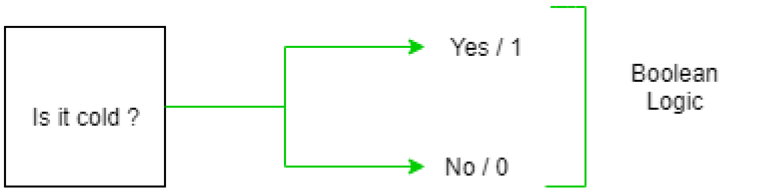
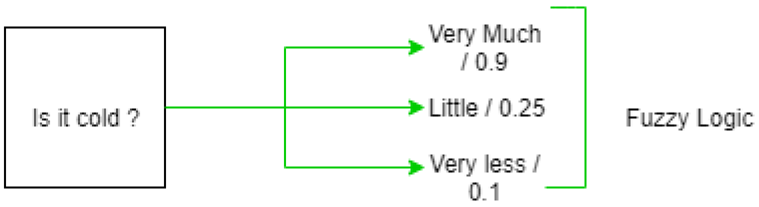
UNIT- 4

FUZZY LOGIC

Introduction of fuzzy logic

The term **fuzzy** refers to things which are not clear or are vague. In the real world many times we encounter a situation when we can’t determine whether the state is true or false, their fuzzy logic provides a very valuable flexibility for reasoning. In this way, we can consider the inaccuracies and uncertainties of any situation.
In Boolean system truth value, 1.0 represents absolute truth value and 0.0 represents absolute false value. But in the fuzzy system, there is no logic for absolute truth and absolute false value. But in fuzzy logic, there is intermediate value too present which is partially true and partially false.





Example of fuzzy logic:

```
if ((speed >= 0.0)&&(speed < 0.25)) {
        //speed is slowest
                }
                else if ((speed >= 0.25)&&(speed < 0.5))
                {
                        //  speed is slow
                }
                else if ((speed >= 0.5)&&(speed < 0.75))
                {
                        //  speed is fast
                }
                else // speed >= 0.75 && speed < 1.0
                {
                        //  speed is fastest
                }
```

Architecture

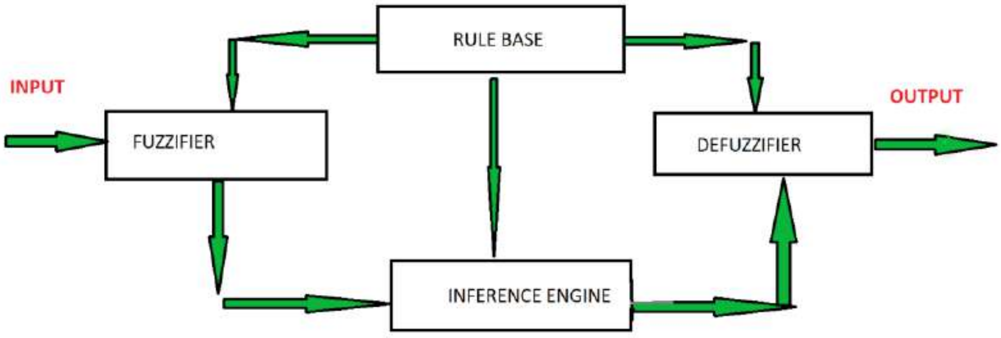
Its Architecture contains four parts:

- RULE BASE: It contains the set of rules and the IF-THEN conditions provided by the experts to govern the decision making system, on the basis of linguistic information. Recent developments in fuzzy theory offer several effective methods for the design and tuning of fuzzy controllers. Most of these developments reduce the number of fuzzy rules.

- FUZZIFICATION: It is used to convert inputs i.e. crisp numbers into fuzzy sets. Crisp inputs are basically the exact inputs measured by sensors and passed into the control system for processing, such as temperature, pressure, rpm's, etc.

- INFERENCE ENGINE: It determines the matching degree of the current fuzzy input with respect to each rule and decides which rules are to be fired according to the input field. Next, the fired rules are combined to form the control actions.

- DEFUZZIFICATION: It is used to convert the fuzzy sets obtained by inference engine into a crisp value. There are several defuzzification methods available and the best suited one is used with a specific expert system to reduce the error.



FUZZY LOGIC ARCHITECTURE

Membership function

Definition: A graph that defines how each point in the input space is mapped to membership value between 0 and 1. Input space is often referred as the universe of discourse or universal set (u), which contain all the possible elements of concern in each particular application. There are largely three types of fuzzifiers:

- Singleton fuzzifier
- Gaussian fuzzifier
- Trapezoidal or triangular fuzzifier

What is Fuzzy Control?

- It is a technique to embody human-like thinkings into a control system.
- It may not be designed to give accurate reasoning but it is designed to give acceptable reasoning.
- It can emulate human deductive thinking, that is, the process people use to infer conclusions from what they know.
- Any uncertainties can be easily dealt with the help of fuzzy logic.

Advantages of Fuzzy Logic System

- This system can work with any type of inputs whether it is imprecise, distorted or noisy input information.
- The construction of Fuzzy Logic Systems is easy and understandable.
- Fuzzy logic comes with mathematical concepts of set theory and the reasoning of that is quite simple.
- It provides a very efficient solution to complex problems in all fields of life as it resembles human reasoning and decision making.
- The algorithms can be described with little data, so little memory is required.

Disadvantages of Fuzzy Logic Systems

- Many researchers proposed different ways to solve a given problem through fuzzy logic which lead to ambiguity. There is no systematic approach to solve a given problem through fuzzy logic.
- Proof of its characteristics is difficult or impossible in most cases because every time we do not get mathematical description of our approach.
- As fuzzy logic works on precise as well as imprecise data so most of the time accuracy is compromised.

Application

- It is used in the aerospace field for altitude control of spacecraft and satellite.
- It has used in the automotive system for speed control, traffic control.
- It is used for decision making support systems and personal evaluation in the large company business.
- It has application in chemical industry for controlling the pH, drying, chemical distillation process.
- Fuzzy logic are used in Natural language processing and various intensive applications in Artificial Intelligence.
- Fuzzy logic are extensively used in modern control systems such as expert systems.
- Fuzzy Logic is used with Neural Networks as it mimics how a person would make decisions, only much faster. It is done by Aggregation of data and changing into more meaningful data by forming partial truths as Fuzzy sets.

Classical and fuzzy sets:

Set: A set is defined as a collection of objects, which share certain characteristics.

Classical set:

1. Classical set is a collection of **distinct** objects. For example, a set of students passing grades.
2. Each individual entity in a set is called a **member** or an **element** of the set.
3. The classical set is defined in such a way that the universe of discourse is spitted into two groups' **members** and **non-members**. Hence, in case classical sets, **no partial membership exists**.

4. Let A is a given set. The membership function can be used to define a set A is given by:

$$\mu A(x)=\{\ 1 \text{ if } x \in A$$

$$0 \text{ if } x \notin A\ \}$$

5. **Operations on classical sets**: For two sets A and B and Universe X:

- **Union**:

  $$A \cup B = \{x \mid x \in A \text{ or } x \in B\}$$

  This operation is also called **logical OR**.
- **Intersection**:

  $$A \cap B = \{x \mid x \in A \text{ and } x \in B\}$$

  This operation is also called **logical AND**.
- **Complement**:

  $$A' = \{x \mid x \notin A,\ x \in X\}$$

- **Difference**:

  $$A \mid B = \{x \mid x \in A \text{ and } x \notin B\}$$

6. **Properties of classical sets**: For two sets A and B and Universe X:

- **Commutativity**:

  $$A \cup B = B \cup A$$

  $$A \cap B = B \cap A$$

- **Associativity**:

  $$A \cup (B \cup C) = (A \cup B) \cup C$$

  $$A \cap (B \cap C) = (A \cap B) \cap C$$

- **Distributivity**:

  $$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$$

  $$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$$

- **Idempotency**:

$$A \cup A = A$$

$$A \cap A = A$$

- **Identity**:

$$A \cup \emptyset = A$$

$$A \cap X = A$$

$$A \cap \emptyset = \emptyset$$

$$A \cup X = X$$

- **Transitivity**:

If $A \subseteq B$ and $B \subseteq C$, then $A \subseteq C$

<u>Fuzzy set</u>:

1. **Fuzzy set** is a set having **degrees of membership** between 1 and 0. Fuzzy sets are represented with tilde character (~). For example, Number of cars following traffic signals at a particular time out of all cars present will have membership value between [0,1].
2. Partial membership exists when member of one fuzzy set can also be a part of other fuzzy sets in the same universe.
3. The degree of membership or truth is not same as probability, fuzzy truth represents membership in vaguely defined sets.
4. A fuzzy set A~ in the universe of discourse, U, can be defined as a set of ordered pairs and it is given by

$$\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) | x \in X\}$$

5. When the universe of discourse, U, is **discrete and finite**, fuzzy set A~ is given by

$$\tilde{A} = \sum_{i=1}^{n} \frac{\mu_{\tilde{A}}(x_i)}{x_i} = \frac{\mu_{\tilde{A}}(x_1)}{x_1} + \frac{\mu_{\tilde{A}}(x_2)}{x_2} + \ldots +$$

$$\tilde{A} = \int \frac{\mu_{\tilde{A}}(x)}{x}$$

Where “n” is a finite value.

