Fuzzy Logic and Knowledge Based Systems

1 Fuzzy Logic - A Brief Overview

Several years ago, Steve Marsh, director of strategic operations at Motorola (Austin), visited Japan. Having checked into his hotel, he had only enough time to drop his baggage in his room before hurrying to dinner with a customer. He entered the elevator and pushed the button of his floor. The elevator doors closed, and, in a few seconds, opened again. Thinking that someone had pushed the elevator-request button, Steve held the door open and looked out, but no one was waiting. When he reached again to push the button for his floor, he discovered he was already there. The elevator had transported him several floors without his feeling any motion.

Being an engineer, Marsh was intrigued, so he commandeered the elevator and played awhile, moving between floors, both short and long distances, both up and down. Concentrating now, he could feel the car moving but was amazed at the lack of its jerk in both starting and stopping and at its smooth acceleration and deceleration. I don't know whether Steve made it to his dinner on time, but he later had Motorola personnel inquire about why the elevator's movement was so smooth. The explanation was simple: Fuzzy logic controls the elevators. Steve became a believer.

This half of the module deals with the highly successful practical technique of fuzzy logic. Before we start, it is worth mentioning that the term *fuzzy logic* is interpreted differently by different people. There is a whole mathematical logic based on fuzzy sets that is not covered by this module (you may or may not be pleased to know!!). I use the term in a colloquial sense, since you will be learning more than just fuzzy sets but will be getting exposure as to how they may be used to aid decision making, control equipment and so on. The intent is to expose you to enough 'fuzzy logic' to be able to implement a fuzzy system (software that deploys fuzzy sets) in Matlab. Before joining the fuzzy world let's explore why we may need something like fuzzy logic.

1.1 Uncertainty, Imprecision and Vagueness

Many problems in industry and commerce can be modelled mathematically and/or statistically. However, in practice, many applications can only be modelled in this way very poorly - there are important assumptions made that are often not valid and/or the results are 'brittle' and lack robustness. Standard control applications, for example, often work poorly under certain conditions or they are not smooth in their movement. Physical measurements are, by their nature, imprecise. they are only as good as the instrument doing the measuring. A 2 Kg bag of sugar is never 'exactly' 2 Kg for example. Yet, traditional mathematically based control solutions use these measurements as precise. Experts make decisions with imprecise data in an uncertain world. they work with knowledge that is rarely defined mathematically or algorithmically but uses vague terminology with words. Let's briefly consider one example taken from the web[1]

When it comes to heavy Seattle traffic, 'fuzzy logic' smooths the flow

Sure, Seattle was ranked third earlier this month among U.S. cities for the amount of time drivers spend stuck in traffic. And just this week, the intersection of Interstate 5 and I-90 was named one of the worst in the nation. But next time you're suffering oh so slowly through the rush-hour mess, ponder this: It would likely be a lot worse if a new "fuzzy logic" regulating system weren't in the driver's seat.

Although it sounds imprecise, fuzzy logic can deliver exacting answers about the ever-changing status of area freeways to help move traffic more efficiently, according to researchers at the University of Washington. After several months of testing at selected on-ramps last spring, the system performed so well that it now dictates traffic flow on all 126 of the metered freeway on-ramps in the Seattle area. And UW researcher Cynthia Taylor said she's getting calls from traffic engineers around the country interested in learning more about her method.

"There seems to be a lot of interest," said Taylor, a research engineer with the UW's Department of Electrical Engineering. "We are very pleased with how well it seems to be working."

Taylor, working with Deirdre Meldrum, UW associate professor of electrical

[1] http://www.washington.edu/newsroom/news/1999archive/11-99archive/k112499.html

engineering, designed the new fuzzy logic algorithm that has replaced the old "threshold" method that regulated the metered freeway on-ramps. The metered ramps feature traffic lights to control how quickly vehicles leave the ramps to merge with mainline traffic. The old method worked on a sort of "yes-no" scenario, according to Taylor. It generally wouldn't take action until after traffic had reached a threshold limit. By that time, a problem had often already developed and would persist for hours before it could be worked out.

Fuzzy logic, on the other hand, uses smooth, continuous control to prevent or delay congestion.

"It uses linguistic variables and rule-based logic," Taylor said. "Because it is similar to the way we talk and think, it is easier for us to adjust."

One key fuzzy logic feature is its ability to balance conflicting objectives. That's a primary issue in managing traffic. An effective system must be able to strike the precarious balance between keeping freeway traffic flowing and avoiding long lines of vehicles on the ramps, waiting for a chance to merge.

Fuzzy logic can also operate with incomplete data, making decisions based on what information is available. That's critical, says Taylor, because up-to-the-minute traffic information can be spotty.

"We may not get data because of construction, communication problems and hardware failures, and data that we do get may be inaccurate," she said.

In addition, the fuzzy logic algorithm is forward-looking. Information from each on-ramp site is considered with information gathered from known trouble spots farther down the road. As problems begin to develop, fuzzy logic anticipates them by taking action upstream to mitigate the trouble.

Such characteristics set the stage for an impressive showing when the fuzzy logic method was tested beginning in March. Researchers alternately used fuzzy logic and the old system at two sites - Interstate 90 westbound from State Road 900 in Issaquah to Eastgate in Bellevue, and Interstate 405 southbound from N.E. 160th Street in Bothell to N.E. 72nd Place in Kirkland - to make a comparison.

On I-90, the fuzzy logic system produced an 8.2 percent reduction in freeway congestion - significant enough to be noticeable on a day-to-day basis. It also prevented a bottleneck near the Eastgate on-ramp, a task at which the old algorithm failed. Results show that traffic flow was 4.9 percent better with fuzzy logic.

On I-405, fuzzy logic produced slightly higher freeway congestion and slightly higher flow. The trade-off was that lines at the on-ramps were much shorter using fuzzy logic. According to test data, unacceptably long on-ramp lines occurred nearly twice as often using the old system as they did with fuzzy logic. So fuzzy logic significantly decreased the ramp queues while maintaining mainline efficiency.

Numerous factors, including weather, accidents and daily variations in traffic, make an exact comparison difficult, Taylor said. But the test results indicate that, as a whole, fuzzy logic reduces total travel time and increases flow. It's also very easy to adjust, so traffic engineers can readily fine-tune it to meet current needs - a big plus in a freeway system as large and variable as Seattle's.

As a result, transportation officials opted to begin implementing the fuzzy logic algorithm systemwide after test results were available at the end of the summer.

Taylor said she believes the key to fuzzy logic's success lies in its flexibility - its ability to cope with imperfect input and adapt as the situation changes.

