The ITB Journal

Volume 1 | Issue 1 Article 4

2000

Fuzzy Logic and Neural Networks - a Glimpse of the Future

Raymond Manley

Recommended Citation

Manley, Raymond (2000) "Fuzzy Logic and Neural Networks - a Glimpse of the Future," *The ITB Journal*: Vol. 1: Iss. 1, Article 4.
doi:10.21427/D7D333
Available at: https://arrow.tudublin.ie/itbj/vol1/iss1/4

Fuzzy Logic and Neural Networks - a glimpse of the future.

By Raymond Manley,

School of Informatics and Engineering. ITB Blanchardstown, Dublin, Ireland

Introduction

During the nineties I was interested and involved in the areas mentioned above. The practical application of my interest was in my masters thesis and in the design and delivery of a similarly named subject on the B.Eng. degree course in Product Design Engineering at Dundalk IT. My original interest was sparked in the area by the notion that engineers and scientists were starting to look at problems from a more 'human' perspective.

Since the Newtonian split each of the disciplines of human behaviour have more or less headed off in their own direction. Before this time we had people like Leonard de Vinci who could paint, design the first helicopter and probably wrote some philosophic and theological works as well. Thus, nowadays, engineers tend to stick to engineering in a narrow way and so on for other disciplines with the result that perhaps some of the narrower implementations of areas such as computers stem from a lack of a 'bigger picture'.

With my involvement in teaching digital logic on engineering courses over the years, it was not too difficult to see the organisation of computers and digital logic as existing only in terms of Boolean logic. Here we limit ourselves to two possible logic states a 1 and a 0, yes or no. This is still the case with all computers. However, this does appear quiet limiting given that we can appreciate as humans that our thought processes are infinitely more complex than a yes/no reasoning. (even for those professions where we think it exists!). So it was with interest that I started reading about two areas where efforts were being made to bring our human thought process into the engineering realm. Fuzzy logic allows a decision making process like our own and neural networks provide systems that can learn what to do in certain situations in a manner mimicking our own neural structures.

So to look to the future we may be seeing a trend where by the bigger picture and the reintegration of various disciplines will come about as art, medicine, the sciences and engineering come closer in the pursuit of more advanced and useful technological advances. If our technology is more integrated according to a human model it follows that it would more likely be more in tune with our way of doing things and thus be more appropriate. It is hard to envisage computers trotting along in a 100 years time still restricted to binary logic which originated in finding a system that was simple to build at the time.

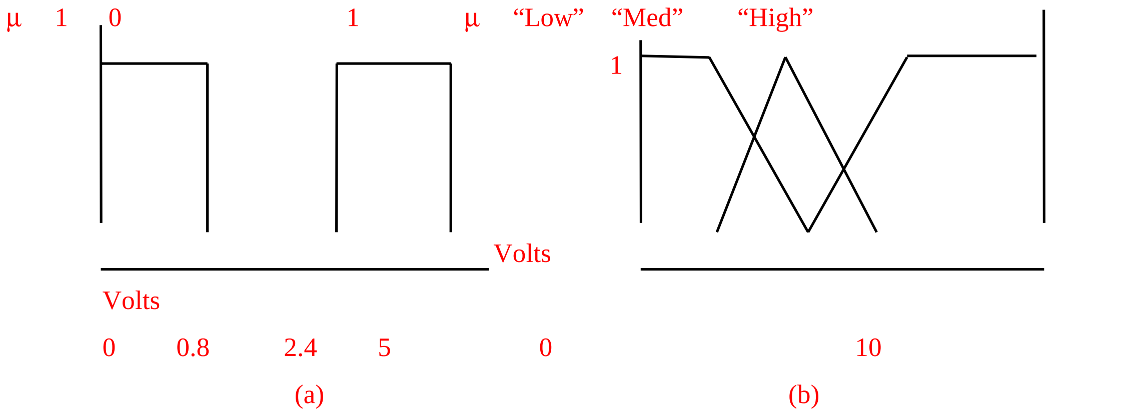
The rest of this article is a reediting of various pieces of work from the last decade. The first is an article I wrote for the journal of the Institution of Engineers of Ireland in November 1998 and will be somewhat dated at this stage. The other two pieces offer an introduction to fuzzy set theory in some more detail and also an introduction to neural networks to give a flavour of these two areas. In terms of what may exist in the future, you can use your imagination in the quiet contentment that it won't be mimicked by machines - well just yet.

Fuzzy Logic Made Clear !

These words, ‘fuzzy logic’, are becoming more a part of everyday speech as they start to appear in the specifications of a whole range of consumer products from washing machines to cameras. Most people usually refer to it in some joking manner but what is it all about? Hopefully it will be a little less ‘fuzzy’ at the end of this article!

Origins of Fuzzy Logic

In 1965 Lofti Zadeh[1] published his paper on fuzzy set theory , putting it forward as a way of more closely realising the human thought process. Many systems developed to aid human activities have been based on definitive , yes/no, type decision making processes. An example is the way all computers are based on the binary logic system where only two possible and separate logic levels are allowed, a logic 1 or logic 0. However, we know from everyday experience that humans think in terms of vague linguistic categories, for example, the weather is fairly good today. "Fairly good" represents a vague category that can be represented by a fuzzy set which allows values to belong to the set by a varying degree from 0 up to 1. The grade of membership is not a probability , it is a measure of the compatibility of an object with the concept represented by the fuzzy set.



(a) Traditional binary logic shown as two non overlapping sets with all values in each set having maximum grade of membership.

(b) Three fuzzy sets shown with overlap. The varying degree of memberships allows us to be vague about whether a value belongs to each of these sets.

Since Zadeh proposed his theory many areas of applications have been considered to assess the suitability of applying fuzzy set theory. Areas include fuzzy logic and approximate reasoning, expert systems, pattern recognition, fuzzy decision making in economics and medicine and fuzzy control. [2] However, it is in the area of fuzzy logic control that most success has been achieved.

Fuzzy Logic Control

Traditional control theory requires that a mathematical model of the process to be controlled be available. Using Laplace and Z transform techniques control systems are designed to control processes from factories to spacecraft. Digital computers have come to be used in control applications with larger and more powerful microprocessors being used. Now it is being found that there are limitations to our traditional control methods.

In some applications no precise mathematical model can be constructed to model the process or such a model proves to be too complex because of the highly non-linear characteristics of the process. Also, where models do exist, i.e. linearization of non-linear problems, the cost of DSP (Digital signal processing) is too high for many consumer applications.

