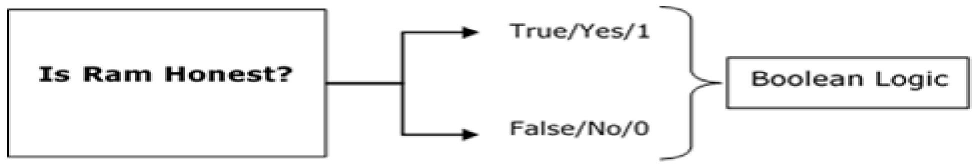
Fuzzy Logic

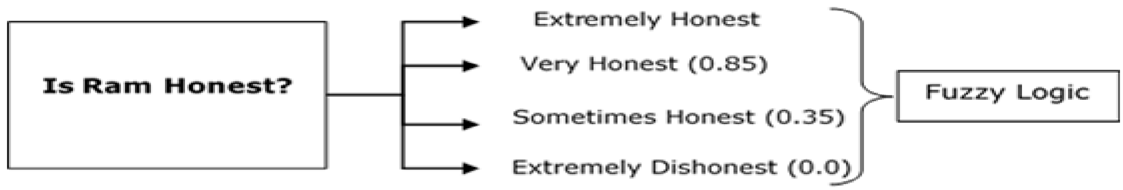
The word **fuzzy** refers to things which are not clear or are vague. Any event, process, or function that is changing continuously cannot always be defined as either true or false, which means that we need to define such activities in a Fuzzy manner.

What is 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.





In other words, we can say that fuzzy logic is not logic that is fuzzy, but logic that is used to describe fuzziness. There can be numerous other examples like this with the help of which we can understand the concept of fuzzy logic. Fuzzy Logic was introduced in 1965 by Lofti A. Zadeh in his research paper "Fuzzy Sets". He is considered as the father of Fuzzy Logic.

Fuzzy Logic - Classical Set Theory

A set is an unordered collection of different elements. It can be written explicitly by listing its elements using the set bracket. If the order of the elements is changed or any element of a set is repeated, it does not make any changes in the set.

Example

- A set of all positive integers.
- A set of all the planets in the solar system.
- A set of all the states in India.
- A set of all the lowercase letters of the alphabet.

Mathematical Representation of a Set

Sets can be represented in two ways − Roster or Tabular Form

In this form, a set is represented by listing all the elements comprising it. The elements are enclosed within braces and separated by commas.

Following are the examples of set in Roster or Tabular Form −

- Set of vowels in English alphabet, A = {a,e,i,o,u}
- Set of odd numbers less than 10, B = {1,3,5,7,9}

Set Builder Notation

In this form, the set is defined by specifying a property that elements of the set have in common. The set is described as A = {x:p(x)}

Example 1 − The set {a,e,i,o,u} is written as A = {x:x is a vowel in English alphabet}
Example 2 − The set {1,3,5,7,9} is written as B = {x:1 ≤ x < 10 and (x%2) ≠ 0}

If an element x is a member of any set S, it is denoted by x∈S and if an element y is not a member of set S, it is denoted by y∉S.
Example − If S = {1,1.2,1.7,2},1 ∈ S but 1.5 ∉ S

Cardinality of a Set

Cardinality of a set S, denoted by |S||S|, is the number of elements of the set. The number is also referred as the cardinal number. If a set has an infinite number of elements, its cardinality is ∞
Example − |{1,4,3,5}| = 4,|{1,2,3,4,5,…}| = ∞

If there are two sets X and Y, *|X| = |Y|* denotes two sets X and Y having same cardinality. It occurs when the number of elements in X is exactly equal to the number of elements in Y. In this case, there exists a bijective function 'f' from X to Y.

|X| ≤ |Y| denotes that set X's cardinality is less than or equal to set Y's cardinality. It occurs when the number of elements in X is less than or equal to that of Y. Here, there exists an injective function 'f' from X to Y.

|X| < |Y| denotes that set X's cardinality is less than set Y's cardinality. It occurs when the number of elements in X is less than that of Y. Here, the function 'f' from X to Y is injective function but not bijective.

If *|X| ≤ |Y|* and *|X| ≤ |Y|* then *|X| = |Y|*. The sets X and Y are commonly referred as **equivalent sets**.

Types of Sets

Sets can be classified into many types; some of which are finite, infinite, subset, universal, proper, singleton set, etc.

Finite Set

A set which contains a definite number of elements is called a finite set.
Example − S = {x|x ∈ N and 70 > x > 50}

Infinite Set

A set which contains infinite number of elements is called an infinite set.
Example − S = {x|x ∈ N and x > 10}

Subset

A set X is a subset of set Y (Written as X ⊆ Y) if every element of X is an element of set Y.
Example 1 − Let, X = {1,2,3,4,5,6} and Y = {1,2}. Here set Y is a subset of set X as all the elements of set Y is in set X. Hence, we can write Y⊆X.

Example 2 − Let, X = {1,2,3} and Y = {1,2,3}. Here set Y is a subset (not a proper subset) of set X as all the elements of set Y is in set X. Hence, we can write $Y \subseteq X$.

Proper Subset

The term "proper subset" can be defined as "subset of but not equal to". A Set X is a proper subset of set Y (Written as $X \subset Y$) if every element of X is an element of set Y and $|X| < |Y|$.

Example − Let, X = {1,2,3,4,5,6} and Y = {1,2}. Here set $Y \subset X$, since all elements in Y are contained in X too and X has at least one element which is more than set Y.

Universal Set

It is a collection of all elements in a particular context or application. All the sets in that context or application are essentially subsets of this universal set. Universal sets are represented as U.

Example − We may define U as the set of all animals on earth. In this case, a set of all mammals is a subset of U, a set of all fishes is a subset of U, a set of all insects is a subset of U, and so on.

Empty Set or Null Set

An empty set contains no elements. It is denoted by Φ. As the number of elements in an empty set is finite, empty set is a finite set. The cardinality of empty set or null set is zero.

Example – $S = \{x \mid x \in N \text{ and } 7 < x < 8\} = \Phi$

Singleton Set or Unit Set

A Singleton set or Unit set contains only one element. A singleton set is denoted by {s}.

Example − $S = \{x \mid x \in N, 7 < x < 9\} = \{8\}$

Equal Set

If two sets contain the same elements, they are said to be equal.

Example − If A = {1,2,6} and B = {6,1,2}, they are equal as every element of set A is an element of set B and every element of set B is an element of set A.

Equivalent Set

If the cardinalities of two sets are same, they are called equivalent sets.

Example − If A = {1,2,6} and B = {16,17,22}, they are equivalent as cardinality of A is equal to the cardinality of B. i.e. $|A| = |B| = 3$

Overlapping Set

Two sets that have at least one common element are called overlapping sets. In case of overlapping sets −

$$n(A \cup B) = n(A) + n(B) - n(A \cap B)$$
$$n(A \cup B) = n(A - B) + n(B - A) + n(A \cap B)$$
$$n(A) = n(A - B) + n(A \cap B)$$
$$n(B) = n(B - A) + n(A \cap B)$$

Example − Let, A = {1,2,6} and B = {6,12,42}. There is a common element '6', hence these sets are overlapping sets.

Disjoint Set

Two sets A and B are called disjoint sets if they do not have even one element in common. Therefore, disjoint sets have the following properties −

$n(A \cap B) = \varphi n(A \cap B) = \varphi$

n(A∪B)=n(A)+n(B)

Example − Let, A = {1,2,6} and B = {7,9,14}, there is not a single common element, hence these sets are overlapping sets.

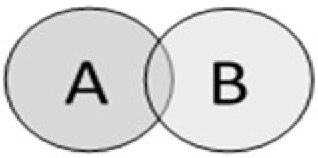
Operations on Classical Sets

Set Operations include Set Union, Set Intersection, Set Difference, Complement of Set, and Cartesian Product.

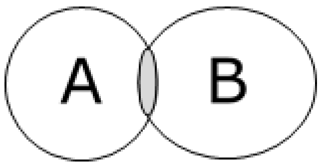
Union

The union of sets A and B (denoted by A ∪ BA ∪ B) is the set of elements which are in A, in B, or in both A and B. Hence, A ∪ B = {x|x ∈ A OR x ∈ B}.

Example − If A = {10,11,12,13} and B = {13,14,15}, then A ∪ B = {10,11,12,13,14,15} – The common element occurs only once.



A U B



A ∩ B

Intersection

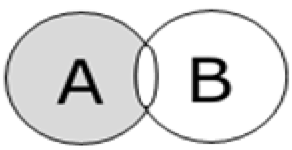
The intersection of sets A and B (denoted by A ∩ B) is the set of elements which are in both A and B. Hence, A ∩ B = {x|x ∈ A AND x ∈ B}.

Difference/ Relative Complement

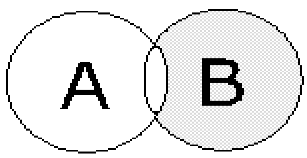
The set difference of sets A and B (denoted by A–B) is the set of elements which are only in A but not in B. Hence, A − B = {x|x ∈ A AND x ∉ B}.

Example − If A = {10,11,12,13} and B = {13,14,15}, then (A − B) = {10,11,12} and (B − A) = {14,15}.

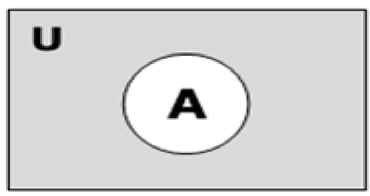
Here, we can see (A − B) ≠ (B − A)



A - B



B - A



Complement of set A

Complement of a Set

The complement of a set A (denoted by A′) is the set of elements which are not in set A. Hence, A′ = {x|x ∉ A}.

More specifically, A′ = (U−A) where U is a universal set which contains all objects.

Example − If A = {x|x belongs to set of add integers} then A′ = {y|y does not belong to set of odd integers}

Cartesian Product / Cross Product

The Cartesian product of n number of sets A1,A2,…An denoted as A1 × A2...× An can be defined as all possible ordered pairs (x1,x2,…xn) where x1 ∈ A1,x2 ∈ A2,…xn ∈ An

Example − If we take two sets A = {a,b} and B = {1,2},

The Cartesian product of A and B is written as − A × B = {(a,1),(a,2),(b,1),(b,2)}

And, the Cartesian product of B and A is written as − B × A = {(1,a),(1,b),(2,a),(2,b)}

Properties of Classical Sets

Properties on sets play an important role for obtaining the solution. Following are the different properties of classical sets −

Commutative Property

Having two sets **A** and **B**, this property states −

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

Associative Property

Having three sets **A**, **B** and **C**, this property states −

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

Distributive Property

Having three sets **A**, **B** and **C**, this property states −

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

Idempotency Property

For any set **A**, this property states −

$$A\cup A=A$$
$$A\cap A=A$$

Identity Property

For set **A** and universal set **X**, this property states −

$$A\cup\phi=A$$
$$A\cap X=A$$
$$A\cap\phi=\phi$$
$$A\cup X=X$$

Transitive Property

Having three sets **A**, **B** and **C**, the property states − **If $A\subseteq B\subseteq C$, then $A\subseteq CA\subseteq C$**

Involution Property

For any set **A**, this property states − $\overline{\overline{A}}=A$

De Morgan's Law

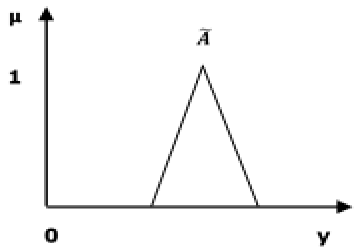
It is a very important law and supports in proving tautologies and contradiction. This law states −

$$\overline{A\cap B} = \overline{A}\cup\overline{B}$$

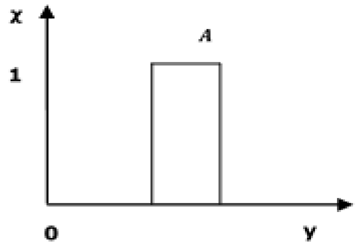
$$\overline{A\cup B} = \overline{A}\cap\overline{B}$$

Fuzzy Logic - Set Theory

Fuzzy sets can be considered as an extension and gross oversimplification of classical sets. It can be best understood in the context of set membership. Basically it allows partial membership which means that it contain elements that have varying degrees of membership in the set. From this, we can understand the difference between classical set and fuzzy set. Classical set contains elements that satisfy precise properties of membership while fuzzy set contains elements that satisfy imprecise properties of membership.



