UNIT- II

Fuzzy sets – Fuzzy rules and fuzzy reasoning – Fuzzy inference system – Mamdani fuzzy model – Sugeno fuzzy model – Tsukamoto fuzzy model.

Introduction

Fuzzy logic is being developed as a discipline to meet two objectives:

1. As a professional subject for building systems of high utility - for example fuzzy control.
2. As a theoretical subject - fuzzy logic is "symbolic logic with a comparative notion of truth developed fully in the spirit of classical logic. It is a branch of many-valued logic based on the paradigm of inference under vagueness."

What is Fuzzy Logic?

Fuzzy Logic is a form of multi-valued logic derived from fuzzy set theory to deal with reasoning that is approximate rather than precise. Fuzzy logic is **not a vague logic system**, but a system of logic for dealing with vague concepts. As in fuzzy set theory the set membership values can range (inclusively) between 0 and 1, in fuzzy logic the degree of truth of a statement can range between 0 and 1 and is not constrained to the two truth values true/false as in classic predicate logic.

Example for Fuzzy Logic

Problem: A real estate owner wants to classify the houses he offers to his clients. One main indicator of comfort of these houses is the **number of bedrooms in them**. Let the available types of houses be represented by the following set.

U = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

The houses in this set are described by **U** number of bedrooms in a house. The realtor wants to describe a "comfortable house for a 4-person family," using a fuzzy set.

Solution: The fuzzy set "comfortable type of house for a 4-person family" may be described using a fuzzy set in the following manner.

HouseForFour =FuzzySet [{{1, 0.2}, {2, .5}, {3, .8}, {4, 1}, {5, .7}, {6, .3}},

Universal Set—>{1,10}];

FuzzyPlot [HouseForFour, ShowDots –> True];

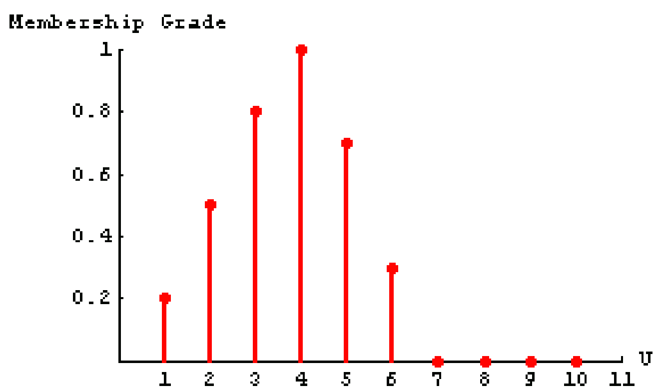

Fig 1. The Plot

Fuzzy System

A Fuzzy System can be contrasted with a conventional - crisp system in three main ways:

1. A **Linguistic Variable** is defined as a variable whose values are sentences in a natural or artificial language. Thus, "if tall", "not tall", "very tall", "very very tall", etc. are values of height, then height is a linguistic variable.

2. **Fuzzy Conditional Statements** are expressions of the form "If A THEN B", where A and B have fuzzy meaning, e.g. "If x is small THEN y is large", where small and large are viewed as labels of fuzzy sets.

3. **A Fuzzy Algorithm** is an ordered sequence of instructions which may contain fuzzy assignment and conditional statements, e.g., “x = very small, IF x is small THEN y is large”. The execution of such instructions is governed by the compositional rule of inference and the rule of the preponderant alternative.

2.1 Fuzzy Sets

A Fuzzy set is a set whose elements have degrees of membership. Fuzzy sets are an extension of the classical notion of set (known as a Crisp Set). More mathematically, a fuzzy set is a pair (A, μA) where A is a set and $\mu A : A \to [0, 1]$. For all $x \in A$, $\mu A(x)$ is called the grade of membership of x. If $\mu A(x) = 1$, we say that x is **Fully Included** in (A, μA), and if $\mu A(x) = 0$, we say that x is Not Included in (A, μA). If there exists some $x \in A$ such that $\mu A(x) = 1$, we say that (A, μA) is **Normal**. Otherwise, we say that (A, μA) is **Subnormal**.

A fuzzy set is denoted as:

$$\mathbf{A = \mu A(x1)/x1 + \cdots + \mu A(xn)/xn}$$

that belongs to a finite universe of discourse: $A \subseteq \{x1, x2, \ldots, xn\} = X$

where $\mu A(xi)/xi$ (a singleton) is a pair “grade of membership element”.

