcomes is available. The second type contains problems in which the relevant world is not random it behaves normally unless there is some exception. Many tasks fall into these category and we call these tasks as intelligent tasks. These tasks can be solved using techniques like Probability, Statistical reasoning, Rule based systems, Bayesian networks, Fuzzy logic, Neural networks, Genetic algorithm and hybrid systems.

Many technologies like expert systems, neural networks, fuzzy logic, cellular automata, probabilistic reasoning exist to solve intelligent problems. Many hybrid technologies like neuro-fuzzy^[7], neuro-genetic and fuzzy-genetic hybrid systems are proposed based on back propagation algorithm, associative methods. Here in our research we implemented a hybrid system to solve uncertain problems in Engineering.

2. METHODOLOGY

2.1 Neural networks

Neural Networks^{[11][12]} are simplified models of nervous system defined from the computing performed by a human brain. Neural networks are interconnected network of a set of processing units called neurons. Neural network exhibits parallel distributed processing. Neural networks exhibit the following characteristics.

Pattern association

Generalization

Robustness

Fault tolerance

Parallel processing

High speed information processing.

Neural networks use two popular learning methods like supervised learning and unsupervised learning. Neural network architectures are classified into feed forward networks and feed backward networks like single layer feed forward networks, multi layer feed forward networks, recurrent networks, black propagation networks, Perceptron, Adaline, associative memory, Boltzmann machine, adaptive resonance theory, self organizing feature map and Hopfield network. Neural networks are efficient in solving various types of problems in different areas of pattern recognition, image processing, data compression, forecasting and optimization.

2.2 Fuzzy Logic

Fuzzy logic has been proposed by Lotfi A. Zadeh in the year 1965 which is a generalization of classical set theory. Fuzzy logic was proposed on fuzzy set theory to capture the way human beings represent and reason real world knowledge to solve uncertain problems. Uncertainty arises due to some peculiar characteristics like generality, vagueness, ambiguity and incomplete knowledge. A fuzzy set is represented mathematically by assigning to each member in the set a value representing its grade of membership in the fuzzy set, where the grade corresponds to the degree to which the member is similar or compatible with the concept represented by the fuzzy set. In mathematical set theory an element exhibits the property of whether it belongs to the set or not and so they are called as crisp sets. But when it comes to fuzzy sets many degrees of freedom are allowed. A membership function is associated with a fuzzy set so that the function maps every element in the set on to a value between 0 and 1. Fuzzy logic is a powerful representation to measure uncertainties. It provides a meaningful representation of vague concepts which are expressed in any natural language. Mathematical set theory has lead to symbolic logic and fuzzy set theory leads to fuzzy logic. Only two values 'true' and 'false' are allowed in symbolic logic but in fuzzy logic multi valued truth values such as true, absolutely true, fairly true, false, absolutely false, partly false are allowed. Fuzzy inference rules are computational procedures used in evaluating linguistic descriptions and fuzzy rule based systems are a set of fuzzy if-then statements used in many applications. Fuzzy logic is applicable in the areas of control systems, pattern recognition applications and decision making.

2.3 Fuzzy neural networks

A Fuzzy Neural Network^[5] is a structure $FNN(U, W, X, Y, Z)$ with the following specification. U is a non empty set of fuzzy neurons and auxiliary units.

- a. The structure and parameters of the fuzzy neural networks are described by the weight matrix W given by Cartesian product
 $U \times U \rightarrow D_w$, where D_w is the domain of weights.
- b. The vector for fuzzy inputs $X \in D_x$ describes the input for the fuzzy neural network.
- c. The vector for fuzzy outputs $Y \in D_y$ describes the output for the fuzzy neural network.
- d. The learning algorithm L describes the mechanism for learning and adaptation to the new information.

2.4 Fuzzy Inference System

Fuzzy inference system maps the given input to output using fuzzy logic. The mapping provides a basis from which decisions can be made and patterns can be matched. The process of fuzzy inference involves membership functions, logical operations and a set of If-Then rules. Fuzzy inference engine can be implemented using either Mamdani type or Sugeno type, where both the systems vary in the way outputs are determined.