6. Fuzzy sets also satisfy every property of classical sets.

7. Some other useful operations on Fuzzy set:

- **Algebraic sum**:

μA+B(x)= μA(x) + μB(x) - μA(x). μB(x)

- **Algebraic product**:

μA.B(x)= μA(x). μB(x)

- **Bounded sum**:

μA⊕B(x)= min{1, μA(x) + μB(x)}

- **Bounded difference**:

μA⊙B(c)= max{0, μA(x) - μB(x)}

Operation of Fuzzy Sets: Union, Intersection and Complement

- Given two fuzzy sets $\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) | x \in X\}$ and $\tilde{B} = \{(x, \mu_{\tilde{B}}(x)) | x \in X\}$ over the same universe of discourse X, we can define operations of union, intersection and complement. We define:

- **The *union* of the fuzzy sets** $\tilde{A}$ si $\tilde{B}$ as the fuzzy set $\tilde{C} = \tilde{A} \cup \tilde{B}$, given by $\tilde{C} = \{(x, \mu_{\tilde{C}}(x)) | x \in X\}$, where

$$\mu_{\tilde{C}}(x) = \max(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x))$$

- **The *intersection* of the fuzzy sets** $\tilde{A}$ and $\tilde{B}$ as the fuzzy set $\tilde{D} = \tilde{A} \cap \tilde{B}$, given by $\tilde{D} = \{(x, \mu_{\tilde{D}}(x)) | x \in X\}$, where

$$\mu_{\tilde{D}}(x) = \min(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x))$$

- **The *complement* of $\tilde{A}$** in X as the fuzzy set $\tilde{E} = C_{\tilde{A}}X$ given by $\tilde{E} = \{(x, \mu_{\tilde{E}}(x)) | x \in X\}$, where

$$\mu_{\tilde{E}}(x) = 1 - \mu_{\tilde{A}}(x)$$

Operation of Fuzzy Logic: Inclusion, Equality

- **Inclusion** of fuzzy sets: given two fuzzy sets $\tilde{A}$ and $\tilde{B}$ included in X , the inclusion $\tilde{A} \subseteq \tilde{B}$ takes place iff $\mu_{\tilde{A}}(x) \leq \mu_{\tilde{B}}(x)$, $(\forall)x \in X$
- **Equality of two fuzzy sets:** two fuzzy sets $\tilde{A}$ and $\tilde{B}$ included in X are equals iff $\mu_{\tilde{A}}(x) = \mu_{\tilde{B}}(x)$, $(\forall)x \in X$
- **Equivalently**, two fuzzy sets $\tilde{A}$ and $\tilde{B}$ included in X are equals iff $\tilde{A} \subseteq \tilde{B}$ and $\tilde{B} \subseteq \tilde{A}$

Extension of Operation with Fuzzy Sets:

Cartesian product: Given the fuzzy sets $A_1, \ldots, A_n$ in the universes $X_1, \ldots, X_n$, the Cartesian product of these fuzzy sets is a fuzzy set the product space $X_1 \times \ldots \times X_n$ with the membership function

$$\mu_{\tilde{A}_1 \times \ldots \tilde{A}_n}(x) = \min[\mu \tilde{A}_i(x_i)],$$

Where $x = (x_1 \ldots x_n)$, $x_i \in X_i$

Example of Cartesian product for n = 2

The m-th power of a fuzzy set A: in the universe of discourse X is the fuzzy set with the membership function.

$$\mu_{A^m}(x) = [\mu_A(x)]^m,$$

Where $x \in X$

Intersection and union operations with fuzzy sets are extended by the triangular, or t-norms (for intersection) and s-norms, named also t-co-norms (the union).

T-norms:

A t-norm is a two valued function t: $[0, 1] \times [0, 1] \rightarrow [0, 1]$ which satisfies the following properties:

1. $t(0, 0) = 0$,
   $t(1, \mu_A(x)) = t(\mu_A(x), 1) = \mu_A(x)$, $(\forall)\ x \in X$, i.e. $(\forall)\ \mu_A(x) \in [0, 1]$

2. **Monotony**: $t(\mu_A(x), \mu_B(x)) \leq t(\mu_C(x), \mu_D(x))$
   If $\mu_A(x) \leq \mu_C(x)$ and $\mu_B(x) \leq \mu_D(x)$

3. **Commutativity**: $t(\mu_A(x), \mu_B(x)) = t(\mu_B(x), \mu_A(x))$

4. **Associativity**: $t(\mu_A(x), t(\mu_B(x), \mu_C(x))) = t(t(\mu_A(x), \mu_B(x)), \mu_C(x))$

S –norms:

An s-norm (t-co norm) is a two valued function
s: $[0, 1] \times [0, 1] \rightarrow [0, 1]$ following properties:

1. $s(1, 1) = 1$,
   $s(0, \mu_A(x)) = s(\mu_A(x), 0) = \mu_A(x)$, $(\forall)\ x \in X$, i.e. $(\forall)\ \mu_A(x) \in [0, 1]$

2. **Monotony**: $s(\mu_A(x), \mu_B(x)) \le s(\mu_C(x), \mu_D(x))$ if $\mu_A(x) \le \mu_C(x)$ and $\mu_B(x) \le \mu_D(x)$

3. **Commutativity**: $s(\mu_A(x), \mu_B(x)) = s(\mu_B(x), \mu_A(x))$

4. **Associativity**: $s(\mu_A(x), s(\mu_B(x), \mu_C(x))) = s(s(\mu_A(x), \mu_B(x)), \mu_C(x))$

Fuzzy Rule Generation:

In fuzzy rule generation, training data are preclustered or postclustered. In preclustering, we cluster the training data in advance and generate a fuzzy rule for each cluster. In postclustering, we start from one fuzzy rule and generate fuzzy rules around the training data with large estimation errors or at the points where the training data gather until the error limit is within the specified limit.

In preclustering, there are two types to generate fuzzy rules. The first type is based on clustering of the input space. Namely, first we divide the output space and consider each divided interval as a class. Then we divide the training data into classes according to the output values of the training data, and extract fuzzy rules for each class according to one of the methods. The second type clusters the input and output spaces simultaneously. Namely, we combine the input values and output values of the training data and consider them as input data. Then we select the cluster centers, and define fuzzy rules for the cluster centers. Since approximation of the multi-dimensional output can be easily realized by an extension of that of the one-dimensional output, in the following we discuss function approximation with the m-dimensional input vector x and the one-dimensional output y.

Clustering of input and output:

Here, we discuss fuzzy rule extraction clustering input and output spaces together. Namely, we first combine the m-dimensional input training data and I-dimensional output training data into (m + I)-dimensional training data. Then, to these data we apply one of the clustering methods. We may choose cluster centers from the training data or we may determine the centers which differ from the training data. Let Ci (i = 1, ... , N) and Oi be the selected inputs and the associated outputs, respectively. Then the fuzzy rules become

Ri: If x is Ci then y = Oi, for i = 1, ... , N.

We set some appropriate membership function, such as the bell-shaped membership function, as the membership function around Ci.

FUZZY NUMBERS:

- Set **A :** ***R tends to [0,1]***
- To qualify as ***fuzzy number,*** a fuzzy set **A on R** must have following

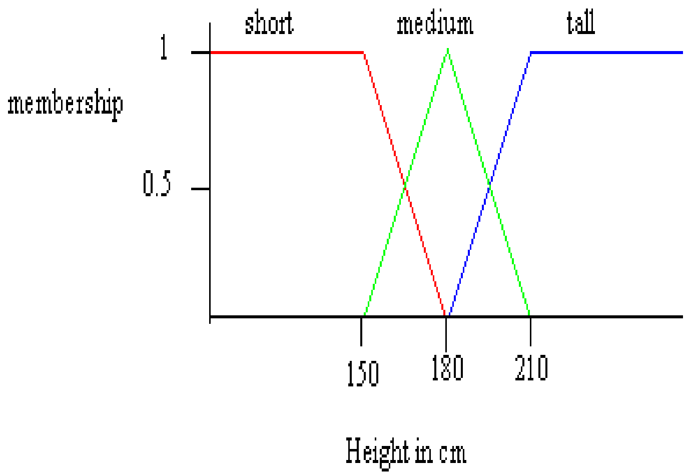
  Properties:

1. Set **A** must be Normal Fuzzy set;

2.α-cut of **A** must be closed interval for every α in (0,1];

3.Support and strong α-cut of **A** must be bounded.