Membership Function of Fuzzy set $\tilde{A}$



Membership Function of classical set A

Mathematical Concept

A fuzzy set **A˜** in the universe of information **U** can be defined as a set of ordered pairs and it can be represented mathematically as – $\tilde{A}=\{(y,\mu_{\tilde{A}}(y)) | y\in U\}$

Here $\mu_{\tilde{A}}(y)$ = degree of membership of **y** in \widetilde{A}, assumes values in the range from 0 to 1, i.e., $\mu_{\tilde{A}}(y)\in[0,1]$

Representation of fuzzy set

Let us now consider two cases of universe of information and understand how a fuzzy set can be represented.

Case 1

When universe of information **U** is discrete and finite –

$$\widetilde{A}=\left\{\frac{\mu_{\widetilde{A}}(y_1)}{y_1}+\frac{\mu_{\widetilde{A}}(y_2)}{y_2}+\frac{\mu_{\widetilde{A}}(y_3)}{y_3}+\dots\right\}$$

$$=\left\{\sum_{i=1}^{n}\frac{\mu_{\widetilde{A}}(y_i)}{y_i}\right\}$$

Case 2

When universe of information **U** is continuous and infinite –

$$\widetilde{A}=\left\{\int\frac{\mu_{\widetilde{A}}(y)}{y}\right\}$$

In the above representation, the summation symbol represents the collection of each element.

Operations on Fuzzy Sets

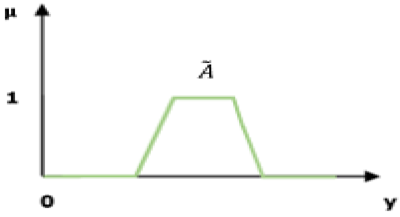
Having two fuzzy sets **A˜** and **B˜**, the universe of information **U** and an element **y** of the universe, the following relations express the union, intersection and complement operation on fuzzy sets.

Union/Fuzzy 'OR'

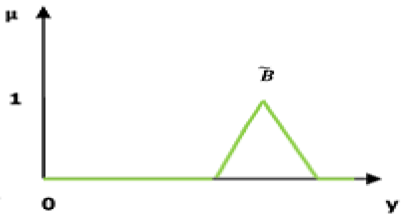
Let us consider the following representation to understand how the Union/Fuzzy 'OR' relation works –

$$\mu_{\widetilde{A}\cup\widetilde{B}}(y)=\mu_{\widetilde{A}}\vee\mu_{\widetilde{B}}\quad\forall y\in U$$

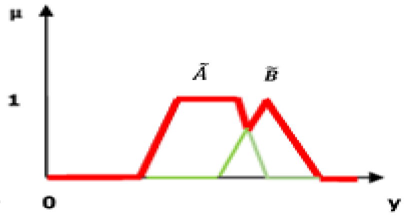
Here $\vee$ represents the 'max' operation.



Fuzzy set $\tilde{A}$



Fuzzy set $\tilde{B}$



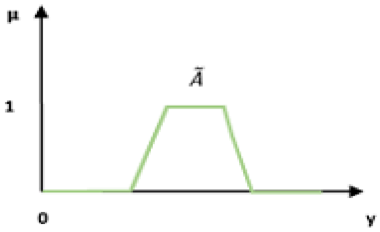
Union of two Fuzzy sets

Intersection/Fuzzy 'AND'

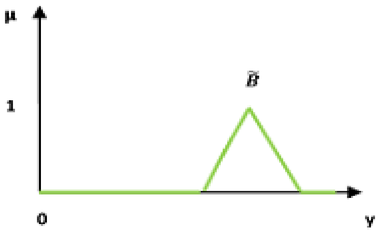
Let us consider the following representation to understand how the Intersection/Fuzzy 'AND' relation works –

$$\mu_{\tilde{A}\cap\tilde{B}}(y) = \mu_{\tilde{A}} \wedge \mu_{\tilde{B}} \quad \forall y \in U$$

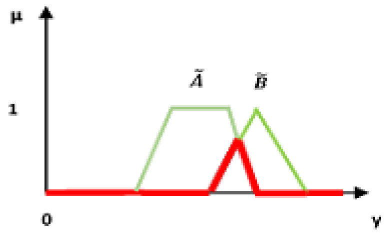
Here $\wedge$ represents the 'min' operation.



Fuzzy set $\tilde{A}$



Fuzzy set $\tilde{B}$

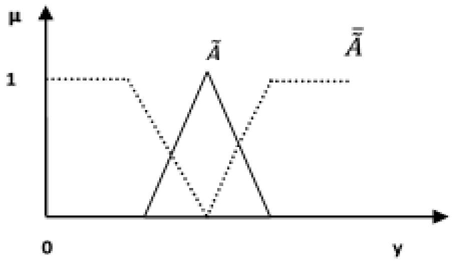


Intersection of two Fuzzy sets

Complement/Fuzzy 'NOT'

Let us consider the following representation to understand how the **Complement/Fuzzy 'NOT'** relation works –

$$\mu_{\bar{\tilde{A}}} = 1 - \mu_{\tilde{A}}(y) \quad y \in U$$



Complement of a fuzzy set

Properties of Fuzzy Sets

Let us discuss the different properties of fuzzy sets.

Commutative Property

Having two fuzzy sets **A˜**and **B˜**, this property states –

$$\tilde{A} \cup \tilde{B} = \tilde{B} \cup \tilde{A} \qquad \tilde{A} \cap \tilde{B} = \tilde{B} \cap \tilde{A}$$

Associative Property

Having three fuzzy sets A˜, B˜ and C˜, this property states –

$$\tilde{A} \cup \left(\tilde{B} \cup \tilde{C}\right) = \left(\tilde{A} \cup \tilde{B}\right) \cup \tilde{C} \quad \tilde{A} \cap \left(\tilde{B} \cap \tilde{C}\right) = \left(\tilde{A} \cup \tilde{B}\right) \cup \tilde{C}$$

Distributive Property

Having three fuzzy sets **A˜, B˜** and **C˜**, this property states –

$$\tilde{A} \cup \left(\tilde{B} \cap \tilde{C}\right) = \left(\tilde{A} \cup \tilde{B}\right) \cap \left(\tilde{A} \cup \tilde{C}\right) \quad \tilde{A} \cap \left(\tilde{B} \cup \tilde{C}\right) = \left(\tilde{A} \cap \tilde{B}\right) \cup \left(\tilde{A} \cap \tilde{C}\right)$$

Idempotency Property

For any fuzzy set **A˜**, this property states –

$$\tilde{A} \cup \tilde{A} = \tilde{A} \quad \tilde{A} \cap \tilde{A} = \tilde{A}$$

Identity Property

For fuzzy set **A˜** and universal set **U**, this property states –

$$\tilde{A} \cup \varphi = \tilde{A} \quad \tilde{A} \cap U = \tilde{A} \quad \tilde{A} \cap \varphi = \varphi \quad \tilde{A} \cup U = U$$

Transitive Property

Having three fuzzy sets **A˜**, **B˜** and **C˜**, this property states –

$$If\ \tilde{A} \subseteq \tilde{B} \subseteq \tilde{C},\ then\ \tilde{A} \subseteq \tilde{C}$$

Involution Property

For any fuzzy set **A˜**, this property states –

$$\overline{\overline{\tilde{A}}} = \tilde{A}$$

De Morgan's Law

This law plays a crucial role in proving tautologies and contradiction. This law states –

$$\overline{\tilde{A} \cap \tilde{B}} = \overline{\tilde{A}} \cup \overline{\tilde{B}} \quad \overline{\tilde{A} \cup \tilde{B}} = \overline{\tilde{A}} \cap \overline{\tilde{B}}$$

Fuzzy Logic vs. Probability

Fuzzy Logic	Probability
Fuzzy: Tom's degree of membership within the set of old people is 0.90.	Probability: There is a 90% chance that Tom is old.
Fuzzy logic takes truth degrees as a mathematical basis on the model of the vagueness phenomenon.	Probability is a mathematical model of ignorance.

Crisp vs. Fuzzy

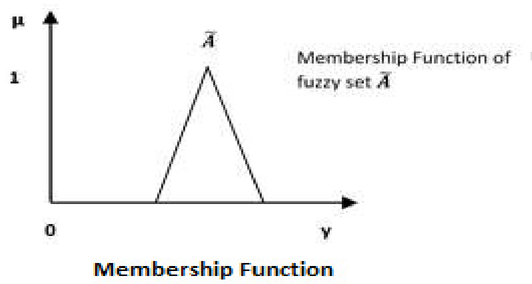
Crisp	Fuzzy
It has strict boundary T or F	Fuzzy boundary with a degree of membership
Some crisp time set can be fuzzy	It can't be crisp
True/False {0,1}	Membership values on [0,1]
In Crisp logic law of Excluded Middle and Non-Contradiction may or may not hold	In the fuzzy logic law of Excluded Middle and Non-Contradiction hold

Classical Set vs. Fuzzy set Theory

Classical Set	Fuzzy Set Theory
Classes of objects with sharp boundaries.	Classes of objects do not have sharp boundaries.
A classical set is defined by crisp boundaries, i.e., there is clarity about the location of the set boundaries.	A fuzzy set always has ambiguous boundaries, i.e., there may be uncertainty about the location of the set boundaries.
Widely used in digital system design	Used only in fuzzy controllers.

Fuzzy Logic - Membership Function

Fuzzy logic is not logic that is fuzzy but logic that is used to describe fuzziness. This fuzziness is best characterized by its membership function. In other words, we can say that membership function represents the degree of truth in fuzzy logic.



Following are a few important points relating to the membership function −

- Membership functions were first introduced in 1965 by Lofti A. Zadeh in his first research paper “fuzzy sets”.
- Membership functions characterize fuzziness (i.e., all the information in fuzzy set), whether the elements in fuzzy sets are discrete or continuous.
- Membership functions can be defined as a technique to solve practical problems by experience rather than knowledge.
- Membership functions are represented by graphical forms.
- Rules for defining fuzziness are fuzzy too.

Mathematical Notation

We have already studied that a fuzzy set $\tilde{A}$ in the universe of information U can be defined as a set of ordered pairs and it can be represented mathematically as −

$$\widetilde{A} = \left\{ \left(y, \mu_{\tilde{A}}(y) \right) | y \in U \right\}$$

Here $\mu_{\tilde{A}}(\cdot)$ = membership function of $\tilde{A}$; this assumes values in the range from 0 to 1, i.e., $\mu_{\tilde{A}}(\cdot) \in [0,1]$.

The membership function $\mu_{\tilde{A}}(\cdot)$ maps **U** to the membership space **M**.

The dot **(·)** in the membership function described above, represents the element in a fuzzy set; whether it is discrete or continuous.

Features of Membership Functions

We will now discuss the different features of Membership Functions.

Core

For any fuzzy set **A~**, the core of a membership function is that region of universe that is characterize by full membership in the set. Hence, core consists of all those elements **y** of the universe of information such that,

μ A~ (y)=1

Support

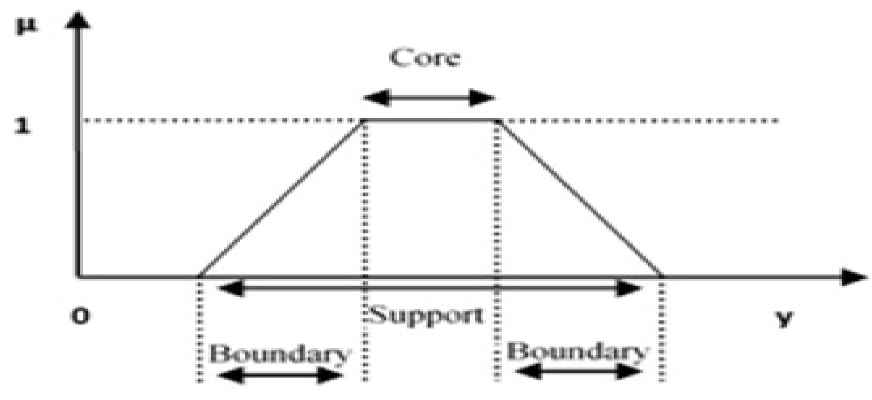
For any fuzzy set **A~**, the support of a membership function is the region of universe that is characterize by a nonzero membership in the set. Hence core consists of all those elements **y** of the universe of information such that,

$$\mu_{\tilde{A}}(y) > 0$$

Boundary

For any fuzzy set **A~**, the boundary of a membership function is the region of universe that is characterized by a nonzero but incomplete membership in the set. Hence, core consists of all those elements **y** of the universe of information such that,

$$1 > \mu_{\tilde{A}}(y) > 0$$



Features of Membership Function

Fuzzification

It may be defined as the process of transforming a crisp set to a fuzzy set or a fuzzy set to fuzzier set. Basically, this operation translates accurate crisp input values into linguistic variables.