Simple Example:

Consider $X = \{1, 2, \ldots, 10\}$.

Suppose a child is asked which of the numbers in X are “large” relative to the others. The child might come up with the following:

Number	Comment	Degree
10	Definitely	1
9	Definitely	1
8	Quite possible	0.8
7	May be	0.5

6	Not usually	0.2
5,4,3,2,1	Definitely Not	0

Fig 2. Possible solution given by the child

Definitions on Fuzzy Sets

Following are the definitions for two fuzzy sets (A, µA) and (B, µB), where $A, B \subseteq X$:

- **Equality:** A = B if µA(x) = µB(x) for all $x \in X$
- **Inclusion:** $A \subseteq B$ iff $\mu A(x) \leq \mu B(x)$ for all $x \in X$
- **Cardinality:** |A| = Xn i=1 µA(xi)
- **Empty Set:** A is empty iff µA(x) = 0 for all $x \in X$.
- **α-Cut:** Given $\alpha \in [0, 1]$, the α-cut of A is defined by $A\alpha = \{x \in X \mid \mu A(x) \geq \alpha\}$

Operations on Fuzzy Sets

Let (A, µA),(B, µB) be a fuzzy sets.

- Complementation: (¬A, µ¬A), where µ¬A = 1 − µA
- Height: h(A) = max $x \in X$ µA(x)
- Support: supp(A) = $\{x \in X \mid \mu A(x) > 0\}$
- Core: C(A) = $\{x \in X \mid \mu A(x) = 1\}$
- Intersection: $C = A \cap B$, where µC = min $x \in X$ {µA, µB}
- Union: $C = A \cup B$, where µC = max $x \in X$ {µA, µB}
- Bounded Sum: C = A + B, where µC(x) = min{1, µA(x) + µB(x)}
- Bounded Difference: C = A − B, where µC(x) = max{0, µA(x) − µB(x)} 7

- Exponentiation: $C = A\alpha$ where $\mu C = (\mu A)\ \alpha$ for $\alpha > 0$
- Level Set: $C = \alpha A$ where $\mu C = \alpha\mu A$ for $\alpha \in [0, 1]$
- Concentration: $C = A\alpha$ where $\alpha > 1$
- Dilation: $C = A\alpha$ where $\alpha < 1$

Note that $A \cap \neg A$ is not necessarily the empty set, as would be the case with classical set theory. Also, if A is crisp, then $A\alpha = A$ for all α. We will define the Cartesian product $A \times B$ to be the same as $A \cap B$.

Membership Functions

A *membership function* (MF) is a curve that defines how each point in the input space is mapped to a membership value (or degree of membership) between 0 and 1. The input space is sometimes referred to as the *universe of discourse*, a fancy name for a simple concept.

The most commonly used MFs are

- Triangles
- Trapezoids
- Bell Curves
- Gaussian and
- Sigmoidal

2.2 Fuzzy Rules

Human beings make decisions based on rules. Although, we may not be aware of it, all the decisions we make are all based on computer like if-then statements. If the weather is fine, then we may decide to go out. If the forecast says the weather will be bad today, but fine tomorrow, then we make a decision not to go today, and postpone it till tomorrow. Rules associate ideas and relate one event to another.

Fuzzy machines, which always tend to mimic the behavior of man, work the same way. However, the decision and the means of choosing that decision are replaced by fuzzy sets and the rules are replaced by fuzzy rules. Fuzzy rules also operate using a series of if-then statements. For instance, if X then A, if y then b, where A and B are all sets of X and Y. Fuzzy rules define **fuzzy *patches***, which is the key idea in fuzzy logic. A machine is made smarter using a concept designed by Bart Kosko called the Fuzzy Approximation Theorem (FAT). The FAT theorem generally states a finite number of patches can cover a curve as seen in the figure below. If the patches are large, then the rules are sloppy. If the patches are small then the rules are fine.

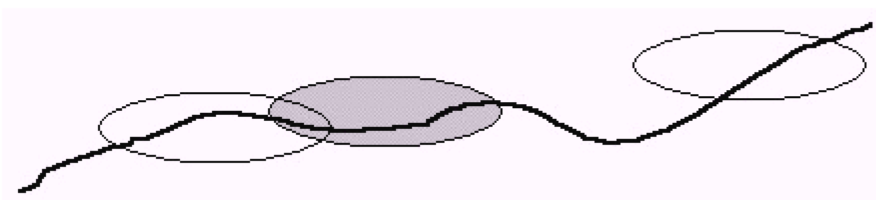

Fig 3. Fuzzy Patches

In a fuzzy system this simply means that all our rules can be seen as patches and the input and output of the machine can be associated together using these patches. Graphically, if the rule patches shrink, our fuzzy subset triangles get narrower. Simple enough? Yes, because even novices can build control systems that beat the best math models of control theory. Naturally, it is math-free system.

