

# Fuzzy Logic and its Application.

*Adeel Nafis, Student Member IEEE.  
Institute of Industrial Electronics Engineering (IIEE), PCSIR.  
ST-22C, Block-6, Gulshan-e-Iqbal, Karachi-75300, Pakistan.  
Tel: 4982353, 4966279  
E-mail: adeelnafis@ieee.org*

## Abstract:

Fuzzy logic has come up as a major contender to conventional approaches of controlling a system. The paper is an effort to introduce Fuzzy Logic and its applications in Control Engineering. The object being to demonstrate is that Fuzzy Logic is a viable alternative to solving complex control problems without developing rigorous mathematical model, which may be too difficult to construct or implement.

First an introduction is given to the basic fundamentals of Fuzzy then the actual implementation is covered, later in the paper, along with the differences in fuzzy and conventional methods.

## What is Fuzzy Logic?

Fuzzy logic is based on the fact that humans are able to handle complex problems in everyday life without having an exact mathematical understanding of the system [1]. For example when we ride a bicycle we are unaware of the value of parameters that are influencing our ride. These parameters may include such things as the cycle's weight velocity and friction. Even then a rider is usually able to handle a bicycle quite precisely in the most extreme of circumstances.

Deriving from the above example we can see that the capability of humans to solve problem is based on a heuristic approach. The accumulated experience is used in such a way as to provide an approximate solution to the problem at hand. Fuzzy logic is all about developing a system based on heuristic knowledge; mathematical models are usually applied to work as a tool but not as the base.

Fuzzy logic is a superset of conventional Boolean logic [2]. Boolean logic is based on the rule that every input and output of a system can have a maximum of two discrete levels of value, usually called '1' or '0'. All processing is carried out on the same bases. Boolean logic naturally assumes that every property of a system can have only fixed discrete levels of information. On the other hand Fuzzy assumes that discrete levels of information are only a special case of information being continuous in nature.

Consider a driver overtaking another one, he knows that for such an attempt there is a need of pressing the gas pedal, but how far and how fast should he step on it. This depends on a number of factors, such as the road condition, the particular vehicle, traffic and whether there is an emergency or not. Again all these particulars usually do not have a fixed single value, for example being late to a dinner may be considered a minor emergency while taking an accident victim to hospital may be a major one. From this we may conclude that an input parameter 'emergency in this case' for a particular system can

have a range of values, unlike in Boolean logic which will assume emergency to be present or not present.

It is possible and indeed preferable in many systems to avoid the vagueness that arises due to fuzzy logic. However the fact is that words such as fast or slow although may not have the sudden transitions that are required by Boolean logic but they do have their own important meaning, which is best left attributed to them when describing a system [3]. These words come important in systems, which, for example, are required to mimic human behavior.

## Fuzzy Vs Conventional Control:

In development of conventional controllers some basic steps are always present.

1. Identification of the system, its inputs and outputs.
2. Development of a mathematical model of the system
3. Development of a controller based on the mathematical model
4. Implementation of the Controller
5. Final testing and tuning.

These steps are well documented and researched. Human experience is helpful through out these steps but the last one requires it the most. Now as humans (operators in a plant e.g.) are able to handle many types of complex control tasks without having a mathematical model can we reason that a controller build on the expertise of a human operator will be able to achieve better results?

*Fuzzy control provides a formal methodology for representing, manipulating, and implementing a human's heuristic knowledge about how to control a system [7].*

## Fuzzy's History:

Traditional mathematical logical is based on the notion that any particular proposition can have only one of two values, true or false. This law was first proposed by Parminedes (400BC). Even then there were quite a few objections to it, and over the years some philosophers and mathematicians gradually realized that there were many instances where this law failed to hold. Pluto was the first one to grab hold of the idea that there was a third region where the opposites (true and false) 'tumbled about [3].

For quite a long time there was no mathematics to support the propositions and theories but in the early 1900's Lukasiewicz described a three valued system with a third term which was proposed to be the possible one between true and false. Later on he provided the detailed mathematics to go along with it. Some more authors started work on it but the proposed mathematics did not catch up [4].