Fuzzy inference systems are popular applications of fuzzy logic and fuzzy set theory. They are mainly used in classification tasks, offline process simulation and diagnosis, online decision support tools and process control. Strength of fuzzy inference system lies on the linguistic concepts and the nonlinear mappings between inputs and outputs. These two characteristics are used in designing two types of fuzzy inference systems. The first type of fuzzy inference system focuses on the ability of fuzzy logic to model natural language problems^[2]. The second type focuses on fuzzy rules built from expert knowledge and are called as fuzzy expert systems. Expert systems are used to solve problems using simulation purposes. These expert systems rely on boolean logic and are not suitable in managing the progressiveness in the underlying process phenomena. Fuzzy deals with the limitations of human knowledge and particularly the difficulties in formalizing interactions in complex processes.

Several fuzzy inference systems have been described by different researchers (Zadeh, L.A., 1965; Mamdani, E.H., 1974; Takagi, T. and M. Sugeno, 1985; Sugeno, M. and G.T. Kang, 1988; Sugeno, M. and K. Tanaka, 1991 ;). The most commonly-used systems are the Mamdani-type and Takagi–Sugeno type, also known as Takagi–Sugeno–Kang type. In the case of a Mamdani-type fuzzy inference system, both premise (if) and consequent (then) parts of a fuzzy if-then rule are fuzzy propositions. In the case of a Takagi–Sugeno-type fuzzy inference system where the premise part of a fuzzy rule is a fuzzy proposition, the consequent part is a mathematical function, usually a zero- or first-degree polynomial function

2.5 Overview of Fuzzy Inference Process

Fuzzy inference process contains different stages like fuzzification of the input variables, application of fuzzy operators in the antecedent, implication from the antecedent to the consequent, aggregation of the consequents across the rules, and defuzzification.

2.5.1 Fuzzify Inputs

The first step is to take the inputs and determine the degree to which they belong to each of the appropriate fuzzy sets via membership functions.

2.5.2 Apply Fuzzy Operator

After the inputs are fuzzified, the degree to which each part of the antecedent is satisfied for each rule is known. If the antecedent of a given rule has more than one part, the fuzzy operator is applied to obtain one number that represents the result of the antecedent for that rule.

2.5.3 Apply Implication Method

Before applying the implication method, the rule's weight will be determined. Every rule has a *weight* (a number between 0 and 1), which is applied to the number given by the antecedent.

After proper weighting has been assigned to each rule, the implication method is implemented. A consequent is a fuzzy set represented by a membership function, which weights appropriately the linguistic characteristics that are attributed to it.

2.5.4 Aggregate All Outputs

Because decisions are based on the testing of all of the rules in a FIS, the rules must be combined in some manner in order to make a decision. Aggregation is the process by which the fuzzy sets that represent the outputs of each rule are combined into a single fuzzy set.

2.5.5 Defuzzify

The input for the defuzzification process is a fuzzy set (the aggregate output fuzzy set) and the output is a single number.

The most commonly used fuzzy inference technique is the so called Mamdani method. In 1975 professor Ebrahim Mamdani of London University built one of the first fuzzy systems to control a steam engine and boiler combination. Mamdani fuzzy inference has 4 steps.

1. Fuzzification of the input variables.
2. Rule evaluation.
3. Aggregation of the rule outputs.
4. Defuzzification.

Another method that exists is the Sugeno fuzzy inference method. Michio Sugeno suggested to use a single spike, a singleton, as a membership function of the rule consequent. A singleton more precisely a fuzzy singleton, is a fuzzy set with a membership function that is unity at a single particular point on the universe of discourse.

2.6 Neuro Fuzzy Modeling Using ANFIS

Neuro fuzzy system is a combination of two different systems and is called as a hybrid system that contains the advantage of both neural networks and fuzzy system. It has a combination of advantages like robustness, learning, training and parallelism that are present in artificial neural networks and uncertainty present in fuzzy logic. Hybrid neuro fuzzy system, ANFIS (Adaptive neuro fuzzy inference system) is a multilayered feed forward artificial neural network consisting of nodes, directional links which nodes are connected. Adaptive means that output of some nodes depend on the input. ANFIS uses either an available fuzzy inference system to train data pairs or constructs a fuzzy inference system whose membership functions are tuned using a learning algorithm. An adaptive network is generated to incorporate the capability of learning from the input/output data sets in fuzzy inference system. Learning rule specifies how the weights should be adjusted to minimize the prescribed error measure. These changes applied changes the shape of membership function associated with fuzzy inference system. Fuzzy inference model is represented in figure 1.