- Every Fuzzy number is a convex fuzzy set, The Inverse is not necessarily true.



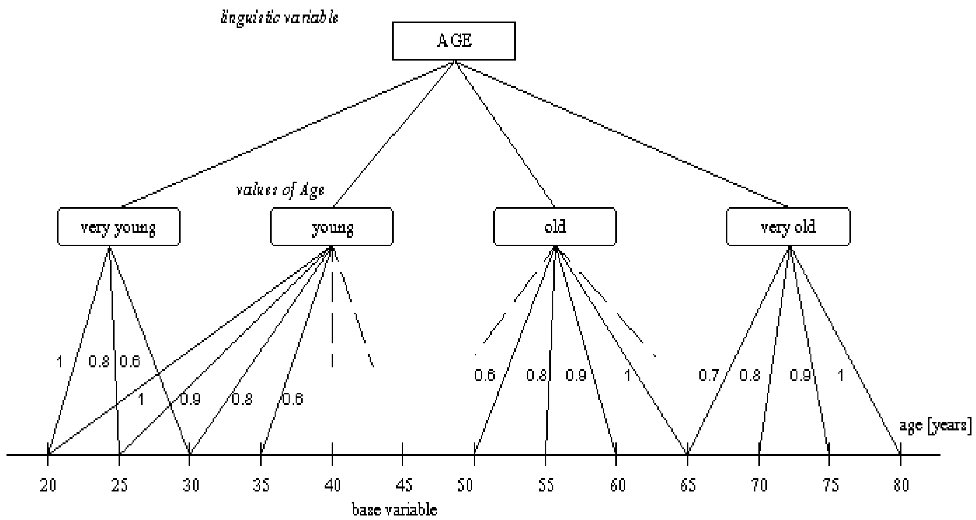
Linguistic Variables:

Linguistic variable is a variable whose values are words in a natural language. Example: speed is a linguistic variable which can take the values as slow, fast, very fast and so on.

Linguistic variables collect elements into similar groups where we can deal with less precisely and hence we can handle more complex systems.

It is a variable made up of number of words with associated degree of membership. It is a mathematical representation of semantic concepts that include more than one fuzzy set.

Linguistic variables is a variable of higher order than fuzzy variable.



Arithmetic Operations on Intervals:

– Each fuzzy set and thus each fuzzy number is uniquely be represented by its α-cuts;

– α-cut of **A** must be closed intervals of real numbers for all α in (0,1).

Using above properties we define ***arithmetic operations on fuzzy numbers in terms of arithmetic operations on their α cuts.***

Let * denotes any of the four arithmetic operations on closed intervals :

- [a, b]*[d, e]={f*g|a $\leq$ f $\leq$ b, d $\leq$ g $\leq$ e}

1.Addition

[a , b]+[d, e]=[a+d, b+e];

2.Subtraction

[a, b]-[d, e]=[a-e, b-d];

3.Multiplication

[a, b].[d, e]=[min(ad,ae,bd,be), max(ad,ae,bd,be)];

4.Division*

[a, b]/[d, e]=[a, b].[1/e, 1/d]

=[min(a/d,a/e,b/d,b/e), max(a/d,a/e,b/d,b/e)].

5.Inverse Interval*

[d,e] inverse =[min(1/d,1/e),max(1/d,1/e)]

* 0 $\notin$ [d,e] i.e excluding the case d=0 or e=0.

Examples:

1. [-1, 3] + [1, 5] = [-1+1, 3+5] = [0, 8]

2. [-1, 3] - [1, 5] = [-1-5, 3-1] = [-6, 2]

3. [-1,3].[1, 5] = [min(-1,-5,3,15),max(-1,-5,3,15)] =[-5,15]

4. [-1,3]/[1, 5] = [-1, 3].[1/1, 1/5]= [min(-1/1,-1/5,3/1,3/5),max(-1/1,-1/5,3/1,3/5)]

   = [-1/1, 3/1] = [-1, 3]

5. [-2, 7] inverse = [min (1/-2, 1/7), max (1/-2, 1/7)] = [-1/2, 1/7]

Properties:

1. Commutativity (+ , .) A*B = B*A
2. Associativity (+ , .) (A*B)*C= A*(B*C)
3. Identity A=0+A=A+0
4. Distributivity A.(B+C)= A.B+A.C
5. Inclusion monotonicity (all). If A $\subseteq$ E & B$\subseteq$ F then A*B $\subseteq$ E*F

6. 0, 1 are included in -,/ operations between same fuzzy intervals respectively. $0 \in A\text{-}A$ and $1 \in A/A$

Arithmetic Operations on Numbers:

Moving from intervals we can define arithmetic on fuzzy numbers based on principles of Interval Arithmetic

Let A and B denote fuzzy numbers and let * denote any of the four basic arithmetic operations. Then, we define a fuzzy set on R, A*B, by defining its alpha-cut as:

$$^{\alpha}(A * B) = {^{\alpha}A} * {^{\alpha}B}$$

$$\mu_{(A)}(x) = \begin{cases} 0, & x < 1 \\ x - 1, & 1 \le x < 2 \\ -\frac{1}{2}x + 2, & 2 \le x < 4 \\ 0, & x \ge 4 \end{cases}$$

$$\mu_{(B)}(y) = \begin{cases} 0, & y < 2 \\ \frac{1}{2}y - 1, & 2 \le y < 4 \\ -\frac{1}{2}y + 3, & 4 \le y < 6 \\ 0, & y \ge 6 \end{cases}$$

Lattice of Fuzzy Logic:

- As is well known, the set **R** of real numbers is linearly ordered. For every pair of real numbers, x and y, either x <=y or y <= x. The pair (**R**, <=) is a lattice, which can also be expressed in terms of two lattice operations,

$$\min(x, y) = \begin{cases} x & \text{if } x \le y \\ y & \text{if } y \le x, \end{cases}$$

$$\max(x, y) = \begin{cases} y & \text{if } x \le y \\ x & \text{if } y \le x \end{cases}$$

- To introduce a meaningful ordering of fuzzy numbers, we first extend the lattice operations min and max on real numbers, to corresponding operations on fuzzy numbers, MIN and MAX. For any two fuzzy numbers A and *B, we* define

$$\text{MIN}(A, B)(z) = \sup_{z=\min(x,y)} \min[A(x), B(y)],$$

$$\text{MAX}(A, B)(z) = \sup_{z=\max(x,y)} \min[A(x), B(y)]$$

For all 'Z ε R'.