The freeway system is chaotic and non-linear, she said. Accidents, special events and bad weather make sudden changes in traffic more the rule than the exception. That underlines the need for a system that can respond to a wide variety of conditions.

"It's really more complex to be vague," Taylor said. "This algorithm is able to use that vagueness to achieve a precise answer. By considering shades of gray and all factors simultaneously, you get a better answer, one that is more suited to the situation."

This is a good example to illustrate how fuzzy logic doesn't ignore vagueness but uses it to produce a better solution.Fuzzy logic relies on the concept of a *fuzzy set*. Zadeh introduced the notion of fuzzy sets in 1965 in a seminal paper - Fuzzy Sets, Information and Control, 8, 338-353. He is a Professor at the University of Southern California and is still active in the field. The idea of fuzzy sets described in his seminal work lays the basis for Fuzzy Logic.

Fuzzy Logic is particularly good at handling uncertainty, vagueness and imprecision. This is especially useful where a problem can be described linguistically (using words) or, as with neural networks, where there is data and you are looking for relationships or patterns within that data. Fuzzy Logic uses imprecision to provide robust, tractable solutions to problems. Applications of fuzzy logic include robotics, washing machine control, nuclear reactors, focusing a camcorder, information retrieval, train scheduling, system modelling, and stock tracking on the Nikkei stock exchange.

Computer systems that employ a fuzzy set approach allow for vagueness and imprecision - the way humans work!! Indeed the latest term deployed by Zadeh is ‘Computing with Words’. This is what makes the approach appealing, exciting and yet practical. In Japan, some consumer products are sold as being ‘fuzzy’.

You should be aware that fuzzy logic has a 30+ year history and there have been tens of thousands of papers published, as well as thousands of applications. This module focuses on how fuzzy logic allows for the development of computer systems that employ fuzzy rules (from now on we shall call these computer systems *fuzzy systems*).

2 Fuzzy Sets

The best way to start the discussion about fuzzy sets is via a simple example. In many circumstances where we are writing software to aid in the decision making process an expert would provide some knowledge that would be considered vague or imprecise. In this first foray into the fuzzy world let’s consider the case of an expert in ‘tallness’ describing people as tall and in some way wanting to use that information to make a decision. For example, we might want to develop a computer system that is a sport adviser which has a rule in it that says

if you are tall and can run fast you should consider basketball

If we wanted to, in some way, implement the concept of tallness (and, indeed, fastness) in a computer without fuzzy logic we would have to have boundaries for the word tall. For example in Figure 1 we see that I have placed a lower boundary on the word tall of 5ft 11ins and an upper boundary of 7ft (this is a simple example and I am assuming we are dealing with men, you can’t be over 7ft tall and I am afraid I can only work in feet and inches!!)The way I have presented this is that we have a function that can only take a value one or zero. In this case the function for tall takes the value one for all points from 5ft 11in to 7ft. This approach follows

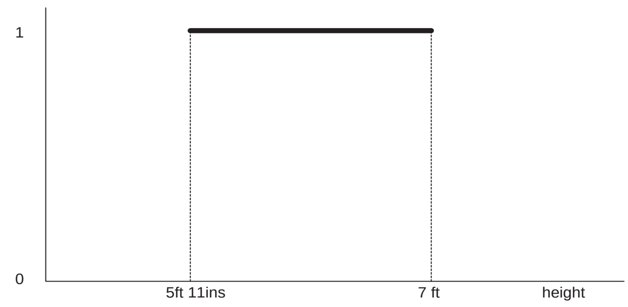

Figure 1: A crisp way of modelling tallness

that of traditional set theory. To find out whether I belong to the set tall I measure my height and if it falls between these two heights, I am tall - otherwise I am not. Now, suppose we wish to consider short in the same way we would perhaps come up with the function represented in Figure 2.Note that in both these examples we

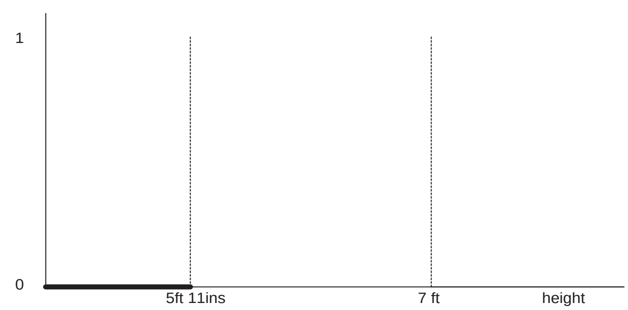

Figure 2: The crisp version of short

have a measurement that maps onto a number (zero or one). I have done this deliberately to help illustrate the notion of a fuzzy set. There are some problems with this (boolean) approach. Firstly it is very crisp. If you are 5ft 10 ins you are short but someone who is only one inch taller than you is tall. This surely isn't right? Also if you are 6ft you are as tall as someone who is 6ft 11 ins. Again, an expert would differentiate between people of different heights in some way. Perhaps for example he might introduce the terms quite tall, tall and very tall and these could be represented as in Figure 3.This doesn't seem satisfactory though

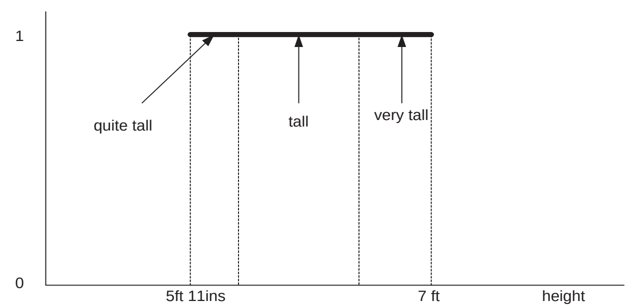

Figure 3: Some crisp tall definitions

because we have these 'cut off' points.

Fuzzy sets treat such an imprecise notion as tallness in a completely different way. The central idea behind fuzzy sets is that something (or someone) can belong to a set to some degree. The notion that somebody is tall or not tall does not make sense. People are tall to some degree. Graphically we might represent the fuzzy set tall as in Figure 4. The shape you see is known as the *membership function* - more of this later. In Figure 5 we see, superimposed on the membership function the heights 5ft 11in and 7ft. In this case someone who has height 5ft 11ins is tall to degree 0.2 (ish!). However someone who has height 7ft is tall to degree 1.