2.3 Fuzzy Reasoning

Single rule with single antecedent

Rule: if x is A then y is B

Fact: x is A’

Conclusion: y is B’

The i-th fuzzy rule from this rule-base

$$R_i : \text{if } x \text{ is } A_i \text{ and } y \text{ is } B_i \text{ then } z \text{ is } C_i \text{ is}$$

implemented by a fuzzy relation Ri and is

defined as

$$Ri(u, v, w)=(Ai \times Bi \to Ci)(u, w)$$

$$= [Ai(u) \times Bi(v)] \to Ci(w) \quad \text{for } i = 1,...,n.$$

2.4 Fuzzy Inference (Expert) system

A fuzzy inference system (FIS) is a system that uses fuzzy set theory to map inputs (features in the case of fuzzy classification) to outputs (classes in the case of fuzzy classification).

Fuzzy inference systems have been successfully applied in fields such as automatic control, data classification, decision analysis, expert systems, and computer vision. Because of its multidisciplinary nature, fuzzy inference systems are associated with a number of names, such as fuzzy-rule-based systems, fuzzy expert systems, fuzzy modeling, fuzzy associative memory, fuzzy logic controllers, and simply (and ambiguously) fuzzy systems.

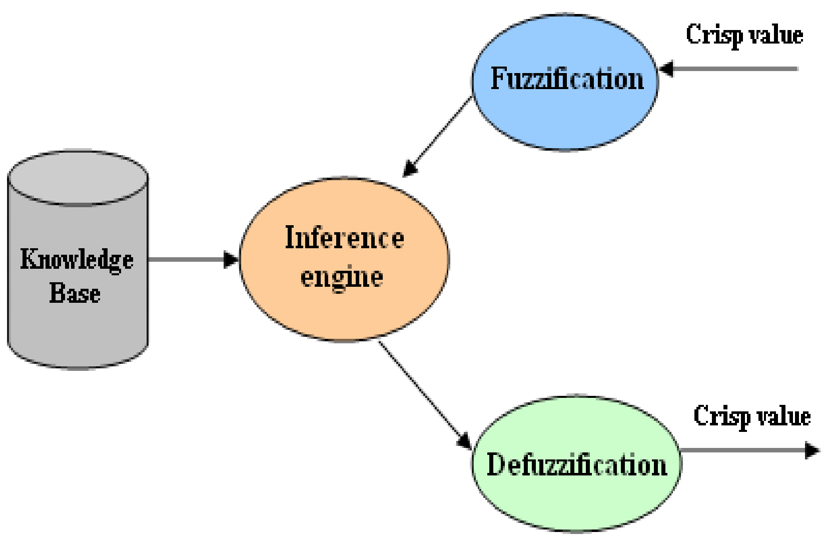

Fig 4. Structure of a Fuzzy Expert System

The rules in FIS (sometimes may be called as fuzzy expert system) are fuzzy production rules of the form:

- *if p then q,* where p and q are fuzzy statements.

For example, in a fuzzy rule

- if x is low and y is high then z is medium.
- Here x is low; y is high; z is medium are fuzzy statements; x and y are input variables; z is an output variable, low, high, and medium are fuzzy sets.

The antecedent describes to what degree the rule applies, while the conclusion assigns a fuzzy function to each of one or more output variables. Most tools for working with fuzzy expert systems allow more than one conclusion per rule.

The set of rules in a fuzzy expert system is known as ***knowledge base***.

The functional operations in fuzzy expert system proceed in the following steps.

- Fuzzification
- Fuzzy Inferencing (apply implication method)
- Aggregation of all outputs
- Defuzzification

Fuzzification

- In the process of fuzzification, membership functions defined on input variables are applied to their actual values so that the degree of truth for each rule premise can be determined.
- Fuzzy statements in the antecedent are resolved to a degree of membership between 0 and 1.

- If there is only one part to the antecedent, then this is the degree of support for the rule.
- If there are multiple parts to the antecedent, apply fuzzy logic operators and resolve the antecedent to a single number between 0 and 1.