Thus, in recent years, the focus is changing to look at alternative methods of control where control can be achieved at lower costs and without recourse to complex mathematical models. In applying themselves to this problem many researchers, such as Mamdani [3], noticed that in control problems where no automatic electronic control was traditionally available, the control was provided by human operators who used their skill, tacit knowledge and experience of the system to produce an acceptable result.

This led to renewed interest in fuzzy set theory which Zadeh had developed in the 1960's. Using fuzzy logic it becomes possible to quantify with fuzzy sets the vague linguistic categories that a skilled operator uses in their control of a process. For example a certain range of input values could be assigned to a fuzzy set labelled " LOW", individual values in this fuzzy set can belong to it by a varying degree from 0 to 1.

The resulting controllers allow expert knowledge to be built into them without having to describe the process mathematically. This allows problems to be addressed that in the past were too complicated.

First Use of Fuzzy Logic Control

The first real example of the possibilities of using fuzzy logic for control applications was presented by Mamdani[3] in 1974 with the control of a model steam engine. Mamdani showed that Zadeh's approach provided a convenient way of expressing the linguistic rules of a human controller in a form easily processed by computer. This allowed a complex, non-linear dynamic plant to be controlled with a simpler control strategy which would be more viable in practical situations.

Application of fuzzy logic control to an industrial process was first achieved in 1982 with the control of a cement kiln in Denmark [4] and 1988 [5], a highly non-linear process and one which traditionally could only be controlled by a skilled operator as a good mathematical model was not available to allow conventional control. These early examples of fuzzy control were applied to problems where a good base of operator knowledge could easily be transferred into the IF..THEN rules of the fuzzy logic controller.

Fuzzy Logic finds Acceptance in Japan

Despite a promising start, acceptance of fuzzy logic control was slow. This no doubt was due in part to the negative connotations of the word 'fuzzy' in the English language denoting something imprecise and unreliable. However, fuzzy logic control had a better reception in Japan where the word 'fuzzy' was just seen as a label and the technology was investigated on its own merits. Thus many of the early uses of this new control method were to be found in Japan.

The Sendai Subway Automatic Operations Controller, possibly the best known example of fuzzy control, was introduced in 1986 and the strategies of experienced operators were implemented in fuzzy rules. The system claims smoother braking at stations and speed control, however some commentators [6] question the standard of conventional controller that the fuzzy logic controller was compared with.

Similar reactions have been reported [7], which explains the slowness of acceptance of fuzzy logic control outside of Japan. However, the number of applications has increased from 8 in 1986 to 1,500 in 1993 [4]. Since 1990, many Japanese home appliances from washing machines to video cameras contain fuzzy logic control to give cost-effective control of such parameters as focusing and sharpness in cameras. [8]. Fuzzy control was introduced to cruise control in Japanese cars in 1991. Western countries tried hard to catch up on Japan at the start of the nineties and during NASA space shuttle flights in 1992/1993, temperature control of some experiments, where the temperature needed to be kept within 0.1 ^{o}C over a range of 1 to 40, was achieved using fuzzy control.

As the interest in fuzzy logic control continues to grow, many control situations are being re-investigated to see if improvements can be achieved over existing methods. The area of controlling non-linear systems has been mentioned. Another aspect of fuzzy logic control being investigated is the improvement in control action where noise, load or parameter changes occur.[9] Yet another area is where consumer products are required with greater control requirements and with a faster development cycle to meet market demands. With this last point in mind, fuzzy logic control, and its related area of neural networks, have been included in the new B.Eng. in Product Design Engineering at the Regional Technical College, Dundalk.

Summary

In summary, the use of fuzzy logic controllers has proven successful and practical in a range of commercial applications, particularly where non-linear systems or systems which are subject to noisy, erroneous information are involved and where no easy mathematical models are to be found. With increasing uses in consumer products, fuzzy logic is set to become a more accepted design method.

Fuzzy Set Theory

Fuzzy Sets - a definition

A **fuzzy set** differs from a crisp set in that the elements of such a set can have a varying degree of membership dependant on how strongly they match the set membership criteria.

For example, for a fuzzy set labelled " Tall Men", a height of 5ft would not have as strong a membership as a height of 7ft.

> Definition:
>
> Thus a fuzzy set can be defined as a set of ordered pairs where each element is assigned a membership value between 0 and 1.

The fuzzy set "Tall Men" could thus be described as:

$$TM = \{ (4,0.2) , (5,0.7) , (6,0.8), (7,1) \}$$

This grading of membership mirrors the vagueness that characterises human thought processes.
If asked, could a 6ft man be included in a group labelled "Tall Men", a person would most likely respond that a 6ft person was 'fairly tall'. This vagueness (or fuzziness as Zadeh called it) is reflected in the fuzzy set by the fact that 6ft has a membership weighting of 0.8 i.e. 80% true.

Important distinction with Probability Theory

The grade of membership is not a probability, it is a measure of the compatibility of an object with the concept represented by the fuzzy set.

Fuzzy Sets - examples

A classical or crisp set can be defined as a grouping of elements which either belong to or do not belong to that particular set. The set can be described by listing the elements of the set or by stating the conditions for membership.

Ex 1 (a):

A set **A** is defined as:

$$A = \{x \mid x \leq 6\} \quad x \in X \quad ,$$

Where X contains all possible integers.

or $A = \{ 0,1,2,3,4,5,6\}$

A fuzzy set can be defined as a group of ordered pairs denoting the element and it's degree of membership. The set can be described by listing the ordered pairs or stating the conditions of membership of the fuzzy set.

Ex 1 (b):

If X is a collection of elements , then a fuzzy set A is defined as:

$$A = \{ x, \mu_A(x) \} \quad x \in X$$

where $\mu_A(x)$ is known as the degree of membership (or degree of truth) of x in fuzzy set A, which maps X into the membership space of all values between 0 and 1.

Ex 1 (c):

If , in a garage, all cars are classified by engine size in litres then

E = { 1, 1.2, 1.3,1.4, 1.6, 1.8, 1.9, 2,2.5,2.8 } is the crisp set of all available engine sizes.

A fuzzy set to describe 'comfortable size cars' could be defined as follows;

C = { (1.3,0.3), (1.3,0.7), (1.4,0.9), (1.6,1), (1.8,0.8), (1.9, 0.4), (2,0.3)}

In general fuzzy sets will be denoted as ordered pairs of the element followed by it's degree of membership of the set.

A fuzzy set can also be defined by a function determining the degree of membership of its elements.