What about the fuzzy set short? Figure 6 gives a possible membership function. We can now put the membership functions for tall and short on the same diagram as in Figure 7. Notice here that someone could belong to both the fuzzy sets at the same time. This is particularly attractive in real applications since this reflects real life. We are (almost) all tall AND short - to some degree. This is a difficult concept to grasp and flies in the face of the traditional logic based paradigm

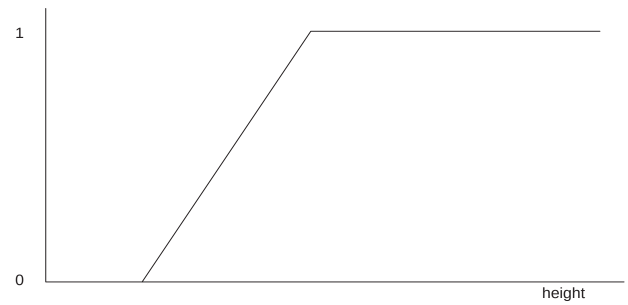

Figure 4: A possible fuzzy set tall

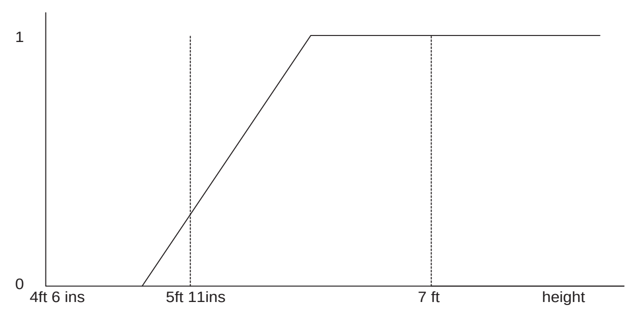

Figure 5: A possible fuzzy set tall with some example heights

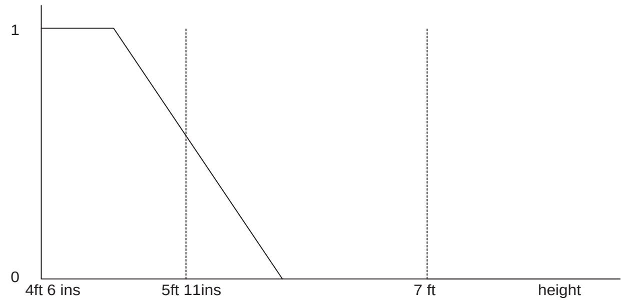

Figure 6: A possible fuzzy set short

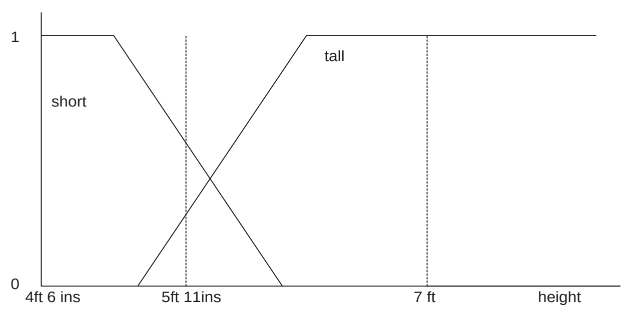

Figure 7: The membership functions for tall and short

that we have been brought up with - the law of excluded middle.

In other words you can belong to a fuzzy set to some degree. So, Michael Jordan might belong to the fuzzy set tall to degree 0.9 (on a scale of zero to one) where as I might be tall to degree 0.6 and Danny DeVito might be tall to degree 0. We now introduce a formal definition of a fuzzy set.

For any fuzzy set, A, the function μ_A represents the membership function for which $\mu_A(x)$ indicates the degree of membership that x, of the universal set X, belongs to set A and is, usually, expressed as a number between 0 and 1:

$$\mu_A(x) : X \rightarrow [0,1]$$

Fuzzy sets can be either discrete or continuous. The notation for fuzzy sets can be confusing initially. For the member, x, of a discrete set with membership μ we use the notation μ/x . In other words, x is a member of the set to degree μ.

Discrete sets are written as:

$$A = \mu_1/x_1 + \mu_2/x_2 + \ldots\ldots\ldots + \mu_n/x_n$$

or

$$A = \sum_{i=1,n} \mu_i/x_i$$

where $x_1, x_2 \ldots . x_n$ are members of the set A and $\mu_1, \mu_2, \ldots ., \mu_n$ are their degrees of membership. A continuous fuzzy set A is written as:

$$A = \int_X \mu(x)/x.$$

This notation can be confusing and takes some getting used to but is the accepted notation in the fuzzy community.
The key points to draw from the discussion to this point are:

- The members of a fuzzy set are members to some degree, known as a *membership grade* or *degree* of membership.
- A fuzzy set is fully determined by the membership function.
- The membership grade is the degree of belonging to the fuzzy set. The larger the number (in [0,1]) the more the degree of belonging.

- The translation from x to $\mu_A(x)$ is known as *fuzzification*.
- A fuzzy set is either continuous or discrete.
- Graphical representation of membership functions is very useful.

Let us now look at some more simple examples. Suppose we want to define the set of integers "close to 1". The membership function might be:

$$A = 0.0/-2 + 0.3/-1 + 0.6/0 + 1.0/1 + 0.6/2 + 0.3/3 + 0.0/4$$

In other words 0 has a degree of membership in the fuzzy set closetoone of 0.6 which indicates that it is more closetoone than, say 3 which has a degree of membership of 0.3. This membership function is represented graphically in Figure 8.

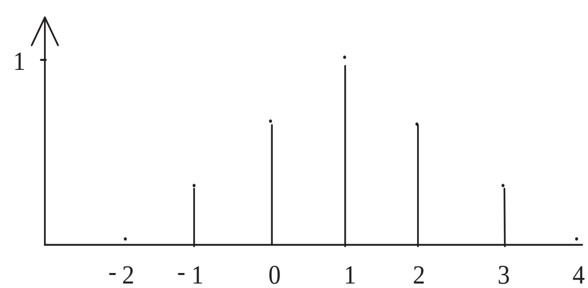

Figure 8: The Membership Function "Close to 1"

Let's look at another example. Suppose we wish to describe people as "Young", "Middle-aged" and "Old". The membership functions might look like those in Figure 9. Again, notice the overlapping of the sets reflecting the real world more accurately than if we were using a traditional approach.