2.6.1 Fuzzy inference system has 6 functional blocks.

- 1) Fuzzifier: Fuzzifier converts real numbers of inputs to fuzzy sets.
- 2) Database: Database contains the membership function of fuzzy sets

3) Membership function: Membership functions provides flexibility to fuzzy sets in modeling

4) Rule base: Rule base contains a set of statements in the form of 'If and Then' statements.

5) Inference engine: Inference engine performs inference operations on the rules to infer output by a fuzzy reasoning method.

6) Defuzzifier: Defuzzifier converts the fuzzy output into a non fuzzy output real number domain

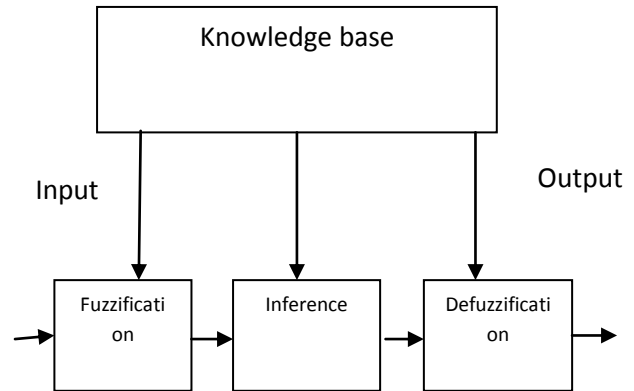


Fig.1. Fuzzy Inference system.

2.7 Modeling Process in ANFIS:

ANFIS process has two important phases namely system identification, determining the network structure and learning algorithm

1. System Identification: Identification of input and output variables
2. Determining the network structure: once the variables are identified the six layered network is constructed.

Layer 1: Input layer

Each node in input layer represents an input variable. This layer simply transmits these input variables to fuzzification layer.

Layer 2: Fuzzification layer

This layer describes membership function of each input fuzzy set, where membership functions are used to characterize fuzziness in fuzzy sets.

Layer 3: Inference layer

Each node in this layer is a fixed node that represents the 'if part' of a fuzzy rule. This layer aggregates membership grades using any fuzzy intersection operation, which can perform fuzzy AND operation. The fuzzy intersection operators are commonly referred to as T-norm operators.

Layer 4: Normalization Layer:

The i^{th} node of this layer is also a fixed node and calculates ratio of i^{th} rule's firing strength in inference layer to sum of the entire rule's firing strength.

Layer 5: Output layer:

This layer represents the THEN part of the fuzzy rule. The operation performed by the nodes in this layer is to generate

qualified consequent of each rule depending on firing strength. Every node in this layer is an adaptive node. Output of the node computed as

$$O_i = w_i f_i$$

where w_i is the normalized firing strength from layer I, f_i is the linear function of the input

Layer 6. Defuzzification layer:

This layer aggregates qualified consequents to produce a crisp output. The single node in this layer is a fixed node. It computes weighted average of output signals of output layer as

$$O = \sum O_i = \sum w_i f_i / \sum w_i$$

3. Learning algorithm and parameter tuning.

ANFIS model represented in figure 2, fine tunes parameters of membership function using either back propagation learning or hybrid learning rule. Back Propagation algorithm is an error based supervised learning algorithm that was gradient descent method to update parameters. New output is compared with desired output values.