Ex 1 (d):

A fuzzy set B = ' real numbers close to 15' can be described as :

$$B = \{ x, \mu_B(x) \mid \mu_B(x) = ((1+(x-15)^2)^{-1} \}$$

Fuzzy Sets - graphical representation

Fuzzy sets when used in engineering are more usually represented graphically. Example 1(d) can be described graphically by plotting element values versus membership values as follows:

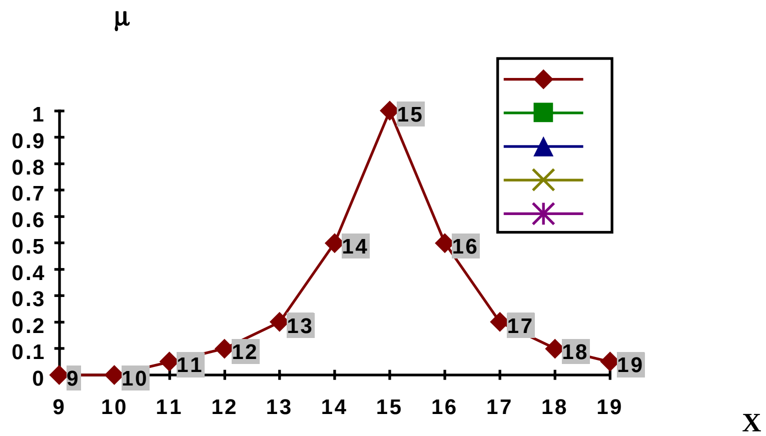


Figure 1

This example illustrates two important points about fuzzy sets. Firstly, a fuzzy set usually has associated with it a vague linguistic term which describes it in human terms. i.e. ' real numbers close to 15' or 'fairly warm' etc. Secondly, fuzzy sets can be described graphically to show how the membership function varies with element value. This graphical representation can be used when designing systems to show what values belong to which fuzzy set and to show the overlap between sets.

NB: The overlap is used to give smooth control or decision changes as input variables change, i.e. no abrupt changes from one set of conditions to another which would cause a step response in the controlled plant. An example is shown in Fig 2,

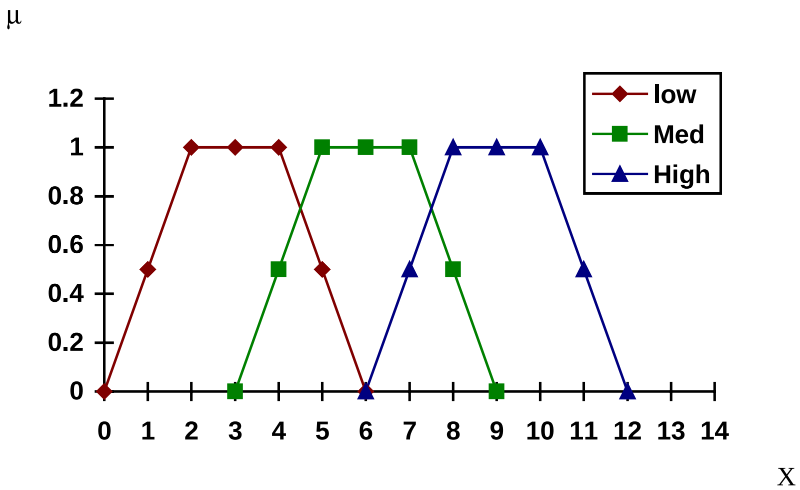


Figure 2

Research has shown that for most situations where fuzzy sets are used, the shape of the fuzzy set is always convex as shown in Figures 1 and 2.

Non-convex sets can lead to instability in certain situations and erroneous results in such areas as decision making and expert systems.

A test can be carried out on a fuzzy set to ensure that it is convex by comparing the elements to fit the following criteria.

$\mu_A\ (\lambda x_1 + (1-\lambda)x_2) \geq \text{MIN}\ (\mu_A(x_1)\ ,\ \mu_A(x_2))\ \ x_1,x_2 \in X\ ,\ \lambda \in [0,1]$

Perform this check on the fuzzy set below:

fuzzy set B = ' real numbers close to 15' can be described as :

$B = \{\ x,\ \mu_B(x)\ \mid\ \mu_B(x)= ((1+(x-15)^2)^{-1}\ \}$

Basic Set-Theoretic Operations on Fuzzy Sets

Just as we defined new crisp sets by applying set-theoretic operations we can apply such operations to fuzzy sets to create a new set. For example the intersection of two crisp sets gave a new set which was the set of all elements that both original sets had in common. How does this work with fuzzy sets?

The Intersection of two fuzzy sets

The MIN operator governs the intersection of two fuzzy sets.

Definition:

$C = A \cap B$, is defined pointwise by the function:

$\mu_C\ (x)\ = \min\ (\ \mu_A\ (x),\ \mu_B\ (x)\ )\ \ \ x \in X$

i.e. the lowest value of membership function is mapped into the fuzzy set C for that element.

This can be shown graphically by the following example.

Ex 1 (e): Example of the MIN operator.

Let C = { (1.2,0.3) , (1.3,0.7) , (1.4,0.9) , (1.6,1) , (1.8,0.8) , (1.9,0.4) , (2, 0.3)}

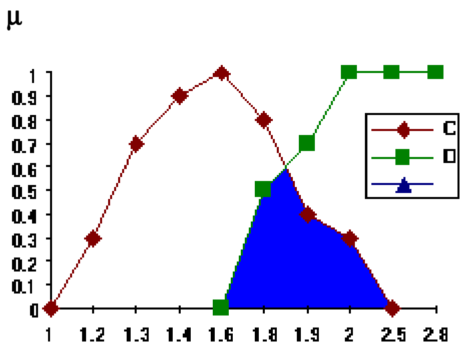
be the set of 'comfortable size car' from Ex 1 (c).

and D be the set of 'cars with large engines' defined as:

D = { (1.8,0.5) , (1.9.0.7) , (2,1) , (2.5,1) , (2.8,1) }

Then $F = C \cap D =$ { (1.8,0.5) , (1.9,0.4) , (2,0.3) }

The fuzzy set F describes 'comfortable size cars with a large engine' or it could be described as 'comfortable size cars and with a large engine'. The intersection is shown graphically below as a shaded area.



Litres.

Figure 3

Intersection of two fuzzy sets.

So in fuzzy sets, an "AND" function describes the area that both sets have in common i.e. intersection.