Fuzzy logic allows for modelling of linguistic terms using linguistic variables and linguistic values. The fuzzy sets 'Young', 'Middle-aged' and 'Old' are fully

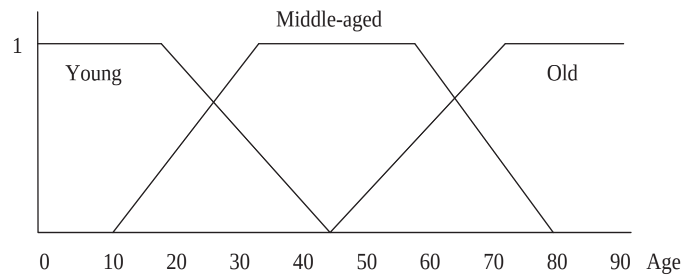

Figure 9: The Membership Functions Young, Middle-aged and Old

defined by their membership functions . The *linguistic variable* “Age” can therefore take linguistic values. Let’s briefly look at how this approach helps with imprecision and linguistic uncertainty.

2.0.1 Imprecision

As has already been discussed, in many physical systems measurements are never precise (a physical property can always be measured more accurately). There is imprecision inherent in measurement. Fuzzy numbers are one way of capturing this imprecision by having a fuzzy set representing a real number where the numbers in an interval near to the number are in the fuzzy set to some degree. So, for example, the fuzzy number ‘About 35’ might look like the fuzzy set in Figure 10 where the numbers closer to 35 have membership nearer unity than those that are further away from 35. The number 35.2 might belong to the fuzzy set ‘About 35’ to degree 0.8 whereas 35.8 might have a membership grade of 0.3 in the same fuzzy set.

2.0.2 Vagueness or Linguistic Uncertainty

Another use of fuzzy sets is where words have been used to capture imprecise notions, loose concepts or perceptions. We use words in our everyday language

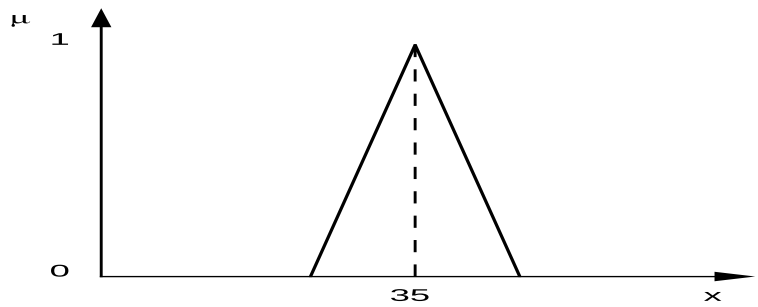

Figure 10: The fuzzy number 'About 35'

that we, and the intended audience, know what we want to convey but the words cannot be precisely defined. For example, where a bank is considering a loan application somebody may be assessed as a good risk in terms of being able to repay the loan. Within the particular bank this notion of a good risk is well understood. So, for example, on a scale of one to one hundred the fuzzy sets 'Good Risk' might look like the membership function in Figure 11. In other words it is not a

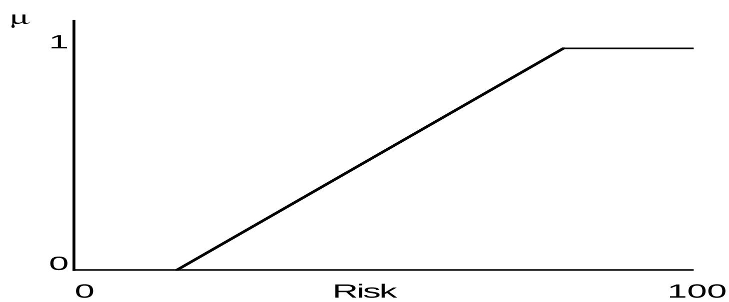

Figure 11: The fuzzy set 'Good Risk'

black and white decision as to whether someone is a good risk or not - they are a good risk *to some degree*. This use of fuzzy sets to capture the uncertainty in linguistic terms is the most common in decision making situations. For a particular bank someone, an expert in banking perhaps, would have to assign a number in [0,100] to that bank if the fuzzy set 'Good Risk' is to be used in a real system about the bank. Risk in a banking environment is a good example of a concept or perception that cannot be directly measured necessarily. An expert weighs up a number of factors - some directly measurable, some not (e.g. strong management of the company) - to arrive at the risk associated with the loan.

2.1 Fuzzy sets and probabilities

Fuzzy sets are not probabilities. The difference is best explained by an example. Suppose you have a bottle of liquid where you don't know the contents. You are told that there is a probability of 0.7 that it is drinkable. A fuzzy enthusiast also tells you that the liquid is drinkable to degree 0.7 where drinkable is a fuzzy set. What is the difference? In the first instance there is a 0.3 chance that it is not drinkable - this is risky. In the fuzzy situation it just means that it might not be as nice as if it were drinkable to degree 0.9! It measures in some way how drinkable it is whereas, in the probabilistic case, it is saying that if you had ten bottles of the same liquid then, on average, three would be disgusting or poisonous!

2.2 Summary

The strength of fuzzy logic is that we are able to model words by the use of fuzzy sets. Since the world is inherently imprecise or vague then it has much to offer. It is used in many applications from consumer devices to controlling subway trains.

3 Union, Intersection and Complement of Fuzzy Sets

Fuzzy logic has a well-established theoretical base. However, for practical implementations there are a relatively small number of operations required. This section concentrates on providing enough of a theoretical base for you to be able to implement computer systems that use fuzzy logic. Two particularly important operations are intersection and union which correspond to 'AND' and 'OR' respectively. These two operators will become important later as they are the building blocks for us to be able to compute with fuzzy if-then rules. First let us look at boolean AND and OR. Table 1 is the truth table for AND and Table 2 provides the truth table for OR.

Table 1: Logical AND

A	B	A and B
0	0	0

Table 2: Logical OR

A	B	A or B
0	0	0

Another term for AND is the intersection and for OR is the union. Figures 12 and 13 provide a diagrammatic representation of intersection and union for crisp (traditional) sets.

The shapes represents crisp sets. If an object falls within the shape it belongs to that set. The shading represents the union or intersection.