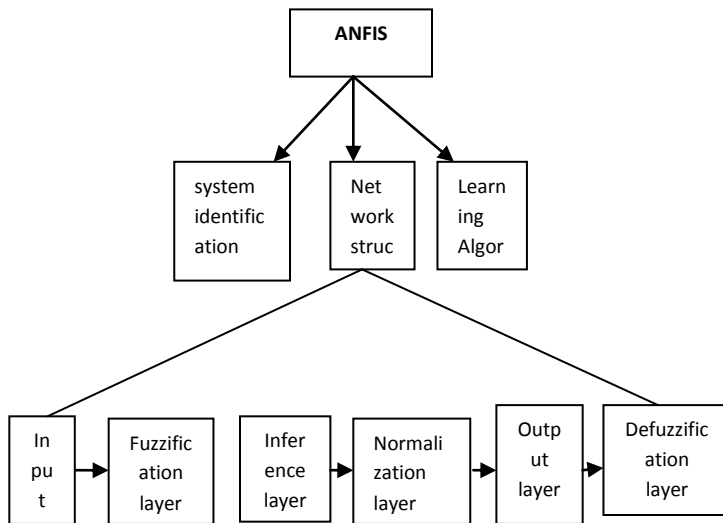


Fig.2 ANFIS Model

The advantage of using ANFIS is a hybrid learning rule for estimation of the premise and consequent parameters, in this process ANFIS keeps the premise parameters fixed and estimates them in a forward pass and then in a backward pass by keeping fixed the consequent parameters the process would be continued. In the forward pass the input is forwarded and propagated and then by applying the least square method the error would be calculated and forwarded to the next layer. In the second pass the error would be backward to and the premise parameters are updated by a gradient method. The advantages of Fuzzy logic for grade estimation are clear because it is a powerful tool that is flexible and in lack of data with its ability which is if-then rules would be able to solve the problems. One of the biggest problem in fuzzy logic application is the shape and location of the membership function used for each fuzzy variable which can be solved by trail and error method. In contrast numerical computation and learning are the advantages

of neural network; however it is not easy to obtain the optimal structure using a number of hidden layers and number of neurons in each hidden layer. Both fuzzy logic and neural networks have their advantages therefore it is a good idea to combine their ability and make a strong tool and also a tool which improves their weaknesses. Jang combined both fuzzy logic and neural networks to produce a powerful processing tool named Neuro-Fuzzy System which is a powerful tool that have both advantages and the most common one is ANFIS.

3. ENGINEERING EXAMPLE

3.1 Problem description and scope

As an example to solve uncertain problems, a rather simple but typical problem of engineering is considered. It consists of developing a control system which controls the personal computer in spite of power failures.

3.2 Fuzzy Inference System to control power failures in Personal Computers

Electronically controlled equipment is frequently prone to failures due to the disturbances caused in electrical power^[4]. They cause harm to industries, commercial and residential people. Blackout, brownout, sag, spike and swing are some examples of disturbances related to power supply problems^[8]. The disturbance caused by power may be accidental or incidental. Some problems may be due to voltage fluctuations, lightning, thunderstorms, poor wiring, grounding problems, load variations etc. To analyze the voltage fluctuations in a personal computer it is important to consider the components that can be affected due to power failures. These real uncertainties make hard the decision task of forecasting the success or failure of computer based loads against possible power quality related problems. There for fuzzy logic is the appropriate solution to represent load sensitivity to voltage fluctuations. Most of the power problems are due to the susceptibility of current loads with present power systems normal operations.

3.3 Problems in Power Supply

Power supply in a system is failing needs troubleshooting. Troubleshooting the power supply basically means isolating the supply as the cause of problems within a system and, if necessary, replacing it. The following is a list of PC problems that often are related to the power supply:

- Any power-on or system start up failures or lockups
- Spontaneous rebooting or intermittent lockups during normal operation
- Intermittent parity check or other memory-type errors
- Hard disk and fan simultaneously failing to spin (no +12V)
- Overheating due to fan failure
- Small blackout and brownouts that cause the system to reset, blackout is presented in figure 3.
- Electric shocks felt on the system case or connectors
- Slight static discharges that disrupt system operation
- Erratic recognition of bus-powered USB peripherals