The union of Two fuzzy sets

The union of two fuzzy sets and is governed by the MAX operator.

Definition:

$C = A \cup B$ is pointwise defined by the function:

$\mu_C(x) = \max(\mu_A(x), \mu_B(x)) \quad x \in X$

i.e.. the highest value of membership is mapped into the fuzzy set C for that element.

Ex 1 (f): An example of the MAX operator.

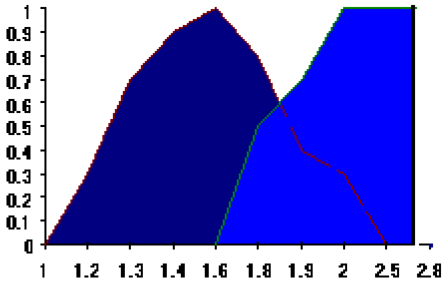
Using the same fuzzy sets C and D as in Ex 1 (e) , then

$G = C \cup D$

G = { (1.2,0.3) , (1.3,0.7) , (1.4,0.9) , (1.6,1) , (1.8,0.8) , (1.9,0.7) , (2,1) , (2.5,1) , (2.8,1) }

The fuzzy set G now describes ' comfortable size cars or cars with a large engine'.

The union of sets C and D is shown graphically below as the shaded area.



μ Litres

Figure 4

Union of two fuzzy sets

So, in fuzzy sets an "OR" function describes the area which meets the requirements of both sets to the highest degree of truth.

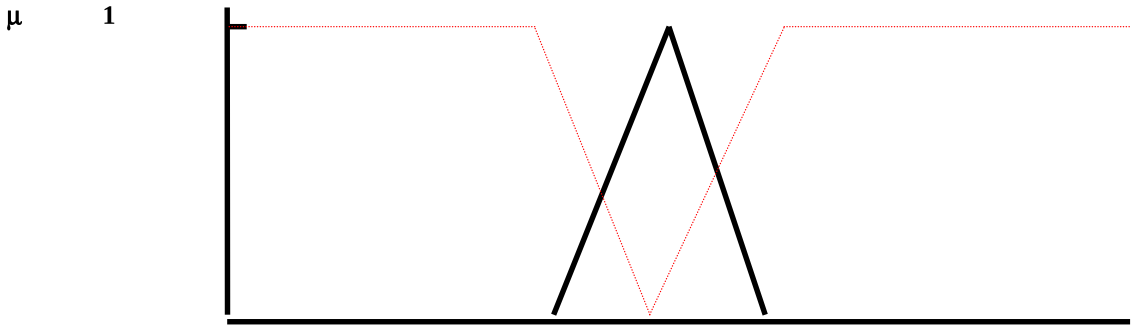
The Complement of a Fuzzy Set

The complement of a fuzzy set defines the degree of truth of all elements that are not compatible with the concept represented by the fuzzy set e.g. NOT A.

Definition:

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x) \quad x \in X$$

Graphically, this can be represented as all the area that is not part of the fuzzy set.



It is useful in clearly representing what is NOT membership of a fuzzy set in fuzzy control rules.

Other Operations on Fuzzy Sets

Algebraic Sum

Definition:

The algebraic sum of two fuzzy sets A and B is defined as :

$C = A + B$, $C = \{ x, \mu_{A+B}(x) \mid x \in X$

where $\mu_{A+B}(x) = \mu_A(x) + \mu_B(x) - \mu_A(x) . \mu_B(x)$.

Bounded Sum

Definition:

The bounded sum of two fuzzy sets A and B , $C = A \oplus B$, is defined as :

$$C = \{ x, \mu_{A\oplus B}(x) \mid x \in X$$

where $\mu_{A\oplus B}(x) = \min\{1, \mu_A(x) + \mu_B(x)\}$

Bounded Difference

Definition:

The bounded difference of two fuzzy sets A and B , $C = A \bigcirc B$, is defined as:

$$C = \{(x, \mu_{A\bigcirc B}(x)) \mid x \in X\}$$

where $\mu_{A\bigcirc B}(x) = \max\{ 0, \mu_A(x) + \mu_B(x) -1\}$

Algebraic Product

Definition:

The algebraic product of two fuzzy sets A and B , C = A•B, is defined as:

$$C = \{ (x, \ \mu_A (x) . \ \mu_B (x)) \mid x \in X \}$$

Ex 1 (g): An example of the other operators.

These operators are illustrated for the following example for fuzzy sets A and B.

A = { (2,0.4) , (3,0.7) , (4,1) , (5,0.6)}

B = { (3,0.5) , (4,0.8) , (5,1) , (6,0.6)}

A + B = { (2,0.4) , (3,0.85) , (4,1) , (5,1) , (6,0.6) }

A ⊕ B = { (2,0.4) , (3,1) , (4,1) , (5,1) , (6.0.6) }

A O B = { (3,0.2) , (4,0.8) , (5,0.6) }

A•B = { (3,0.35) , (4,0.8) , (5,0.6) }

NB: It should be noted that where an element has no ordered pair in a fuzzy set , it's existence with a membership function of 0 is implied.

These operators have been used to model the intersection and union of fuzzy sets and have provided alternatives to the MIN/MAX operators. Their use is dependent on the particular application and their adaptability to it. However, certain operators who exhibit common properties are often grouped together.

Fuzzy Logic and Bivalent Logic

How do these operators relate to the traditional binary logic operations?

Fuzzy Union and the OR Function

In traditional logic equations the OR operator returns the value of input that has the highest value i.e. a logic 1 on any input of an OR function ensures that the output is a logic 1.

The fuzzy union returns the value which meets the requirements of both sets to the highest degree of truth and is governed by the MAX operator.

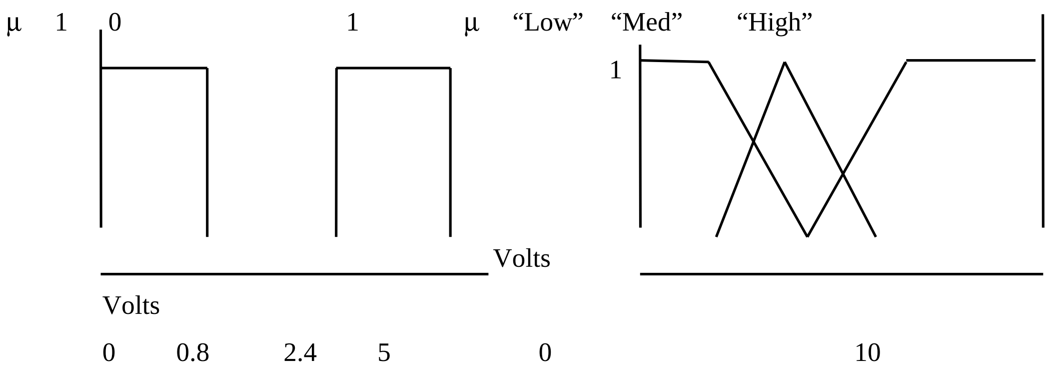
Fuzzy Intersection and the AND Function

In the bivalent logical AND operation the lowest value , 0 , controls the output condition. So if one of the inputs is logic 0 , the output is logic 0.