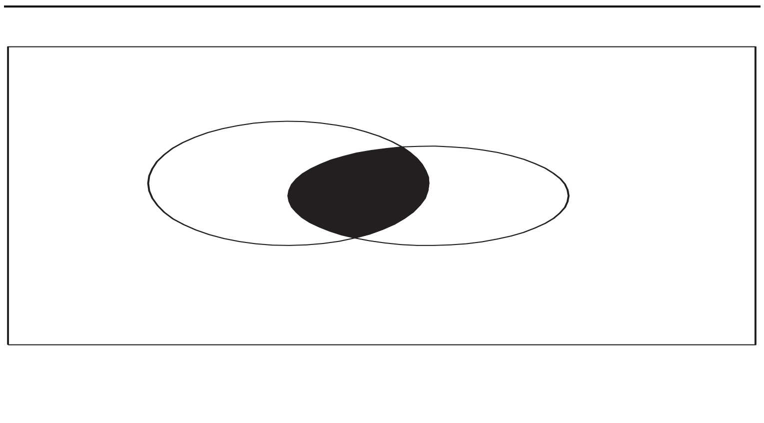

Figure 12: Crisp Intersection

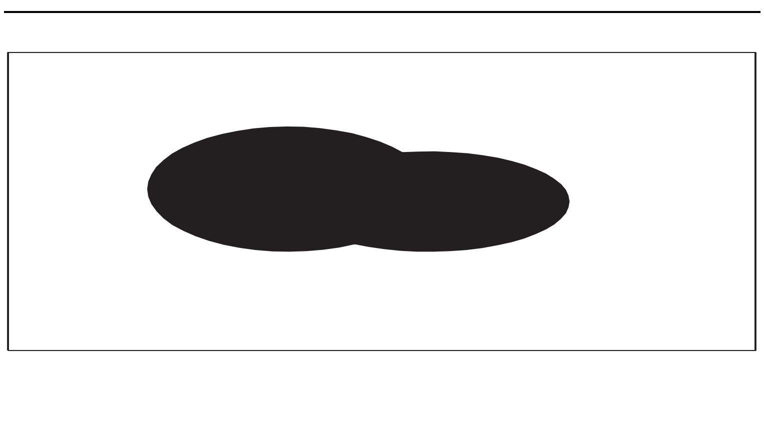

Figure 13: Crisp Union

So, what is the fuzzy equivalent?

3.1 Fuzzy Union

The union (OR) is calculated using t-conorms. A t-conorm operator is a function s(.,.) where the following holds:

s(1,1) = 1, s(a,0) = s(0,a) = a (boundary)
s(a,b)$\leq$ s(c,d) if a $\leq$ c and b $\leq$ d (monotonicity)
s(a,b) = s(b,a) (commutativity)
s(a,s(b,c)) = s(s(a,b),c) (associativity)

By far and away, the most commonly used for fuzzy union is to take the maximum. That is, given two fuzzy sets A and B with membership functions $\mu_A(x)$ and $\mu_B(x)$

$$\mu_{A\cup B}(x) = max(\mu_A(x), \mu_B(x))$$

3.2 Fuzzy Intersection

The intersection (AND) is calculated using t-norms. A t-norm operator is a function t(.,.) where the following holds:

t(0,0) = 0, t(a,1) = t(1,a) = a (boundary)
t(a,b)$\leq$ t(c,d) if a $\leq$ c and b $\leq$ d (monotonicity)
t(a,b) = t(b,a) (commutativity)
t(a, t(b,c)) = t(t(a,b),c) (associativity)

The most commonly adopted t-norm is the minimum. That is, given two fuzzy sets A and B with membership functions $\mu_A(x)$ and $\mu_B(x)$

$$\mu_{A\cap B}(x) = min(\mu_A(x), \mu_B(x))$$

Let us illustrate union and intersection of fuzzy sets by some simple examples. Figure 14 presents a diagram of two fuzzy sets A and B. Draw a bold line to represent the union of A and B ($A \cup B$). In Figure 15 draw a bold line to represent the intersection of A and B ($A \cup B$).
Now consider a discrete example. Suppose we have the following fuzzy sets:

$$A = 0.4/1 + 0.6/2 + 0.7/3 + 0.8/5$$

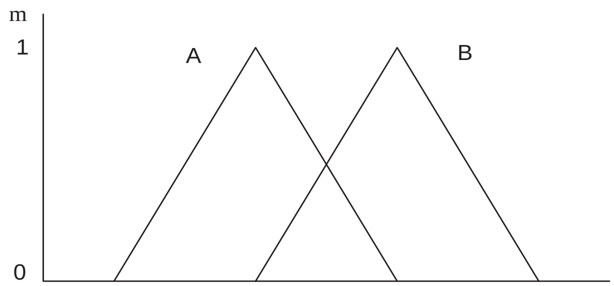

Figure 14: The Union of 2 Fuzzy Sets

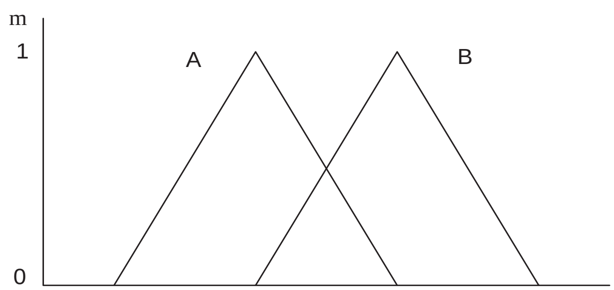

Figure 15: The Intersection of 2 Fuzzy Sets

$$B = 0.3/1 + 0.65/2 + 0.4/3 + 0.1/4$$

The union and intersection of A and B are given by:

$$A \cup B =$$

$$A \cap B =$$

3.3 Fuzzy Complement

To be able to develop fuzzy systems we also have to deal with NOT or complement. This is the same in fuzzy logic as for boolean logic. For any fuzzy set A the complement $\overline{A}$ is given by:

$$\mu_{\overline{A}}(x) = 1 - \mu_A(x)$$

We now have the building blocks for fuzzy systems in the form of union, intersection and complement.

4 Some more on membership functions

The membership function fully defines the fuzzy set. There is no limit to the form that membership functions can take but there are some common examples that appear in real applications. By far and away the most popular are triangular membership functions defined by:

$$\mu_A(x,a,b,c) = \begin{cases} 0 & if\ x \leq a \\ \frac{x-a}{b-a} & if\ a \leq x \leq b \\ \frac{c-x}{c-b} & if\ b \leq x \leq c \\ 0 & if\ x \geq c \end{cases}$$

where a, b and c represent the x coordinates of the three vertices of $\mu_A(x)$. Another

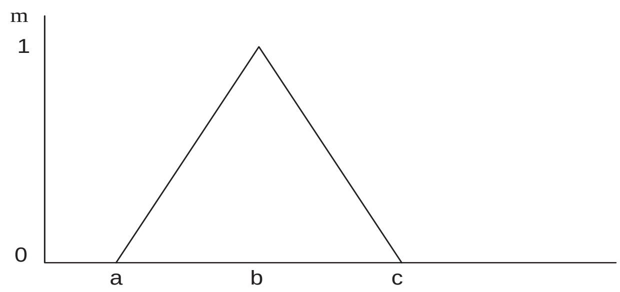

Figure 16: A Triangular Membership Function

popular type of membership function is Gaussian, defined by

$$\mu_A(x,c,s) = exp\left[-\frac{1}{2}\left\{\frac{x-c}{s}\right\}^2\right]$$

where c represents the centre and s its width. See Figure 17 for a schematic diagram (produced by using gaussmf in Matlab with c = 5 and s = 1.5).