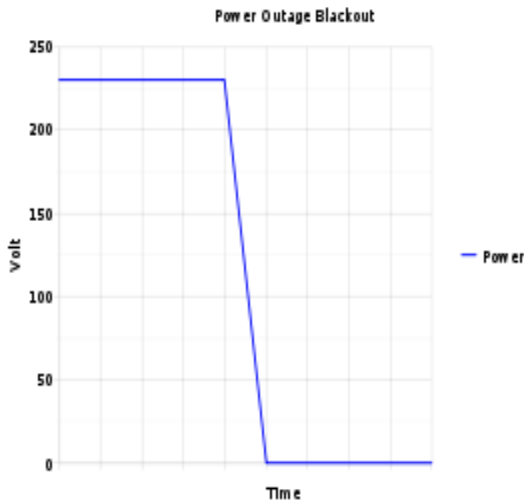


Fig 3. Blackout

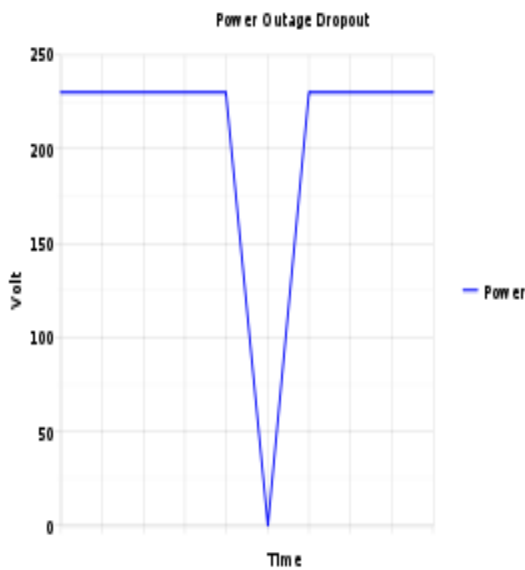


Fig 4. Dropout.

The following are some symptoms point right to the power supply as a possible cause of failure

- System that is completely dead
- Smoke
- Blown circuit breakers

Power supplies should stay within operating specifications and continue to run a system even if any of these power line disturbances occur:

- Voltage drop to 80V for up to 2 seconds

- Voltage drop to 70V for up to .5 seconds, dropout is presented in figure 4.
- Voltage surge of up to 143V for up to 1 second

3.4 Computer Power Supply Failure Symptoms

Power supplies are essential hardware components that draw energy from an electrical outlet and convert it into electricity at the proper voltages to power a computer's motherboard and other devices. Power supplies can become overstressed if the system demands more power than the supply is able to put out, which can result in power supply failure. Fortunately, there are a number of telltale signs that can help identify power supply failure.

A. Crashes

The first tip-off that your power supply is failing is likely to be unexplained computer crashes or restarts. When the motherboard, processor and RAM receive too little power, they experience errors that lead to crashing. The main problem with diagnosing power supply failure as an issue after crashes is that many other computer errors can lead to crashes. Viruses, processor overheating, RAM issues and software driver problems can all result in crashes.

B. Heat

Another important symptom of computer power supply failure is excessive heat coming from the power supply. Power supplies are usually warm to the touch when operating normally but won't be so hot that they're uncomfortable or burn the skin. If your computer crashes, touch the power supply or the part of the case where the power supply is located during operating to see how hot it feels. If it's very hot, there's a good chance the crash was caused by a power supply failure. Another issue with heat is that as the power supply fails, internal fans run slower. This raises the CPU and RAM temperature, which can also lead to crashing.

C. Noise and Smell

Noise and smell can indicate power supply failure. When a power supply is severely overstressed, it may become so hot that components inside it burn. If you smell smoke or burning coming from your computer, there's a good chance the power supply is burning out. Noise can also be a key indicator, as power supplies contain cooling fans that often become noisier or groan when the device is failing. Cooling fans within the computer case may also be noisier when the power supply is failing.

D. Start-up Failure

If a power supply has suffered total failure, the computer will not be able to start up. In addition, any peripheral devices will not receive power, so no hard drives or fans will spin up. Because motherboard issues can also cause start up failure, it's useful to open a computer and check whether any lights on the motherboard turn on when you attempt to start up. If the motherboard isn't receiving power, chances are the power supply has died.

E. Over heating

Power supplies should be replaced before they reach failure to protect the system's other components from possible damage

due to crashes, lack of power and overheating. Most manufactured computer systems come with a power supply adequate to run the system's basic devices. But when upgrades are added such as new video cards, RAM or extra hard drives, the energy demands may exceed the output of a base line power supply.

4. EXPERIMENTAL RESULTS

ANFIS has been applied to Pc power failures. Two input variables and one output variables are used to develop the neural network. Input variables are current and voltage. Output variable is resistance. All these variables are used to control the power failures in personal control. Five triangular membership functions are used to train the data. 100 data sets are used in ANFIS application, out of which nearly twenty five data sets are used in learning and seventy one data sets are used in training for 200 epochs. The error after 200 epochs has been calculated as 0.00097045. For the two input variables and one output variables 25 rules are defined in the rule editor to map input variables to a single output variable. Plot for the trained data set is represented in figure 5. Neural network for the two input variable, one output variable and 25 rules is represented in figure 6. ANFIS information has been represented in figure 7. It includes the number of linear and non linear parameters used, number of epochs and the final error value. Rule editor is represented in figure 8, rule viewer in figure 9, surface viewer in figure 10. The training error has been represented in figure 11.