Following are the two important methods of fuzzification –

Support Fuzzification(s-fuzzification) Method

In this method, the fuzzified set can be expressed with the help of the following relation –

$$\widetilde{A} = \mu_1 Q(x_1) + \mu_2 Q(x_2) + \ldots + \mu_n Q(x_n)$$

Here the fuzzy set **Q(xi)** is called as kernel of fuzzification. This method is implemented by keeping **μi** constant and **xi** being transformed to a fuzzy set $Q(x_i)$.

Grade Fuzzification (g-fuzzification) Method

It is quite similar to the above method but the main difference is that it kept **xi** constant and **μi** is expressed as a fuzzy set.

Defuzzification

It may be defined as the process of reducing a fuzzy set into a crisp set or to convert a fuzzy member into a crisp member.

Fuzzification process involves conversion from crisp quantities to fuzzy quantities. In a number of engineering applications, it is necessary to defuzzify the result or rather "fuzzy result" so that it must be converted to crisp result. Mathematically, the process of Defuzzification is also called "rounding it off".

Different Defuzzification Methods

The following are the known methods of defuzzification.

- Center of Sums Method (COS)
- Center of gravity (COG) / Centroid of Area (COA) Method
- Center of Area / Bisector of Area Method (BOA)
- Weighted Average Method
- Maxima Methods
  - First of Maxima Method (FOM)
  - Last of Maxima Method (LOM)
  - Mean of Maxima Method (MOM)

Center of Sums (COS) Method

This is the most commonly used defuzzification technique. In this method, the overlapping area is counted twice.

The defuzzified value x^* is defined as : =

$$x^* = \frac{\sum_{i=1}^{N} x_i \cdot \sum_{k=1}^{n} \mu_{A_k}(x_i)}{\sum_{i=1}^{N} \quad \sum_{k=1}^{n} \mu_{A_k}(x_i)} ,$$

Here, n is the number of fuzzy sets, N is the number of fuzzy variables, $\mu_{A_k}(x_i)$ is the membership function for the k-th fuzzy set.

The defuzzified value x^* is defined as :

$$x^* = \frac{\sum_{i=1}^{k} A_i \times \bar{x}_i}{\sum_{i=1}^{k} A_i} ,$$

Here, A_i represents the firing area of i^{th} rules and k is the total number of rules fired and $\bar{x}_i$ represents the center of area.

The aggregated fuzzy set of two fuzzy sets C_1 and C_2 is shown in Figure 1. Let the area of this two fuzzy sets are A_1 and A_2.

A_1 = ½ * [(8-1) + (7-3)] * 0.5 = ½ * 11 * 0.5 = 55/20=2.75

A_2 = ½ * [(9-3) + (8-4)] * 0.3 = ½ * 10 * 0.3 = 3/2 =1.5

Now the center of area of the fuzzy set C_1 is let say $\overline{x_1}$ = (7+3)/2= 5 and

the center of area of the fuzzy set C_2 is $\overline{x_2}$= (8+4)/2=6.

Now the defuzzified value $x^* = \frac{(A_1\overline{x_1}+A_2\overline{x_2})}{A_1+A_2} = \frac{(2.75 * 5 + 1.5 * 6)}{(2.75 + 1.5)}$ = 22.75/4.25 = 5.35

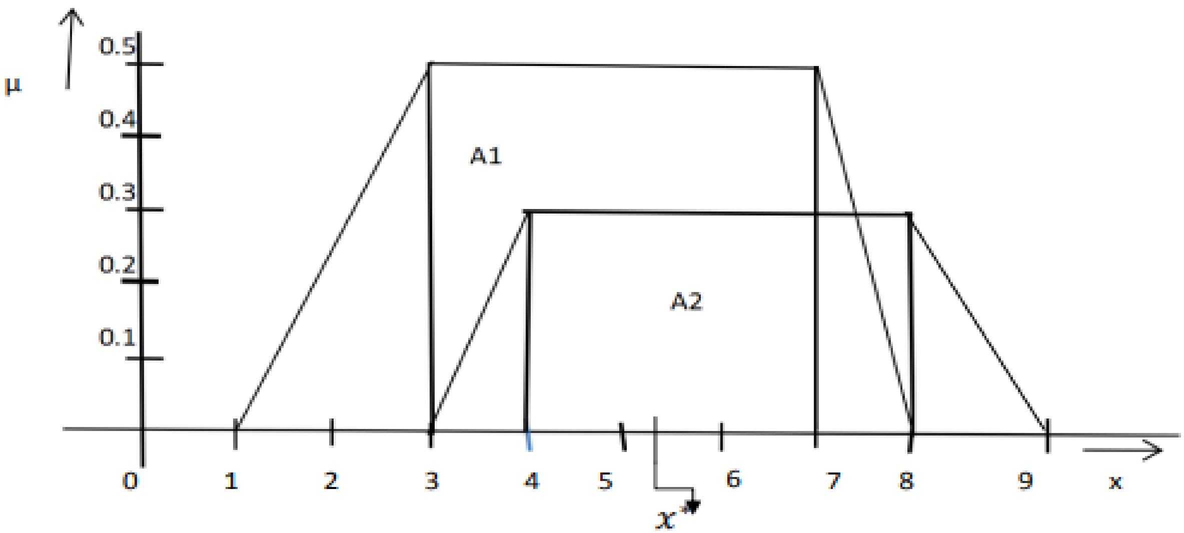


Figure 1 : Fuzzy sets $C_1$and C_2

Center of gravity (COG) / Centroid of Area (COA) Method

This method provides a crisp value based on the center of gravity of the fuzzy set. The total area of the membership function distribution used to represent the combined control action is divided into a number of sub-areas. The area and the center of gravity or centroid of each sub-area is calculated and then the summation of all these sub-areas is taken to find the defuzzified value for a discrete fuzzy set.

For discrete membership function, the defuzzified value denoted as x^* using COG is defined as:

$$x^* = \frac{\sum_{i=1}^{n} x_i \cdot \mu(x_i)}{\sum_{i=1}^{n} \mu(x_i)}$$

indicates the sample element, $\mu(x_i)$ is the membership function, and n represents the number of elements in the sample. For continuous membership function, x^* is defined as :

$$x^* = \frac{\int x\, \mu_A(x)\, dx}{\int \mu_A(x)\, dx}$$

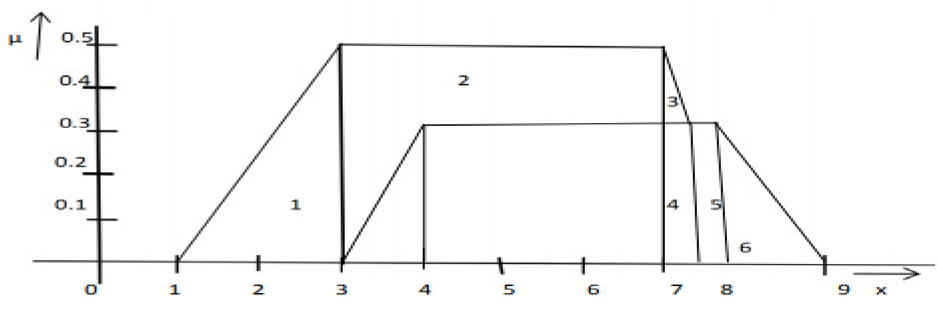


Figure 2 : Fuzzy sets C1 and C2

Center of Area / Bisector of Area Method (BOA)

This method calculates the position under the curve where the areas on both sides are equal. The BOA generates the action that partitions the area into two regions with the same area.

$$\int_{\alpha}^{x^*} \mu_A(x)\; dx = \int_{x^*}^{\beta} \mu_A(x)dx \text{ , where } \alpha = \min\{x \mid x \in X\} \text{ and } \beta = \max\{x \mid x \in X\}$$

Weighted Average Method

This method is valid for fuzzy sets with symmetrical output membership functions and produces results very close to the COA method. This method is less computationally intensive. Each membership function is weighted by its maximum membership value. The defuzzified value is defined as :

$$x^* = \frac{\sum \mu(x).x}{\sum \mu(x)}$$

Here $\sum$ denotes the algebraic summation and x is the element with maximum membership function.

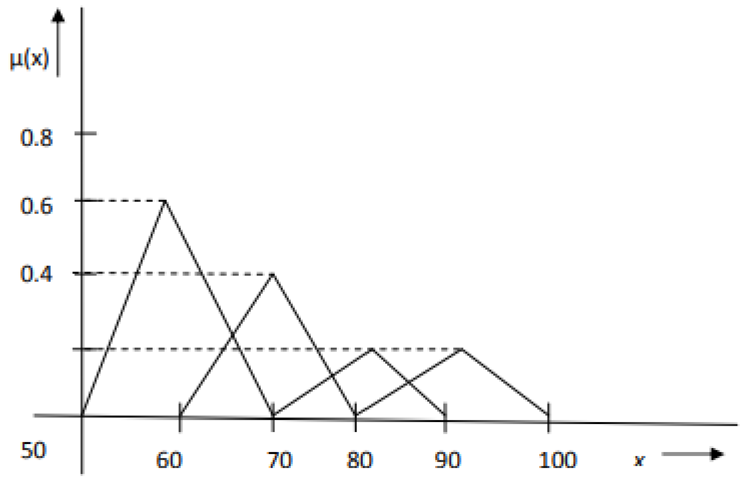


Figure 3: Fuzzy set A

Let A be a fuzzy set that tells about a student as shown in figure 3 and the elements with corresponding maximum membership values are also given. A = {(P, 0.6), (F, 0.4),(G, 0.2),(VG, 0.2), (E, 0)}

Here, the linguistic variable P represents a Pass student, F stands for a Fair student, G represents a Good student, VG represents a Very Good student and E for an Excellent student.

Now the defuzzified value **x** $*$for set A will be

$$x^* = \frac{(60*0.6+70*0.4+80*0.2+90*0.2+100*0)}{0.6+0.4+0.2+0.2+0}$$

$= 98/1.4=70$

The defuzzified value for the fuzzy set A with weighted average method represents a Fair student.

Maxima Methods

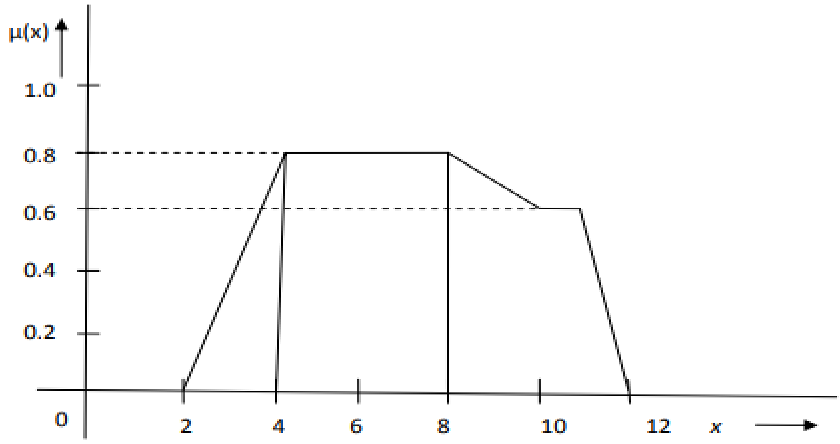
This method considers values with maximum membership.

There are different maxima methods with different conflict resolution strategies for multiple maxima.

- First of Maxima Method (FOM)
- Last of Maxima Method (LOM)
- Mean of Maxima Method (MOM)

First of Maxima Method (FOM)

This method determines the smallest value of the domain with maximum membership value. Example: The defuzzified value x*of the given fuzzy set will be x*=4.



Last of Maxima Method (LOM)

Determine the largest value of the domain with maximum membership value.

In the example given for FOM, the defuzzified value for LOM method will be x*=8

Mean of Maxima Method (MOM)

In this method, the defuzzified value is taken as the element with the highest membership values. When there are more than one element having maximum membership values, the mean value of the maxima is taken.

Let A be a fuzzy set with membership function $\mu_A(x)$ defined over $x \in X$, where X is a universe of discourse.