Finally in 1965, Lotfi A. Zadeh described the mathematics of fuzzy logic in his seminal work. The theory showed how *membership functions* could be developed having values between true '1' and false '0'. This provided the stimulus to Fuzzy Logic as calculus operations on fuzzy numbers were also developed. The proposed system is what the world has come to identify as fuzzy logic.

## Implementation of Fuzzy Logic

Fuzzy logic works on a set of rules. The set of rules that are applicable for a particular set of inputs generate an output, which is then processed to produce a particular solution. There are three basic processes involved [5]

1. Fuzzification
2. Inference
3. Defuzzification.

These processes can be best understood if we remember that any state of input cannot change states immediately [6]. The range of the value of a particular input or output is called its 'Universe of Discourse'. The input/ output variable can have many states in this universe of discourse. For example the states for a temperature input a system may be 'Hot', 'Very Hot', 'Cool', etc. for each value of the input to the system there will be a corresponding value of each state. An example may be.

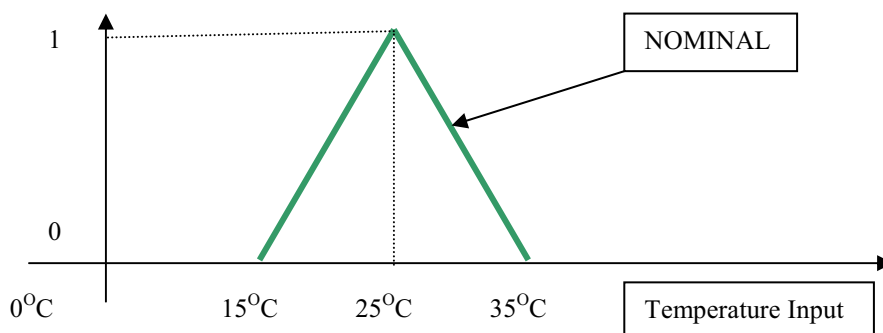
Consider the same temperature input, being applied to a car air conditioning system. The temperature sensor provides an input signal, which corresponds to say 40°C; this value might be regarded in terms of a state of the input variable as 'VERY HOT'.

As the inputs and outputs are being defined on heuristic approach and so are the values we call the input/output variables the *linguistic variable* and the states that it may have as the *linguistic terms or linguistic values*. In this way we are not concerned with the actual value of the temperature but the value [0,1] of the linguistic terms at a particular temperature. This assignment of *Degree of Membership* (DOM) to a variable is done through *Membership Functions (MFs)*. MFs are the functions, which provide the DOM of a term for every point on the Universe of Discourse of the Variable.

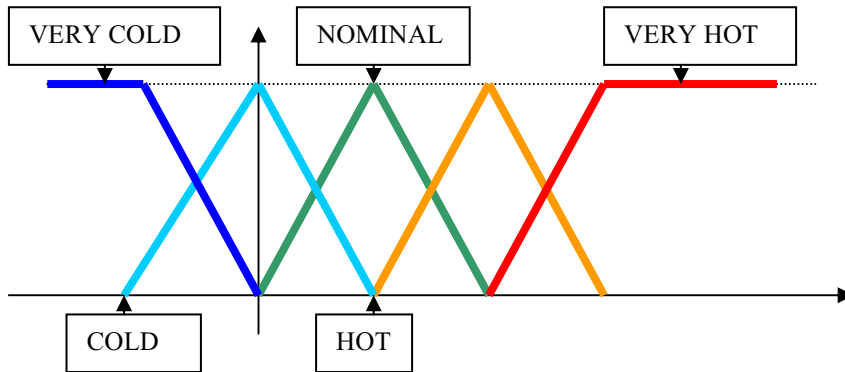
Let us say that the temperature drops to 25°C, we would most probably not consider it 'VERY HOT' and let us assign it an arbitrary linguistic value, 'NOMINAL', and another one for a slightly higher temperature but less than VERY HOT as 'HOT'. In normal Boolean logic these states will change state abruptly. Say that 30°C is hot, but is 31°C VERY HOT?