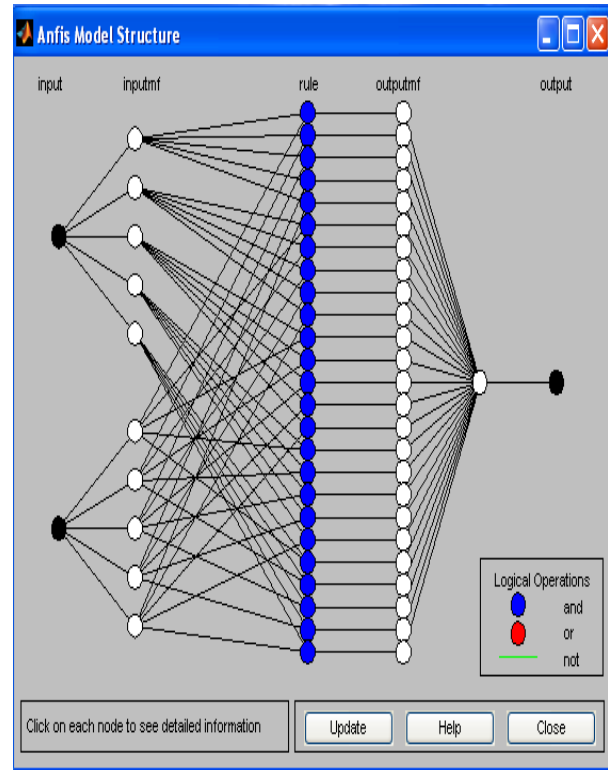


Fig.6 The neural network

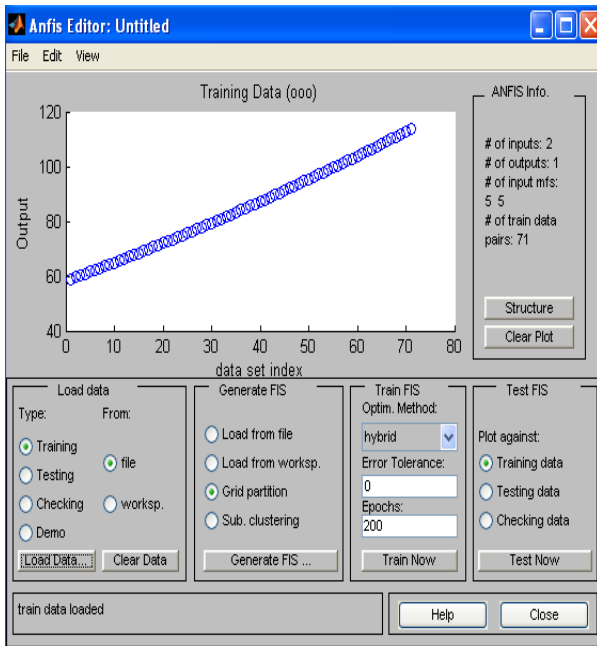


Fig.5 The plot for the trained data set.

The neural network for the neuro-fuzzy model is as follows.

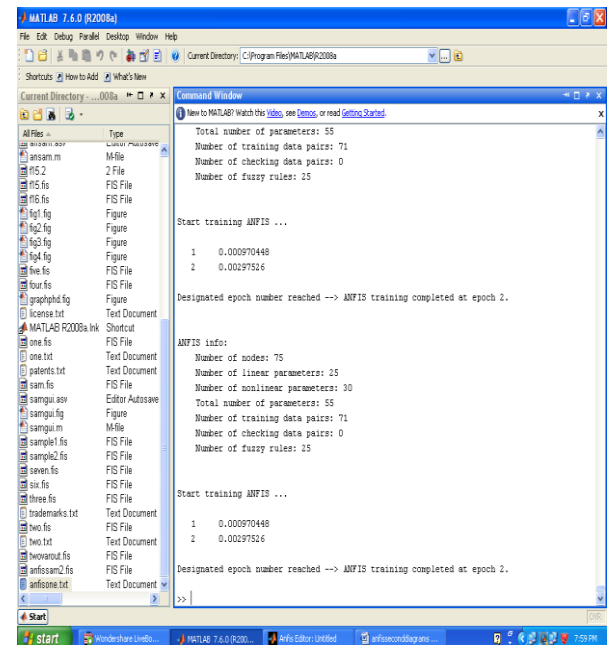


Fig.7 ANFIS information.