The defuzzified value is let say **x*** of a fuzzy set and is defined as,

$$x^* = \frac{\sum_{x_i \in M} x_i}{|M|},$$

Here, $M = \{x_i | \ \mu_A(x_i)$ is equal to the height of the fuzzy set A$\}$ and $|M|$ is the cardinality of the set M.

Example

In the example as shown in Fig. , x = 4, 6, 8 have maximum membership values and hence |M| = 3

According to MOM method, $$x^* = \frac{\sum_{x_i \in M} x_i}{|M|}$$

Now the defuzzified value x^* will be $x^* = \frac{4+6+8}{3} = \frac{18}{3} = 6.$

Fuzzy Logic - Traditional Fuzzy Refresher

Logic, which was originally just the study of what distinguishes sound argument from unsound argument, has now developed into a powerful and rigorous system whereby true statements can be discovered, given other statements that are already known to be true.

Predicate Logic

This logic deals with predicates, which are propositions containing variables.

A predicate is an expression of one or more variables defined on some specific domain. A predicate with variables can be made a proposition by either assigning a value to the variable or by quantifying the variable.
Following are a few examples of predicates −

- Let E(x, y) denote "x = y"
- Let X(a, b, c) denote "a + b + c = 0"
- Let M(x, y) denote "x is married to y"

Propositional Logic

A proposition is a collection of declarative statements that have either a truth value "true" or a truth value "false". A propositional consists of propositional variables and connectives. The propositional variables are dented by capital letters (A, B, etc). The connectives connect the propositional variables.
A few examples of Propositions are given below −

- "Man is Mortal", it returns truth value "TRUE"
- "12 + 9 = 3 – 2", it returns truth value "FALSE"

The following is not a Proposition −

- **"A is less than 2"** − It is because unless we give a specific value of A, we cannot say whether the statement is true or false.

Connectives

In propositional logic, we use the following five connectives −

- OR (∨)
- AND (∧)
- Negation/ NOT (¬)
- Implication / if-then (→)
- If and only if (⇔)

OR (V)

The OR operation of two propositions A and B (written as A∨B) is true if at least any of the propositional variable A or B is true.

The truth table is as follows –

A	B	A V B
TRUE	TRUE	TRUE
TRUE	FALSE	TRUE
FALSE	TRUE	TRUE
FALSE	FALSE	FALSE

AND (∧)

The AND operation of two propositions A and B (written as A∧B) is true if both the propositional variable A and B is true.

The truth table is as follows –

A	B	A ∧ B
TRUE	TRUE	TRUE
TRUE	FALSE	FALSE
FALSE	TRUE	FALSE
FALSE	FALSE	FALSE

Negation (¬)

The negation of a proposition A (written as ¬A) is false when A is true and is true when A is false.

The truth table is as follows −

A	¬A
TRUE	FALSE
FALSE	TRUE

Implication / if-then (→)

An implication A→B is the proposition “if A, then B”. It is false if A is true and B is false. The rest cases are true. The truth table is as follows −

A	B	A→B
TRUE	TRUE	TRUE
TRUE	FALSE	FALSE
FALSE	TRUE	TRUE
FALSE	FALSE	TRUE

If and only if (⇔)

A⇔B is a bi-conditional logical connective which is true when p and q are same, i.e., both are false or both are true. The truth table is as follows −

A	B	A⇔B
TRUE	TRUE	TRUE
TRUE	FALSE	FALSE
FALSE	TRUE	FALSE
FALSE	FALSE	TRUE

Well Formed Formula

Well Formed Formula (wff) is a predicate holding one of the following −

- All propositional constants and propositional variables are wffs.
- If x is a variable and Y is a wff, ∀xY and ∃xY are also wff.
- Truth value and false values are wffs.
- Each atomic formula is a wff.
- All connectives connecting wffs are wffs.

Quantifiers

The variable of predicates is quantified by quantifiers. There are two types of quantifier in predicate logic −

- Universal Quantifier
- Existential Quantifier

Universal Quantifier

Universal quantifier states that the statements within its scope are true for every value of the specific variable. It is denoted by the symbol ∀.

∀xP(x) is read as for every value of x, P(x) is true.

Example − "Man is mortal" can be transformed into the propositional form **∀xP(x)**. Here, P(x) is the predicate which denotes that x is mortal and the universe of discourse is all men.

Existential Quantifier

Existential quantifier states that the statements within its scope are true for some values of the specific variable. It is denoted by the symbol **∃.**

∃xP(x) for some values of x is read as, P(x) is true.

Example – "Some people are dishonest" can be transformed into the propositional form $\exists x P(x)$ where P(x) is the predicate which denotes x is dishonest and the universe of discourse is some people.

Nested Quantifiers

If we use a quantifier that appears within the scope of another quantifier, it is called a nested quantifier.

Example

$\forall a \exists b P(x,y)$ where P(a,b) denotes a+b = 0

$\forall a \forall b \forall c P(a,b,c)$ where P(a,b) denotes a+(b+c) = (a+b)+c

Note – $\forall a \exists b P(x,y) \neq \exists a \forall b P(x,y)$

Approximate Reasoning

Following are the different modes of approximate reasoning –

Categorical Reasoning

In this mode of approximate reasoning, the antecedents, containing no fuzzy quantifiers and fuzzy probabilities, are assumed to be in canonical form.

Qualitative Reasoning

In this mode of approximate reasoning, the antecedents and consequents have fuzzy linguistic variables; the input-output relationship of a system is expressed as a collection of fuzzy IF-THEN rules. This reasoning is mainly used in control system analysis.

Syllogistic Reasoning

In this mode of approximation reasoning, antecedents with fuzzy quantifiers are related to inference rules. This is expressed as –

x = S1A's are B's
y = S2C's are D's
----
z = S3E's are F's

Here A,B,C,D,E,F are fuzzy predicates.

S1 and S2 are given fuzzy quantifiers.

S3 is the fuzzy quantifier which has to be decided.

Dispositional Reasoning

In this mode of approximation reasoning, the antecedents are dispositions that may contain the fuzzy quantifier "usually". The quantifier Usually links together the dispositional and syllogistic reasoning; hence it pays an important role.

For example, the projection rule of inference in dispositional reasoning can be given as follows –

usually((L,M) is R) ⇒usually (L is [R ↓ L])

Here **[R ↓ L]** is the projection of fuzzy relation R on L

Fuzzy Logic Rule Base

It is a known fact that a human being is always comfortable making conversations in natural language. The representation of human knowledge can be done with the help of following natural language expression –

IF antecedent THEN consequent

The expression as stated above is referred to as the Fuzzy IF-THEN rule base.

Canonical Form

Following is the canonical form of Fuzzy Logic Rule Base –

Rule 1 – If condition C1, then restriction R1

Rule 2 – If condition C1, then restriction R2

.

.

Rule n – If condition C1, then restriction Rn

Interpretations of Fuzzy IF-THEN Rules
Fuzzy IF-THEN Rules can be interpreted in the following four forms −

Assignment Statements
These kinds of statements use "=" (equal to sign) for the purpose of assignment. They are of the following form −

a = hello
climate = summer

Conditional Statements
These kinds of statements use the "IF-THEN" rule base form for the purpose of condition. They are of the following form −

IF temperature is high THEN Climate is hot
IF food is fresh THEN eat.

Unconditional Statements
They are of the following form −

GOTO 10
turn the Fan off

Linguistic Variable

Fuzzy logic uses linguistic variables which are the words or sentences in a natural language. For example, if we say temperature, it is a linguistic variable; the values of which are very hot or cold, slightly hot or cold, very warm, slightly warm, etc. The words very, slightly are the linguistic hedges.

Characterization of Linguistic Variable
Following four terms characterize the linguistic variable −

- Name of the variable, generally represented by x.
- Term set of the variable, generally represented by t(x).
- Syntactic rules for generating the values of the variable x.
- Semantic rules for linking every value of x and its significance.

Propositions in Fuzzy Logic
As we know that propositions are sentences expressed in any language which are generally expressed in the following canonical form − **s as P**
Here, s is the Subject and P is Predicate.

For example, *"Delhi is the capital of India"*, this is a proposition where "Delhi" is the subject and "is the capital of India" is the predicate which shows the property of subject.

Logic is the basis of reasoning and fuzzy logic extends the capability of reasoning by using fuzzy predicates, fuzzy-predicate modifiers, fuzzy quantifiers and fuzzy qualifiers in fuzzy propositions which creates the difference from classical logic.

Propositions in fuzzy logic include the following −
Fuzzy Predicate
Almost every predicate in natural language is fuzzy in nature hence, fuzzy logic has the predicates like tall, short, warm, hot, fast, etc.

Fuzzy-predicate Modifiers
We discussed linguistic hedges above; we also have many fuzzy-predicate modifiers which act as hedges. They are very essential for producing the values of a linguistic variable. For example, the words very, slightly are modifiers and the propositions can be like "water is slightly hot."

Fuzzy Quantifiers
It can be defined as a fuzzy number which gives a vague classification of the cardinality of one or more fuzzy or non-fuzzy sets. It can be used to influence probability within fuzzy logic. For example, the words many, most, frequently are used as fuzzy quantifiers and the propositions can be like "most people are allergic to it."

Fuzzy Qualifiers
Let us now understand Fuzzy Qualifiers. A Fuzzy Qualifier is also a proposition of Fuzzy Logic. Fuzzy qualification has the following forms −
Fuzzy Qualification Based on Truth
It claims the degree of truth of a fuzzy proposition.
Expression − It is expressed as x is t. Here, t is a fuzzy truth value.
Example − (Car is black) is NOT VERY True.

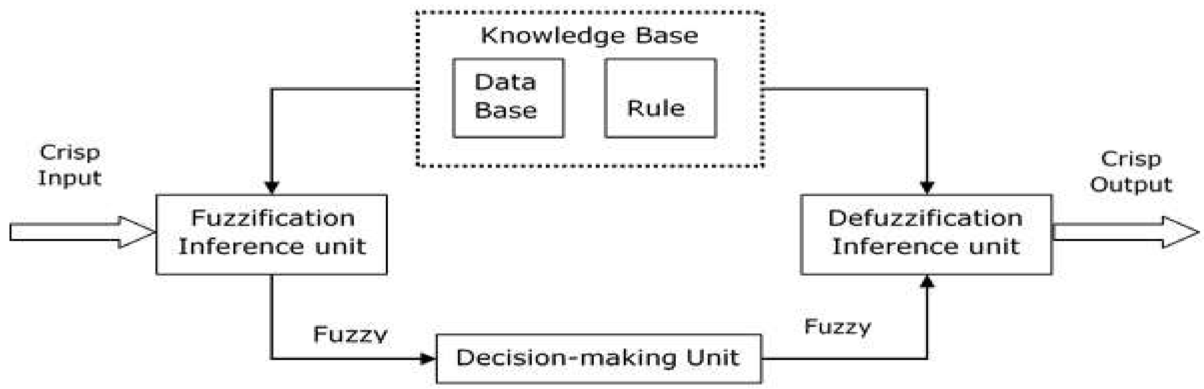
Fuzzy Qualification Based on Probability
It claims the probability, either numerical or an interval, of fuzzy proposition.
Expression − It is expressed as x is λ. Here, λ is a fuzzy probability.
Example − (Car is black) is Likely.

Fuzzy Qualification Based on Possibility
It claims the possibility of fuzzy proposition.
Expression − It is expressed as x is π. Here, π is a fuzzy possibility.
Example − (Car is black) is Almost Impossible.

Fuzzy Logic - Inference System
Fuzzy Inference System is the key unit of a fuzzy logic system having decision making as its primary work. It uses the "IF…THEN" rules along with connectors "OR" or "AND" for drawing essential decision rules.

Characteristics of Fuzzy Inference System
Following are some characteristics of FIS −
- The output from FIS is always a fuzzy set irrespective of its input which can be fuzzy or crisp.
- It is necessary to have fuzzy output when it is used as a controller.
- A defuzzification unit would be there with FIS to convert fuzzy variables into crisp variables.

Functional Blocks of FIS
The following five functional blocks will help you understand the construction of FIS −
- Rule Base − It contains fuzzy IF-THEN rules.
- Database − It defines the membership functions of fuzzy sets used in fuzzy rules.
- Decision-making Unit − It performs operation on rules.
- Fuzzification Interface Unit − It converts the crisp quantities into fuzzy quantities.
- Defuzzification Interface Unit − It converts the fuzzy quantities into crisp quantities. Following is a block diagram of fuzzy interference system.