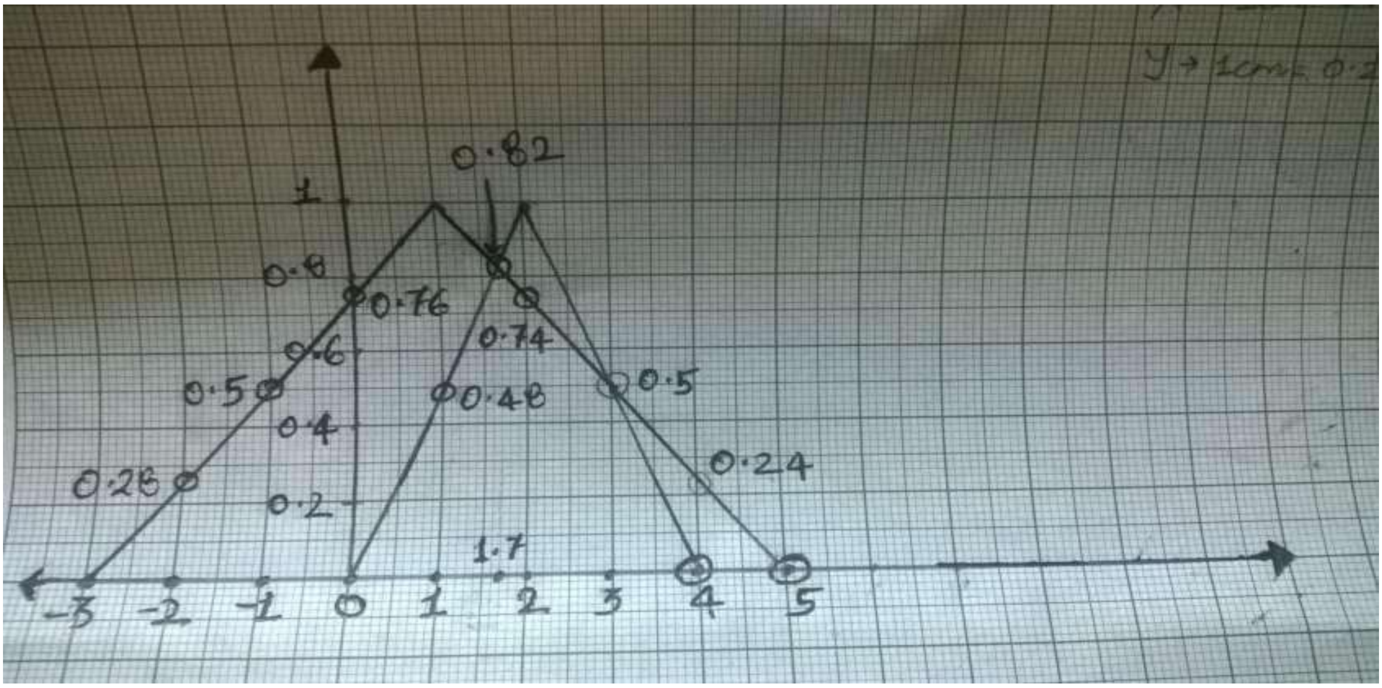


Figure: Question of lattice of fuzzy logic

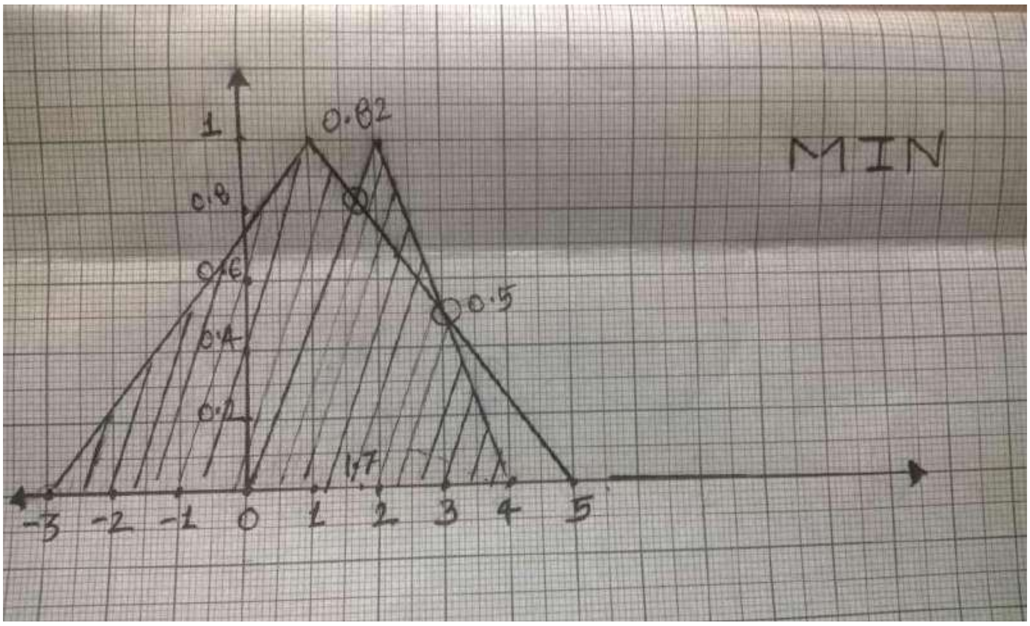


Fig: Min of lattice system

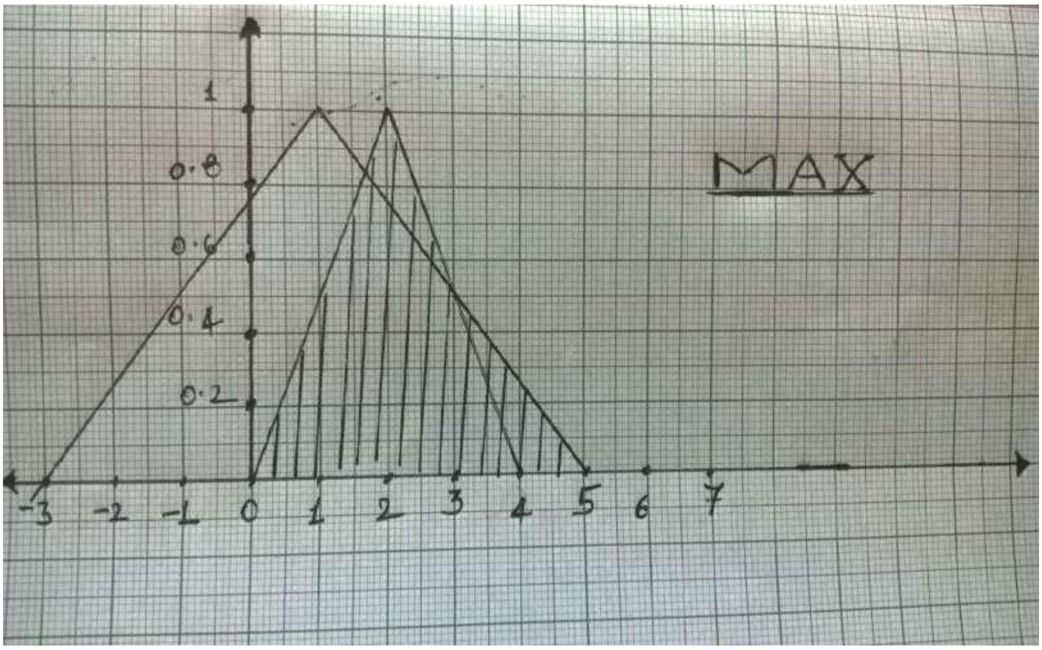


Fig: Max of lattice system

Fuzzy Equations:

These are equations in which coefficients and unknowns are fuzzy numbers, and formulas are constructed by operations of fuzzy arithmetic.

A+X =B and A.X= B, where A and B are the fuzzy numbers and X is an unknown fuzzy number for which either of the equations is to be satisfied.

Equation A + X= B

The difficulty of solving this fuzzy equation is caused by the fact that X=B-A is not the solution. To see this, let us consider two closed intervals, $A=[a_1,a_2]$ and $B=[b_1,b_2]$,which may be viewed as special fuzzy numbers.

$$B - A = [b_1 - a_2 , b_2 - a_1]$$

$$\begin{aligned} A + (B - A) &= [a_1, a_2] + [b_1 - a_2, b_2 - a_1] \\ &= [a_1 + b_1 - a_2, a_2 + b_2 - a_1] \\ &\neq [b_1, b_2] = B, \end{aligned}$$

- whenever $a_1 \neq a_2$. *Therefore, $X = B - A$ is not a solution of the equation.*

- Let $X = [x_1, x_2]$. Then, $[a_1 + x_1, a_2 + x_2] = [b_1, b_2]$ follows immediately from the equation. This results in two ordinary equations of real numbers,

$$a_1+x_1=b_1$$
$$a_2+x_2=b_2$$

- whose solution is $X_1 = b1 - a_1$ and $x_2 = b_2 - a_2$. Since X must be an interval, it is required that $x_1 <= x_2$. That is, the equation has a solution iff $b_1 - a_1 < b_2 - a_2$. If this inequality is satisfied, the solution is $X = [b_1 - a_1, b_2 - a_2]$.

- **Equation A • *X = B***
- Let us assume, for the sake of simplicity, that *A, B are fuzzy numbers on R^+. It is easy to* show that *X = B/A is not a solution of the equation. For each a ε(0,1], we obtain the* interval equation

$$^{\alpha}A \bullet {}^{\alpha}X = {}^{\alpha}B$$

- Our fuzzy equation can be solved by solving these interval equations for all *αε(0,1).*

  Let $^{\alpha}A =[^{\alpha}a_1, {}^{\alpha}a_2]$, $^{\alpha}B_1 = [^{\alpha}b_1, {}^{\alpha}b_2]$, and $^{\alpha}X = [^{\alpha}x_1\ {}^{\alpha}x_2]$. Then, the solution of the fuzzy equation exists iff:

(i) $^{\alpha}b_1/^{\alpha}a_1 \leq {}^{\alpha}b_2/^{\alpha}a_2$ for each $\alpha \in (0, 1]$, and

(ii) $\alpha \leq \beta$ implies $^{\alpha}b_1/^{\alpha}a_1 \leq {}^{\beta}b_1/^{\beta}a_1 \leq {}^{\beta}b_2/^{\beta}a_2 \leq {}^{\alpha}b_2/^{\alpha}a_2$.

If the solution exists, it has the form

$$X = \bigcup_{\alpha\in(0,1]} {}_{\alpha}X.$$

16.7 Applications

Fuzzy logic is well-suited for systems that require the ability to handle vague data and/or model imprecise reasoning procedures. Many commercial applications of fuzzy logic relate to "process control", which refers to the management of a mechanical or environmental process. Examples of fuzzy logic technology include:

- air conditioning: gradually slows down the cooling system as the room temperature approaches the desired setting.
- cruise control: determines ambient acceleration or deceleration and controls the countering application of gas and brake.
- ship boilers: monitors the temperature, pressure, and chemical content to ensure stability.
- video cameras: identifies when the subject of a video shot is moving and when motion is caused by the cameraman's vibrations.
- washing machines: optimizes the wash cycle by examining the load size, fabric mix, and quantity of detergent.