The same idea is covered in fuzzy set theory by the intersection of two fuzzy sets, which returns the value which meets the requirements of both sets to the lowest degree of truth and is governed by the MIN operator.

Bivalent Logic vs. Fuzzy Logic

If we were to compare graphically traditional bivalent logic with fuzzy logic , they would look as follows.



(a) (b)

Traditional binary logic shown as two non-overlapping sets with all values in each set having maximum grade of membership.

(a) i.e. it is a particular case of fuzzy set theory.

(b) Three fuzzy sets shown with overlap. The varying degree of memberships allows us to be vague about whether a value belongs to each of these sets.

Neural Networks

Introduction

What are neural networks?

What are they used for?

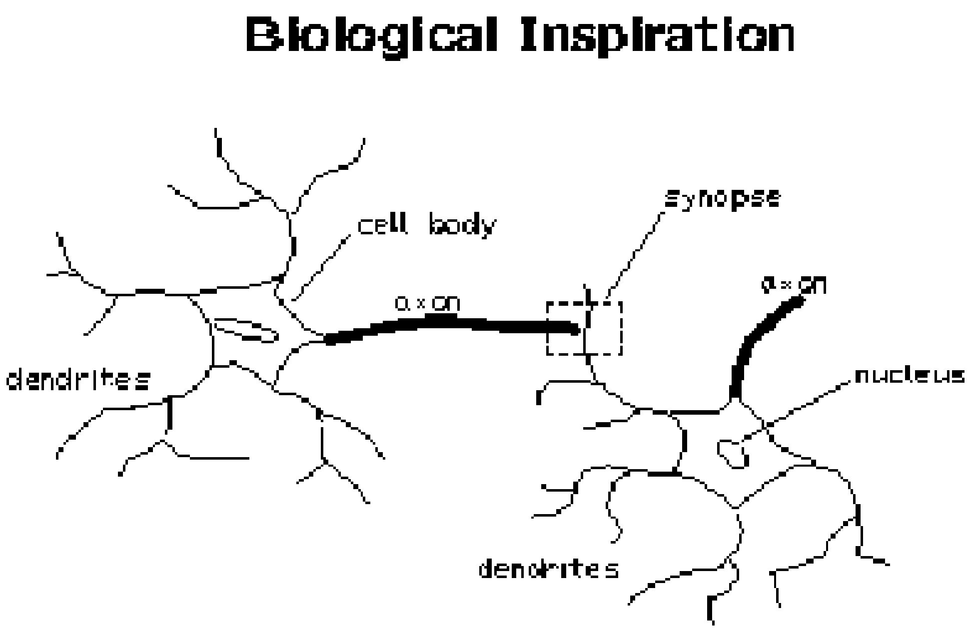
How do they operate?

What different types are there?

How are they related to fuzzy logic?

Human Brain

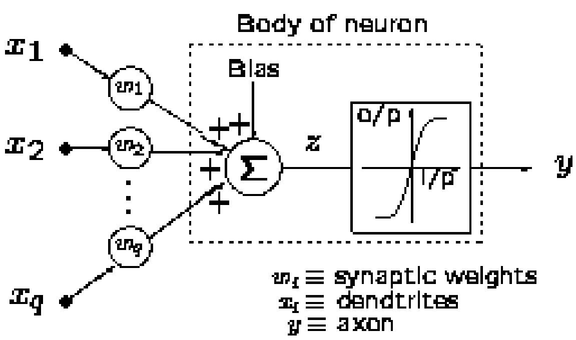
The human brain represents the best computational "machine". The brain consists of many neurons which when working together performs tasks way beyond what computers can do. Scientists since the 1940's have been fascinated with creating a machine that would work like the human brain neurons.



Characteristics of Biological Neurons.

- A typical brain contains between 10^{10} and 10^{12} neurons.
- Neurons are connected to each other in a complex spatial arrangement to and from the central nervous system.
- The cell body is typically a few microns in diameter.
- The ‘hair like’ dendrites are tens of microns in length and receive incoming signals.
- The axon is the neuron output and is 1mm to 1m in length. It can branch at its extremity, allowing it to connect with a number of other neurons.
- A single neuron may be directly connected to hundreds or even tens of thousands of other neurons.
- Signals are transmitted by electromechanical means.
- Pulse propagation speed ranges from 5 to 125 m/s .
- A delay of 1ms exists for pulses to traverse the synapse, via the generation of chemical substances called ‘*neurotransmitters*’.
- It is thought that, due to increased cell activity, metabolic growth takes place at the synaptic junction, which can increase its area and hence its ‘weight’. (So practice does pay off!)
- If the ‘sum’ of the signals received by a neuron exceeds a threshold, the neuron ‘fires’.
- Neurons can fire over a wide range of frequencies but always fire with the same amplitude.
- After carrying a pulse, an axon fiber is in a state of complete non-excitability for the duration of the ‘refractory period’, about 10ms.
- Information is frequency encoded on the emitted signals.

A Model of an Artificial Neuron.



ANN mimic of biological neuron

The following diagram is the common model used to mimic a brain’s neuron.

Formula:

$Y = f(\sum_{i=1} w_i x_i + w0)$

The inputs, X*i*, are analogous to the dendrites, *y* to the axon and *wi* to the synaptic weights.

These models are a much simplified version of a real neuron.

These ANNs are joined together to form ***networks*** which can then perform tasks.