The Rule Editor for the neuro-fuzzy model is

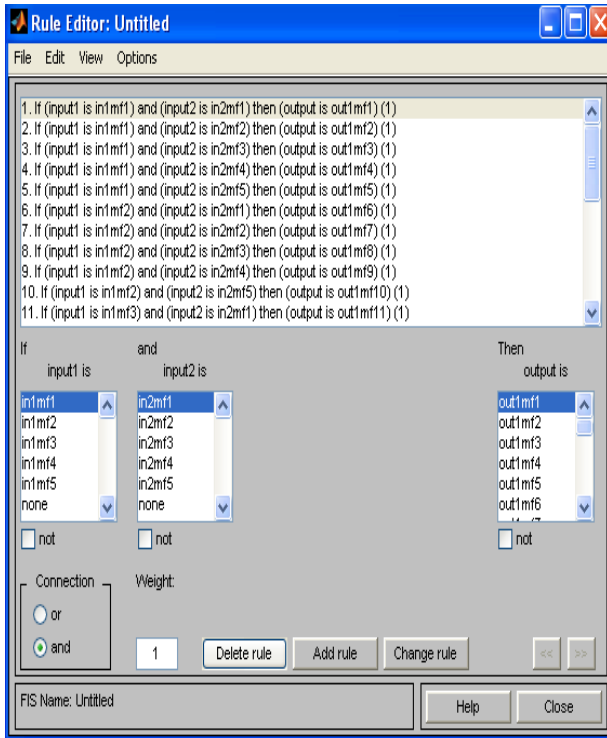


Fig 8. The Rule Editor

The surface viewer is as follows.

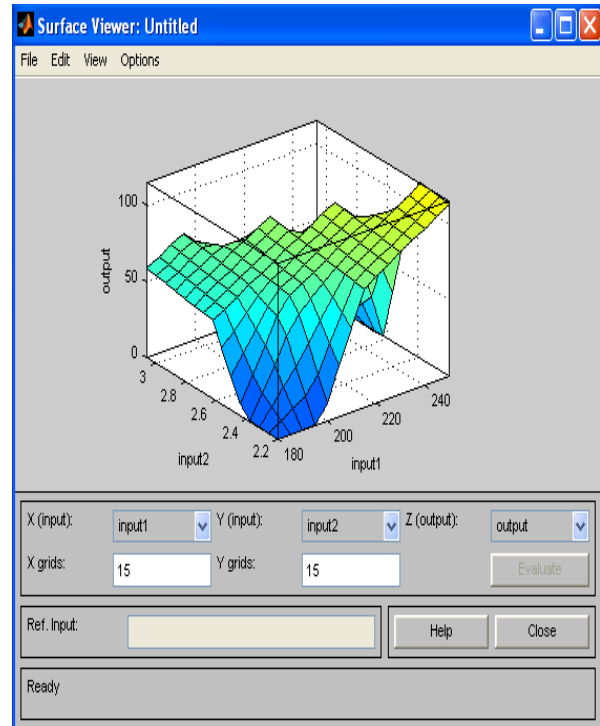


Fig.10 The Surface Viewer

The rule viewer for the neuro-fuzzy model is as follows.