This problem is avoided if we say that 31°C is more on the VERY HOT side than to HOT state. The value (degree of membership) of these linguistic values can range from 0 to 1, and at 31°C we may assign a value of 0.7 to VERY HOT and 0.3 to HOT. We decide these values for every point till from 35°C and above we may safely say that it surely is warm enough that VERY HOT reaches a maximum value and HOT should not be considered. So from 35°C onwards VERY HOT stays at 1 while HOT stays at 0. If we define the function mathematically we get the MF. These MFs are defined for each linguistic value and shows the weight of each linguistic value for a particular value of the input.

In graphical form we may show the air conditioners temperature input's NOMINAL state MF as:



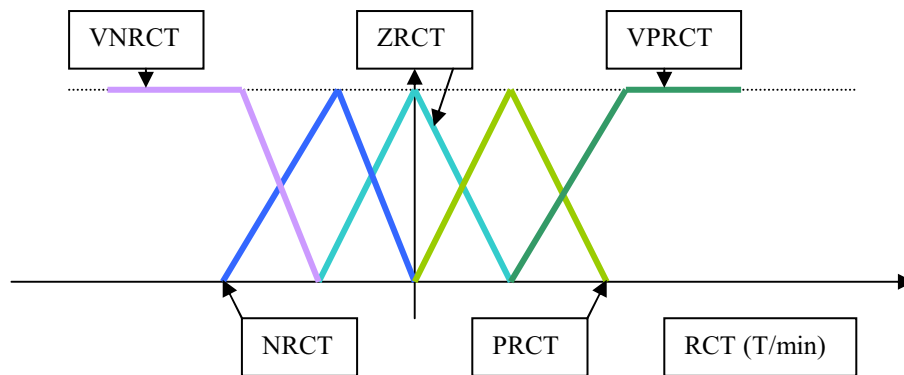
We may add as many linguistic values as suitable, with their corresponding membership functions, if we add 5 the graph becomes.



Although the graphs shown are drawn with straight lines, generating triangular MF (NOMINAL, HOT and COLD) or ramp (VERY HOT and VERY COLD) one's, but they can have different shapes such as Gaussian, Sinusoidal, trapezoidal etc.

These MFs are defined for each input and output. The number of which depends on different factors, however the more the MFs generally the more precision we get. For the air-conditioner example we may define another input which may be the Rate of Change of Temperature (RCT), with its own distinct MFs.

Membership functions of Rate of Change of Temperature.



- KEY:
- RCT : Rate of Change of Temperature
  - VNRCT : Very Negative RTC
  - NRCT : Negative RCT
  - ZRCT : Zero RTC
  - PRCT : positive RTC
  - VPRCT : Very Positive

These MFs may be assigned values as ascertained to meet the requirements. If the speed of the fan over the evaporator is the output of the system, then it can similarly be assigned MFs over its universe of discourse. This process of assigning Membership Functions to an input or output linguistic value is called the Fuzzification process.

Although this Fuzzification may lead to drawing a parallel with probability, but it is not so [2]. Probability theory gives the relative chance of something being or not being present. Fuzzy logic on the other hand assigns a weight to each linguistic value, everything is always present but its weight is different for different input conditions.

There are two basic types of probabilities [8] one is the *Stochastic* and the other *lexical*. The first one is what we usually use in mathematics, in stochastic probability the process or event is well defined. E.g. the probability of getting '1' from a dice throw, is 1 out of 6. In lexical uncertainty the event is not well defined. This type of uncertainty is what we are accustomed in to everyday life. For example a teacher might say that the probability of a student performing well is quite high. In this example 'performing well' is an event not exactly defined, as it is based on a number of factors and there are no formal criteria of a person performing well. Even though we are able to use these words precisely, the meaning being usually clear from the context.

## Developing a Rule Base

After deciding on the number of inputs and outputs and their relative MFs the next task is to manipulate the inputs to provide an output. Developing a Rule Base does this. The rule base is the incorporation of human expertise into a set of rule, and is simply the process of determining the output for each particular set of input conditions.

For example:

Rule #1

If INPUT1 is THIS (1) and INPUT2 is THIS (2) then OUTPUT is THIS (3)

Rule #2

If INPUT1 is THIS (1) or INPUT2 is THIS (2) then OUTPUT is THIS (3)

Rule #3

If INPUT1 is THIS (1) and INPUT2 is not THIS (2) then OUTPUT is THIS (3)

In general every rule follows a pattern, which is:

IF Antecedent THEN Consequent

In this way rules are defined for each particular input of conditions. This is the most critical part of the system. The systems output is intimately linked to the Rule Base. For our particular air-conditioner example there will be 25 Rules in the Rule Base. For a complete Climate Control system there will be two outputs, one for heat and the other one for cooling. The rules will address each output also.