Brief History of ANN Development

Origins

1943 McCullough-Pitts neuron

1949 Hebbian learning

The First Golden Age

1958 von Neumann's input

1958 Introduction of Perceptron (Rosenblatt, Block, Min- sky & Papert

1960 Widrow-Hoff learning rule (Delta rule, LMS) and Ada- line

The Quiet Years

1972 Kohonen - associative memory neural nets.

1977 Anderson - associative memory nets.

1967-88 Grossberg - 146 publications with math. and bio- logical treatment

‘Renewed enthusiasm'

1983 Bolzmann machine

1985 Carpenter - adaptive resonance theory (ART)

1986 Backpropagation training rule (Werbos, McClelland & Rummelhart)

1987 Hopfield networks (associative memory nets)

1987 VLSI and optical implementations of ANNs

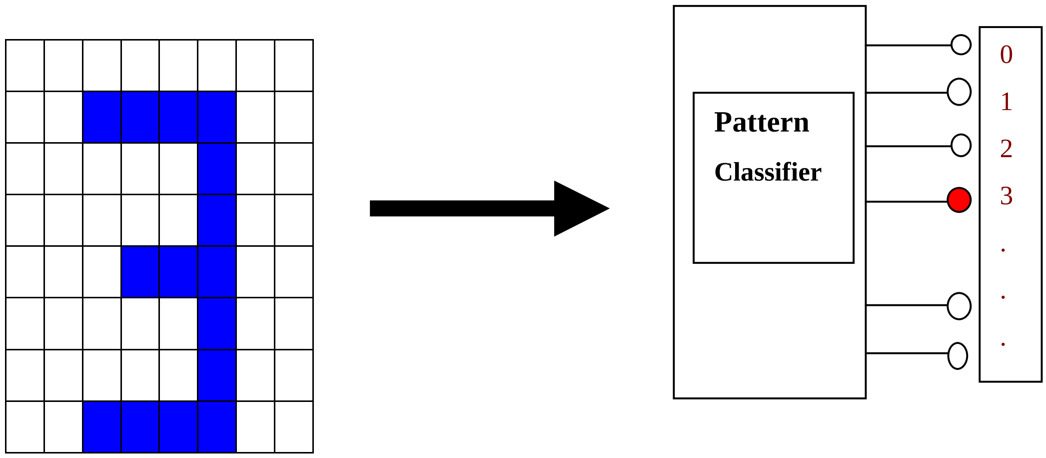
1988 Fukushima - neocognitron

What does a neural network do?

The function of a neural network is to produce an output pattern when presented with an input pattern.

Pattern Classification

Your brain is interpreting these words and letters as patterns, which is easier as they are typed. Trying to interpret hand written text is a pattern recognition problem that is hard but can be implemented using an ANN. Using an electronic memory to build a pattern recogniser for say 10 classes to recognise the numbers from 0-9 is not feasible.



There are 64 inputs (8x8) so there are $2^{64} = 1.8 \times 10^{19}$ different input patterns.

So a memory with 1.8×10^{19} locations would be needed.

Each location must be 10 bits to represent the 10 different output values.

This would require something like 170 million 1Gbyte chips!

If 1 pattern is processed each second it would take 600,000,000,000 years to program! Not feasible.

How is a neural network better?

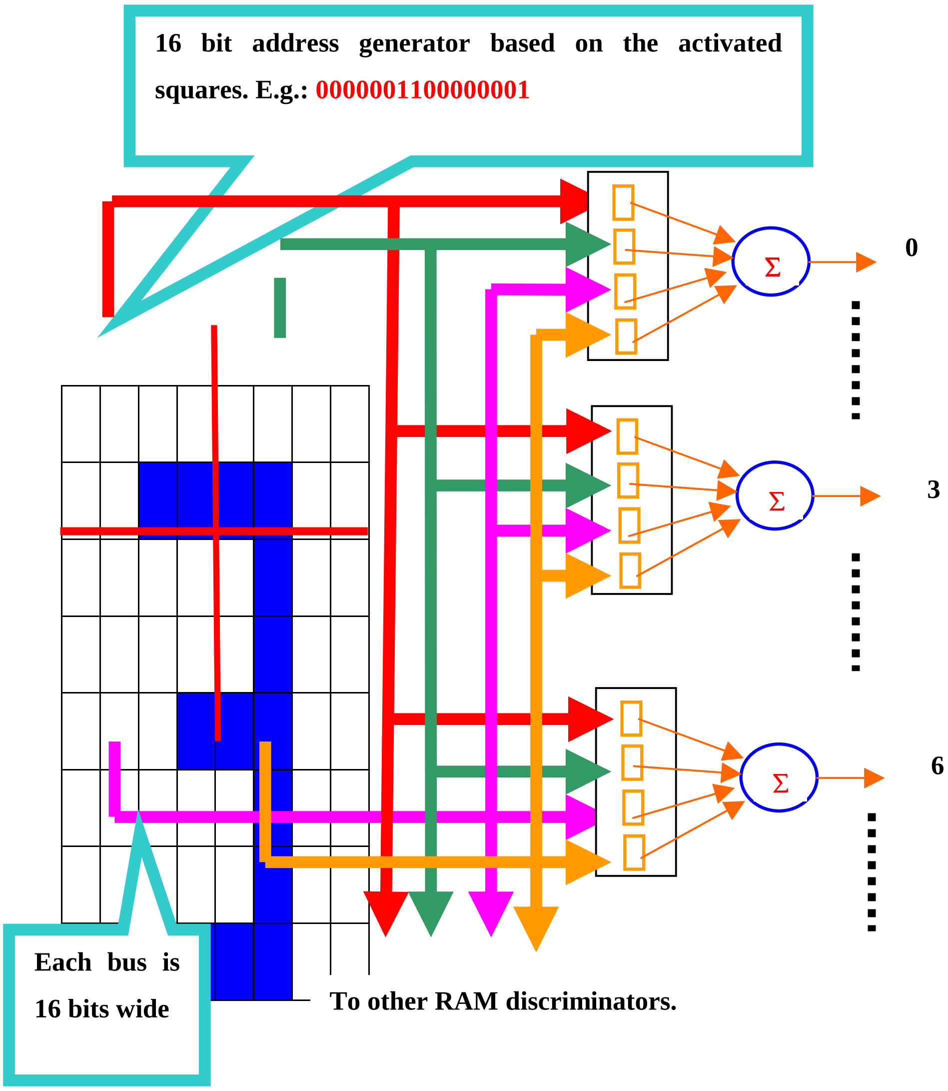
An ANN is better because it can:

- Learn from a small number of input patterns
- Generalise from the examples it has been shown
- Be small enough to be physically realisable.

Ram Discriminators

These are an example of a system that can recognise patterns, learns by example and can generalise. However, they are implemented using Boolean logic circuits and are thus known as Boolean Neural Networks. They allow the last example to be realisable.

Let's take our 8x8 image for classifying numbers 0-9.