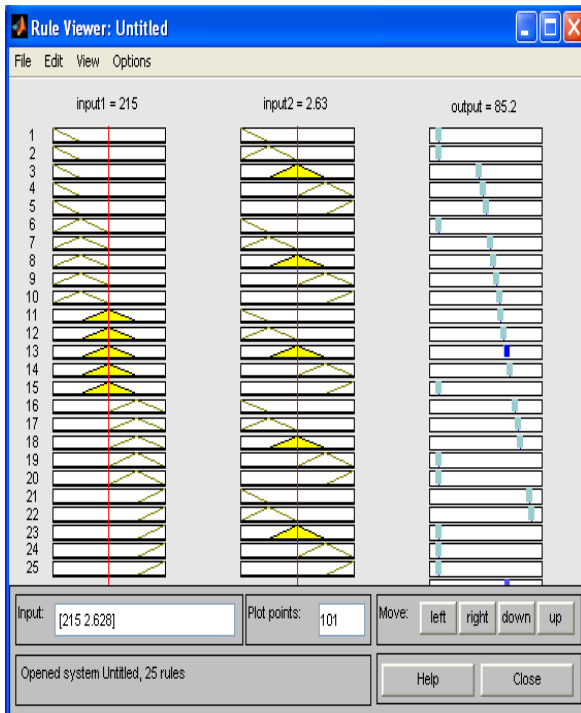


Fig.9 The Rule Viewer

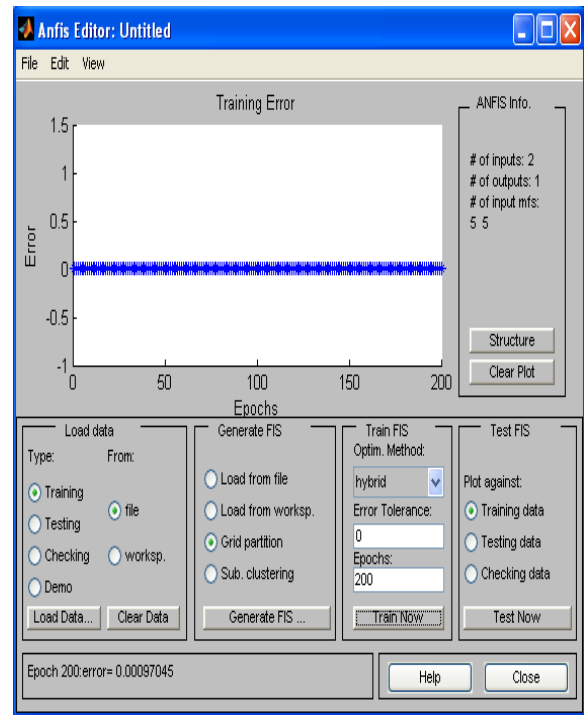


Fig.11 The average error for the trained data set is 0.0009

5. CONCLUSION

This paper has presented an application of fuzzy logic in power failures. The uncertainty in power failures was formulated by fuzzy logic theory through membership functions, which usually do not use strict boundaries, and such a feature was found suitable in dealing with the vagueness associated with the power failure system. The results for diagnosing electrical disturbances giving answers to the need of the manufacturers, utilities and electric energy customers

6. REFERENCES

- [1] Klir,G.J. and Yuan,B.(2000).*Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Prentice-Hall, India
- [2] Kukolj,D.(2000).Design of adaptive Takagi-Sugeno-Kang Fuzzy models.*Appl.Soft.Comput.*,2, 89-103
- [3] Rutkowaska,D.(2000).*Neuro_fuzzy Architectures and Hybrid Learning*. Physica-Verlag,U.S.A.
- [4] Wang, Z.Y., “Artificial Intelligence Applications in the diagnosis of Power Transformer Incipient Faults”, Ph.D Dissertation submitted to Virginia Polytechnic Institute and State University, August 2000.
- [5] W. Pedrycz and F. Gomide, *Fuzzy Systems Engineering Towards Human-Centric Computing*, John Wiley & Sons, New Jersey, 2007
- [6] J.S.R.Jang, C.T. Sun, E. Mizutani, *Neuro-Fuzzy and Soft Computing*, Prentice Hall, New jersey, 1997.
- [7] D. Nauck, F. Klawonn, R. Kruse, *Foundations of Neuro-Fuzzy Systems*, Wiley, England, 1997.
- [8] Yong Liu, Chanan Singh, Evaluation of the Failure Rates of transmission Lines During Hurricanes Using a Neuro – Fuzzy System, submitted to the 11th International IEEEConference PMAPS, 2010, Pages 569-574
- [9] File : PowerOutageBlackout.svg, available online at <http://en.Wikipedia.org/wiki/File:PowerOutageBlackout.svg>
- [10] File : Dropout.svg, available online at <http://en.Wikipedia.org/wiki/File:Dropout.svg>
- [11] Rathinam, A.; Padmini, S.; Composite Counter propagation Neural Networks for Solving Power Flow Problem, Conference on Computational Intelligence and Multimedia Applications. 2007, Volume 1, Digital Object Identifier: 10.1109/ICCIMA.2007.348, Pages 212-216
- [12] Rathinam, A.; Padmini, S.; Ravikumar, V.; Application of supervised learning artificial neural networks [CPNN, BPNN] for solving power flow problem. Information and Communication Technology in Electrical Sciences. 2007, ICTES, IET_UK International Conference, Pages 156-160.