At this point it is important to know what a Fuzzy Subset is. A fuzzy subset shows linguistic values degree of membership [0 1] on the Universe of Discourse of a linguistic variable [2]. The fuzzy subset defines the mapping of a value at each point, between the universe of discourse and a value from 0 to 1, or for visualization it is the area under the MF of that particular input. The ordered pair is generated by having the value on the

universe of discourse as the first ordinate and the value between 0 and 1 as the second one. In our example HOT is the fuzzy subset of the universe of discourse of Temperature. Let these subsets be represented by  $A_i^j$ .  $A$  is the  $j^{\text{th}}$  values fuzzy subset of a linguistic variable  $i$ . Mathematically  $A_i^j \in A_i$ . The membership function is represented by  $U_{A_i^j}$ , this implies a MF for the fuzzy subset  $A_i^j$ .

Now how do we assign numerical values to Consequent. Usually for a particular set of input more then one rule may fire. For example if the temperature input is  $30^{\circ}\text{C}$  then the temperature input will have weight of both NOMINAL and HOT. The values of each MF for the particular input are found simply by putting the value of the input in the MF or by drawing a perpendicular from the input value and seeing the corresponding value of each MF that is cut.

The process of determining the extent to which each rule is relevant to the current set of inputs and then determining the conclusion from the inputs and the rule base is collectively called *inference*. Say the value of NOMINAL is 0.7 and of HOT is 0.3. Now the RCT input will also have a particular value at that instant say it is such that PRCT and VPRCT both have a value other then zero. This will mean that the temperature is somewhere between NOMINAL and HOT and is increasing with a rate in between POSITIVE and VERY POSITIVE. Let the values of PRCT and VPRCT be 0.4 and 0.6 respectively.

Now the rule present which ‘ANDS’ temperature inputs NOMINAL and POSITIVE will fire. Similarly other rules satisfying the conditions will fire and each fired rule will assign a particular value to the corresponding MF of output (i.e. is speed of the fan). We may have a rules such as:

RULE #1      If (Temp is HOT) and (RCT is VPRCT) then (Speed is HIGH)  
 RULE #2      If (Temp is HOT) and (RCT is PRCT) then (Speed is HIGH)  
 RULE #3      If (Temp is NOMINAL) and (RCT is VPRCT) then (Speed is HIGH)  
 RULE #4      If (Temp is NOMINAL) and (RCT is PRCT) then (Speed is HIGH)

As is evident from the above rules for a particular set of conditions more then one rule may fire. There may also be multiple assignments made to the same output variable, this may call for storing the different values assigned to the same term in an array. In our

In this way HIGH and VERY HIGH will each get a particular value for a particular set of inputs. We may store the value of each HIGH as HIGH1, HIGH2, HIGH3 and so on, so that later we are able to apply mathematical functions to get a final value of speed. The problem is what does AND,OR, NOT stand for in terms of fuzzy MFs as they do not have a true or false value only.

In Fuzzy Logic terminology AND is usually represented by the ‘\*’ operator. It can have two different meanings and one is usually decided on before implementation.

Let the antecedent of a rule be shown as  $(Aa^j \times Ab^k \times \dots \times An^l)$  –the operator may be either AND ‘\*’ or OR ‘ $\oplus$ ’.

$$1. UAa^j * UA_b^k * \dots * UAn^l = \min(UAa^j, UA_b^k, \dots, UAn^l)$$

$$2. UA_a^j * UA_b^k * \dots * UA_n^l = UA_a^j \times UA_b^k \times \dots \times UA_n^l$$

This means \* can mean either the min of the terms ANDed together or the product of the terms.

Similarly OR ‘ $\oplus$ ’ has two meanings, either the maximum of the ORed terms or the algebraic sum of the terms.