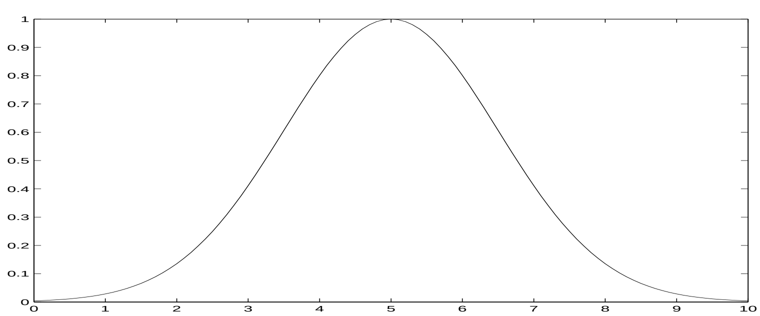

Figure 17: A Gaussian Membership Function

5 Fuzzy Systems

The question we now tackle is "how can we use this notion of fuzzy logic in real problem solving?" Fuzzy sets can be used as part of a computer system for decision making. These systems are variously described as Fuzzy Systems, Fuzzy Knowledge Based Systems and Fuzzy Inferencing Systems. These terms are interchangeable but we shall use Fuzzy System. A Fuzzy System (FS) uses fuzzy sets and fuzzy if-then rules to make decisions or draw conclusions.
A fuzzy system is a computer system that uses fuzzy sets in either the antecedent and/or the consequent of fuzzy if-then rules. It consists of

- The 'base' fuzzy sets that describe the problem;
- The if-then rules;
- Rule Composition;
- Defuzzification.

(see Figure 18). Fuzzy Systems use fuzzy rules and fuzzy reasoning. Fuzzy reasoning is based on the extension principle which allows us to map a function between two fuzzy sets. Suppose we have a fuzzy set A. The extension principle states that if there is a function, f, then the fuzzy set, B, is given by

$$B = f(A) = \sum_i \mu_A(x_i)/f(x_i)$$

This principle lays at the root of all fuzzy inferencing. For our purposes, however, we will not go into the mathematical detail but look at the practical effect in computer systems.

In a FS there are a variety of forms that the rules can take. We will talk about this later. However, they are often of the form

IF x is A THEN y is B

where A and B are fuzzy sets defined on the universes of discourse X and Y respectively. Some examples might be:

- if pressure is high then volume is small;

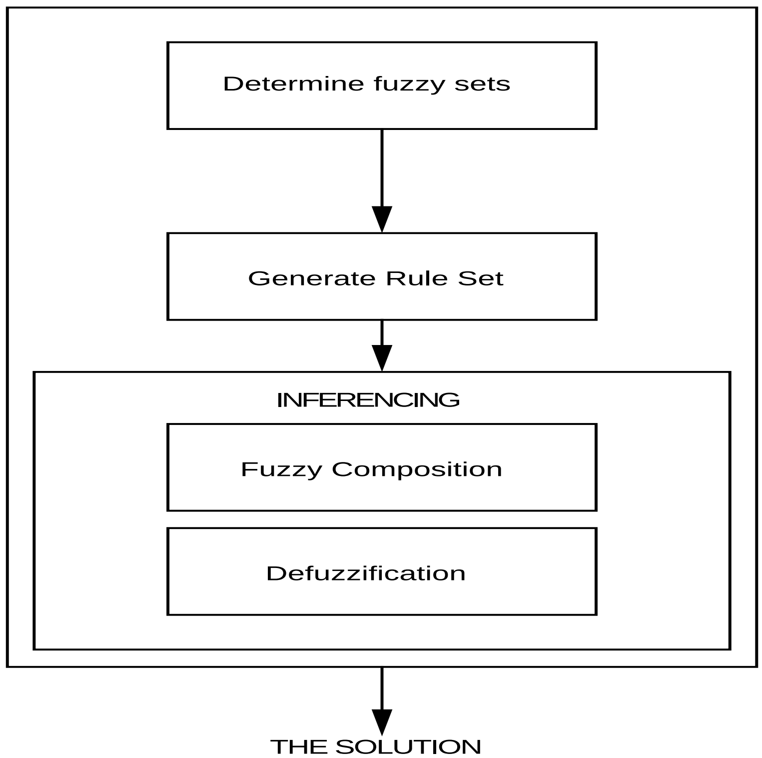

Figure 18: A Fuzzy System

- if a tomato is red then a tomato is ripe.

where high, small, red and ripe are fuzzy sets. Fuzzy Reasoning employs the idea of the compositional rule of inference, which uses the extension principle described earlier.

There are a number of problems and issues that need to be addressed by the developer of a Fuzzy System:

- Determining the Membership Function. Since fuzzy sets are fully determined by their membership function then to be able to use fuzzy sets in a practical application requires the developer to determine the membership function. There is no standard way of doing this but the most common is a 'heuristic' approach where the developer sits down with an expert in the domain and draws the shape of the function arriving at a final shape by discussion. Statistical techniques are often deployed where a number of experts are asked a series of questions to determine the shape. Neural networks and genetic algorithms have also been used. We discuss this in more detail later.

- Determining the Rules. The first question to answer is what type of rules should be used (more on this later). Determining the content of the rules is usually achieved by knowledge acquisition with an expert which, for a large application, can be very time consuming. Some automatic methods, such as neural networks and genetic algorithms, have been suggested.

- Composition operators. There are a variety of ways to combine the rules in a FS. Most applications use the 'max' operator (see later). Again there is no best approach, and trial and error in a particular application will help.

- Defuzzification. Most applications require a 'crisp' solution to a problem yet the various methods for combining fuzzy sets lead to a fuzzy solution. This solution has to be defuzzified and there are a number of possible techniques.

Now we look at the detail of working with fuzzy rules. Suppose we have the following rule:

$$IF\ x\ is\ A\ THEN\ z\ is\ B$$

where x ranges from 0 to 10 and z ranges from 0 to 100. Suppose now we are given a value for x and we wish to use this rule. Figure 19 shows the process. The value of x is used to find its membership in A ($\mu_A(x)$). This is known as fuzzification. This is then used to truncate the set B. The result of the inference on this single antecedent rule is the truncated set. How does this work if we have two antecedents connected by AND?

$$IF\ x\ is\ A\ and\ y\ is\ B\ THEN\ z\ is\ C$$

See Figure 20. These examples are of the type Mamdani. we now look at the two main models used in fuzzy systems.