Working of FIS

The working of the FIS consists of the following steps −

- A fuzzification unit supports the application of numerous fuzzification methods, and converts the crisp input into fuzzy input.
- A knowledge base - collection of rule base and database is formed upon the conversion of crisp input into fuzzy input.
- The defuzzification unit fuzzy input is finally converted into crisp output.

Methods of FIS

Let us now discuss the different methods of FIS. Following are the two important methods of FIS, having different consequent of fuzzy rules −

- **Mamdani Fuzzy Inference System**
- **Takagi-Sugeno Fuzzy Model (TS Method)**

Mamdani Fuzzy Inference System

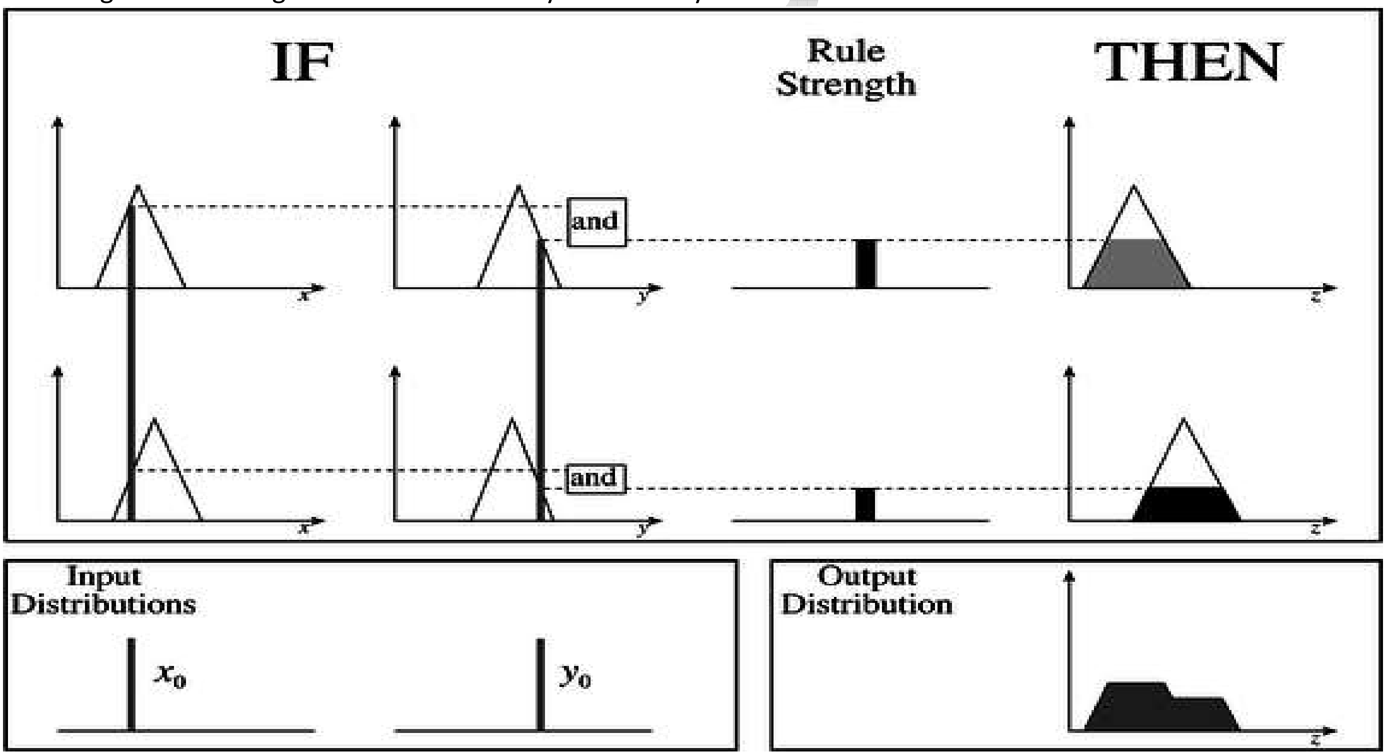
This system was proposed in 1975 by Ebhasim Mamdani. Basically, it was anticipated to control a steam engine and boiler combination by synthesizing a set of fuzzy rules obtained from people working on the system.

Steps for Computing the Output

Following steps need to be followed to compute the output from this FIS −

1. determining a set of fuzzy rules
2. fuzzifying the inputs using the input membership functions,
3. combining the fuzzified inputs according to the fuzzy rules to establish a rule strength,
4. finding the consequence of the rule by combining the rule strength and the output membership function,
5. combining the consequences to get an output distribution, and
6. defuzzifying the output distribution (this step is only if a crisp output (class) is needed).

Following is a block diagram of Mamdani Fuzzy Interface System.



Takagi-Sugeno Fuzzy Model (TS Method)

This model was proposed by Takagi, Sugeno and Kang in 1985. Format of this rule is given as −

IF x is A and y is B THEN Z = f(x,y)

Here, AB are fuzzy sets in antecedents and z = f(x,y) is a crisp function in the consequent.

Fuzzy Inference Process

The fuzzy inference process under Takagi-Sugeno Fuzzy Model (TS Method) works in the following way −

- Step 1: Fuzzifying the inputs − Here, the inputs of the system are made fuzzy.
- Step 2: Applying the fuzzy operator − In this step, the fuzzy operators must be applied to get the output.

Rule Format of the Sugeno Form

The rule format of Sugeno form is given by − **if 7 = x and 9 = y then output is z = ax+by+c**

Comparison between the two methods

Let us now understand the comparison between the Mamdani System and the Sugeno Model.

- Output Membership Function − The main difference between them is on the basis of output membership function. The Sugeno output membership functions are either linear or constant.
- Aggregation and Defuzzification Procedure − The difference between them also lies in the consequence of fuzzy rules and due to the same their aggregation and defuzzification procedure also differs.
- Mathematical Rules − More mathematical rules exist for the Sugeno rule than the Mamdani rule.
- Adjustable Parameters − The Sugeno controller has more adjustable parameters than the Mamdani controller.

Fuzzy Logic - Database and Queries

Fuzzy Logic is an approach to computing based on "degrees of truth" rather than the usual "true or false" logic. It deals with reasoning that is approximate rather than precise to solve problems in a way that more resembles human logic, hence database querying process by the two valued realization of Boolean algebra is not adequate.

Fuzzy Scenario of Relations on Databases

The Fuzzy Scenario of Relations on Databases can be understood with the help of the following example −

Example

Suppose we have a database having the records of persons who visited India. In simple database, we will have the entries made in the following way −

Name	Age	Citizen	Visited Country	Days Spent	Year of Visit
John Smith	35	U.S.	India	41	1999
John Smith	35	U.S.	Italy	72	1999
John Smith	35	U.S.	Japan	31	1999

Now, if anyone queries about the person who visited India and Japan in the year 99 and is the citizen of US, then the output will show two entries having the name of John Smith. This is simple query generating simple output.

But what if we want to know whether the person in the above query is young or not. According to the above result, the age of the person is 35 years. But can we assume the person to be young or not? Similarly, same thing can be applied on the other fields like days spent, year of visit, etc.

The solution of the above issues can be found with the help of Fuzzy Value sets as follows −

- FV(Age){ very young, young, somewhat old, old }
- FV(Days Spent){ barely few days, few days, quite a few days, many days }
- FV(Year of Visit){distant past, recent past, recent }
- Now if any query will have the fuzzy value then the result will also be fuzzy in nature.

Fuzzy Query System
A fuzzy query system is an interface to users to get information from the database using (quasi) natural language sentences. Many fuzzy query implementations have been proposed, resulting in slightly different languages. Although there are some variations according to the particularities of different implementations, the answer to a fuzzy query sentence is generally a list of records, ranked by the degree of matching.

Fuzzy Logic – Quantification
In modeling natural language statements, quantified statements play an important role. It means that NL heavily depends on quantifying construction which often includes fuzzy concepts like “almost all”, “many”, etc. Following are a few examples of quantifying propositions −

- Every student passed the exam.
- Every sport car is expensive.
- Many students passed the exam.
- Many sports cars are expensive.

In the above examples, the quantifiers “Every” and “Many” are applied to the crisp restrictions “students” as well as crisp scope “(person who)passed the exam” and “cars” as well as crisp scope ”sports”.

Fuzzy Events, Fuzzy Means and Fuzzy Variances
With the help of an example, we can understand the above concepts. Let us assume that we are a shareholder of a company named ABC. And at present the company is selling each of its share for ₹40. There are three different companies whose business is similar to ABC but these are offering their shares at different rates - ₹100 a share, ₹85 a share and ₹60 a share respectively.

Now the probability distribution of this price takeover is as follows −

Price	100	85	60
Probability	0.3	0.5	0.2

Now, from the standard probability theory, the above distribution gives a mean of expected price as below −
100×0.3+85×0.5+60×0.2=84.5
And, from the standard probability theory, the above distribution gives a variance of expected price as below −
(100−84.5)2×0.3+(85−84.5)2×0.5+(60−84.5)2×0.2=124.825

Suppose the degree of membership of 100 in this set is 0.7, that of 85 is 1, and the degree of membership is 0.5 for the value 60. These can be reflected in the following fuzzy set −

$$\left\{ \frac{0.7}{100}, \frac{1}{85}, \frac{0.5}{60}, \right\}$$

The fuzzy set obtained in this manner is called a fuzzy event.
We want the probability of the fuzzy event for which our calculation gives −

0.7×0.3+1×0.5+0.5×0.2=0.21+0.5+0.1=0.81

Now, we need to calculate the fuzzy mean and the fuzzy variance, the calculation is as follows −

Fuzzy_mean =(1/0.81)×(100×0.7×0.3+85×1×0.5+60×0.5×0.2=85.8
Fuzzy_Variance =7496.91−7361.91=135.27

Fuzzy Logic - Decision Making
It is an activity which includes the steps to be taken for choosing a suitable alternative from those that are needed for realizing a certain goal.

Steps for Decision Making

Let us now discuss the steps involved in the decision making process –

- Determining the Set of Alternatives – In this step, the alternatives from which the decision has to be taken must be determined.
- Evaluating Alternative – Here, the alternatives must be evaluated so that the decision can be taken about one of the alternatives.
- Comparison between Alternatives – In this step, a comparison between the evaluated alternatives is done.

Types of Decision

Making We will now understand the different types of decision making.

Individual Decision Making

In this type of decision making, only a single person is responsible for taking decisions. The decision making model in this kind can be characterized as –

- Set of possible actions
- Set of goals $G_i\,(i\,\in\,X_n)\,;$
- Set of Constraints $C_j\,(j\,\in\,X_m)$

The goals and constraints stated above are expressed in terms of fuzzy sets.

Now consider a set A. Then, the goal and constraints for this set are given by –

$G_i\,(a)$ = composition $[G_i\,(a)]$ = $G_i^1\,(G_i\,(a))$ with G_i^1

$C_j\,(a)$ = composition $[C_j\,(a)]$ = $C_j^1\,(C_j\,(a))$ with C_j^1 for $a\,\in\,A$

The fuzzy decision in the above case is given by –

$$F_D = min[i \in X_n^{in}\, f G_i\,(a)\,, j \in X_m^{in}\, f C_j\,(a)]$$

Multi-person Decision Making

Decision making in this case includes several persons so that the expert knowledge from various persons is utilized to make decisions.

Calculation for this can be given as follows –

Number of persons preferring x_i to x_j = $N\,(x_i,\,x_j)$

Total number of decision makers = **n**

Then, $SC\,(x_i,\,x_j) = \frac{N(x_i,\,x_j)}{n}$

Multi-objective Decision Making

Multi-objective decision making occurs when there are several objectives to be realized. There are following two issues in this type of decision making –

- To acquire proper information related to the satisfaction of the objectives by various alternatives.
- To weigh the relative importance of each objective.

Mathematically we can define a universe of n alternatives as –

A=[a1,a2,...,ai,...,an]A=[a1,a2,...,ai,...,an]

And the set of “m” objectives as O=[o1,o2,...,oi,...,on]O=[o1,o2,...,oi,...,on]

Multi-attribute Decision Making

Multi-attribute decision making takes place when the evaluation of alternatives can be carried out based on several attributes of the object. The attributes can be numerical data, linguistic data and qualitative data.
Mathematically, the multi-attribute evaluation is carried out on the basis of linear equation as follows -

$$Y=A1X1+A2X2+...+AiXi+...+ArXr$$

Fuzzy Logic - Control System

Fuzzy logic is applied with great success in various control application. Almost all the consumer products have fuzzy control. Some of the examples include controlling your room temperature with the help of air-conditioner, anti-braking system used in vehicles, control on traffic lights, washing machines, large economic systems, etc.