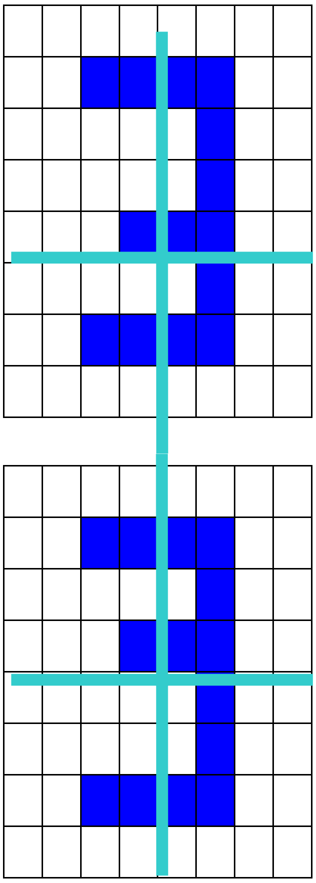
Process for the above example.

- We break the image into 4 quadrants.
- Each quadrant has 16 pixels which address a RAM memory device. Therefore, each RAM needs 2^{16} = 64K locations.
- Instead of storing 10 bits, we now only store 1 bit but have 10 different RAM chips.
- Before 2^{64} locations with 10 bits were needed. Now we only need 10 sets of $4x2^{16}$ locations i.e. 70 billion times smaller!
- The outputs of all 4 RAMs are connected to a summing junction, so if all RAMs produce a 1 we get a maximum output of 4.
- Training is by showing the system a pattern such as 3. Then place a 1 in the 4 memory locations addressed in the '3' discriminator. The memory devices in the other discriminators are unchanged.
- If the system is shown several examples of the number 3, each example puts 1s in different memory locations.
- Now, if a pattern is shown to the system that matches one of these, that discriminator will give a maximum output, 4. Others should be less than the maximum.

An interesting feature of this system is that it can **generalise.**

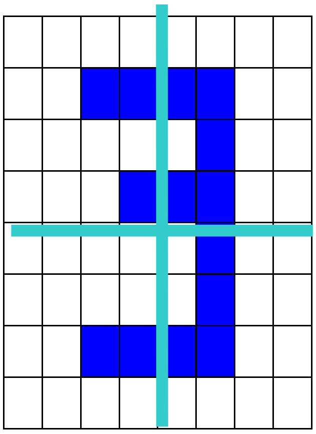
An example:

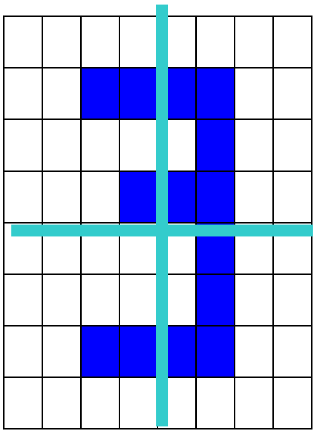
Let's say the following two patterns are used to train for a 3.

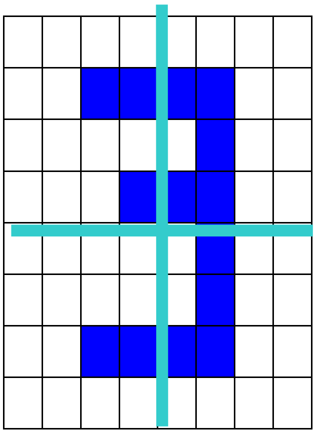


Even though we have only shown the system two patterns, there are 16 patterns that would give a maximum of 4 from the '3' discriminator. 14 of whom it never saw before i.e. it can generalise.

Here are 3 of those 14 other possibilities. They do look closer to a 3 than any other number so it is right that they should be classified as '3's.







So this is an example of a Pattern Classifier which can be *trained* and can *generalise*.

Biologically Inspired Neural Networks

Most of the different neural networks, apart from Boolean ones, are descended in some way from the McCulloch-Pitts neuron (1943).

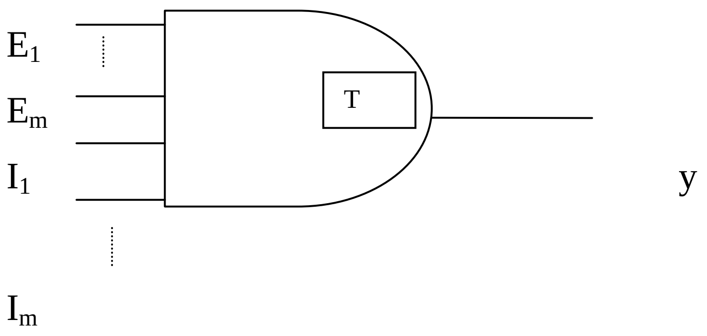


Figure 1 McCulloch-Pitts neuron (1943)

Figure 1 shows the basic idea behind the original neuron. It has excitory and inhibitory inputs. If any of the inhibitory inputs are active, the neuron won't fire. If none are and if the sum of the excitatory is greater than the threshold, we get an output. Mathematically:

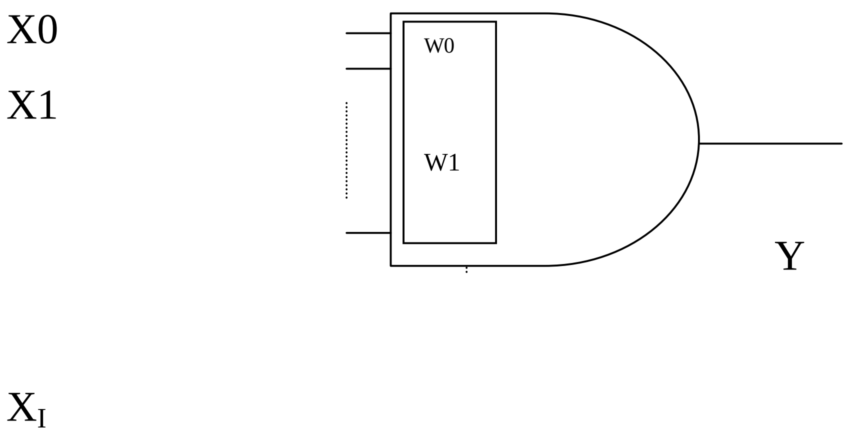
$$Y = 1 \text{ if } \sum_{i=1} I_i = 0 \text{ and } \sum_{j=1} Ej \geq T$$

$$Y = 0 \text{ otherwise.}$$

The problem with this early neuron model was that it lacked the ability to learn or generalise. Various modifications were made to it over the years including making the inhibitory inputs negative and adding weightings to the inputs. One of the earliest useful neuron models was the ADALINE. ADAptive LINear Elements developed by Windrow and Hoff(1960).