1.  $UA_a^j \oplus UA_b^k = \max(UA_a^j, UA_b^k)$
2.  $UA_a^j \oplus UA_b^k = UA_a^j + UA_b^k - (UA_a^j \times UA_b^k)$

Lastly NOT specifies subtraction from 1. For example if the value of HOT is 0.7 then NOT (HOT) = 1 - 0.7 = 0.3.

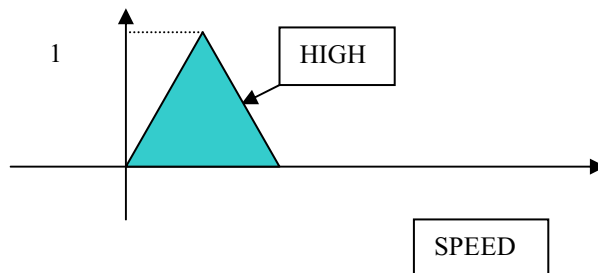
Say that HOT has a value of 0.3 and VPRCT has 0.6, now if we take the meaning of \* to be min then, for RULE#1, as HOT is lower 0.3 will be assigned to HIGH1. Similarly HIGH2 will also get 0.3 while RULE#3 will assign a value of 0.6 to HIGH3. In this way we will get a value of speed in terms of its states’ MFs.

- HIGH1= 0.3
- HIGH2= 0.3
- HIGH3=0.6
- HIGH4=0.4

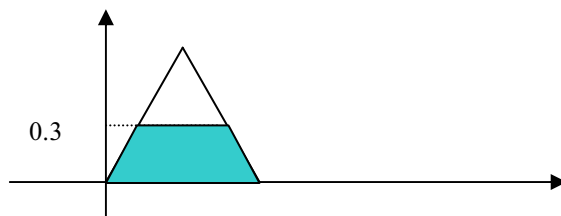
Once we get the value we want to assign to the consequent, there are two ways of mapping that output to the fuzzy subset. In the MIN method the fuzzy subset is simply clipped at the particular value and in the PRODUCT method the height is multiplied by the value.

For example from RULE#1 (above) HIGH1 = 0.3, therefore HIGH1 will clip HIGH at 0.3 in the MIN method.

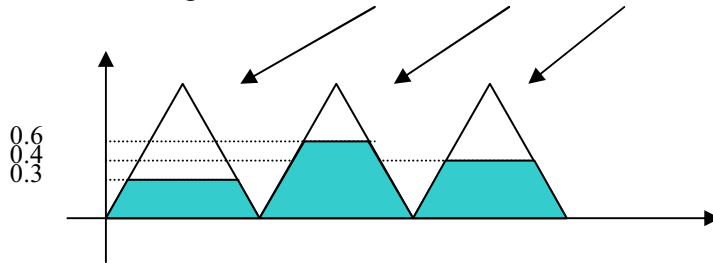
HIGH subset:



HIGH after assignment from HIGH1

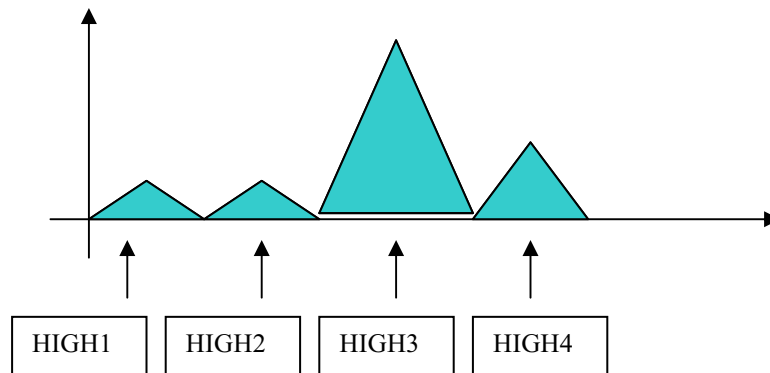


HIGH after assignment from HIGH2, HIGH3, HIGH4



If the same results was obtained and the PRODUCT method was used then the tops of the HIGH will be scaled according to the signal.