16.8 Introduction to Neuro Fuzzy Systems

During the last few years there has been a large and energetic upswing in research efforts aimed at synthesizing fuzzy logic with neural networks. Neural networks possess advantages in the areas of learning, classification, and optimization, whereas fuzzy logic has advantages in areas such as reasoning on a high (semantic or linguistic) level. The two technologies nicely complement each other.

Neural networks and fuzzy systems individually have reached a degree of maturity where they are applied to real world situations. Researchers often utilize these two technologies in series, using one as preprocessor or postprocessor for the other. Examples include the use of fuzzy inputs and outputs for neural networks, the use of neural networks to quantify the shape of a fuzzy membership function and so on.

At present, there is no generally accepted theory or methodology for the design of neural networks, and the process used is generally a trial-and-error approach based on the experience of the designer. The complexity of neural network design arises from the high dimensional, heterogeneous space that must be explored by the system. The primary features that are of concern in the design of neural networks are the structure of the network, the inputs to the network, and the specification of the learning algorithm parameters. All of these quantities are problem specific. While there are guidelines based on experience that can be very helpful in design, some mathematical based procedure would also be very helpful. Optimization of the design based on the use of Genetic Algorithms offers such a methodology.

There are other hybrid combinations of the different elements of soft computing that are also synergistic. Perhaps next in importance is the combination of genetic algorithms with neural networks and/or fuzzy systems. The ability to carry out near-global optimization on any problem for which an objective function can be defined is an incredibly powerful tool that can enhance the capabilities of any technology.

Fuzzy systems are made up of fuzzy sets, defined by their membership functions and fuzzy rules that determine the action of the fuzzy systems. Fuzzy systems can model general non-linear mappings in a manner similar to feed forward neural networks since it is a well-defined function mapping of real valued inputs to real valued outputs. Kosko has shown that fuzzy systems, like feed forward neural networks, are universal approximators in that they are capable of approximating general non-linear functions to any desired degree of accuracy.

Fuzzy neural networks aspiring to integrate neural learning with the knowledge representation capabilities of fuzzy systems have been actively investigated in recent years. The terms fuzzy-neural or neuro fuzzy networks very often in the literature refers to hybrid combination of fuzzy logic and neural tools – for example, giving fuzzy inputs to a crisp network and extracting fuzzy outputs as well.

Fuzzy systems have been successfully applied to many problems in decision making and incorporated in control technologies. Hybrid systems that fuse fuzzy control and neural networks (NN) have been propounded for utilizing numerical data. NN models which adopt learning methods using numerical data as well as expert knowledge represented by fuzzy IF–THEN rules have been discussed by Ishibuchi *et al.*

The neural methods and fuzzy systems are brought in both for the purpose of identifying (extracting) rules and membership functions and for adaptation of a fuzzy system (or linguistic description) to a changing physical system and its environment. The approach is known in the literature as neural network driven fuzzy reasoning. For both expert knowledge elicitation and adaptation, the underlying strategy is, in essence, to identify certain parameters of fuzzy systems and use neural networks to induce and/or adjust them.

Fuzzy neural approaches have found their way in a variety of engineering applications, including, but not limited to, consumer electronics, various aspects of control, diagnostics, industrial production lines, biotechnology, power generation, chemical processes, power electronics, communications and software resource management. It is expected that the applications of fuzzy-neural synergism will increasingly move toward computer applications as well, such as machine learning.

The term 'fuzzy neural network (FNN)' has existed more than a decade. However, the recent resurgence of interest in this area is motivated by the increasing recognition of the potential of fuzzy logic and neural networks are two of the most promising approaches for exploring the functioning of human brains. Many researchers are currently investigating ways and means of building fuzzy neural networks by incorporating the notion of fuzziness into a neural network framework.

Ambiguity is always present in any realistic process. This ambiguity may arise from the interpretation of the data inputs and in the rules used to describe the relationships between the informative attributes. Fuzzy logic provides an inference structure that enables the human reasoning capabilities to be applied to artificial knowledge-based systems. The efficient working the artificial knowledge-based systems depends upon algorithms, which are cumbersome to implement and require extensive computational time. On the other hand, the human brain, which performs approximate reasoning, employs simple information processing elements called neurons. The paradigm of artificial neural networks, developed to emulate some of the capabilities of the human brain, has demonstrated a great potential in terms of learning and adaptation for various low-level computations and embodies salient features such as learning, fault-tolerance, parallelism and generalization. On the other hand, fuzzy logic provides a means for converting linguistic strategy into control actions and thus offering a high-level computation. Although, fuzzy logic and artificial neural networks are both functionally and structurally different, it is envisaged that the synthesis of these two areas will give rise to a new paradigm called fuzzy neural networks. The latter have the potential to capture the benefits of both the fields, fuzzy logic and neural networks, into a single paradigm.

Over the past two decades or so, several parallel advances have been made in the two distinct disciplines: fuzzy logic and neural networks. Fuzzy logic provides a mathematical strength to the emulation of certain perceptual and linguistic attributes associated with human cognition, whereas the science of neural networks provides a new computing tool with learning and adaptation capabilities. While the theory of fuzzy logic provides an inference mechanism under cognitive uncertainty, computational neural networks offer exciting advantages such as learning, adaptation, fault-tolerance, parallelism and generalization. Neural networks, comprising of processing elements called neurons, are capable of coping with computational complexity, non-linearity and uncertainty. In view of this versatility of neural networks, it is believed that they hold great potential as building blocks for a variety of behaviors associated with human cognition. However, the subjective phenomena such as reasoning and perceptions are often regarded beyond the domain of neural network theory.

The Fuzzy Neural hybrid system can be realized as both

(i) Fuzzy Neural Hybrids

(ii) Neuro Fuzzy Hybrids

The difference in these two hybrids is in terms of the system which forms the input and which forms the output.

16.8.1 Fuzzy Neural Hybrids

It is interesting to note that fuzzy logic and neural networks complement each other, and their unification provides the benefits of both the technologies. Neural networks can deal with imprecise data and ill-defined activities; thus they offer low-level computational features. On the other hand, fuzzy logic provides higher-level cognitive features as it can deal with issues such as approximate reasoning and natural language processing. A brief comparison between fuzzy logic and neural networks from the point of knowledge acquisitions, uncertainty, reasoning, adaptation and natural language processing is shown in Table 16.1. The merging of these two fields results in an emerging paradigm called 'fuzzy neural networks' or 'fuzzy-neural systems'. Fuzzy neural networks are believed to have considerable potential in the areas of expert systems, medical diagnosis, control systems, pattern recognition and system modeling.

Table 16.1 *A Comparative Study between Fuzzy Systems and Neural Networks*

	Skills	*Fuzzy Systems*	*Neural Networks*
Knowledge acquisition	Inputs	Human experts	Sample sets
	Tools	Interaction	Algorithms
Uncertainty	Information	Quantitative and qualitative	Quantitative
	Cognition	Decision making	Perception
Reasoning	Mechanism	Heuristic search	Parallel computations
	Speed	Low	High
Adaptation	Fault-tolerance	Low	Very high
	Learning	Induction	Adjusting synaptic weights
Natural language	Implementation	Explicit	Implicit
	Flexibility	High	Low

To enable a system to track real-life situations in a manner more like humans, one may incorporate the concept of fuzzy sets into the neural network. Although fuzzy logic is a natural mechanism for propagating uncertainty, it may involve, in some cases, an increase in the amount of computation required (compared with a system using classical binary logic). Using fuzzy neural network models having the potential for parallel computation with high flexibility can readily offset this.

The structure of this type of fuzzy neural system is shown in Fig. 16.2.