* Antecedent may be joined by OR; AND operators.
  - For OR -- max
  - For AND -- min

Fuzzy Inferencing

In the process of inference

- Truth value for the premise of each rule is computed and applied to the conclusion part of each rule.
- This results in one fuzzy set to be assigned to each output variable for each rule.

The use of degree of support for the entire rule is to shape the output fuzzy set. The consequent of a fuzzy rule assigns an entire fuzzy set to the output. If the antecedent is only partially true, (i.e., is assigned a value less than 1), then the output fuzzy set is truncated according to the implication method. If the consequent of a rule has multiple parts, then all consequents are affected equally by the result of the antecedent. The consequent specifies a fuzzy set to be assigned to the output. The implication function then modifies that fuzzy set to the degree specified by the antecedent.

The following functions are used in inference rules.

* *min* or *prod* are commonly used as inference rules.
* *min*: truncates the consequent's membership function
* *prod*: scales it.

Aggregation of all outputs

It is the process where the outputs of each rule are combined into a single fuzzy set.

- The input of the aggregation process is the list of truncated output functions returned by the implication process for each rule.
- The output of the aggregation process is one fuzzy set for each output variable.
  - Here, all fuzzy sets assigned to each output variable are combined together to form a single fuzzy set for each output variable using a fuzzy aggregation operator.

Some of the most commonly used aggregation operators are

- the maximum : point-wise maximum over all of the fuzzy sets
- the sum : (point-wise sum over all of the fuzzy) the probabilistic sum.

Defuzzification

In Defuzzification, the fuzzy output set is converted to a crisp number.

Some commonly used techniques are the *centroid* and *maximum* methods.

- In the *centroid method*, the crisp value of the output variable is computed by finding the variable value of the centre of gravity of the membership function for the fuzzy value.
- In the *maximum method*, one of the variable values at which the fuzzy set has its maximum truth value is chosen as the crisp value for the output variable.

Some other methods for defuzzification are:

- bisector, middle of maximum (the average of the maximum value of the output set), largest of maximum, and smallest of maximum, etc.

There are two types of fuzzy inference systems that can be implemented in the Fuzzy Logic Toolbox: Mamdani-type and Sugeno-type.

2.5 Mamdani Fuzzy Model

Mamdani's fuzzy inference method is the most commonly seen fuzzy methodology. Mamdani's method was among the first control systems built using fuzzy set theory. It was proposed in 1975 by Ebrahim Mamdani as an attempt to control a steam engine and boiler combination by synthesizing a set of linguistic control rules obtained from experienced human operators.

To compute the output of this FIS given the inputs, one must go through six steps:

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).

The following is a more detailed description of this process.

The following is a more detailed description of this process.

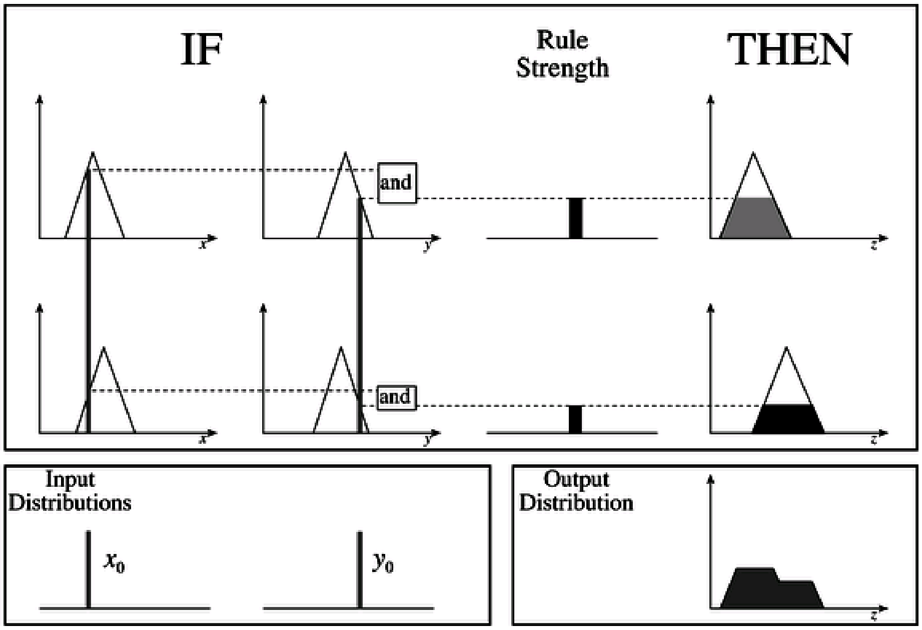

Fig 5. A two input, two rule Mamdani FIS with crisp inputs

Creating fuzzy rules

Fuzzy rules are a collection of linguistic statements that describe how the FIS should make a decision regarding classifying an input or controlling an output. Fuzzy rules are always written in the following form:

if *(input1 is membership function1)* ***and/or*** *(input2 is membership function2)* ***and/or then*** *(output$_n$ is output membership function$_n$).*