Why Use Fuzzy Logic in Control Systems

A control system is an arrangement of physical components designed to alter another physical system so that this system exhibits certain desired characteristics. Following are some reasons of using Fuzzy Logic in Control Systems −

- While applying traditional control, one needs to know about the model and the objective function formulated in precise terms. This makes it very difficult to apply in many cases.
- By applying fuzzy logic for control we can utilize the human expertise and experience for designing a controller.
- The fuzzy control rules, basically the IF-THEN rules, can be best utilized in designing a controller.

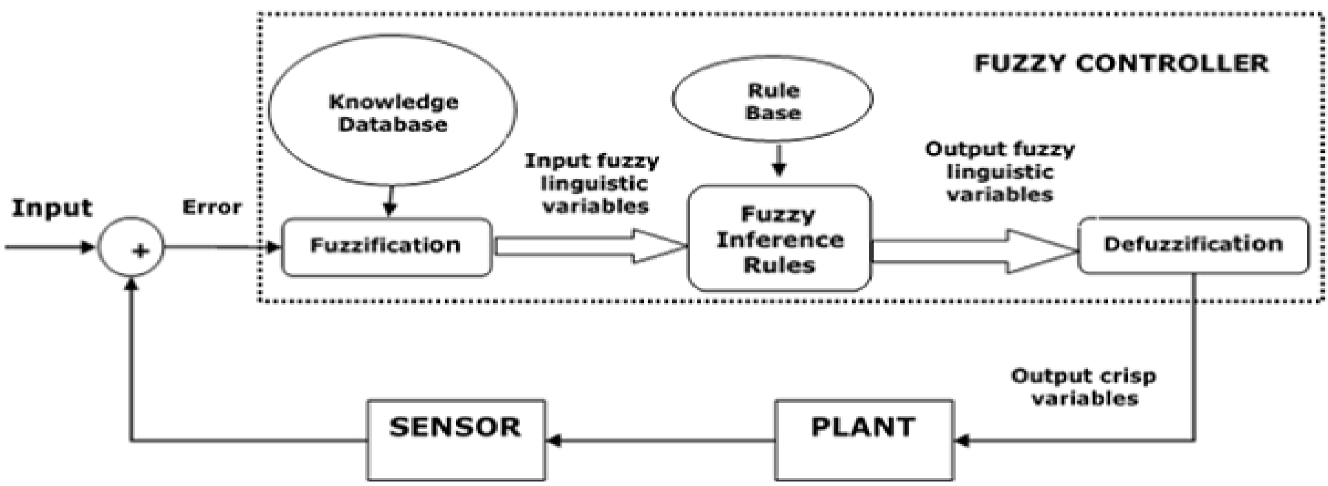
Assumptions in Fuzzy Logic Control (FLC) Design

While designing fuzzy control system, the following six basic assumptions should be made −

- The plant is observable and controllable − It must be assumed that the input, output as well as state variables are available for observation and controlling purpose.
- Existence of a knowledge body − It must be assumed that there exist a knowledge body having linguistic rules and a set of input-output data set from which rules can be extracted.
- Existence of solution − It must be assumed that there exists a solution.
- ‘Good enough’ solution is enough − The control engineering must look for ‘good enough’ solution rather than an optimum one.
- Range of precision − Fuzzy logic controller must be designed within an acceptable range of precision.
- Issues regarding stability and optimality − The issues of stability and optimality must be open in designing Fuzzy logic controller rather than addressed explicitly.

Architecture of Fuzzy Logic Control

The following diagram shows the architecture of Fuzzy Logic Control (FLC).



Major Components of FLC

Followings are the major components of the FLC as shown in the above figure −

- Fuzzifier − The role of fuzzifier is to convert the crisp input values into fuzzy values.
- Fuzzy Knowledge Base − It stores the knowledge about all the input-output fuzzy relationships. It also has the membership function which defines the input variables to the fuzzy rule base and the output variables to the plant under control.
- Fuzzy Rule Base − It stores the knowledge about the operation of the process of domain.
- Inference Engine − It acts as a kernel of any FLC. Basically it simulates human decisions by performing approximate reasoning.
- Defuzzifier − The role of defuzzifier is to convert the fuzzy values into crisp values getting from fuzzy inference engine.

Steps in Designing FLC

Following are the steps involved in designing FLC −

- Identification of variables − Here, the input, output and state variables must be identified of the plant which is under consideration.
- Fuzzy subset configuration − The universe of information is divided into number of fuzzy subsets and each subset is assigned a linguistic label. Always make sure that these fuzzy subsets include all the elements of universe.
- Obtaining membership function − Now obtain the membership function for each fuzzy subset that we get in the above step.
- Fuzzy rule base configuration − Now formulate the fuzzy rule base by assigning relationship between fuzzy input and output.
- Fuzzification − The fuzzification process is initiated in this step.
- Combining fuzzy outputs − By applying fuzzy approximate reasoning, locate the fuzzy output and merge them.
- Defuzzification − Finally, initiate defuzzification process to form a crisp output.

Advantages of Fuzzy Logic Control

Let us now discuss the advantages of Fuzzy Logic Control.

- Cheaper − Developing a FLC is comparatively cheaper than developing model based or other controller in terms of performance.
- Robust − FLCs are more robust than PID controllers because of their capability to cover a huge range of operating conditions.
- Customizable − FLCs are customizable.
- Emulate human deductive thinking − Basically FLC is designed to emulate human deductive thinking, the process people use to infer conclusion from what they know.
- Reliability − FLC is more reliable than conventional control system.
- Efficiency − Fuzzy logic provides more efficiency when applied in control system.

Disadvantages of Fuzzy Logic Control

We will now discuss what are the disadvantages of Fuzzy Logic Control.

- Requires lots of data − FLC needs lots of data to be applied.
- Useful in case of moderate historical data − FLC is not useful for programs much smaller or larger than historical data.
- Needs high human expertise − This is one drawback as the accuracy of the system depends on the knowledge and expertise of human beings.
- Needs regular updating of rules − The rules must be updated with time.

Adaptive Fuzzy Controller

Adaptive Fuzzy Controller and how it works. Adaptive Fuzzy Controller is designed with some adjustable parameters along with an embedded mechanism for adjusting them. Adaptive controller has been used for improving the performance of controller.

Basic Steps for Implementing Adaptive Algorithm

Let us now discuss the basic steps for implementing adaptive algorithm.

- **Collection of observable data** – The observable data is collected to calculate the performance of controller.
- **Adjustment of controller parameters** – Now with the help of controller performance, calculation of adjustment of controller parameters would be done.
- **Improvement in performance of controller** – In this step, the controller parameters are adjusted to improve the performance of controller.

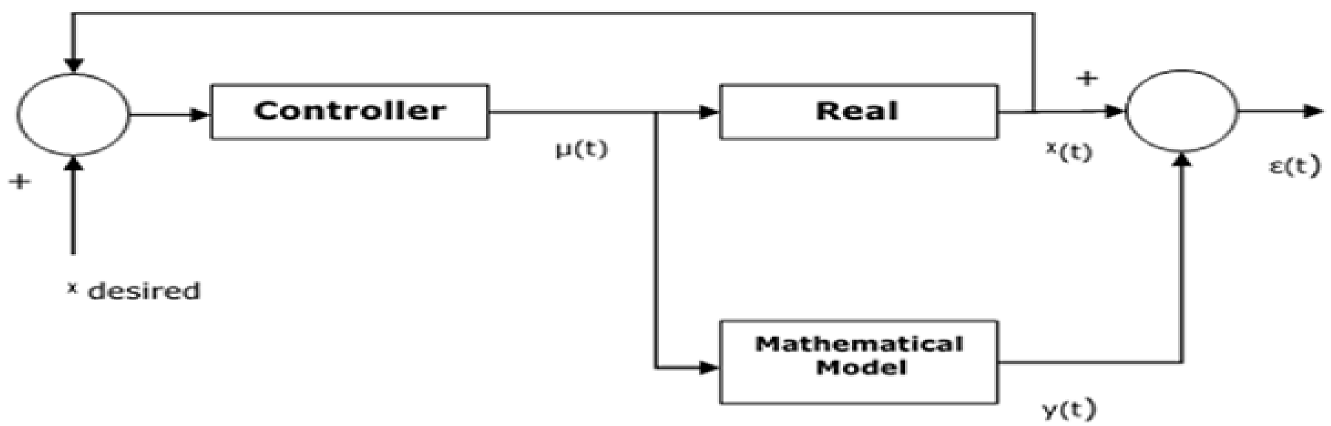
Operational Concepts

Design of a controller is based on an assumed mathematical model that resembles a real system. The error between actual system and its mathematical representation is calculated and if it is relatively insignificant than the model is assumed to work effectively.

A threshold constant that sets a boundary for the effectiveness of a controller, also exists. The control input is fed into both the real system and mathematical model. Here, assume **x(t)** is the output of the real system and **y(t)** is the output of the mathematical model. Then the error **ϵ(t)** can be calculated as follows – **$\epsilon(t)=x(t)-y(t)$**

Here, **x** desired is the output we want from the system and **μ(t)** is the output coming from controller and going to both real as well as mathematical model.

The following diagram shows how the error function is tracked between output of a real system and Mathematical model –



Parameterization of System

A fuzzy controller the design of which is based on the fuzzy mathematical model will have the following form of fuzzy rules –

Rule 1 – IF $x_1(t_n) \in X_{11}\ AND \ldots AND\ x_i(t_n) \in X_{1i}$

THEN $\mu_1(t_n) = K_{11}x_1(t_n) + K_{12}x_2(t_n)\ + \ldots + K_{1i}x_i(t_n)$

Rule 2 – IF $x_1(t_n) \in X_{21}\ AND \ldots AND\ x_i(t_n) \in X_{2i}$

THEN $\mu_2(t_n) = K_{21}x_1(t_n) + K_{22}x_2(t_n)\ + \ldots + K_{2i}x_i(t_n)$

Rule j – IF $x_1(t_n) \in X_{k1}\ AND \ldots AND\ x_i(t_n) \in X_{ki}$

THEN $\mu_j(t_n) = K_{j1}x_1(t_n) + K_{j2}x_2(t_n) + \ldots + K_{ji}x_i(t_n)$

The above set of parameters characterizes the controller.

Mechanism Adjustment

The controller parameters are adjusted to improve the performance of controller. The process of calculating the adjustment to the parameters is the adjusting mechanism.

Mathematically, let **θ(n)** be a set of parameters to be adjusted at time **t=tn**. The adjustment can be the recalculation of the parameters, **θ(n)=Θ(D0,D1,...,Dn)**

Here **Dn** is the data collected at time **t=tn**.

Now this formulation is reformulated by the update of the parameter set based on its previous value as,

$$\theta^{(n)} = \phi(\theta^{n-1},\ D_n)$$

Parameters for selecting an Adaptive Fuzzy Controller

The following parameters need to be considered for selecting an adaptive fuzzy controller −

- Can the system be approximated entirely by a fuzzy model?
- If a system can be approximated entirely by a fuzzy model, are the parameters of this fuzzy model readily available or must they be determined online?
- If a system cannot be approximated entirely by a fuzzy model, can it be approximated piecewise by a set of fuzzy model?
- If a system can be approximated by a set of fuzzy models, are these models having the same format with different parameters or are they having different formats?
- If a system can be approximated by a set of fuzzy models having the same format, each with a different set of parameters, are these parameter sets readily available or must they be determined online?

PID Control with Fuzzy Logic

Proportional integral derivative (PID) control is a well established way of driving a system towards a target position or level. It's practically ubiquitous as a means of controlling temperature, and finds application in myriad chemical and scientific processes as well as automation. PID control is however not without problems. It can yield less than ideal results in situations where the target value changes, whether as a step function or as part of a "ramp & soak" profile.

In an effort to improve performance, some instrumentation manufacturers are exploring the value of using "fuzzy logic" for process control. This OMEGA Engineering White Paper explores both the weaknesses of PID systems and the potential benefits of fuzzy logic, with particular reference to issues in temperature control. Individual sections address:

- Basics of PID control loops
- PID challenges
- Introduction to fuzzy logic for control
- PID plus adaptive fuzzy logic
- Applications

Basics of PID Control Loops
In the most basic form of control loop feedback, an output such as temperature is measured and compared to a target value. Based on the difference between these values a correction factor is calculated and applied to the input. If an oven is cooler than required, the heat will be increased. In proportional control, (the first term in the PID acronym) the correction factor is proportional to the difference. In consequence, the target value is never achieved because as the difference approaches zero, so too does the applied correction.