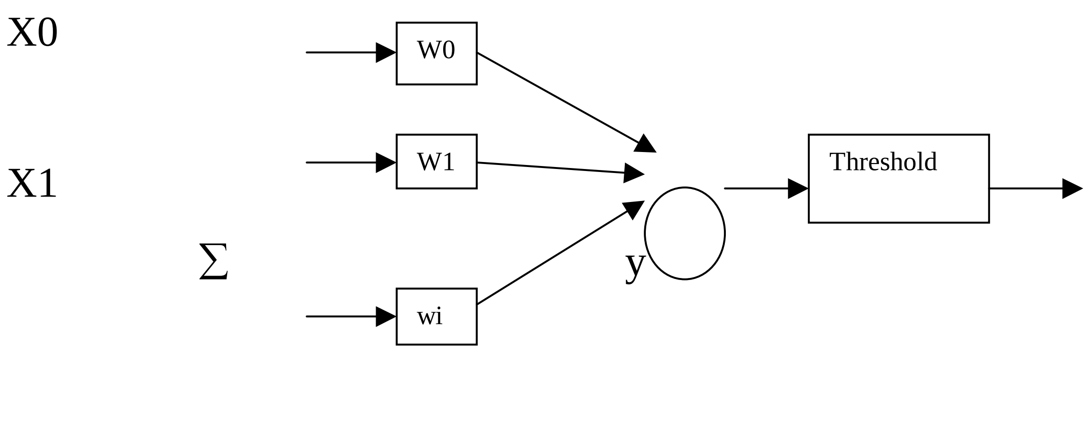
ADALINE

Figure 2 shows the basic Adaline model.



Each input can have a value of +1 or –1. Each input has a weight associated with it which is a real number and can be positive or negative.

Figure 3 Alternate Adeline Representation



Xi

Figure 3

It can also be represented as Figure 3 above. When a pattern is presented to the input, the weighted sum, called the net input, *net*, is found by multiplying the inputs with their corresponding w2eights and adding all the products together.

$net = \sum_{i=0} w_i x_i$

The additional input and weight, x0 and w0, provide a constant offset with x0 permanently set to +1.

This replaces the threshold that was used in the McCullough-Pitts neuron.

The value, *net*, is transformed into the output, y, by a non-linear output function. This function gives a +1 if the weighted sum is > 0. If sum ≤ 0 then the output is –1.

This is known as a Hard-Limeter.

Training this and other neural networks involves finding values for the weights.

Representing the No. 3 Problem with ADALINE

10 Neurons are required, one for each numeral.

When a "3" appears at the input of the 3 neuron, it should fire.

The diagram below shows whites represented by a –1 and blacks by a +1.

-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	**+1**	**+1**	**+1**	**+1**	-1	-1
-1	-1	-1	-1	-1	**+1**	-1	-1
-1	-1	-1	**+1**	**+1**	**+1**	-1	-1
-1	-1	1-1	-1	-1	**+1**	-1	-1
-1	-1	-1	-1	-1	**+1**	-1	-1
-1	-1	**+1**	**+1**	**+1**	**+1**	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1

One way to train a neural network to recognise this pattern as a 3 is to assign values to the weights so that for white cells it is –1 and for black cells the weight is +1.

So the value of *net* will be 64 when presented with a perfect 3

If the offset, w0, is set to –63, then the weighted sum becomes 64-63 = 1. This being >0 we get a +1 when passed through the hard-limiter.

If another pattern is presented the weighted sum will be at most –1 and possibly as low as –127, so the hard-limiter will give a –1 for all other patterns.

What if a pattern of 3 with 1 bit different appears?

With 1 bit different we get (63x+1) + (1x-1) = 62

With an offset, w0, set to 63 we get a sum of –1 which is less than 0 so that the hard-limiter will give a –1 out. It is rejected.

If the offset is adjusted to 61 then *net* sum for the corrupted 3 becomes +1 and it is accepted. All patterns with one corrupted pixel will be accepted.

Thus adjusting the weights adjusts the sensitivity of the neuron.

Different Learning Methods

When a neural network such as ADALINE is in its learning phase there are 3 things that have to be taken into account:

- The inputs that are applied are chosen from a training set where the desired response of the system to these inputs is known.
- The actual output generated when an input pattern is applied is compared with the desired output and used to calculate an error.
- The weights are adjusted to reduce the error.

This kind of training is called **supervised learning.**

Conclusion

The above examples give a flavour and an introduction to the areas of fuzzy set theory/logic and an example of basic neural network The area of fuzzy logic controllers which uses fuzzy set theory in a particular way to control electromechanical devices, such as a washing machine, is not covered. In the area of neural networks many other examples could be looked at to cover areas such as weighting, self-learning and indeed other models of neural networks. Anyone interested can contact me for more!

References

[1] Zadeh, L.A.**; " Fuzzy Sets" , Information and Control , 1965 pp 338-353.**

[2] Zimmermann, H.J.**; "Fuzzy Set Theory and it's applications", Kluwer Academic Publishers, 1990 . ISBN 0-7923-9075-X.**

[3] Mamdani E.H**.; "Application of fuzzy algorithms for control of simple dynamic Plant", Proc. IEEE Vol 121 , No. 12 , 1974, pp 1585-1588.**

[4] Toshinoro Munakata, Yashvant Jani**; "Fuzzy Systems: An Overview", Communications of the ACM, March 1994 Vol 37, No. 3 pp 69-76.**

[5] King, R.E., Karonis, F.C.**; "Multi-level expert control of a large Industrial Process"; Fuzzy Computing Theory, hardware and Applications, Amsterdam 1988 pp 323-340.**

[6] Pease, RA**; "Whats all this fuzzy logic, stuff, anyhow?", Electronic Design May and Nov 1993.**

[7] Nass R.**; "Fuzzy logic finally gets acceptance in the US", Electronic Design June 25 1992. Page 37-40.**

[8] Berardinis, LA**; "It washes, it rinses, it talks", Machine Design, Vol 63 Sept 1991.**

[9] Braae M., Rutherford D.A**.; "Theoretica & Linguistic Aspects of Fuzy Logic Controller", Automatica 15, 5 Logic Controller , Mar 1979 Pages 553- 577.**