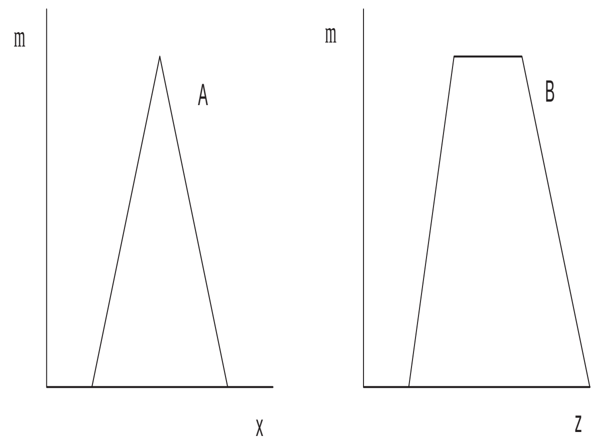

Figure 19: A single antecedent rule

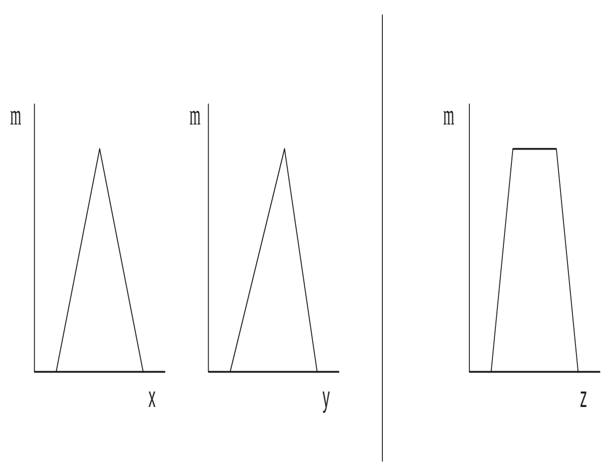

Figure 20: A two antecedent rule

5.1 The Mamdani Model

The Mamdani model uses the rules where the antecedent and consequent are both fuzzy. To illustrate the Mamdani approach a two rule system is considered:

$$IF\ x\ is\ A_1\ and\ y\ is\ B_1\ THEN\ z\ is\ C_1$$

$$IF\ x\ is\ A_2\ and\ y\ is\ B_2\ THEN\ z\ is\ C_2$$

where $A_1, B_1, C_1, A_2, B_2, C_2$ are fuzzy sets. Figure 21 shows how a Mamdani model inferences with two rules. For given values of x and y the procedure is as follows

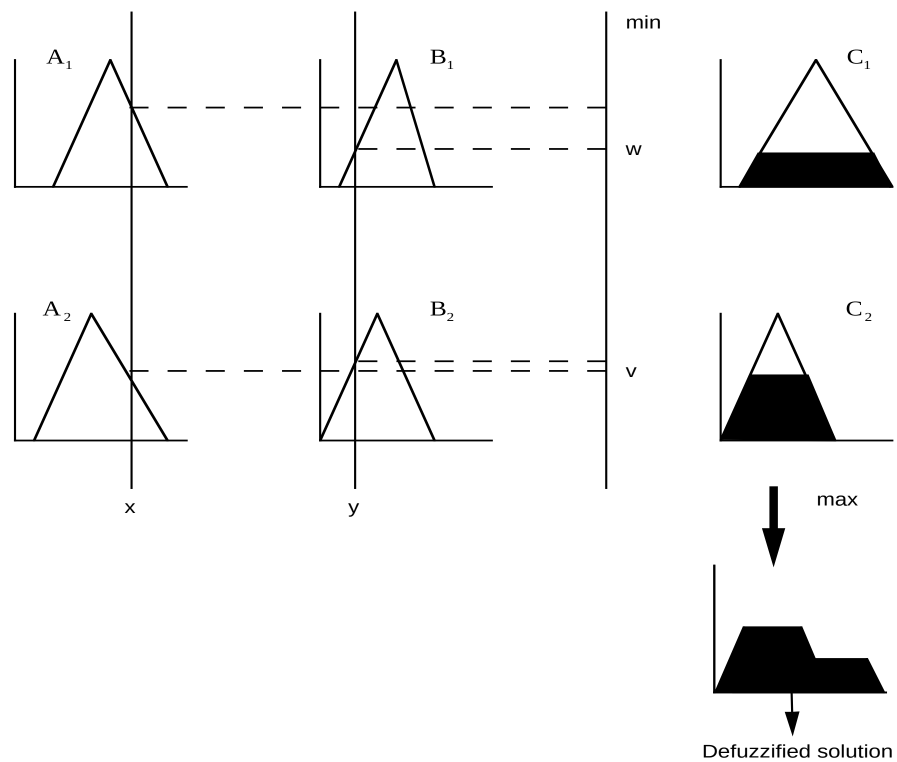

Figure 21: The Mamdani Model for two rules

(using min for AND and max for OR):

1. For the given x find the membership values in A_1 and A_2.
2. For the given y find the membership values in B_1 and B_2.
3. For each rule take the minimum of the membership values in A_i and B_i.
4. Use this value to 'truncate' the fuzzy set $C_i(i = 1,2)$ to produce a new set $C_i'(i = 1,2)$
5. For each value of z in the truncated sets take the maximum to produce the final output fuzzy set.
6. Optionally, 'defuzzify' the output set to produce a single number. This last procedure is known as *defuzzification*.

5.2 The Takagi-Sugeno Model

The Takagi-Sugeno model has if-then rules of the form

$$\textit{IF } x \textit{ is } A \textit{ and } y \textit{ is } B \textit{ THEN } z = f(x,y) \tag{1}$$

where A and B are fuzzy sets but $z = f(x,y)$ is a crisp function in x, y. The antecedent could obviously be more complex with many 'AND's. The function in the consequent can be any function but in a first order Takagi-Sugeno model $f(x,y)$, typically, takes the form $f(x,y) = px + qy + r$ where p, q and r are constants. So, this type of rule has a fuzzy antecedent and crisp consequent. Figure 22 shows how two Takagi-Sugeno rules are combined to produce an output. The two rules are:

$$\textit{IF } x \textit{ is } A_1 \textit{ and } y \textit{ is } B_1 \textit{ THEN } z_1 = p_1x + q_1y + r_1$$

$$\textit{IF } x \textit{ is } A_2 \textit{ and } y \textit{ is } B_2 \textit{ THEN } z_2 = p_2x + q_2y + r_2$$