Example

$$\begin{cases} \text{If } X \text{ is small then } Y \text{ is small.} \\ \text{If } X \text{ is medium then } Y \text{ is medium.} \\ \text{If } X \text{ is large then } Y \text{ is large.} \end{cases}$$

Another Example

if *temperature is high* ***and*** *humidity is high* ***then*** *room is hot.*

There would have to be membership functions that define what we mean by high temperature (input1), high humidity (input2) and a hot room (output1). This process of taking an input such as temperature and processing it through a membership function to determine what we mean by "high" temperature is called fuzzification.Also, we must define what we mean by "and" / "or" in the fuzzy rule. This is called fuzzy combination.

Fuzzification

The purpose of fuzzification is to map the inputs from a set of sensors (or features of those sensors such as amplitude or spectrum) to values from 0 to 1 using a set of input membership functions. In the example shown in the above figure, there are two inputs, x0 and y0 shown at the lower left corner. These inputs are mapped into fuzzy numbers by drawing a line up from the inputs to the input membership functions above and marking the intersection point.

These input membership functions, as discussed previously, can represent fuzzy concepts such as "large" or "small", "old" or "young", "hot" or "cold", etc. When choosing the input membership functions, the definition of what we mean by "large" and "small" may be different for each input.

Fuzzy Combinations

In making a fuzzy rule, we use the concept of "and", "or", and sometimes "not". The sections below describe the most common definitions of these "fuzzy combination" operators. Fuzzy combinations are also referred to as "T-norms".

a) Fuzzy "and"

The fuzzy "and" is written as:

$$u_{A \cap B} = T(u_A(x), u_B(x))$$

where μ_A is read as "the membership in class A" and μ_B is read as "the membership in class B". There are many ways to compute "and". The two most common are:

1. Zadeh - min($u_A(x)$, $u_B(x)$) This technique, named after the inventor of fuzzy set theory simply computes the "and" by taking the minimum of the two (or more) membership values. This is the most common definition of the fuzzy "and".

2. Product - $u_a(x)$ times $u_b(x)$) This techniques computes the fuzzy "and" by multiplying the two membership values.

Both techniques have the following two properties:

T(0,0) = T(a,0) = T(0,a) = 0

T(a,1) = T(1,a) = a

One of the nice things about both definitions is that they also can be used to compute the Boolean "and". The table below shows the Boolean "and" operation. Notice that both fuzzy "and" definitions also work for these numbers. The fuzzy "and" is an extension of the Boolean "and" to numbers that are not just 0 or 1, but between 0 and 1.

Input1 (A)	**Input2 (B)**	**Output (A "and" B)**
0	0	0
0	1	0
1	0	0
1	1	1

The Boolean "and"

b) Fuzzy "or"

The fuzzy "or" is written as:

$$u_{A \cup B} = T(u_A(x), u_B(x))$$

Similar to the fuzzy "and", there are two techniques for computing the fuzzy "or":

1. Zadeh - max($u_A(x)$, $u_B(x)$) This technique computes the fuzzy "or" by taking the maximum of the two (or more) membership values. This is the most common method of computing the fuzzy "or".

2. Product - $u_A(x) + u_B(x) - u_A(x)\, u_B(x)$ This technique uses the difference between the sum of the two (or more) membership values and the product of the membership values.

Both techniques have the following properties:

$T(a,0) = T(0,a) = a$

$T(a,1) = T(1,a) = 1$

Similar to the fuzzy "and", both definitions of the fuzzy "or" also can be used to compute the Boolean "or". The table below shows the Boolean "or" operation. Notice that both fuzzy "or" definitions also work for these numbers. The fuzzy "or" is an extension of the Boolean "or" to numbers that are not just 0 or 1, but between 0 and 1.