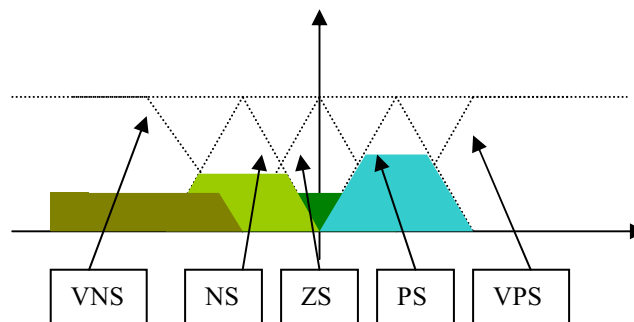
HIGH after assignment to HIGH1, HIGH2, HIGH3, HIGH4, using the PRODUCT method.



Now, the problem remains of how to get a value from this data for each of the linguistic values of speed (such as HIGH, LOW). This is taken care of by ORing the fuzzy subsets together. The usual process is to take the highest one. This is called the sup-star rule of inference. Sup standing for  $\oplus$  and the star for  $*$ .

After getting all these values the last process left is of **Defuzzification**, that is getting a crisp value from all the new fuzzy subsets.

Say the shape of output becomes  
 Final fuzzy subsets of speed (from the MAX-MIN method).  
 (Here MIN is of  $*$  and MAX is of  $\oplus$ )



KEY:	
VNS:	Very Negative Speed
NS:	Negative Speed.
ZS:	Zero Speed.
PS:	Positive Speed.
VPS:	Very Positive Speed.

There are different methods for getting a crisp output from these subsets. One of the most common is to find the center of gravity of the subsets and then take the horizontal coordinate to that point (center of gravity) as the output.

The most commonly used one is the Center of Gravity, this method requires relatively simple mathematical functions and is therefore less computational power. The Center of Gravity (COG) is taken as the Crisp output for the system. the sum of the product area obtained after inference and the point at which the MF is maximum on the Universe of Discourse of the output variable, is divided by the sum of the area after inference.

In the Center of Average method the sum of the product of the heights with the value where the MF is maximum is divided by the sum of the heights of the outputs. As the height is already known not many computations are needed and therefore this method is one of the fastest, and widely used.

### Future and Problems:

Fuzzy logic although a young science shows a lot of promise for the future. The continuous development and advancement of its related mathematics and set theory guarantees a bright future for it. Fuzzy logic is a science that is applicable to a broad field of problems, from programs to assigning appointment times at a doctor's office to flight simulation, and reduction of oscillations in a space projectile.

The most difficult thing in the whole process is the development of a valid rule base as it is on which the whole systems response depends. For a small number of input and outputs and limited Membership Functions the rule base is easy to handle and manipulate. However, the complexity increases with the rise in these parameters.

To handle this problem work is being carried out on the development of Evolutionary Logic Fuzzy Logic controllers. These systems are capable of developing a rule base on the basis of an algorithm that mimics evolution. Due to this much of the hard work in the development of controllers of complex system should be reduced.

## Conclusion:

In this paper I have tried to present the fundamentals of the Fuzzy Logic Control system. a central design idea for an Air conditioning system was followed. It was shown as an aid to the paper, so that readers may get a firm grasp on how to implement such a controller in practice.

## Acknowledgements:

This paper would not have been possible without the help of Professor Ashab Mirza and Mr. Ahmad Hasan. They both took time out to explain concepts and basics, and provide books and material essential for this paper.

## References:

- [1] IEEE Transactions on FUZZY SYSTEMS. February 2000 V8 No. 1.
- [2] <http://www.austinlinks.com/Fuzzy/overview.html>
- [3] Fuzzy Systems a Tutorial James F. Brule
- [4] C. Lejewski "Jan Lukasiewicz", Encyclopedia of Philosophy, Vol 5 ,Macmillan, NY:1967, pp 104- 107.
- [5] <http://www.austinlinks.com/Fuzzy/expert-systems.html>
- [6] <http://www.geocities.com/CapeCanaveral/Launchpad/6000/ttfuzzy.html>.
- [7] Handbook of Control Engineering. © 1999 CRC Press, INC ISBN 81-7224-785-0
- [8] Fuzzy Logic & NeuroFuzzy Applications in Business & Finance, Constantin Von Altrock, © 1995, Prentice Hall PTR, ISBN – 0-13-368465-2
- [9] MultStage Fuzzy Control, Janusz Kacprzyk, © 1997, John Wiley & Sons Ltd.