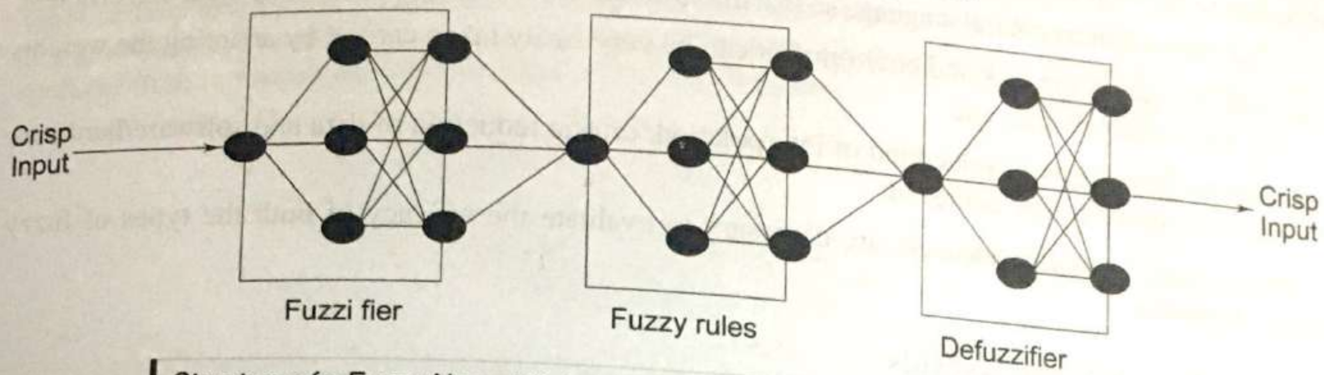


Fig. 16.2 *Structure of a Fuzzy Neural Network. In this System, Neural Networks Essentially Implement the Functions of a Fuzzy System. One Network Performs the Fuzzification of Crisp Input Data, the Fuzzy Rules are Implemented by a Second Network, and Finally the Operation of Defuzzification is Performed to Provide Crisp Output Data*

A fuzzy neural architecture can be visualized in two possible models as depicted in Fig. 16.2. In the first model, Fig. 16.3(a), linguistic statements are inputs to a multi-layered neural network through a fuzzy interface. In the second model, Fig. 16.3(b), a multi-layered neural network drives the fuzzy inference mechanism.

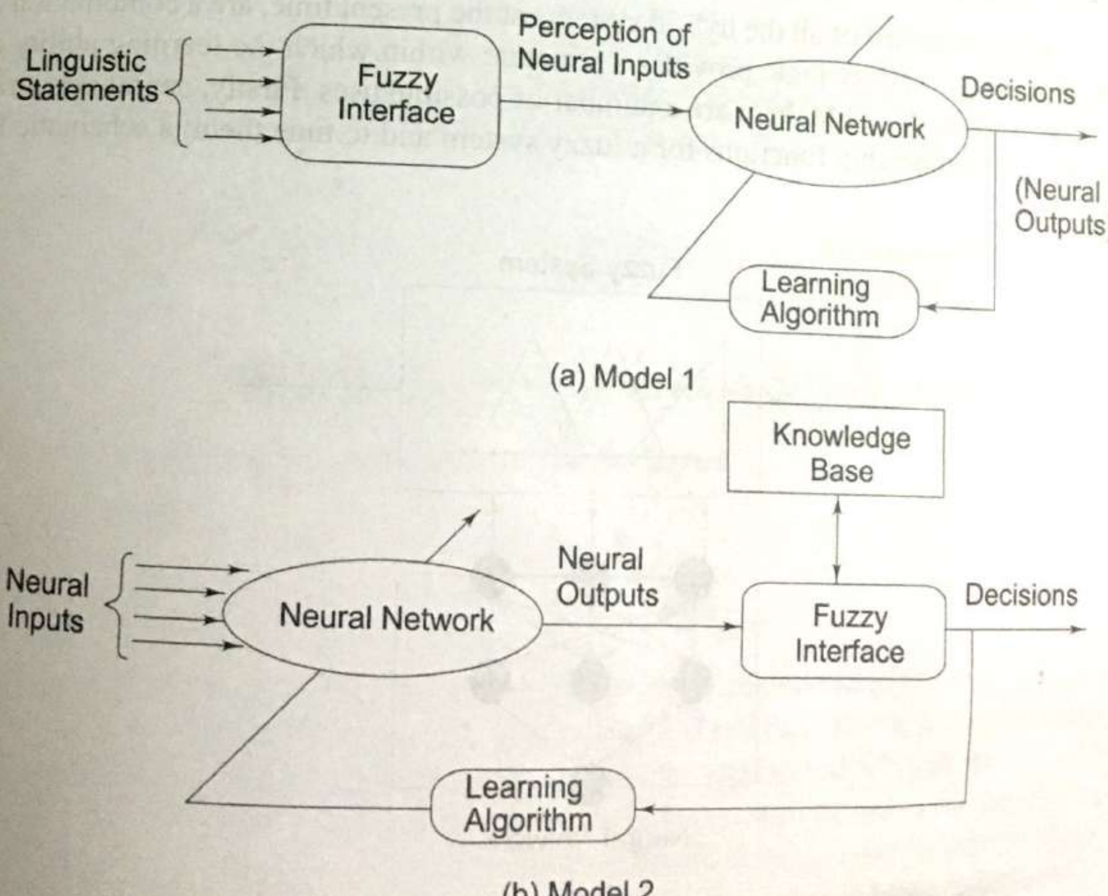


Fig. 16.3 *Two Models of Fuzzy Neural Systems*

Using fuzzy operators and connectives in the mathematical operations of an artificial neuron can develop fuzzy neural networks. Irrespective of what type of fuzzy neural networks are used, it is envisaged that they offer the following features:

1. Easy to implement natural language so that the structure of knowledge base is very clear and efficient.
2. Any changes in the task and environment can be very easily taken care of by adapting the weights of the neural network, and
3. Since a fuzzy system is one kind of interpolation, drastic reduction of data and software/hardware and overheads can be achieved.

Hence more research endeavors are necessary to evaluate the efficacy of both the types of fuzzy neural networks.

16.8.2 Neuro Fuzzy Hybrids

Presently, the neuro-fuzzy approach is becoming one of the major areas of interest because it gets the benefits of neural networks as well as of fuzzy logic systems and it removes the individual disadvantages by combining them on the common features. Different architectures of neuro-fuzzy system have been investigated by number of researchers such as Lin (1994), Medsker (1995) and Jana (1996). These architectures have been applied in many applications especially in the process control.

Neural networks and fuzzy logic have some common features such as distributed representation of knowledge, model-free estimation, ability to handle data with uncertainty and imprecision etc. Fuzzy logic has tolerance for imprecision of data, while neural networks have tolerance for noisy data.

The most widely researched of all the hybrid systems at the present time, are a combination of neural networks and fuzzy logic. Fuzzy logic provides a structure within which the learning ability of neural networks is employed. In this field there are a number of possible uses. Firstly, neural networks can be used to generate the membership functions for a fuzzy system and to tune them; a schematic for this is shown in Fig. 16.4.

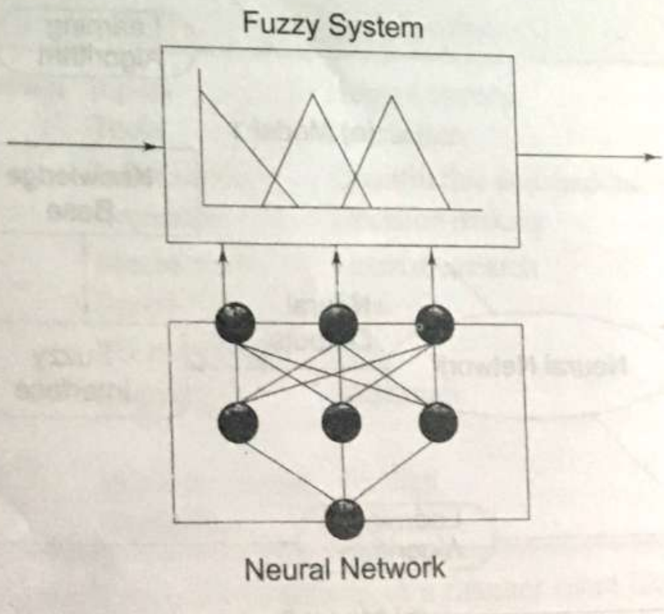


Fig. 16.4 | *Neuro-fuzzy System for Tuning a Membership Function*

Neuro-fuzzy hybrid systems combine the advantages of fuzzy systems, which deal with explicit knowledge, which can be explained and understood, and neural networks, which deal with implicit knowledge which can be acquired by learning. Neural network learning provides a good way to adjust the expert's knowledge and automatically generate additional fuzzy rules and membership functions, to meet certain specifications and reduce design time and costs. On the other hand, fuzzy logic enhances the generalization capability of a neural network system by providing more reliable output when extrapolation is needed beyond the limits of the training data.

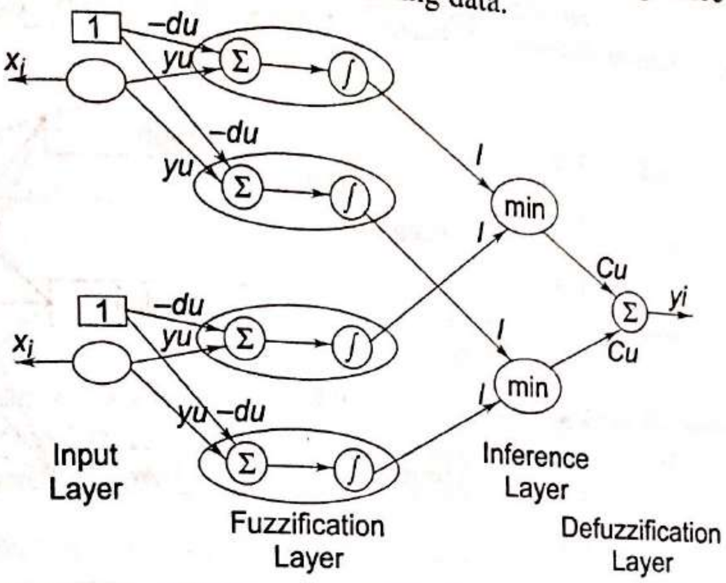


Fig. 16.5 *Architecture of Neuro-fuzzy System. The System has Two Inputs, One Output, and Two Rules*

The neuro-fuzzy system consists of the components of a conventional fuzzy system except that computations at each stage are performed by a layer of hidden neurons and the neural network's learning capacity is provided to enhance the system knowledge. The other architectures of neuro-fuzzy system are available out of which two prominent types are shown in Figs. 16.5 and 16.6.