PID Challenges
The mathematics in a PID control equation is complex with multiple variables and constants interacting. In any given application these are selected to follow the target value as closely as possible, within the constraints imposed by the process itself and the instrumentation.

Three issues common to almost every process control application are:
- Time delays or lag
- Step function response
- “Ramp & Soak” function response

In many situations the output can take a long, and perhaps also variable, time to react to input changes. To give one example, a furnace will cool when “charged” with new metal and could take several minutes to come back up to temperature. This can lead to temperature overshoots which may damage the contents. Alternatively, the heating may be too slow, reducing process efficiency and causing deleterious effects to the product or material.

When the target value changes instantaneously PID forces the system to apply a large correcting factor, which again can lead to overshoot. Alternatively, the system may become saturated, unable to supply sufficient correction, adding to the impact of the “I” term.

These problems also occur in “ramp & soak” situations where temperature is increased gradually then held. Tracking a gradual change in setpoint can challenge PID control systems.

Introduction to Fuzzy Logic for Control
Conventional computing is based on Boolean logic, meaning everything is represented as either zero or one. In some situations this leads to oversimplification and inadequate results. Fuzzy logic, and by extension, fuzzy control, seeks to deal with complexity by creating heuristics that align more closely with human perception of problems.

Fuzzy logic provides a way of dealing with imprecision and nonlinearity in complex control situations. Inputs are passed to an “inference engine” where human or experienced-based rules are applied to produce an output.

PID Plus Adaptive Fuzzy Logic
Tuning of PID loops depends on heuristics yet often ends up being sub-optimal. Fuzzy logic provides an alternative to approaches such as Ziegler Nichols, and a growing body of research suggests it yields superior results. Thus it would seem an ideal way to control many complex processes is with a PID controller tuned with fuzzy logic.

One commercially available product incorporating such an approach is the OMEGA® Platinum Series of temperature and process controllers. This family of compact microprocessorbased PID controllers, available in three DIN sizes, is designed to be easy to set-up and use. All common thermocouples and

RTDs can be connected, with the system automatically enabling only the relevant functions for the input type selected. Voltage and current inputs are also available, allowing use with almost any engineering units. These controllers provide a complete PID solution, supporting complex programs with up to 16 Ramp & Soak sequences. Auto tune is available for PID applications with adaptive fuzzy logic to help attain optimal results.

Applications

Unless open-loop control is acceptable, almost every process control application would benefit from PID control. In terms of temperature control, good examples are:

- Heat treatment of metals. "Ramp & Soak" sequences need precise control to ensure desired metallurgical properties are achieved.
- Drying/evaporating solvents from painted surfaces. Over-temperature conditions can damage substrates while low temperatures can result in product damage and poor appearance.
- Curing rubber. Precise temperature control ensures complete cure is achieved without adversely affecting material properties.
- Baking. Commercial ovens must follow tightly prescribed heating and cooling sequences to ensure the necessary reactions take place.
- Ceramics. Continuous kilns must deliver high levels of heat yet are subject to varying thermal loads. This makes them an ideal application for PID control.

$$u(t) = K_p e(t) + K_i \int_0^t e(x)\,dx + K_d \frac{de(t)}{dt}$$

Where K_P — proportional gain, a tuning parameter

K_i — integral gain, a tuning parameter

K_d — derivative gain, a tuning parameter

e — error present in the controller

t — time or instantaneous time

x — variable of integration, taken from time 0 to present 1

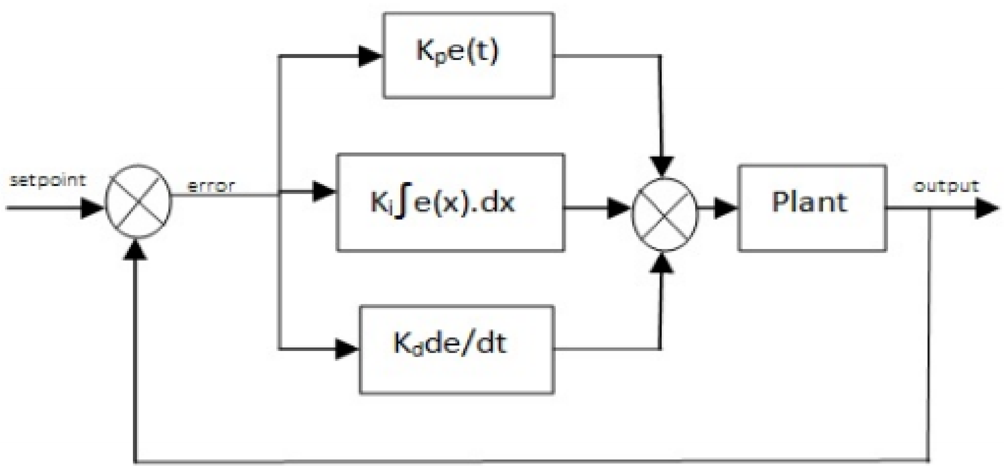


Fig 1:Block diagram of conventional PID controller

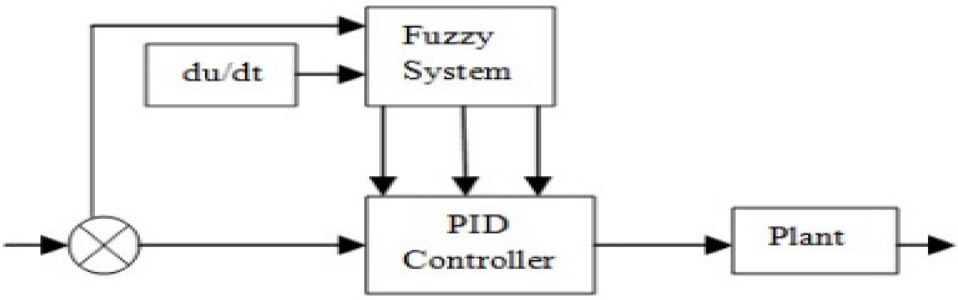


Fig.4 Block diagram of fuzzy PID controller

Fuzzy Controlled Anti-lock Braking System

Stopping a car in a hurry on a slippery road can be very challenging. **Anti-lock braking systems (ABS)** take a lot of the challenge out of this sometimes nerve-wracking event. In fact, on slippery surfaces, even professional drivers can't stop as quickly without ABS as an average driver can with ABS. Anti-lock Brake improves the controllability of vehicles in compare with brake systems lacking ABS. Fuzzy are a multi-valued logic developed to deal with imprecise or vague data. Classical logic holds that everything can be expressed in binary terms: 0 or 1; in terms of Boolean algebra, everything is in one set or another but not in both. Fuzzy logic allows for partial membership in a set, values between 0 and 1. When the approximate reasoning of fuzzy logic is used with an expert system, logical inferences can be drawn from imprecise relationships.

Fuzzy anti-lock Braking systems were developed to reduce skidding and maintain steering control when brakes are used in an emergency situation. Fuzzy controllers are potential candidates for the control of non-linear, time variant and complicated systems. There are many control algorithms for ABS systems and they are partially responsible for their performance.

Fuzzy logic, a more generalized data set, allows for a “class” with continuous membership gradations. It is rigorously structured in mathematics. One advantage is the ability to describe systems linguistically through rule statements.

FUZZIFICATION

The fuzzy controller takes input values from real world. These values referred to as “crisp” values. Since they are represented as single number, not a fuzzy one. In order for the fuzzy controller to understand the input, the crisp input has to be converted to a fuzzy number. This process is called fuzzification.

DEFUZZIFICATION

It is the process of producing a quantifiable result in fuzzy logic. Typically, a fuzzy system will have a number of rules that transform a number of variables into a “Fuzzy” result, that is result is described in terms of membership in fuzzy set. Simplest but least useful defuzzification method is to choose the set with highest membership.Once all the rules are evaluated, their output are combined in order to provide a single value that will be fuzzified.

FUZZY CONTROL

A fuzzy control system is a real-time expert system, implementing a part of human operator’s which does not lend itself to being easily expressed in PID- parameters or differential equations but rather in situation/action rules. Fuzzy control has been so successful in areas where classical control theory has been dominant for so many years. It differs from classical control theory in several aspects. One main feature of fuzzy control system is their existence at two distinct levels: First, there are symbolic if- then rules and qualitative, fuzzy variables and values such as

“If pressure is high and slightly increasing then energy supply is medium negative”

Here ‘slightly increasing’ and ‘pressure is high’ are fuzzy values and ‘and’ is a fuzzy operator. The IF part is called the **"antecedent"** and the THEN part is called the **"consequent".**

Fuzzy control aims at replacing differential equation based techniques and solving the whole problem with artificial intelligence methods. One way to combine fuzzy and PID- control is to use a linear PID system around the set-point, where it does its job, and to ‘delinearize’ the system in other areas by describing the desired behavior or control strategy with fuzzy rules. Fuzzy controllers are very simple conceptually. They consist of-

- an input stage,
- a processing stage and
- an output stage

The input stage maps sensor or other inputs, such as switches, thumbwheels, and so on, to the appropriate membership functions and truth values. The processing stage invokes each appropriate rule and generates a result for each, then combines the results of the rules. Finally, the output stage converts the combined result back into a specific control output value.

Typical fuzzy control systems have dozens of rules. There are several different ways to define the result of a rule, but one of the most common and simplest is the "max-min" inference method, in which the output membership function is given the truth value generated by the premise. Rules can be solved in parallel in hardware, or sequentially in software. The results of all the rules that have fired are "defuzzified" to a crisp value by one of several methods. The "centroid" method is very popular, in which the "center of mass" of the result provides the crisp value. Another approach is the "height" method, which takes the value of the biggest contributor. The centroid method favors the rule with the output of greatest area, while the height method obviously favors the rule with the greatest output value.

Anti-Lock Braking System

Anti-Lock Braking System is a safety system on automobiles which prevents wheels from locking while braking. It is also known as CAB-Controller Anti- Lock Brake. ABS is implemented in automobiles to ensure optimal vehicle control and minimal stopping distances during hard or emergency braking. The theory behind anti- lock brakes is simple. A skidding wheel (where the tire contact patch is sliding relative to the road) has less traction than a non-skidding wheel. ABS are non-linear and dynamic in nature. ABS is now accepted as an essential contribution to vehicle safety. The methods of control utilized by ABS are responsible for system performance.

COMPONENTS OF ABS

There are four main components in ABS. They are

- Wheel speed sensors
- Electronic Controller Units (ECU’s)
- Hydraulic valves
- Pumps

The anti-lock braking system needs some way of knowing when a **wheel** is about to lock up. The wheel speed sensors, which are located at each wheel provides this information.

The **ECU** constantly monitors the rotational speed of each wheel and controls the valves. Wheel speed sensors transmit pulses to the ECU with a frequency proportional to wheel speed. The ECU then processes this information and regulates the brake accordingly.

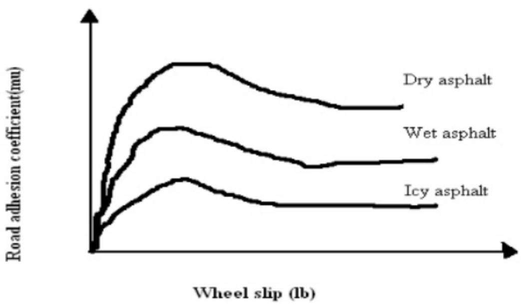
There is a **valve** in the brake line of each brake controlled by the ABS. On some systems, the valve has three positions:

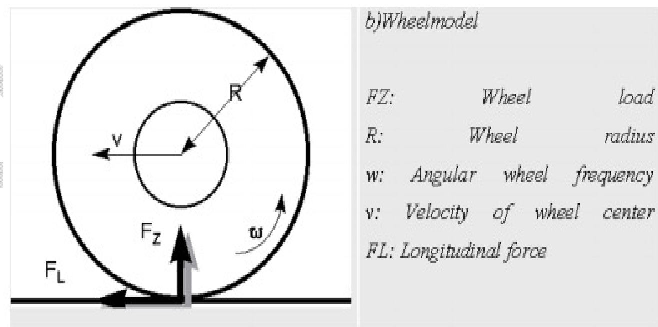
- In position one, the valve is open; pressure from the master cylinder is passed right through to the brake.
- In position two, the valve blocks the line, isolating that brake from the master cylinder. This prevents the pressure from rising further should driver push the brake pedal harder.
- In position three, the valve releases some of the pressure from the brake. Since the valve is able to release pressure from the brakes, there has to be some way to put that pressure back. So when a valve reduces the pressure in a line, the pump is there to get the pressure back up. ABS significantly improves safety and control for drivers in most on-road situations.