Given a value for x and y the membership values are found in A_1, B_1, A_2, B_2. For each rule the antecedent AND is then found by, for example, taking the minimum of the membership grades in each rule (any t-norm would suffice). This gives two values w_1 and w_2 which are the weightings for each function. A weighted average of the two functions z_1 and z_2 produces the final output

$$f = \frac{w_1z_1 + w_2z_2}{w_1 + w_2} \tag{2}$$

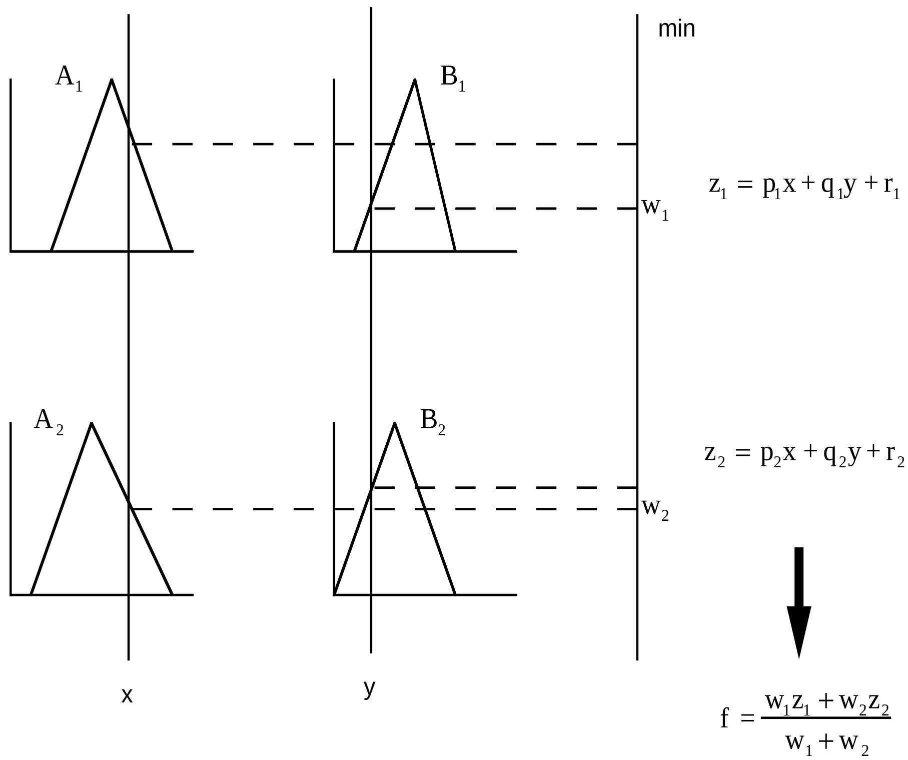

Figure 22: The Takagi-Sugeno Model for two rules

5.3 Defuzzification

Without the defuzzification phase, the final output from the inferencing is, in the Mamdani approach, a fuzzy set. For most applications (especially in control) there is a need for a 'crisp' decision. This is where defuzzification (as its name implies) reduces the fuzzy set to a single number. There are a number of defuzzification techniques available. Two particular methods (Figure 23 provides a schematic) will be described here using the notation used in Jang *et al*(1997):

- The Centre of Area Method.
- The Mean of Maximum Method.

5.3.1 The Centre of Area(COA) Method

The COA method is essentially a technique for finding a mid-point of the final output fuzzy set using a weighted average of the membership grades. Suppose there is a fuzzy set A over universe of discourse Z (the output fuzzy set here) then the Centre of Area z_{COA} is given by:

$$z_{COA} = \frac{\int_{COA} \mu_A(z) z \, dz}{\int_{COA} \mu_A(z) \, dz} \tag{3}$$

5.3.2 The Mean of Maximum(MOM) Method

The MOM method finds the average z where the membership of A is at a maximum. Where the maximum value of $\mu_A(z)$ is denoted by μ^* the Mean of Maximum z_{MOM} is given by:

$$z_{MOM} = \frac{\int_{Z'} z \, dz}{\int_{Z'} \, dz} \tag{4}$$

where $Z' = \{z \mid \mu_A(z) = \mu^*\}$

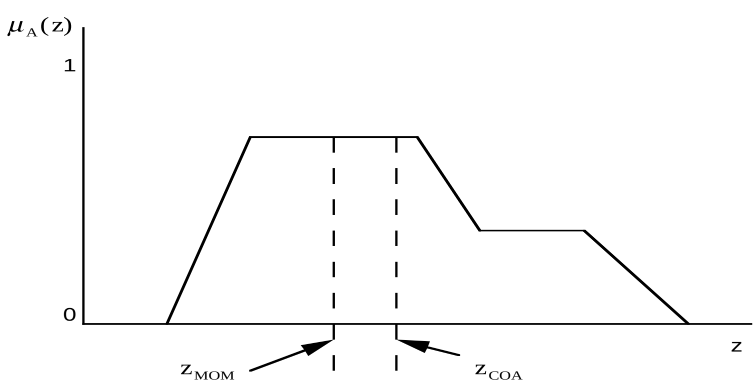

Figure 23: The Centre of Area and Mean of Maximum Methods for Defuzzification of a Type-1 Fuzzy Set.

5.4 Determining Membership Functions

We have already seen that a fuzzy set is fully determined by its membership function. Since fuzzy sets lay the basis for any fuzzy system, the issue of determining membership functions is an important one. There is no accepted method and the literature reports many different approaches. For many applications (especially control) the developer uses triangular or Gaussian functions and in many instances these appear to work satisfactorily. However, for many applications, a good deal of thought and effort is required. Most methods favour eliciting the membership functions from an expert although automatic methods that learn from data have been used. This section considers a number of approaches where the membership function is to be determined by discussion with an expert or group of experts.

Exemplification builds a membership function from a number of samples. So, for example, we might wish to define a membership function ’tall’. The expert would be asked to describe a number of heights as tall using linguistic terms such as true, more or less true, etc. and these would be assigned membership values. For instance ’more or less true’ might be given a value 0.8.

Direct rating presents randomly selected members of the fuzzy set where some measure is available for that member. The expert is then asked a question along the lines ’How tall is Michael Jordan?’ where tall is the fuzzy set on the domain height. S(he) responds by using a sliding scale to indicate the tallness.

Polling asks a different question: ”Do you agree that Michael Jordan is tall?”. This question is asked of a number of experts and the ratio of yes responses to the total responses provides the membership value for Michael Jordan. Reverse rating takes a different approach. The task here would be to ’Identify a man whose height indicates that they possess the degree 0.5 of membership in the fuzzy set tall’. Providing the expert has an understanding of the upper and lower limits, this allows for ready representation.