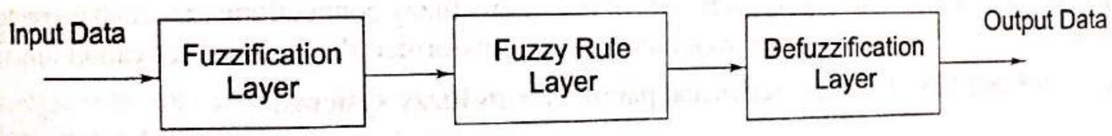


Fig. 16.6 *Schematic Diagram of a Neuro-Fuzzy System*

The system contains the following three different layers:

- Fuzzification layer
- Fuzzy rule layer
- Defuzzification layer

In a *fuzzification layer* each neuron represents an input membership function of the antecedent of a fuzzy rule. In the *fuzzy inference layer* fuzzy rules are fired and the value at the end of each rule represents the initial weight of the rule, and will be adjusted to its appropriate level at the end of training. In the *defuzzification layer* each neuron represents a consequent proposition and its membership function can be implemented by combining one or two sigmoid functions and linear functions. The weight of each output link here represents the centre of gravity of each output membership function of the consequent and is trainable. After getting the corresponding output the adjustment is made in the connection weights and the membership functions in order to compensate the error and produce a new control signal.

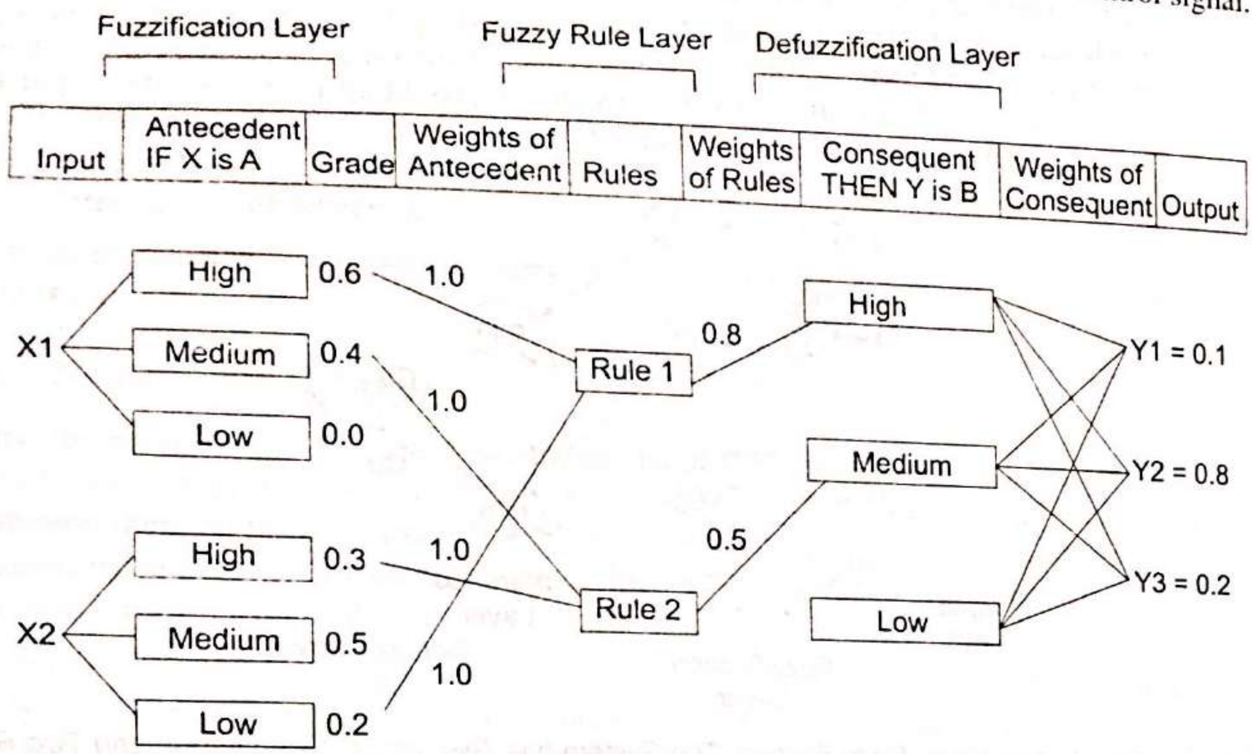


Fig. 16.7 | *Neuro-fuzzy System Structure*

The neuro-fuzzy control architecture proposed by Lin (1994) has five layers, each of which performs fuzzy rule-based system operations (Fig. 16.7). This belongs to the fully integrated hybrid intelligent system. Computer simulations satisfactorily verified the performance of this structure of a neuro-fuzzy control. Like a typical neural network, each node of this neuro-fuzzy connectionist model has connecting weights for incoming variables and the weighted sum is transformed through the activation function.

Figures 16.8 and 16.9 shows the series and parallel neuro fuzzy systems.

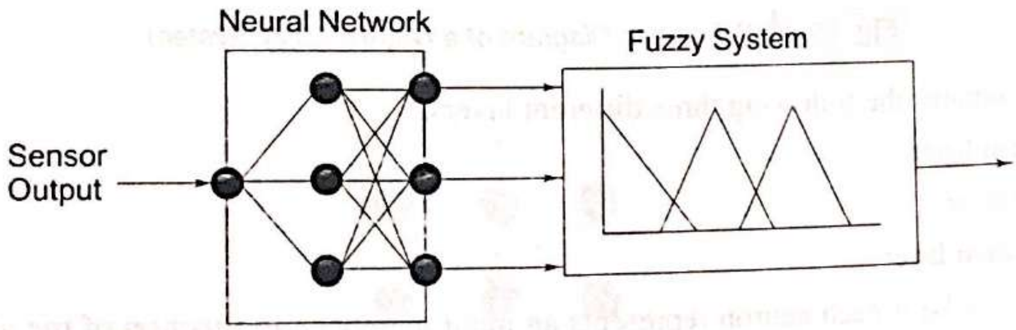


Fig. 16.8 | *Neuro-Fuzzy System with Neural Network in Pre-processing Role. This is a Series System, and would be used if the Sensor Output is not Suitable for Direct Connection to the Input of the Fuzzy System. Post-processing Systems also Exist, in which the Output of a Fuzzy System is not Suitable for Direct Connection to External Devices, and therefore a Neural Network Provides an Interface that Performs a Mapping which could not Easily be Carried out using Analytical Techniques*

to p...

(H) GENETIC ALGORITHMS :- (GA)

→ GA is a search- based optimization technique based on the principles of genetics and natural selection.

→ It is frequently used to find optimal or near optimal solutions to difficult problems which otherwise would take lifetime to solve.

Optimization : process of making something better.

→ GA was developed by John Holland.

→ $\boxed{\text{GA Terminologies}}$ → i) Population : subset of all possible solutions to given problem.

ii) Genotype :- It is the population in computation space. In the computation space, the solutions are represented in a way which can be easily understood and manipulated using a computing system.

iii) Phenotype :- It is the population in actual real-world solution space in which solutions are represented in a way they are represented in real world situations.

iii) Decoding : It is a process of transforming a solution from the genotype to phenotype space.

iv) Encoding : It is a process of transforming from the phenotype to genotype space.

vi) Fitness function : It is a function which takes the solution as input and produces the suitability of the solution as the output.