FUZZY CONTROL OF ABS

Fuzzy controllers are potential candidates for the control of non-linear, time variant and complicated systems. ABS which is a non-linear system may not be easily controlled by classical control methods. An intelligent fuzzy control method is very useful for this kind of non linear system. An intelligent fuzzy ABS controller can adjust the slipping performance for variety of roads. The main disadvantage of ordinary brakes is that a driver cannot precisely control the brake torque applied to the brake. Moreover, as the driver does not have enough information of road condition, he may cause locking of wheels by applying extra pressure on brake pedal. Anti-lock Brake improves the controllability of vehicles in compare with brake systems lacking ABS.

When a vehicle accelerates or brakes, the tractive force Ftf and Ftr developed by front and rear tyre, respectively are proportional to the normal forces of the road acting on the tyre. The coefficient of proportionality mu, is called road coefficient of adhesion and it varies depending on road surface type.





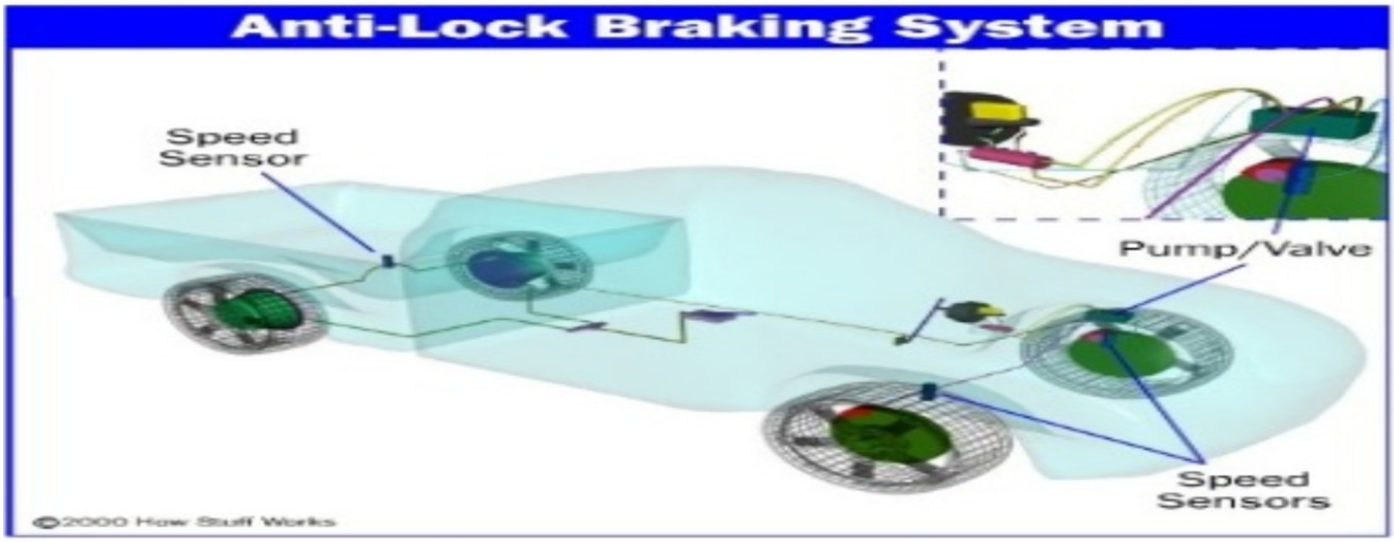
The wheel slip, denoted by lb, is the ratio of difference between the velocity of vehicle and the translational velocity of wheel to the velocity of vehicle. The goal of the ABS is to hold each tyre of the vehicle operating near the peak of mu-lb curve for that tyre, which implies the performance of ABS is strongly related to the surface condition.

WORKING OF FUZZY ABS

There are many different variations and control algorithms for ABS systems. Control algorithm is partially responsible for ABS performance. The controller monitors the speed sensors at all times. It is looking for decelerations in the wheel that are out of the ordinary. Right before a wheel locks up, it will experience a rapid deceleration. If left unchecked, the wheel would stop much more quickly than any car could. It might take a car five seconds to stop from 60 mph (96.6 kmph) under ideal conditions, but a wheel that locks up could stop spinning in less than a second. The ABS controller knows that such a rapid deceleration is impossible, so it reduces the pressure to that brake until it sees an acceleration, then it increases the pressure until it sees the deceleration again. It can do this very quickly, before the tyre can actually significantly change speed. The result is that the tyre slows down at the same rate as the car, with the brakes keeping the tyres very near the point at which they will start to lock up. This gives the system maximum braking power. Fuzzy ABS require more complex control constructs than simple “if-then” rules. In “if-then” rules, input variables marks directly to the output variables. It is possible to build a control with intermediate fuzzy variables. One such rule using fuzzy variables is as follows:

“If the rear wheels are turning slowly and a short time ago the vehicle speed was high then reduces rear brake pressure”

When the ABS system is in operation a pulsing in the brake pedal is experienced; this comes from the rapid opening and closing of the valves. Some ABS systems can cycle up to 15 times per second. The figure below describes the working of ABS.

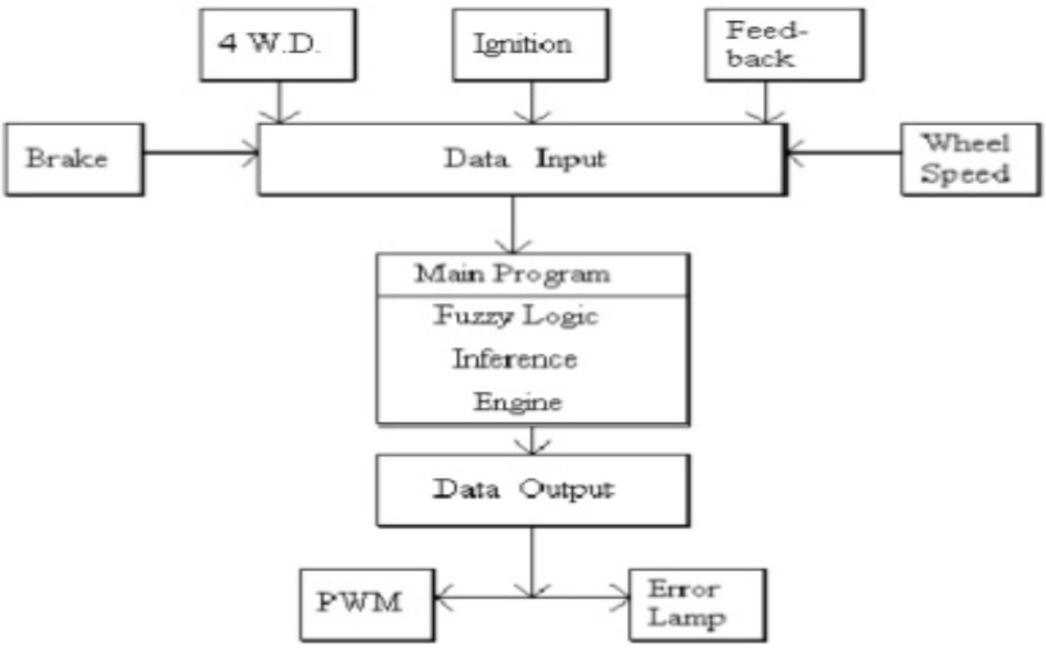


Electronic stability control

Modern Electronic Stability Control (ESC or ESP) systems are an evolution of the ABS concept. Here, a minimum of two additional sensors are added to help the system work: these are a steering wheel angle sensor, and a gyroscopic sensor. The theory of operation is simple: when the gyroscopic sensor detects that the direction taken by the car does not coincide with what the steering wheel sensor reports, the ESC software will brake the necessary individual wheel(s), so that the vehicle goes the way the driver intends. The rule governing ESC is

“If steering wheel sensor reports are not equal to gyroscopic sensor reports then brake the necessary individual wheel”

BLOCK DIAGRAM OF FUZZY ABS



Anti-Lock Brake Types
Anti-lock braking systems use different schemes depending on the type of brakes in use.

Four-channel, four-sensor ABS –
This is the best scheme. There is a speed sensor on all four wheels and a separate valve for all four wheels. With this setup, the controller monitors each wheel individually to make sure it is achieving maximum braking force.

Three-channel, three-sensor ABS –
This scheme, commonly found on pickup trucks with four-wheel ABS, has a speed sensor and a valve for each of the front wheels, with one valve and one sensor for both rear wheels. The speed sensor for the rear wheels is located in the rear axle. This system provides individual control of the front wheels, so they can both achieve maximum braking force.

One-channel, one-sensor ABS - This system is commonly found on pickup trucks with rear-wheel ABS. It has one valve, which controls both rear wheels, and one speed sensor, located in the rear axle.

Benefits of fuzzy ABS:
- Automobiles can be stopped faster and the distance of braking is reduced.
- Steering is possible while braking.

CONCLUSION
Fuzzy ABS has reduced the percentage of road crashes by a considerable amount. Experts predict that 35% to 50% of all cars built worldwide in five years will have ABS as standard equipment. Though it has many advantages, there are few disadvantages such as increased braking distance in snow/icy regions at present. These drawbacks can be overcome in future by fuzzy controlled ABS.

Fuzzy Logic - Applications
Aerospace
In aerospace, fuzzy logic is used in the following areas −
- Altitude control of spacecraft
- Satellite altitude control
- Flow and mixture regulation in aircraft deicing vehicles

Automotive
In automotive, fuzzy logic is used in the following areas −
- Trainable fuzzy systems for idle speed control
- Shift scheduling method for automatic transmission
- Intelligent highway systems
- Traffic control
- Improving efficiency of automatic transmissions

Business
In business, fuzzy logic is used in the following areas −
- Decision-making support systems
- Personnel evaluation in a large company

Defense
In defense, fuzzy logic is used in the following areas −
- Underwater target recognition

- Automatic target recognition of thermal infrared images
- Naval decision support aids
- Control of a hypervelocity interceptor
- Fuzzy set modeling of NATO decision making

Electronics

In electronics, fuzzy logic is used in the following areas −

- Control of automatic exposure in video cameras
- Humidity in a clean room
- Air conditioning systems
- Washing machine timing
- Microwave ovens
- Vacuum cleaners

Finance

In the finance field, fuzzy logic is used in the following areas −

- Banknote transfer control
- Fund management
- Stock market predictions

Industrial Sector

In industrial, fuzzy logic is used in following areas −

- Cement kiln controls heat exchanger control
- Activated sludge wastewater treatment process control
- Water purification plant control
- Quantitative pattern analysis for industrial quality assurance
- Control of constraint satisfaction problems in structural design
- Control of water purification plants

Manufacturing

In the manufacturing industry, fuzzy logic is used in following areas −

- Optimization of cheese production
- Optimization of milk production

Marine

In the marine field, fuzzy logic is used in the following areas −

- Autopilot for ships
- Optimal route selection
- Control of autonomous underwater vehicles
- Ship steering

Medical

In the medical field, fuzzy logic is used in the following areas −

- Medical diagnostic support system
- Control of arterial pressure during anesthesia
- Multivariable control of anesthesia
- Modeling of neuropathological findings in Alzheimer's patients
- Radiology diagnoses
- Fuzzy inference diagnosis of diabetes and prostate cancer

Securities

In securities, fuzzy logic is used in following areas −

- Decision systems for securities trading
- Various security appliances

Transportation

In transportation, fuzzy logic is used in the following areas −

- Automatic underground train operation
- Train schedule control
- Railway acceleration
- Braking and stopping

Pattern Recognition and Classification

In Pattern Recognition and Classification, fuzzy logic is used in the following areas −

- Fuzzy logic based speech recognition
- Fuzzy logic based
- Handwriting recognition
- Fuzzy logic based facial characteristic analysis
- Command analysis
- Fuzzy image search

Psychology

In Psychology, fuzzy logic is used in following areas −

- Fuzzy logic based analysis of human behavior
- Criminal investigation and prevention based on fuzzy logic reasoning