Input1 (A)	**Input2 (B)**	**Output (A "or" B)**
0	0	0
0	1	1
1	0	1
1	1	1

The Boolean "or"

c) Consequence

The consequence of a fuzzy rule is computed using two steps:

1. Computing the rule strength by combining the fuzzified inputs using the fuzzy combination process discussed previously. This is shown in Fig 5. Notice in this example, the fuzzy "and" is used to combine the membership functions to compute the rule strength.

2. Clipping the output membership function at the rule strength. Once again, refer to Fig 5. to see how this is done for a two input, two rule Mamdani FIS.

d) Combining Outputs into an Output Distribution

The outputs of all of the fuzzy rules must now be combined to obtain one fuzzy output distribution. This is usually, but not always, done by using the fuzzy "or". Figure 5 shows an example of this. The output membership functions on the right hand side of the figure are combined using the fuzzy "or" to obtain the output distribution shown on the lower right corner of the figure.

e) Defuzzification of Output Distribution

In many instances, it is desired to come up with a single crisp output from a FIS. For example, if one was trying to classify a letter drawn by hand on a drawing tablet, ultimately the FIS would have to come up with a crisp number to tell the computer which letter was drawn. This crisp number is obtained in a process known as defuzzification. There are two common techniques for defuzzifying:

1. **Center of mass** - This technique takes the output distribution found previously and finds its center of mass to come up with one crisp number. This is computed as follows:

$$z = \frac{\sum_{j=1}^{q} Z_j u_c(Z_j)}{\sum_{j=1}^{q} u_c(Z_j)}$$

where z is the center of mass and u_c is the membership in class c at value z_j. An example outcome of this computation is shown in Fig 6.

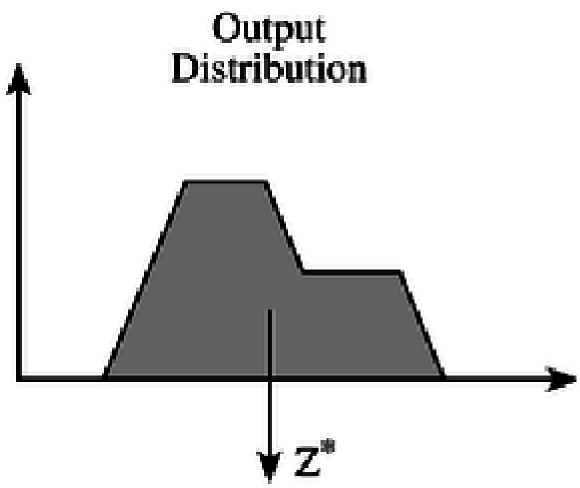

Fig 6. Defuzzification Using the Center of Mass

2. **Mean of maximum** - This technique takes the output distribution found previously and finds its mean of maxima to come up with one crisp number. This is computed as follows:

$$z = \sum_{j=1}^{l} \frac{z_j}{l}$$

where z is the mean of maximum, z_j is the point at which the membership function is maximum, and l is the number of times the output distribution reaches the maximum level. An example outcome of this computation is shown in Figure 7.

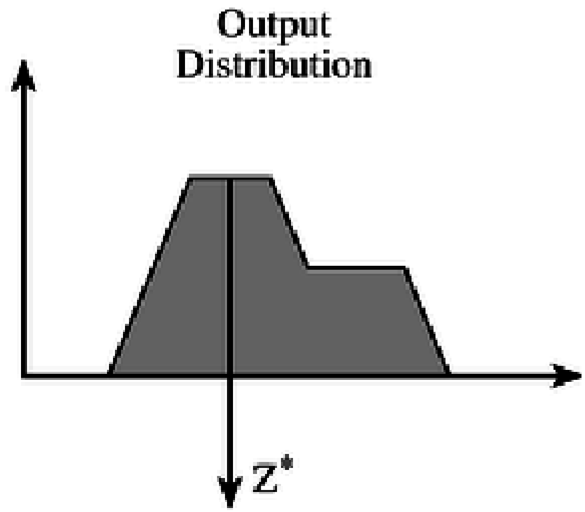

Fig 7. Defuzzification Using the Mean of Maximum

Fuzzy Inputs

In summary, Fig 5 shows a two input Mamdani FIS with two rules. It fuzzifies the two inputs by finding the intersection of the crisp input value with the input membership function. It uses the minimum operator to compute the fuzzy "and" for combining the two fuzzified inputs to obtain a rule strength. It clips the output membership function at the rule strength. Finally, it uses the maximum operator to compute the fuzzy "or" for combining the outputs of the two rules.