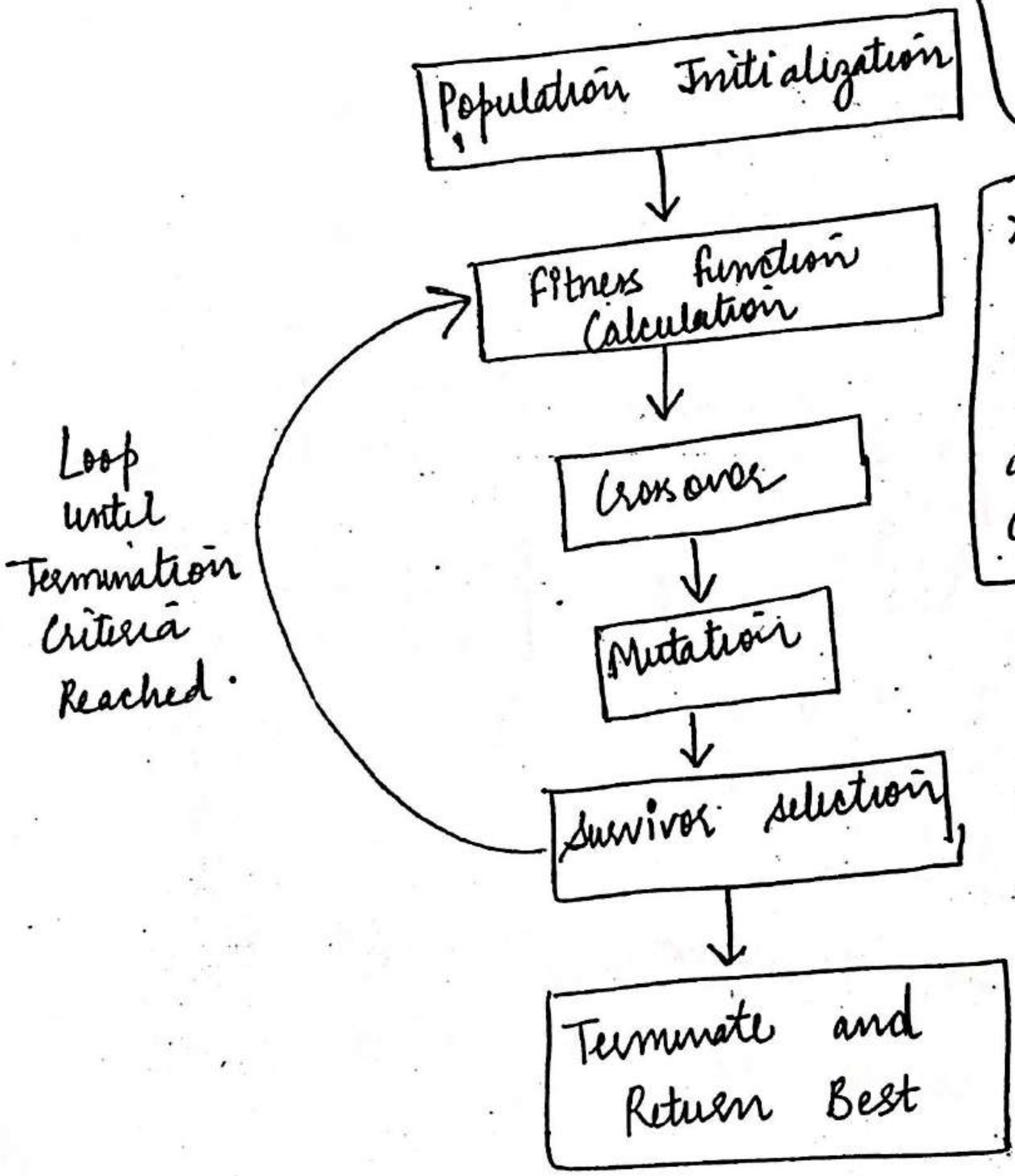
vii) Chromosome : A chromosome is one such solution to given problem.

viii)

viii) Genetic operators: These alter the genetic composition of the offspring. These include cross over, mutation, selection etc.

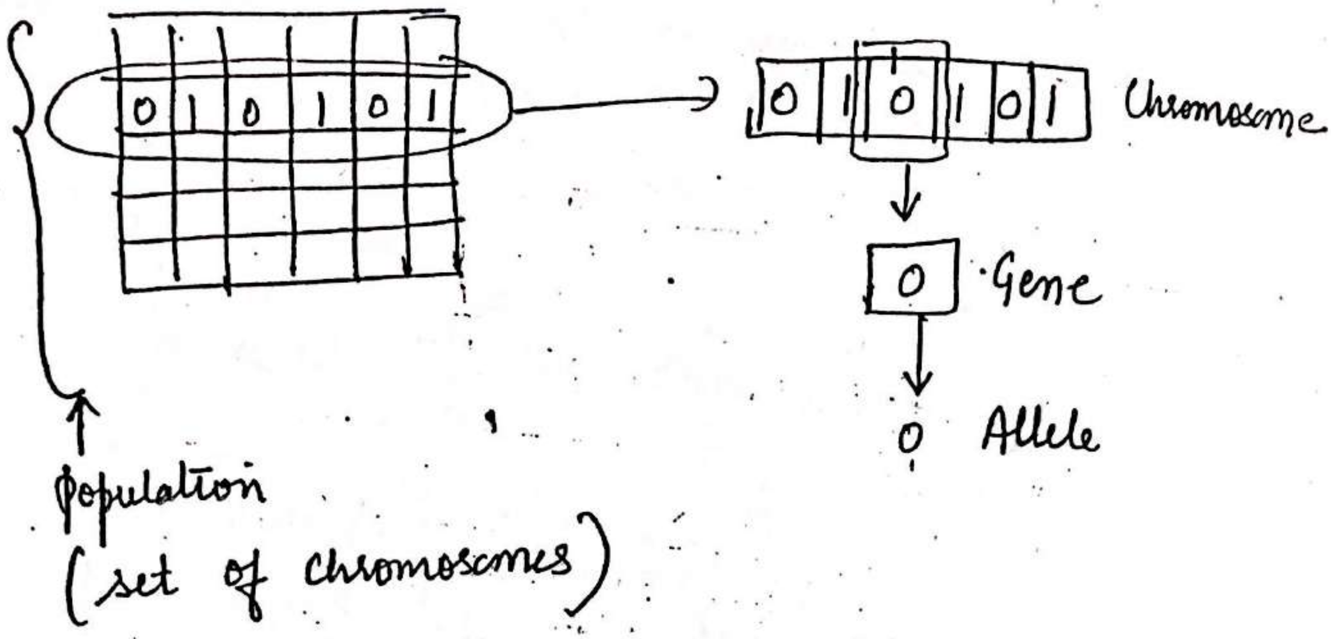
ix) Gene.: A gene is one element position of chromosome.

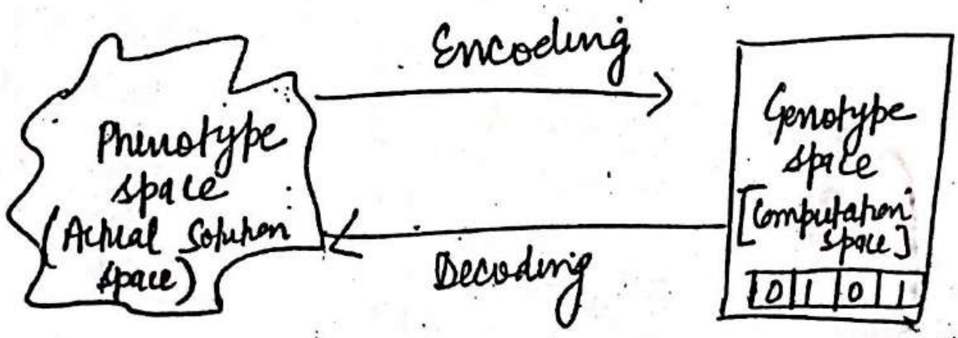
→ Basic structure of GA :-



x) Allele: It is the value a gene takes for a particular chromosome

Representation of GA Terminologies :





Working of GA :-

i) Firstly the populations are selected
ii) Every chromosome is assigned some fitness value.
iii) Crossover (Reproduction of offspring) is done between diff chromosome.

iv) During the creation of offspring, recombination occurs (due to cross over) and in that process genes from parents form a whole new chromosome in some way.

v) The new created offspring can then be mutated.

vi) Mutation means element of DNA is modified.

vii) The fitness of an organism is measured by means of success of organism in life.

→ Representation of GA :

i) Binary Representation : Eg: 0/1 knapsack problem

ii) Real valued Representation : when we want to define the genes using continuous rather than discrete variables.

0.5	0.2	0.6	0.8	0.7

iii) Integer Representation :
Eg: If you want to encode 4 distances – North, South, East, West we can encode them as {0, 1, 2, 3}.

For such cases integer representation is desirable

1	2	3	4	3	2	1

iv) Permutation Combination :-

The solution is represented by order of elements.

Eg: Travelling Salesman Problem : visiting each city exactly once + coming back to starting city.

1	5	4	2	3

v) Hexa decimal Representation : (0123456789ABCDEF)

vi) Octal Representation.

→ The various methods for selection of chromosome for parents to crossover are :

i) Roulette- wheel Selection
ii) Boltzmann Selection
iii) Tournament Selection
iv) Rank selection
v) Steady - state Selection.

No need for going in details